A Socio-Technical Analysis of Information Systems Security Assurance

A Case Study for Effective Assurance

Job Asheri Chaula

Stockholm University
Department of Computer and Systems Sciences

Stockholm 2006

Report series / DSV No. 06-016
ISBN 91-7155-339-8
ISSN 1101-8526
ISRN SU-KTH/DSV/R-06/16-SE

Printed in Sweden by Universitetsservice US-AB, Stockholm 2006
Distributor: Department of Computer and Systems Sciences

"Efficiency is concerned with doing things right.
Effectiveness is doing the right things"

-Peter Drucker-

Abstract

This thesis examines the concepts of Information System (IS) security assurance using a socio-technical framework. IS security assurance deals with the problem of estimating how well a particular security system will function efficiently and effectively in a specific operational environment. In such environments, the IS interact with other systems such as ethical, legal, operational and administrative. Security failure in any of these systems may result in security failure of the whole system.

In this thesis a socio-technical framework is used to examine culture, usability problems, security internal controls, security requirements and re-use of security requirements of TANESCO IS systems. TANESCO is the energy utility company in Tanzania where the case study was conducted. Results show that culture affects the way people approach IS security. Also results show that the socio-technical framework is effective in modeling systems security and its environment. The re-use of security requirements is also shown to significantly minimise the time taken when developing and improving security requirements for an IS.

The overall purpose of this thesis has been to develop a framework for information systems security assurance. The resulting framework of thinking brings together numerous assurance concepts into a coherent explanation that should be useful for any organisation or evaluators seeking to understand the underlying principals of systems security assurance. It contains organisational, cultural, and technical issues that should be looked at when considering and applying systems security assurance methods and techniques.

Acknowledgement

Writing a thesis for a PhD degree is not something one can do alone. This thesis is a research effort in which many people contributed. It is my great pleasure to take this opportunity to express my gratitude to them all for their generous support.

My gratitude first goes to my mentor Prof. Louise Yngström whose critique, guidance and mental support helped me throughout the research period. Thanks to my second supervisor, Dr. Stewart Kowalski for his valuable seminars and guidance. I would like also to thank Prof. Istvan Orci for being ready to supervise my work in the first one and half year of my research. Through out the trials and tribulations of my Ph.D. studies at UCLAS, Dr Gerald Elifuraha Mtalo, my local supervisor, was always willing to offer me mental and material support, for this I thank him very much.

Thanks to the TANESCO's management for generously accepting my request to conduct a case study at the Directorate of Information Systems. Specifically, my gratitude goes to Nanzarius Chonya, Yona Makala and Tabu Tenga for their support in the process of collecting data and conducting security usability experiment of the electricity prepayment system (LUKU). Also my gratitude goes to my friends Jeffy Mwakalinga for his fellowship, Charles Tarimo and Jabiri Bakari for their contributions in planning and execution of security seminars we conducted in Tanzania.

The Swedish International Development Agency (SIDA/SAREC) entirely funded my research. I would like to extend my thanks SIDA/SAREC and to all personnel at the Department of Computer and Systems Sciences (DSV) of Stockholm University and Royal Institute of Technology for their support. Thanks to Rodolfo Candia, Fatima Santala, Prof. Love Enkenberg, and Birgitta Olsson for assisting in administrative matters. Thanks to Asheri Chaula, Anziwike Chambilo, Elia Chaula, Pudenciana Mlanji, Nicas and Sara Yabu, Geoffrey Massawe, Paul Mulokozi and Ubungo TAG for support and prayers.

I am very grateful to my supportive family, my wife Jennifer for being prudent and loving, my sons Jotham and Jeftah for enduring.

To my wife Jennifer and my sons Jotham and Jeftah

Contents

Table of contents

List of Figures

List of Tables

Chapter 1

1. Introduction

This thesis examines the concepts of Information Systems Security Assurance, (ISSA) using a socio-technical framework. ISSA deals with the problem of estimating how well a particular security system will function efficiently and effectively in a specific operational environment.

Schneier (2000) asserts that most Information Systems (IS) security products on the market are not secure because of lack of assurance. It is one thing to model security trust, threats, design security policy, and build counter measure mechanisms such as firewalls, antivirus, VPN, biometric systems, crypto products, digital certificates and public key infrastructure, but estimating how efficient and effective these systems are is not trivial.

Generally, IS security deals with the prevention, detection and response to adversaries' attacks. In addition, IS security deals with recovery from successful attacks. Attacks to IS may originate from within a computer system or from outside the computer system (Gollmann, 1999; Bishop, 2002). This implies that IS security involves procedural and administrative processes that are implemented in order to protect IS systems.

The security requirements and services of those systems seem to be straightforward and summarized with a few words: confidentiality, authentication, integrity, non-repudiation, access control and availability. However, the mechanisms used to address these security requirements and services can be quite complex and expensive, and understanding them may require rather reasonable security knowledge (Stallings, 1999).

In addition to systems complexity, criminals seek to penetrate security mechanisms in order to commit fraud, steal or modify information, spread viruses and worms, send spam mails or perform illegal actions such as social engineering, terrorism, and industrial espionage. Therefore, the need for reliable security mechanisms and tools for protecting information technology resources has become evident. One way to achieve this goal is for systems development engineers to treat security engineering as part of the whole system's engineering process (SSE-CMM, 2003). That is to say it must be integrated into the organisations' internal controls throughout the system's life cycle.

It has also become evident that any technical system is a part of a larger system; thus there is an environment of each system that interacts and interferes with it. To catch the notion of assurance of information systems security one needs to include environmental aspects of security. Therefore, this thesis examines the processes and parts of information systems security assurance from a socio-technical perspective, including in particular

aspects of culture and usability. This is conducted within the tradition of Computer and Systems Sciences applied to the IS security area.

1.1 Research Background and Motivation

The motivation behind this research is based on the researcher's own experience of teaching, certificate, diploma and graduate level computer courses in addition to working in Tanzania for more than five years with information systems which were used for critical purposes such as Tanzania government integrated financial management systems, energy utility revenue management system, accounting systems, etc. In these processes, we came to understand that when vendors claim that a system is secure they meant that some security feature such as a login module or an encryption module is available in the system. In reality, they provided no evidence of the effectiveness of the security features in the system.

Also generally, there was a lack of basic security understanding. On one hand, the security of the systems deserved one's doubt and on the other hand, users of the systems lacked necessary basic security skills to understand and establish requirements, perform risk analysis, and be aware of attacker potential. Further more, they lacked the necessary skills of making use of information systems in a secure manner.

In addition, there was lack of national and organisational IS security policies; computers' importation was banned in the period between years 1974 and 1993, there were poor harmonization of computerization initiatives, unnecessary duplication of efforts, and waste of scarce financial resources through maintenance and training (eSecretariat, 2001; NICT, 2003).

1.2 Security Implications for Tanzania

There are serious security implications because of Tanzanian's background that is briefly described in section 1.1. The Tanzania Government is the largest market of the IS products supplied in Tanzania by local and foreign vendors. Local vendors, in most cases, act as partners for foreign vendors so in reality most hardware and software systems are imported. Significant effort in the implementation process is on training users of the systems and support staff. Greater emphasis is on the adoption of IS as quickly as possible. Consequently, there are serious systems insecurities. In our view, systems insecurities are due to the following factors:

- Environmental factors such as culture and usability
- Lack or poor internal organisational security controls
- Lack of proper security requirements such as Protection Profiles (PP) for procured systems
- Lack of systems evaluation and testing
- Lack of harmonised security and systems standards
- Lack of security awareness for systems' users, support personnel and other stake holders such as decision makers
- Lack of IS security policy
- Lack of legal environment that addresses issues related to computer fraud, crime and misuse, and privacy issues etc.

The result of the factors listed above, is that the government might loose revenue due to undetected fraudulent and faulty systems, citizen's privacy infringement, loss of revenue due to maintenance costs caused by sub-standard systems, and exposure of government secrets. These problems are multiple and addressing them all is not a straightforward and simple process. However, the use of assurance techniques and methods such as testing and evaluation, understanding the environment in which systems are used, conducting proper systems usability analyses and implementing internal security controls may alleviate the systems insecurity problems.

1.3 The Problem at Hand

Information systems security assurance is difficult to achieve because of a number of issues; for instance, systems complexity, evolving properties of systems and misunderstanding between developers and owners. As a result of this misunderstanding, systems might be used in environments not intended. Other critical issues include usability problems in critical mission systems such as financial management systems. In many such cases, there exist sub-systems that are interoperating with each other and may evolve into individual systems. The security requirements for each system can be different. Users may not understand how to use security features of all these individual systems. Third party modules added after the system is designed further cause complexity. In this view, information systems insecurities result from environmental factors such as human cultural factors and usability. In addition, insecurities result from poor or lack of internal organisational security controls, poor or lack of complete and consistent security requirements, lack of security culture in organisations and costs which are associated with requirements engineering and performing usability analysis.

1.3.1 Culture

Culture can be regarded as a system of values, norms and beliefs that influence society and political systems. Culture has to do with power distance, individualism, collectivism, quantity and quality of life, uncertainty avoidance and orientation for short term and long term (Robbins, 2005). These attributes of culture influence behaviour, tolerance, expressions, and motivations. The cultural forces influence the political system that plays important roles in adjudication of resource allocation. In addition, organisations can establish a security culture by motivating their staff through training and using internal controls to adhere to various security principles. Some of the critical security cultural aspects are trust, adhering to privacy principles, and participation in security making processes and risk analyses, including management's commitment to security, security plans and budgeting.

1.3.2 Usability

Traditionally developers treat usability as an end user problem. Programmers place much effort making sure that interfaces and reports are user friendly. However, from an assurance point of view, usability is a problem not only to the end user, but also to developers, evaluators and testers. Developers of security functionality may develop the wrong security mechanism or protect the right function in the wrong way (Anderson, 2001). In scenarios where testers constitute a different team from the team that actually develops the system, it will take some time for the tester to understand how the system was developed. Consequently, there is a possibility to misunderstand the system and introduce faults or use the wrong test data. Similar problems may face evaluators who may not understand the system as perceived by developers.

1.3.3 Internal security controls

Internal organisational controls may significantly improve the overall systems security even though they may not directly relate to the systems security functionality. They may help to minimise security risks that were the result of lack of such controls as background checks on security personnel. Another example of required internal controls is auditing systems for issues such as patching, documentation and adoption of best practices etc. Availability of internal controls gives first hand information about the level of awareness and security measures that a particular organisation has put in place in order to protect its' IS assets.

1.3.4 Development and re-use of security requirements

End users need to drive their systems requirements engineering and determine how effective those systems are in their operating environments. Evaluation of systems against requirements must take into account the evolving nature of systems. For instance, when systems are patched to fix bugs or modify/upgrade functionality there are risks that new bugs are introduced and the product complexity is increased. The increased complexity and need to identify newly introduced bugs becomes a challenge to the evaluation process. This problem applies for both customised and off-the-shelf products, even though mass-market products debugging can be rapid due to their mass usage.

The re-use of existing security requirements could minimize time and costs of developing security requirements. There exist hundreds of evaluated security requirements publicly available at sites such as NIAP (2005) even though in some cases, re-use of evaluation evidence is limited for reasons of ownership of intellectual property and proprietary information (Bishop, 2002).

1.3.5 Evaluation methods

The use of formalised assurance techniques for systems security specification and validation is required for high assurance levels (CCIMB2, 2005) and may be used for lower assurance levels. In addition, formal techniques are useful for cryptographic algorithms specifications and verification. However, formal techniques are not infallible; the wrong model may be proved, proofs may have errors and unrealistic and overtime proof assumptions may no longer be valid (Anderson, 2001). There is no single evaluation method that addresses all evaluation problems. For instance, the Common Criteria, CC is limited to evaluation of products and does not directly address environmental assurance aspects (CCIMB1, 2005).

1.4 Research Purpose

The purpose of this research is to examine social and technical aspects of information systems security assurance dimensions to be included into an information systems security assurance framework.

1.5 Research Goal and Questions

The research goal is to develop an information systems security assurance framework, which includes social and technical aspects. Addressing these aspects will be conducted within the tradition of Computer and Systems Sciences applied to the IS security area. We expect the research process and the resulting framework that should be helpful for those who seek to examine how technical and non-technical assurance issues are related.

Specifically we seek to focus on how culture relates to security of information systems, how to perform usability evaluation in a cost-effective way, how to measure organisational security controls and how to re-use security requirements. In the framework, we do not aim at providing one short solution to information security assurance but rather provide a structure that combines these aspects of information systems security assurance.

Research questions we are attempting to answer through out our research are:

1 How does culture affect/relate to IS security aspects?
2 How is security culture evaluated?
3 How do organisations develop a security culture?
4 Can re-use of security requirements save time and money?
5 How can usability problems affect the security of information systems?
6 How can usability be evaluated in a cost-effective way?
7 How can effectiveness of internal security controls be measured?

1.6 Research Limitations

Given the depth and breadth of the topic and issues being discussed, we need to define the boundaries and limitations of our research. The boundaries are further described in Figure 2-5 "Approach using the Socio-technical Model", the restrictions are as follows:

Firstly, we are using a specific method and model for our socio-technical investigations (Kowalski, 1994). Secondly, in the socio-technical analysis we are restricting the analyses to deal directly with only three out of four subsystems. This means that we are not focusing on the social subsystem "Structure". However, when discussing and researching issues involved in Security culture and Dimensions of culture we do unintentionally touch upon interactions between "Structure" and "Culture". Thirdly, for questions about Security Internal Controls and Security Requirements and their re-use we used established standards such as Common Criteria (CCIMB2, 2005) and Systems Security Engineering Capability Maturity Model (SSE-CMM, 2003). Fourthly, questions related to usability analyses were directed by demands from ISSA methodology and software development procedures and practices. Fifth, in our studies of cultural issues we were inspired by the GLOBE framework (GLOBE, 2003) and its relations to models developed by Robbins (2005). Other more general and acknowledged information security problems that we did not include into our various studies were the composition assurance problem and the relation between ISSA and business models. Further, security culture maturity has not been covered in this thesis. All our empirical studies were conducted in one, albeit large organization, TANESCO.

1.7 Socio-technical Approach

A socio-technical approach (Kowalski, 1994) will be used to address the research questions. This approach is used here to analyse insecurities of IS systems at TANESCO in Tanzania. TANESCO is a government company that generates and supplies electricity. This case study is used throughout the research to study the issues of security culture, usability analysis, security requirements, and security internal controls.

Kowalski in his work applied General Systems Theory (GST) (Bertalanffy, 1968) to develop a socio-technical security system for protecting information (Kowalski, 1994). The model is depicted in Figure 1-1.

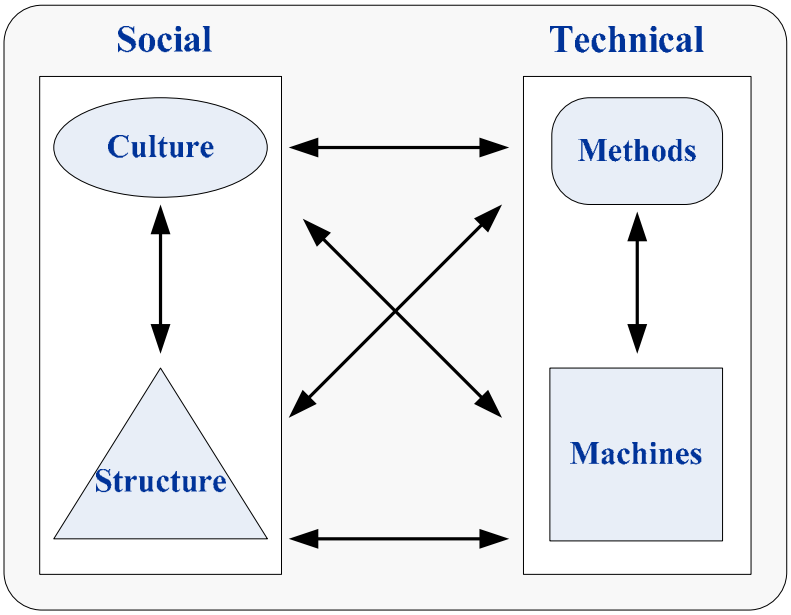


Figure 1-1 Socio-technical System (Kowalski, 1994, p. 10)

He argues that a change in "Machines" does not only affect the "Methods" used but also "Culture" and "Structure" as the system tries to attain balance (or homeostasis). He used this model to focus the analyses on ethics, politics and law, operations and management, and the technology. In our research, we use the model as a thinking aid in efforts to attempt to address the issues of security usability and security culture. We examine the use and re-use of "Methods", the usability of "Machines" and the role of "Structure" in developing a security "Culture".

Figure 1-1 depicts how the system's internal state can be analysed. The arrows depict the flow of analysis of systems stability. A more detailed part of the model includes the Security By Consensus (SBC) parts shown in Figure 1-2 which attempts to model IS systems security by dividing security measures into social and technical categories (Kowalski, 1994).

In Figure 1-2 arrows indicate the flow of analysis of the system whenever there is a change in any of the subsystem of "Methods", "Machines", "Structure" and "Culture". For example if a new security method is introduced in the organisation, then the analysis to determine how the change affects the entire system must consider every layer of the SBC model. Similarly, if a new hardware is introduced the analysis must involve looking at how the new hardware fits in with the existing hardware security mechanisms, operating systems, applications and this process continues until all layers in the SBC model are covered.

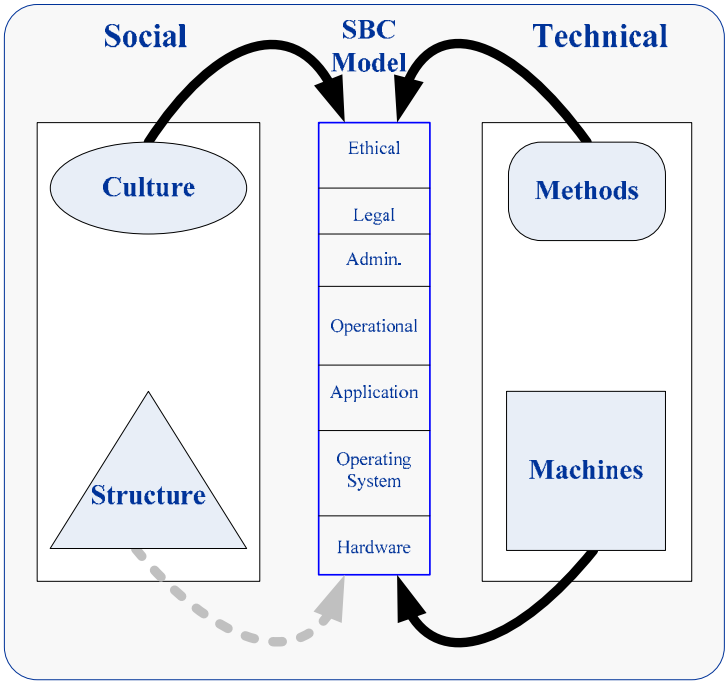


Figure 1-2 SBC Model and Technology and Social Change (adapted from Kowalski, 1994, p. 27)

1.8 The Research Process

Our research process towards the framework includes activities performed during years 2001 – 2006, depicted in Figure 1-3. The process began with a literature review of IS security in general and courses which involved formal lectures on IS security, scientific theory and research methodology. Various literatures clearly indicated that more work is needed to be done on information systems security assurance, especially system evaluation, security usability and non-technical aspects of security (Bishop, 2002; Schneier, 2000; Anderson, 2001; Herrmann, 2001; Herrmann, 2003). In the first phase of our research we attempted to address technically oriented assurance activities; internal security controls, security metrics and security testing of information system. This work was reported and examined in a licentiate thesis (Chaula, 2003) and is included as Appendix G.

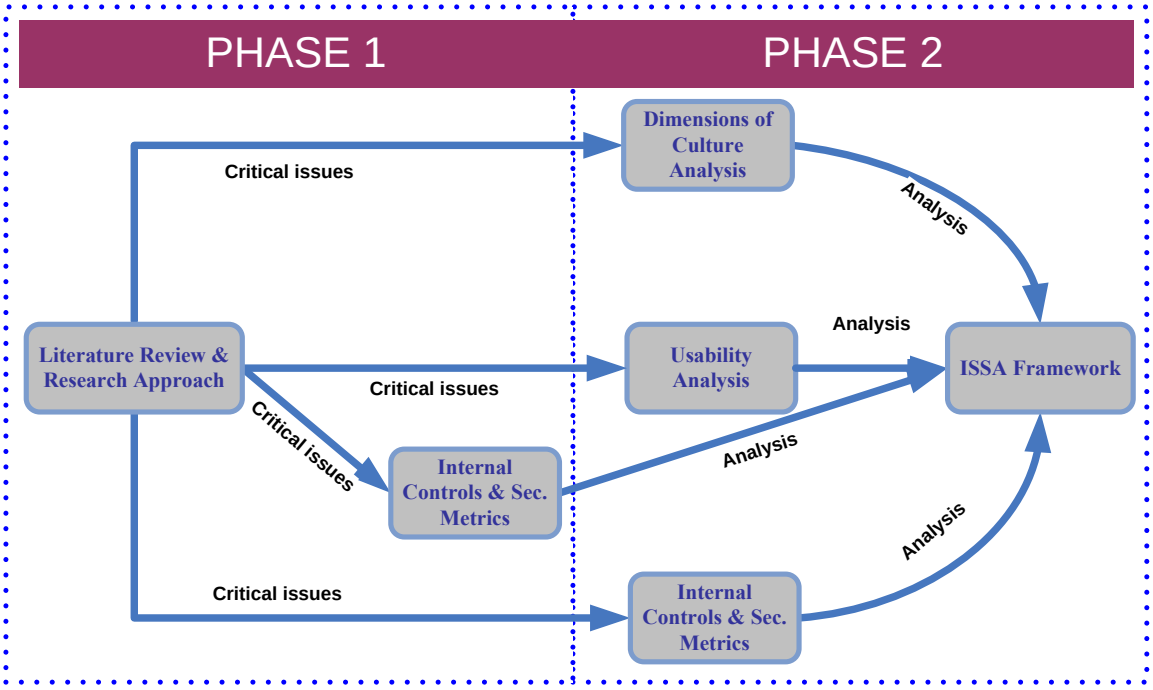


Figure 1-3 Research process: Two phases of the research process

In phase 2 we extended the technical analysis into security requirements and its re-use and included the social oriented analyses of culture and usability. All throughout our research, we had the Tanzanian situation in mind using the case study of Tanzania National Electricity Supplies Company (TANESCO) as our research vehicle. The research purpose and goal – to develop an information systems security assurance framework of thinking was developed taking into consideration security issues addressed in both phases.

1.9 Contributions

The major contributions of this thesis are primarily the application of a socio-technical approach in the tradition of Computer and Systems Sciences to analyse the implications of culture to IS security. In particular, we provide insights on how to evaluate culture and how to use the results to define and make assumptions about organisational behaviour. Organisational behaviour traditionally focuses on defining, predicting and controlling behaviour to achieve objectives such as increased productivity, improved ethical conduct, maintained and improved satisfaction, etc. We argue that understanding organisational behaviour is central when making assumptions about the security environment in which systems are used.

The secondary contribution of this thesis is on how to conduct usability evaluations and develop security requirements in a cost-effective way. This is significant because assurance is perceived to be expensive and time-consuming, hence the need for methods that might be used to analyse systems usability problems and develop requirements within minimum of costs and time.

The third and final contribution is a framework of thinking that provides a coherent explanation on how to address technical and non-technical information systems security assurance problems.

1.10 Thesis Outline

The logical flow of the chapters in this thesis is presented in Figure 1-4 and summaries of the content of each chapter are provided below.

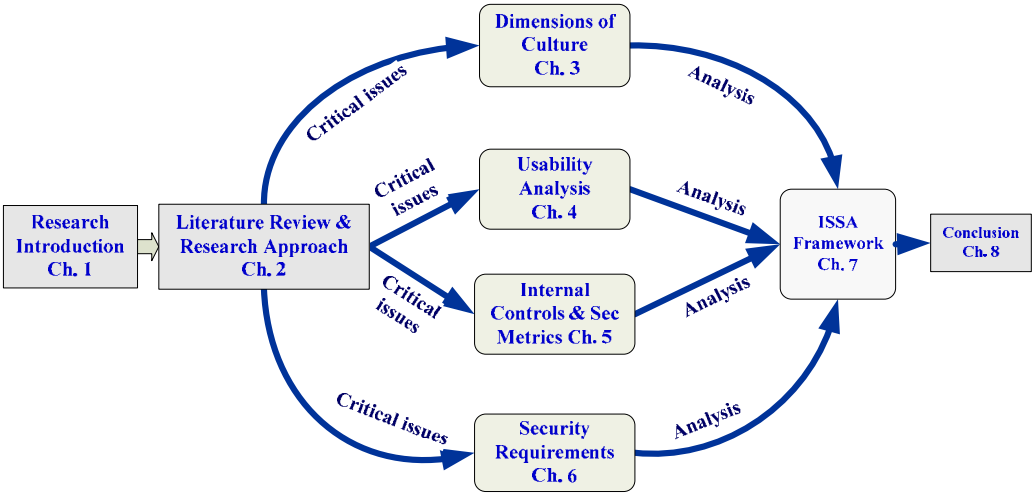


Figure 1-4 Logical flow of chapters

Chapter 1
This chapter presents the IS security overview, the background and the context of this thesis, the purpose of our research, limitations of our work, and the report structure. This chapter will help readers to understand the research area and purpose of this work.

Chapter 2
This chapter presents systems theory, security research and models, information systems security assurance, discussions on appropriate research approaches, research orientation and the case study

Chapter 3
This chapter presents an attempt to identify the key cultural dimensions that make up a security culture. We start off by examining the national dimensions to culture. We then extract the most important of these national dimensions for security and combine them with the most important dimension of organisational culture. We finalise this process by further adding security specific dimensions to create a dimensions inventory for an IS security culture.

Chapter 4
In this chapter, we present usability aspects of information systems security assurance. The focus is on analysing usability problems that are related to the interaction with the system interface. Usability problems occurring in the Man-Machine Interface (MMI) may render a system with well-designed and implemented security assurance policies insecure. We also attempt to address problems that may result in users making mistakes due to poor systems interface design of the security functions.

Chapter 5
This chapter presents and discusses internal security controls and security metrics that can be used in organisations to indicate the maturity of various security processes. This chapter is related to Chapter 4 in that we examine two more levels of the SBC model which are administrative and procedures levels.

Chapter 6
In this chapter the security requirements development process is presented. The security requirements for the LUKU system are then used to analyse the process of re-using security requirements. This analysis is carried out to investigate how to minimize time and costs in the requirement development process.

Chapter 7
In this chapter, we present a framework of thinking about information systems security assurance. The purpose is to present information systems security assurance as a support structure that can be useful when thinking and carrying out ISSA. The development of this structure has taken into considerations issues and challenges that we examined in the previous chapters.

Chapter 8
This chapter presents conclusions and reflections of all the previous chapters. The purpose of the chapter is to show how the research goals were met, what the major contributions to the area of information systems security assurance were and what further research could be performed.

References

Appendix
Appendix A: list of acronyms
Appendix B: National-Organisational culture
Appendix C: Organisational security culture
Appendix D: Tables of data
Appendix E: Security requirements for TANESCOS's PKI
Appendix F: Related publications
Appendix G: Licentiate

Chapter 2

2. Research Background and Approach

The purpose of this chapter is to give an overview of the literature review and research approach. This chapter is divided into six sections which are systems theory, security research and models, information systems security assurance, discussions on appropriate research approaches, research orientation and case study

Our research is mainly qualitative as it aims towards understanding the technical and non-technical aspects of information systems security assurance. In this endeavour we chose to conduct a socio-technical analysis which includes the aspects of security culture, usability testing, specification of security requirements and re-use of such requirements, and the establishment and measuring of internal security controls. All analyses were made on our unit of analysis: the electricity prepayment system LUKU of TANESCO in Tanzania. LUKU is an acronym made out of Swahili words that mean "pay for electricity as you use it".

Our research started by focusing on technical aspects of preventing fraud in the LUKU system. We were looking for efficient ways of decreasing security risks by improving the system testing (Chaula, 2003). After focusing on improving the efficiency of the LUKU system, we then turned our focus towards improving effectiveness. Peter Drucker says "Effectiveness is doing the right things. Efficiency is concerned with doing things right" (Drucker, 1973; Schoderberk, 1990, p. 45). In order to figure out what doing the right things mean, we applied a number of research methodologies. From Checkland (1981) we picked up the notion that formal systems once implemented start being deformed by social forces. From Boulding (1956) we acquired the notion that all theories must have empirical referents. From Kowalski (1994) we were made aware of how technical and non-technical security measures interact, affect and change each other, as the total system drives for a balance between "Culture" – "Structures" – "Methods" – "Machines". From General Systems Theory (Skyttner, 1996; Yngström, 1996) we understood that we had to adopt a holistic approach and that various security models, methods and tools need to be used in order to understand the system and its boundaries.

2.1 Systems Theory and Security

Information systems security research builds on established systems properties, principles, laws and theories that have been developed and refined over a period of time using empirical data. This research builds on the General systems theory (Bertalanffy, 1968), GST basic concepts (Skyttner, 1996), Soft systems methodology (Checkland, 1998; 1981), Systemic holistic approach (Yngström, 1996) and the Socio-technical system (Kowalski, 1994).

Systems theory is a body of concepts and methods for the description, analysis and design of complex entities (Finkelstein, 1988). The classical domain in which systems theory is applicable is that of the engineering of control, information processing and computing systems, all of which consist of component equipments functioning together as a whole. Boulding (1956) classified systems in nine hierarchies as depicted in Figure 2-1

The first level of the hierarchy framework represents static structures, which exhibits static relationships such as the anatomy of cells in living things. The second level clockworks is a level of simple dynamic systems, such as the predetermined motions of the moon around the earth or engine systems we use in cars. The third level cybernetics, encompasses systems which are characterized by feedback mechanisms which regulate the system towards a stable internal state - homeostasis. The fourth level open systems classify systems that are self regulating and self reproductive such as cells. The fifth level is genetic-societal systems characterized by division of labor and slow response to environmental changes. The sixth level is a level of animal system which can be characterized by increased mobility, greater power of storing and processing information, and greater degree of consciousness. The seventh level is of human systems, where an individual is referred to as a system, and adds self consciousness to morality, mobility and goal seeking. On the eighth level social systems are made of humans who are tied together by their roles and channels of communication. The ninth level is a level of systems that are unknowable.

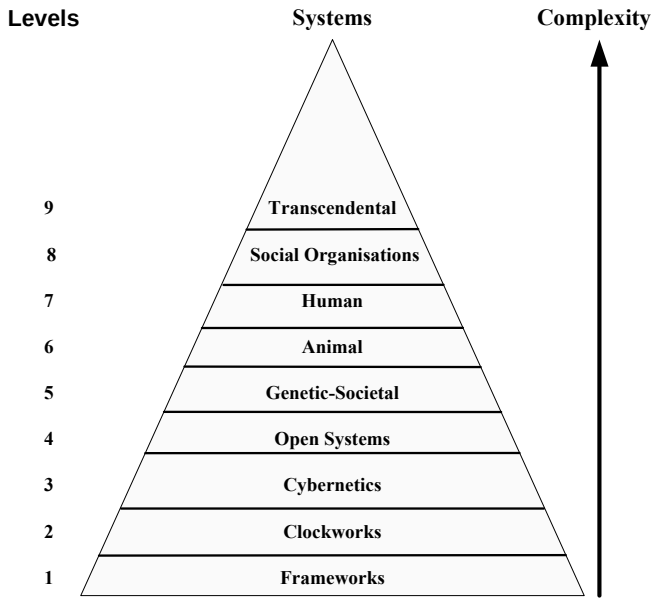


Figure 2-1 Bouldings system of systems, a classification of systems (Boulding, 1956)

Boulding was motivated by the need to present scientific knowledge of his time in the mid 20th century. In other times and with a different motive this classification would have looked quite different. He recognised that all theoretical knowledge must have empirical referents, his classification of systems intended at assessing the gap between theoretical models and empirical knowledge. He stated then that relevant theoretical models existed up to the fourth level, higher levels have insufficient models and empirical referents were deficient within all levels. In our research we focus on level 7 (human) and level 8 (social organisations).

Apart from difficulties we may face in classifying systems, we must also deal with systems principles which are relevant for systems security. A principle is a generalization founded on empirical data not yet qualified into a law (Skyttner, 1996). There exist many such principles. In the section below we outline a few principles which will help us think about systems insecurities.

The darkness principle states that no system can be known completely and the holism principle states that the "*whole is greater than the sum of its parts*" (Bertalanffy, 1968). These two principles are essential features of systems theory. Essentially holism, in the sense of systems theory, means that the modeling and analytical methods of the theory enable all essential effects and interactions in a system and those between a system and its environment to be taken into account.

Other principles are homeostasis and the steady state. Homeostasis principle states that a system survives as long as its variables are maintained within their physical limits and the steady state principle states that every system tries to attain equilibrium (Skyttner, 1996). Systems theory also defines a principle termed as emergent property of systems that is those properties which result from the interaction of system components, properties which are not those of the components themselves.

An information system exhibits principles we mentioned above because it has no existence of its own; it is always a subsystem of some larger system, often called an organisation, or an enterprise. Organisations comprise of people working to achieve certain goals, assisted by a variety of artefacts and constrained by rules and norms of behavior. Information systems exist to support the activities of the organisation, and themselves comprise people and artifacts (Checkland & Holwell, 1998). Information systems, like organisations, are social systems which use technology to help achieve goals. Peter Checkland calls such systems 'human activity systems'. In any organisation there are two types of information systems (Checkland, 1981, p. 317):

> *"Designed systems: Systems that are formally specified, rule-based and purposeful. Most designed information systems of interest are open systems, operating through the interaction of individuals or groups assisted by the use of a variety of tools and instruments.*
>
> *Undesigned systems: Systems that are informal have no specification, may not be authorised and operate through informal and undefined interactions between individuals and groups."*

As soon as a formal system has been implemented, social forces from its environment tend to alter the system by a process of augmentation and replacement. Often, such processes are non-authorised and hence covert, but they may also be the result of properly authorized or semi-authorised actions. Thus, formal systems are fragile, and in their designed form have only a short life. Informal systems, on the other hand, are relatively robust, resist change and must interact with formal system. Informal systems because of robustness can cause difficulties as a result of their behaviour to resist change.

2.2 Security Research and Models

On the IS security research agenda, over the past several decades, there have been areas such as security education, cryptography, security management, systems dependability, legal aspects of IS, forensics, assurance, biometrics, ad-hoc networks security, privacy, etc. Currently research in these areas is still necessary because IS play a central role in organizations, academic institutions, governments and at the family and individual level. Most of these institutions own modern computer networks which are a complex assembly of databases, web and application servers and various network devices that often span across borders of countries and continents. In most cases the convenient solution to achieve this kind of connectivity is connection via open distributed network, the Internet, which is difficult to secure. Consequently, there has been an increase in the number of attacked systems which result in a sense of insecurity and loss of money, reputation and trust.

Current security research has generated many models. They define the philosophy of IS security. Space and time will not allow discussing all of them. However, below we present a few of them which have been widely used in the security community and are used as thinking aids throughout this research. Security models are also the basis for the design, evaluation, and implementation of IS security. Although none of these models in practice can claim to address every aspect of IS security, there are those that have become more widely known and used than others. These are named the Bell and La Padula (BLP) model (Bell & La Padula, 1974), the Biba model (Biba, 1977), the Clark and Wilson model (1998), the Systemic-holistic Model (Yngström, 1996) and the SBC model (Kowalski, 1994). These are presented in this section for completeness and clarity of the security services whose understanding is central when we examine and interpret IS security assurance.

The BLP model is focuses on the assumption that security policies prevent information to flow downwards from a higher security level to a lower security level. It is a model addressing the confidentiality aspects of access control (Bell & La Padula, 1974). The Biba model addresses integrity in terms of how users access objects. In this model, users, processors, and data classification is based on the principle of integrity. In integrity lattice, information may flow downwards (Biba, 1977). The Clark-Wilson model focused on the security requirements of commercial application (Clark & Wilson, 1988). They attempted to address integrity and confidentiality in respect to the differences between military and commercial security requirements. This model defines the concept of the relationship between the system's internal state and the real word. This is referred to as external consistency and is enforced by means outside the computing system, for instance policy. The Systemic-holistic approach is a security model developed by Yngström (1996). It is based on the General System's Theory, Cybernetics and General Living Theory. Using this model one can come to better understanding of where specific details fit into a total system. The model is applicable for security testing and evaluation, security education and many other IS security aspects.

Yngström (1996) examined and addressed the problem of how security knowledge, on an academic level, can be structured and presented. In this endeavour a framework and epistemology collectively called the Systemic-holistic Model was developed. When in use, this model is termed the Systemic-holistic Approach. This approach aims at responding to the need for holistic and interdisciplinary approaches to address security issues.

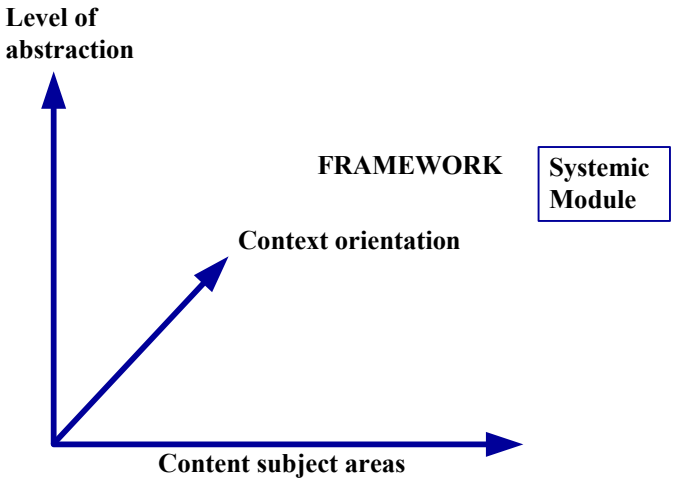


Figure 2-2: The Systemic-holistic Model, Overview (Yngström, 1996, p. 19)

The model is organised into framework and the epistemology, see Figure 2-2. The framework is organised into three dimensions namely content subject areas (technical and non-technical areas), context orientation (geographical, space and time bound), and the level of abstraction (physical constructions, theories/models and designs and architectures). The epistemology part of the model, the systemic module, explains security as a concept of communication and control and acts as meta knowledge to the framework components.

The Systemic-holistic model (Yngström, 1996) and the Socio-technical model (Kowalski, 1994) accentuate the need for holistic and multidisciplinary oriented thinking in addressing systems security issues. Generally it is understood that perfect security is a desirable but unachievable goal. This is due to the systems properties discussed above such as the darkness principal. However, the knowledge about how system security should be organized and how systems (technical artefacts and people) interact within an organisation is central in the assurance process. A user centered security is important because it is one thing to theoretically design a secure system but practically the implementation engineering process faces the reality of design tradeoff and imperfect configuration in the implementation process (Schneier, 2000).

Figure 2-4 depicts how the systems internal state can be analysed using the SBC model. The arrows depict the flow of analysis of systems stability. The SBC model attempts to model IS system security by dividing security measures into social and technical categories that are further divided into subclasses as shown in Figure 2-3.

Social categories include ethical/cultural, legal/contractual, administrative/managerial and operational/procedural. Technical categories include mechanical/electronic, hardware, operating systems, applications and the storage, processing and communication of data. All the above categories can also be grouped into two categories of day to day and emergency.

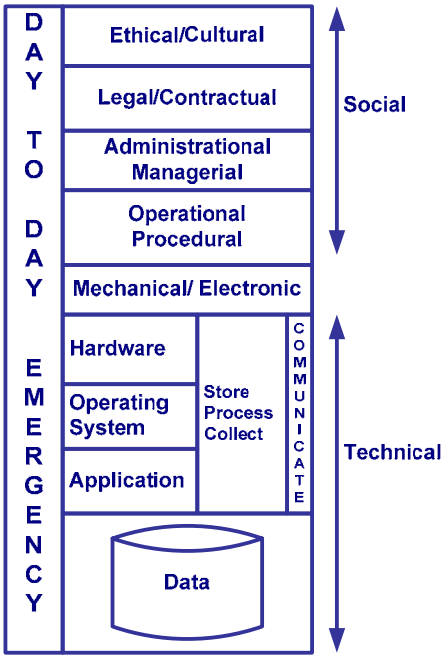


Figure 2-3 The SBC Model (Kowalski, 1994, p. 19)

Arrows in Figure 2-4 indicate the flow of analysis of the system whenever there is a change in any of the subsystems of social/technical. For example, if a new security method is introduced in the organisation, then the analysis to determine how the change affects the entire system must considered every layer of the SBC model. Similarly, if a new hardware is introduced the analysis must involve looking at how the new hardware fits in with the existing hardware security mechanisms, operating systems, applications and this process continues until all layers in the SBC model are covered.

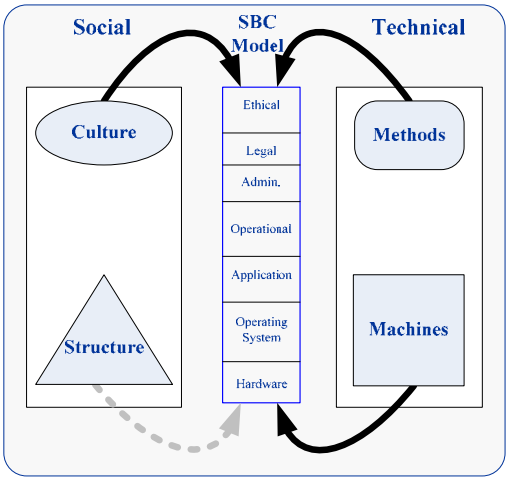


Figure 2-4 SBC Model Technology and technology and social change (adapted from Kowalski, 1994, p. 27)

Social sub-system

The social sub-system is divided in two subsystems, namely culture and structure. This subsystem is used to analyse culture and security culture as shown in Figure 2-5. The culture analysis involves examining how changes in culture affects the security of the system in respect to ethics, legal, administration, organisational, applications, operating system and the hardware. The structure subsystem is concerned with how changes in leadership affect the system security. This subsystem is outside the scope of this research.

Technical sub-system

On the technical sub-system, analysis is done on the methods sub-system where Common Criteria and internal security control methods will be introduced. The machines sub-system analysis will involve usability analysis of the security interface to the security module of the LUKU system and the introduction of seals. Seals are introduced to address physical security problems.

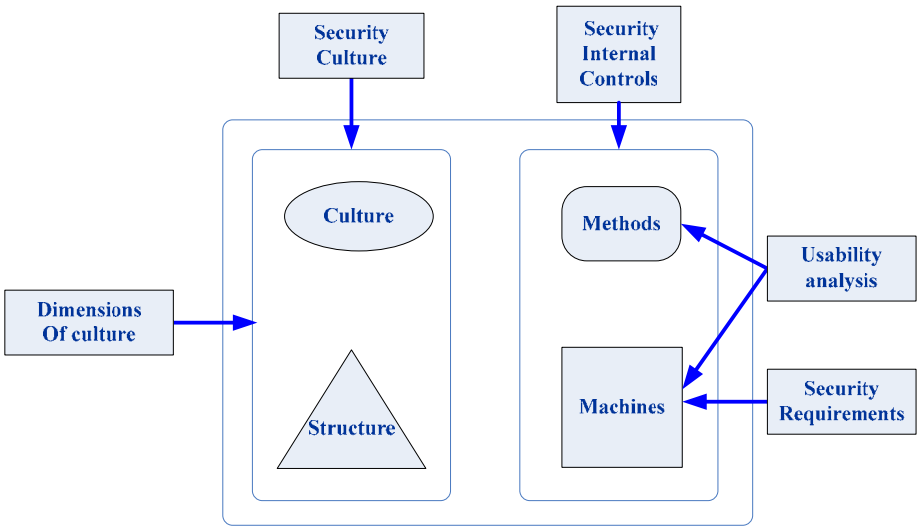


Figure 2-5 Approach using the Socio-technical model

2.3 Information Systems Security Assurance

Information systems security assurance is a process in which evidence that a particular system meets its security requirements are presented. This can be achieved through evaluation and testing. Some authors who examined information systems security assurance include Hermann (2001; 2003), Anderson (2001) and DoD 5-3600.1 (1997).

In order to estimate the confidence or probability that a system will not experience a security failure, there must be evidence that we have applied assurance and evaluation technology. Evaluating a system is a process of gathering evidences for systems correctness and completeness. The evaluation could focus on various aspects of assurance for example, the process used to develop the product (process assurance), the organisational aspects (organisational assurance) and the technical assessment of the product (technical assurance).

The assurance and evaluation techniques can be applied at various stages of a system's life cycle as depicted in Figure 2-6.

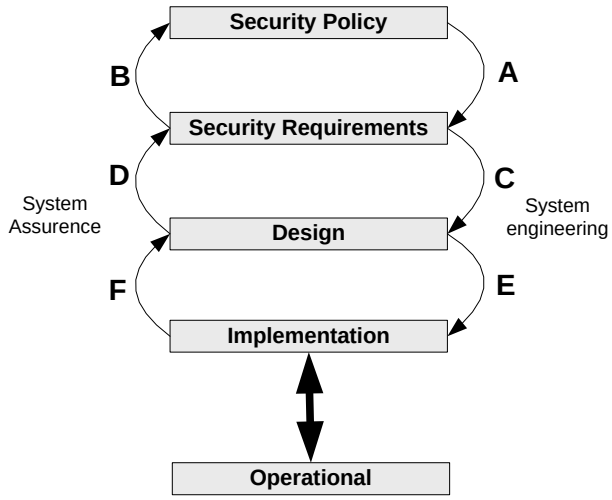


Figure 2-6 System Life cycle assurance (Bishop, 2002)

Security policy can be categorised into three groups namely: general security policy, implementation dependant policy and implementation independent policy (Bishop, 2002). The management usually authorises the general policy that is usually a document in a few pages and the base of all security policies. The implementation dependant policy is a policy for a specific information system such as a smart card. Implementation independent policy could be required to address security requirements for a group of related products such as Public Key Infrastructure (PKI). All arrows labelled A to F in Figure 2-6, indicate the focus of assurance where downward arrows point to implementation and finally operations, and upward arrows point to assurance justifications. Assurance is justified by looking back at policy and requirements to judge whether the implementation and operations fulfil the specification.

Arrow B shows a process to ensure that policies are complete and consistent and address security requirements. Policy assurance process involves examining if policy addresses the threats that are identified and that the policy is suitable for use in the process of developing. Security requirements assurance looks at addressing the question such as are the security requirements sufficient to counter the threats. Arrows D and F shows processes for establishing that the design and implementation are according to the requirements of the security policy. The operational assurance is a process to make sure the system maintains the security requirements during installation, configuration and operation

2.3.1 The concept of the security flaws and security assurance

When conducting security testing it is paramount to keep in mind that security testing is different from normal software testing practices in many ways. This is because security flaws can occur anywhere in the system (Schneier, 2000). They can occur in the design, implementation, source code, platform, interface, protocol, the environment or even in the cryptographic algorithm. Security is a chain and only as secure as the weakest components (Schneier, 2000). Figure 2-7 shows multiple possible sources of security flaws.

The only way to have confidence over any system's security is to over time apply assurance techniques in the system's life cycle. Knowledge about different sources of flaws is necessary when looking at the attacker potential and making assumptions about threats from the system environment. Wrong security assumptions about the environment may result into wrong protection mechanisms (Bishop & Armstrong, 2005). The overall security of any system depends on all areas where a flaw may occur.

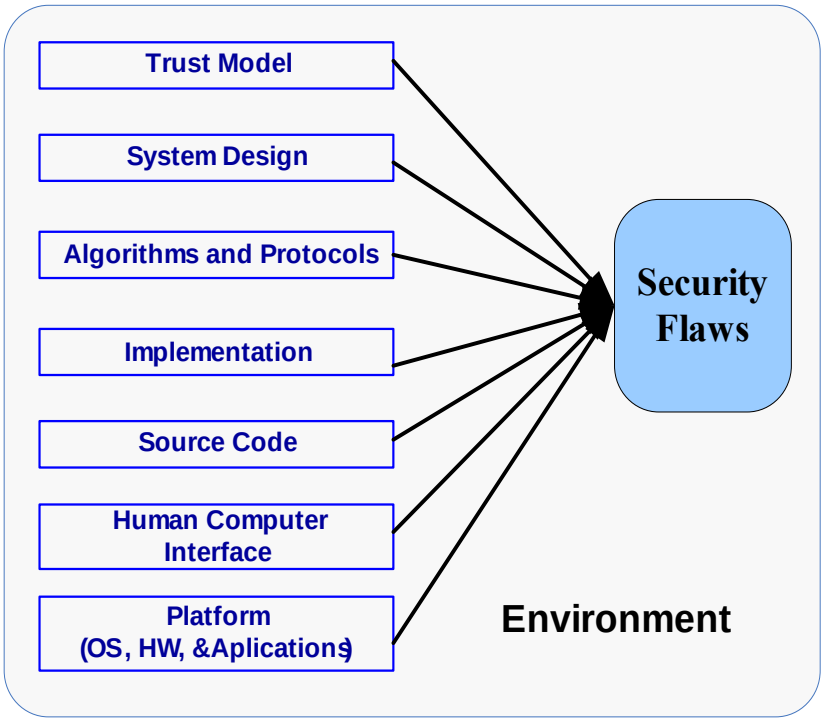


Figure 2-7 Security flaws in environmental contexts

An effective assurance process must take into consideration all sources of flaws. Since this is difficult to achieve, in methods such as the Common Criteria you are allowed to make assumptions about the system's environment in which the Target Of Evaluation (TOE) is placed.

2.3.2 Assurance motives

Anderson (2001) outlines issues for which we may need assurance:

- Functionality
- Strength of mechanism
- Implementation
- Usability

In addition to this list, liability could be one of the most important assurance motives. Liability has to do with addressing legal challenges that may have high potential for damaging the organisation's image as well as causing financial loss. Security functions of information systems can be examined in depth using the Common Criteria (CCIMB2, 2005). The CC is a formalised method used to define security functions such as integrity of stored data, identification and authentication, privacy, resource utilization etc.

Testing and verification of various mechanisms' strength is important because humans are prone to make mistakes, misunderstand, have problem with knowledge acquisition and can even have dishonest motives. Consequently, security mechanisms that may be strong today might not remain strong in the future. In this view, cryptographic algorithms must pass the test of time. As people use it and test it over time and the computing power of for instance PCs increase some cryptographic algorithms become obsolete.

Traditionally, implementation is the focus of assurance. This involves white box testing, black box testing, regression testing and application of different types of review techniques to prove whether the product's implementations are according to specified functionality and strength of mechanism. Common reviews are the inspection of source code for common errors such as buffer overflow, and race conditions. Usability assurance address operational security problem that result due to human errors.

Research by Eloff and Von Solms (20001) on process evaluation/certification and product evaluation shows that security processes assurance is necessary when conducting product evaluation. In this research the focus is on the organisational internal security controls. Martin and Eloff (2001) on the other hand have investigated the assurance aspects of information security culture. The re-ruse of protection profiles are investigated by Jaafar (2004), in which Protection Profiles of the Smart Card for producing the Mobile Phone Digital Rights Management Protection Profile were re-used.

In effort to address the security functionality of IS systems products, Protection Profiles (PP) have been developed and many more PPs are currently under development (ETM, 2003; PPVID, 2004; NIAP, 2005). The NIST website contains a comprehensive list of protection profiles and security targets most of which are for security products such as smart cards, PKI, operating systems and databases (NIAP, 2005). Security usability research focuses on end-user's problems when they using the systems security functions. For instance Liimatainen (2005) focused on end users of distributed systems.

2.3.3 Assurance techniques and methods

Assurance techniques can be categorised as formal, semiformal and informal. Logic, set theory and mathematics are widely used in formal methods. In some cases maximum rigor is necessary in the assurance process, in such cases formalised methods should be used (CCIMB2, 2005). Semiformal methods use natural language with restricted syntax for specification and verification. The sentence structure is restricted and uses meaningful key words or is represented in diagrams (e.g. entity relationships diagrams, data flow diagrams and data structure diagrams) (Sommerville, 2000). Informal methods also use meaningful terms of natural language to convey meaning. This generally imposes minimum rigor on the process used for specification and verification.

Assurance methods provide benchmarks for various aspects of assurance requirements. The focus for various methodologies is mainly on products assurance, process assurance and organisational procedures. Table 2-1 shows some examples of assurance methods.

Table 2-1 Assurance methods

Method	Application area	Reference
CC	Product security assurance	(CCIMB2, 2005)
SSE-CMM	Security process assurance	(SSE-CMM, 2003)
FIPS PUB 140-2	Security requirements for cryptographic modules	(FIPS 1402, 2002)
ISO/IEC 15443-2	A framework for IT security assurance	(ISO 15443-2, 2002)
ISO/IEC 15408-1	Evaluation criteria for IT security	(ISO 15443-1, 2001)
PRISMA	NIST standard that aims to support Critical Infrastructure Protection (CIP)	(PRISMA, 2004)
VNRM	A graphical tool for applying the Network Rating Methodology (NRM)	(VNRM, 1999)
OVAL	Open Vulnerability and Assessment Language	(OVAL, 2006)

Systems security assurance may focus on any of the following:

- Process-based assurance
- Non-technical assurance
- Technical assurance

Process-based assurance focuses on various processes involved in a security project. These might be forming software development team, documentation process or configuration process. Non-technical assurance focus on issues related to legal challenges, human factors, culture and social factors. Technical issues focus on products and systems such as operating systems and applications. Traditionally, systems security assurance focus on technical issues and issues related to organisational challenges. Consequently, less attention is on usability issues.

2.4 Discussion on Appropriate Research Approaches

Research in IS security, has historically, been based on positivist epistemology (Myer, 1997) due to current perception that information systems have to be promoted as a science based on the traditional objectivism associated with the natural sciences (Wynekoop *et al.,* 1997).

However, in recent years increasing interests, needs and awareness for research methods originating from social sciences were pointed out by Anderson (2001). This shift is because security engineering requires cross-disciplinary expertise, ranging from formal methods, cryptography, pedagogical theories, culture, laws, organisational behaviour, systems evaluation and testing, business processes, etc. (Wynekoop, 1993; 1992). Also Viega (2004) points to the needs for systems security engineers to have good and broader knowledge about systems, sources of information and different methods and techniques available for systems security.

Information systems security assurance and evaluation is considered to be one of the hardest topics in security research. System security assurance comes down to the question whether the system will work in its environment and how we convince other people that it works. This raises further questions such as how is the system defined? What is good enough? How do we deal with human errors? How do we deal with wrong requirements? (Anderson, 2001)

Traditionally, the two main components of information systems security assurance are evaluation and testing. Evaluation is the process of establishing evidences that the system meets or fails to meet the security requirements and testing in practice involves black box and white box testing. White box testing includes reviewing product documents and the code and performing test cases while black box testing involves testing the product without documents and the source code.

Both evaluation and testing should not ignore the aspects of usability and internal controls important to ensure that the systems function as specified. Moreover, humans interact with computer systems and in course of this interaction the security of information systems also depends on how humans use it. Taking this view examining information systems security assurance must also include human behaviour and culture.

2.4.1 Underlying epistemology

Epistemology guides qualitative research (Webster, 2005):

> *"Epistemology is the branch of philosophy that studies the nature of knowledge, its presuppositions, and foundations, and its extent and validity."*

Three philosophical epistemological assumptions influence or guide qualitative research in information systems, namely, positivist, interpretative and critical research:

> *"Positivist generally attempt to test theories, in an attempt to increase its predictive understanding of phenomena" (Myers, 1997)*

> *"Interpretative research in information systems aims at producing an understanding of the context of the information system and the process where by the IS influences and is influenced by the context" (Walsham, 1993)*

> *"Critical research focuses on the oppositions, conflicts and contradictions in the contemporary society and seeks to be emancipator" (Myers, 1997)*

2.4.2 Case studies

The term case study has several meanings and definitions. It can describe a unit of analysis (e.g. a case study of a particular organisation) or be a research approach (Myers, 1997). A case study can be seen as an empirical enquiry used for investigating a phenomenon in its real-life context, especially when the boundaries between the phenomenon and context are not known and multiple sources of evidence are used (Yin, 1989). In addition, Stake (1994) identifies three types of case studies namely intrinsic case study, instrumental case study and collective case study. Intrinsic case studies aims at increasing the understanding of a phenomenon and make sense of the case being studied. Instrumental case study aims at refining a theory. In a collective case study a researcher aims at using several case studies to compare and draw general implications of the phenomenon being studied.

2.4.3 Qualitative vs. quantitative research

The validity of qualitative research can be judged by looking at credibility, transferability, dependability and conformability while internal and external validity, reliability and objectivity are used to judge the validity in quantitative research as compiled in Table 2-2 (Lincoln & Gubas, 1985; Hoepfly, 1997).

Table 2-2 The Validity of quantitative vs. qualitative research

Qualitative research	Quantitative Research
Credibility	Internal validity
Transferability	External validity
Dependability	Reliability
Conformability	Objectivity

In qualitative research the *credibility* depends more on the analytical ability of the researcher and the richness of gathered information than on the sample size of data (Patton, 1990). Lincoln and Gubas (1985) suggest that credibility could be enhanced by making raw data available for others to analyse or asking respondents to corroborate the findings. For *transferability of findings*, the researcher must provide sufficient information to the reader who subsequently can determine whether the findings are applicable to his or her environment (Lincoln & Gubas, 1985). This is also related to rigor and relevance as defined by Lincoln and Gubas (1985, pp. 290) as how the enquirer persuades himself and his readers that a study is worth taking into account. *Dependability* and consistency are directly related. *Conformability* refers to the degree that we can demonstrate the research interpretation neutrality using criteria as outlined by Lincoln and Gubas (1985, pp. 320-321) where the research must provide audit trail data such as raw data, analysis notes, preliminary development and analysis information, processes notes, etc. One of the main advantages of using qualitative research methods is the ability to be able to ask the right questions (Myers, 1997; Stake, 1994; Klein, 1999).

In quantitative research internal validity refers to the degree to which the findings describe reality, external validity is shown by the ability to generalize findings in different environments, reliability is shown by the degree to which measurements remain the same when the experiment is repeated and objectivity is shown by quantitative measures which are value-free.

2.5 Research Orientation

Figure 2-8 describes the breadth and depth of our research as also communicated – however differently - in Figure 1-3 “Research process: Two phases of the research process”, Figure 1-4 “Logical flow of chapters” and Figure 2-5 “Approach using the Socio-technical model”. Our main research vehicle is the case study of TANESCO where the investigations of critical security assurance issues were carried out using contemporary methods and in a real life context. The following were key considerations in the research process:

- The method involved participant observation, interviews (oral and questionnaires) and documents review
- The research was carried out in the field in order to develop deep understanding

- The author is used as a primary data collection instrument
- Research questions were modified during the course of study hence questionnaires had to be modified to simplify the language and exclude questions that were too intrusive
- Data includes documents, screen shots, and interview answers
- The subjects/respondents and the observer/researcher negotiated interpretations of research questions and answers
- In the interviews, the meaning of the word ICT security was constantly sought

The considerations highlighted above, are similar with those highlighted by Lincoln and Gubas (1985), Bogdan *et al.* (2002) and Myers (1997) hence supporting the choice of the research methodology. In addition, technical assurance was investigated using traditional positivist-oriented methods.

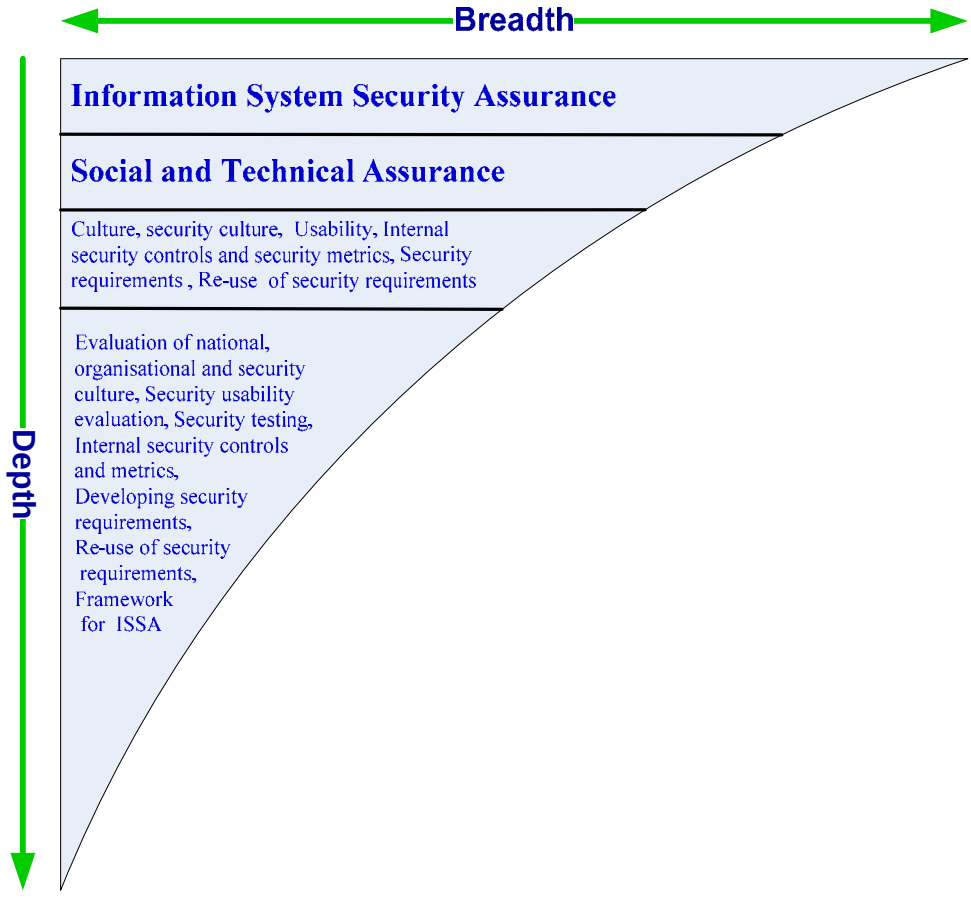


Figure 2-8 Research breadth and depth

2.5 The Case Study

The case study was divided into three activities at TANESCO, the Tanzania electrical supply company. The first activity was a questionnaire including face-to-face interviews with employees. The second activity included a usability study involving other employees. The third activity was a document review performed by the author.

The case study was carried out at the TANESCO because it is one of the largest corporations in Tanzania. Our specific interest in studying LUKU, the electricity-prepaid system, is that it is critical to TANESCO. It is a critical infrastructure because of its use to collect revenue. Consequently, the system has been subject to re-occurring attacks by adversaries who fraud the system with the motive of financial gain.

2.5.1 Background of Tanzania Electric Supply Company Limited (TANESCO) and its electricity prepayment system

The Tanzania Electric Supply Company Limited (TANESCO) is the national parastatal power utility company, which supplies all electricity in Tanzania. The first supply of electricity in Tanzania was established in 1908 at Dar es Salaam. It served the railway workshops and part of the town. In 1920, a Government Electricity Department was formed under the Tanganyika Railways to foresee generation and distribution of electricity (TAN, 2005).

In 1931, the power industry was privatised and two private power companies were formed; one was the Tanganyika Electric Supply Company Limited (TANESCO), which was given a concession area on the Pangani Falls near Tanga. The other company was the Dar es Salaam and District Electric Supply Company Limited (DARESCO) whose concession areas was Dar es Salaam, Dodoma, Tabora and Kigoma; and later expanded to Mwanza, Moshi, Mbeya, Morogoro, Mtwara and other townships. Electricity generation in all these towns greatly depended on imported diesel oil until the first hydro power station was constructed at Kange, Tanga, along Pangani River.

In 1964, the Government bought all the shares from the two private companies and merged them into a single utility under TANESCO. During the past three and half decades, TANESCO has constructed the national grid system; numerous 220 kV, 132 kV, 66 kV, 33 kV and 11 kV transmission distribution lines as well as 400V / 230V lines connecting customers. Only about 11 percent of the country's estimated population of a 34 million, as of 2002, have access to reliable electricity. The national grid spans over a total of 12,934 km. 40 % of the electricity is household consumption, industry and businesses use 50% and 10% is used for public lighting and exports to Zanzibar Island (TAN, 2005).

Information systems at TANESCO are used to control the national grid through phase monitoring, fault detection, communication, operations, meters calibrations and testing, billing and auditing. Computer systems were used since the national grid came into being but extensive computer usage has been since early 1990s when computers also could be used for operational and billing purposes. Recently, TANESCO stated to make extensive use of its IT infrastructure to access the Internet and establish a fibre optics network to connect some of their major generation and transmission stations across the country. The case study was carried out at the electricity prepayment department. The prepayment system, LUKU, was established in 1995 with more than 60,000 customers in Dar es Salaam. Currently the customer base is several hundred thousands and the system is being expanded to other regions such as Arusha and Dodoma. The LUKU system is considered one of the critical systems because it is used to collect and manage revenue.

2.5.2 Electricity Prepayment System (LUKU)

The Electricity Prepayment System, LUKU is depicted in Figure 2-9. LUKU is an abbreviation for the Swahili word "Lipa Umeme Kadiri Unavyotumia", meaning "Pay for

electricity as you consume". The system involves purchases of encrypted magnetic card (token) or keypad token whose application on the LUKU meter provides the customer with electric power as paid for. Since the introduction of the LUKU System in TANESCO, there have been a number of problems -fraud cases- related to the security of the LUKU System.

The system has been subject to attacks despite the presence of security controls such as the installation of a hardware security module that stores the encryption keys and performs the encryption. There have been serious problems of illegal manufacturing of the Coded Electricity Tokens (LUKU Tokens) by insiders collaborating with outsiders, and these tokens are sold at cheaper prices on the streets. As a result, there is a loss of revenue. LUKU reports show that this fraud is even more rampant now than in the past.

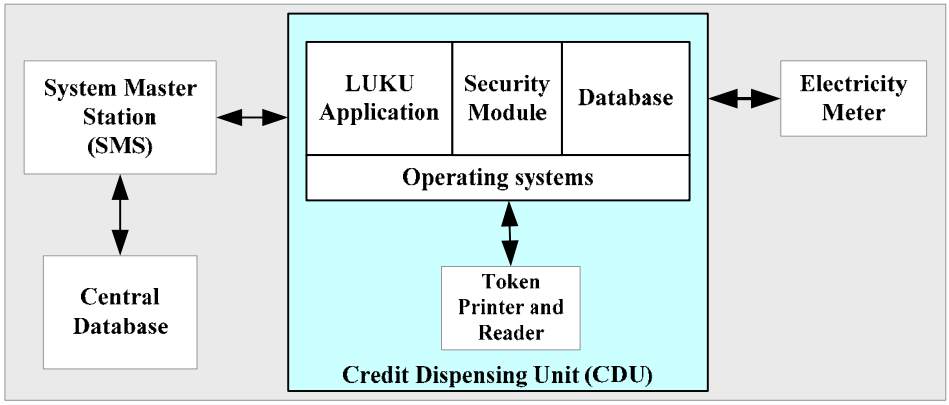


Figure 2-9 LUKU Prepayment system

The motives behind choosing the LUKU system for the case study includes the insecurities that are reported over the period of six years, the fact that the system is subject to upgrades, that it supports a critical mission, and that it is used in other African countries such as South Africa.

2.5.3 The questionnaires and respondents

Three types of questionnaires were used targeting culture, ethics and security culture which are found in appendixes (B & C). Twenty-six participants in the research were staff from operations and information systems departments with the following roles: Information systems director (N=1), security manager (N=1) Systems analysts (N=2), Technicians (N=16), System master station operator (N=1), Developer (N=1), Software development manager (N=2), and Systems administrators (N=2).

2.5.4 The usability experiment

The usability testing experiment was conducted to analyse how usability testing can be conducted in a cost-effective manner using security heuristics and a questionnaire that was designed to cross-check for internal consistency. Five systems administrators were involved. The heuristics were developed using existing security requirements of what a user interface should look like. In this process we derived some questions from the Common Criteria (CCIMB2, 2005).

2.6 Chapter Conclusion

This chapter gave an overview of the literature in systems theory, security research and models and discussed appropriate research approaches for our research including giving background information on the case study. In the next chapter we examine IS security culture.

Chapter 3

3. Information Systems Security Culture

This chapter presents an attempt to identify the key cultural dimensions that make up a security culture. We start off by first examining national dimensions to culture. We then extract the most important of these national dimensions for security and combine them with the most important dimensions of organisational culture. We finalise the process by further adding security-specific dimensions to create a dimensions inventory for an IS security culture.

This chapter has six main sections. The first and second sections present the survey using the GLOBE (2003) nine cultural dimensions for evaluating national culture. The nine dimensions are visualised using spider diagrams and five of these dimensions are proposed as key dimensions when dealing with IS security. The third section examines the implications of culture on IS security. The fourth and fifth section examines how to develop an organisational security culture. In the sixth section, the chapter summary is presented.

3.1 Culture dimensions surveys

The difference between national and organisational culture is that national culture represents values that are dominant in the whole nation and organisational culture represents values that are dominant in a particular organisation. However, Robbins (2005) argues that national culture, organisational culture and employee behaviour can be correlated. He further argues that national culture influences employee more than organisational culture. Therefore, knowledge about national culture is paramount if accurate prediction of employee behaviour in an organisation is sought. In this view, if an organisation plans to develop effective security culture, the security culture should not be developed in isolation of national culture and the organisational culture. National and organisational culture can be described using dimensions. Table 3-1 presents 9 national dimensions (GLOBE, 2003) and 7 organisational dimensions (Robbins, 2005) which we chose as the point of departure for our surveys.

The data was collected in a survey where respondents were arbitrary chosen from the operations and information systems departments at TANESCO. Operations and information systems department represent the core of the company because the two departments comprise all the key projects namely: energy generation, metering and billing, national grid control centre, technical support, software development and auditing. Since people with different roles may have different views on different issues, the survey involved people with different roles across the organisation including technicians, managers, directors, operators, administrators, developers and analysts. Forty questionnaires were distributed but only 26 (65%) were returned. The breakdown of respondents is as follows: technicians (N=16), software development manager (N=2),

information systems director (N=1), system master station operator (N=1), security manager (N=1), systems administrators (N=2) developer (N=1) and systems analysts (N=2). Most of these respondents were males.

The respondents were given three questionnaires covering national culture, organisational security culture and ethics. The national culture questionnaire contained 18 questions covering 9 dimensions (see Table 3-1), and every dimension had two questions. The first question covered "as is" the current perceived attitude. The second question covered "should be", that is the respondents attitude towards how this culture dimension should be in an ideal culture. Respondents answered the statements by using a 7-point bipolar rating scale with the anchor points "Strongly disagree"-"Strongly agree".

The organisational security questionnaire contained 39 statements (see Appendix C and Table 3-7) covering 11 dimensions, see Table 3-7, and each dimension had several questions. The respondents answered the statements by using a 5-point bipolar rating scale with the anchor points "Strongly disagree" – "Strongly agree", see Table 3-8.

The ethics questionnaire contained 22 statements, (see Appendix C). The respondents answered the statements by using a 5-point bipolar rating scale with the anchor points "Strongly disagree" – "Strongly agree", see Table 3-8.

3.2 National Culture Dimension Survey

Table 3-1 shows dimensions that can be used in surveys to evaluate national and organisational culture (Robbins, 2005). In the survey that we conducted, emphasis is on measuring how employees perceive their national culture rather than their organisational culture. Table 3-1 shows the national culture dimensions that are used by the GLOBE (2003) project to evaluate national cultures around the world and the dimensions for measuring organisational behaviour as presented by Robbins (2005).

The organisational culture dimensions in Table 3-1 do not include components such as power distance and the future orientation that we assume necessary parameters in relation to information systems security. We focussed the data analysis on revealing cultural values that might have effect on how the organisation works with strategic contingencies. These are necessary to ensure that business continuity and systems security plans are appropriate, and provide the necessary security level. The emphasis is on measuring how employees perceive shared meaning of their national culture.

Table 3-1 National culture and organisational culture dimensions

National culture (**GLOBE, 2003)**	Organisational culture (**Robbins, 2005)**
• Assertive Orientation • Power Distance • Uncertainty avoidance • Humane Orientation • Institutional Collectivism • In-group collectivism • Gender Egalitarianism • Future orientation and • Performance orientation	• Innovation and risk taking • Attention to details • Outcome orientation • People orientation • Team orientation • Aggressiveness • Stability

Statements and their related dimension are found in Table 3-2.

Table 3-2 National culture dimensions and related statements (GLOBE, 2003)

Dimension	Statements
Assertive orientation	1. People are generally dominant. 2. People should be generally dominant.
Power distance	3. Followers are expected to obey their leaders without question. 4. Followers should be expected to obey their leaders without question
Uncertainty avoidance	5. Most people lead highly structured lives with few unexpected events. 6. Most people should lead highly structured lives with few unexpected events.
Humane orientation	7. People are generally very tolerant of mistakes. 8. People should be generally very tolerant of mistakes
Institutional collectivism	9. Leaders encourage group loyalty even if individual goals suffer. 10. Leaders should encourage group loyalty even if individual goals suffer.
In-group collectivism	11. In this society, children live with parents until they get married. 12. In this society, children should live with parents until they get married.
Gender egalitarianism	13. Boys are encouraged more than girls to attain a higher education. 14. Boys should be encouraged more than girls to attain a higher education.
Future orientation	15. More people live for the present than for the future. 16. More people should live for the present than for the future.
Performance orientation	17. Students are encouraged to strive for continuously improved performance. 18. Students should be encouraged to strive for continuously improved performance.

These statements are designed to obtain information from the respondents with respect to the actual cultural value "as is" and the ideal cultural values "should be". The "should be" values are measured when subjects respond to statements, which describes an ideal cultural value, for instance "more people should live for the present than for the future". A higher score will indicate that the subject believes these values are ideal, a lower score will indicate that the subject does not believe that these values are ideal.

Each of these characteristics exist in a continuum from "Strongly disagree" to "Strongly agree" as shown in Table 3-3.

Table 3-3 Rating scale for measuring culture

1 Strongly Disagree	2 Disagree	3 Tend to Disagree	4 Undecided	5 Tend to Agree	6 Agree	7 Strongly Agree

Appraising the organisation using these measurements, gives a composite picture of how the organisation is influenced by the national culture. Table 3-4 shows the number of respondents who have similar perception on national culture dimension. For instance, if we take the national cultural dimension assertive orientation we see that under the disagree column four respondents disagree that people are dominant and six respondents disagree that people should be dominant.

3.2.1 National culture survey results

Table 3-4 and Table 3-5 give the results of the survey in numeric and percentage form.

Table 3-4 Number of respondents with similar perception about a national culture dimension

Continuum between 1 & 7		1	2	3	4	5	6	7
		Strongly Disagree	Disag.	Tend to Disagr.	Undecided	Tend to Agree	Agree	Strongly Agree
Assertive orientation	As is	0	4	3	5	9	4	1
	should	3	6	3	3	6	3	2
Power distance	As is	5	9	3	1	4	2	2
	should	8	6	2	1	7	1	1
Uncertainty avoidance	As is	3	1	7	5	8	1	1
	should	1	6	8	3	4	3	1
Humane orientation	As is	0	6	1	5	8	6	0
	should	2	7	1	3	8	3	2
In-group collectivism	As is	1	2	0	5	7	10	1
	should	0	9	1	4	4	7	1
Institutional collectivism	As is	3	6	0	5	7	4	1
	should	2	5	4	3	5	6	1
Gender egalitarianism	As is	3	5	2	2	5	9	0
	should	11	7	5	0	0	1	2
Future orientation	As is	1	1	2	3	6	11	2
	should	6	10	4	2	1	2	1
Performance orientation	As is	0	0	0	1	3	13	9
	should	0	2	0	3	3	9	9

Table 3-5 shows the percentage of respondents for each dimension. The percentages are computed from Table 3-4.

Table 3-5 Percentage of respondents for each national culture dimension

Continuum between 1& 7		1	2	3	4	5	6	7
		Strongly Disagree	Disagree	Trend to Disagree	Unsure	Tend to agree	Agree	Strongly Agree
Assertive orientation	As is	0%	15%	12%	19%	35%	15%	4%
	should	12%	23%	12%	12%	23%	12%	8%
Power distance	As is	19%	35%	12%	4%	15%	8%	8%
	should	31%	23%	8%	4%	27%	4%	4%
Uncertainty avoidance	As is	12%	4%	27%	19%	31%	4%	4%
	should	4%	23%	31%	12%	15%	12%	4%
Humane orientation	As is	0%	23%	4%	19%	31%	23%	0%
	should	8%	27%	4%	12%	31%	12%	8%
In-group collectivism	As is	4%	8%	0%	19%	27%	38%	4%
	should	0%	35%	4%	15%	15%	27%	4%

Continuum between 1& 7		1	2	3	4	5	6	7
		Strongly Disagree	Disagree	Trend to Disagree	Unsure	Tend to agree	Agree	Strongly Agree
Institutional collectivism	As is	12%	23%	0%	19%	27%	15%	4%
	should	8%	19%	15%	12%	19%	23%	4%
Gender egalitarianism	As is	12%	19%	8%	8%	19%	35%	0%
	should	42%	27%	19%	0%	0%	4%	8%
Future orientation	As is	4%	4%	8%	12%	23%	42%	8%
	should	23%	38%	15%	8%	4%	8%	4%
Performance orientation	As is	0%	0%	0%	4%	12%	50%	35%
	should	0%	8%	0%	12%	12%	35%	35%

3.2.2 Visualisation of national cultural survey results

It was found during the course of the study that it was difficult to use the tables above to identify any relevant patterns for IS security. To overcome the difficulty we applied a heuristic visualisation technique of spider diagram. Heuristic is a method of solving problems by evaluating past experiences and moving by trial and error to a solution. A visual heuristic method is to evaluate visual experiences by trial and error and move towards a solution.

In the section to follow we visualise each cultural dimension one by one, in order to detect possible gap patterns between "as is" and "should be". We believe that this gap will give an indication of cultural dissonance which might prove to be important to our inventory of security cultural dimensions. We conclude these individual visualisations with a composite visualisation that includes all 9 dimensions.

Assertive orientation

Assertive orientation is the degree to which individuals are assertive, dominant and demanding in their relationships with others. This culture dimension is measured using the following statements: "People are generally dominant" and "People should be generally dominant". The survey results show that respondents tend to agree that individuals in the society are dominant and demanding. Data in Figure 3-1 show that, some people agree and many are undecided.

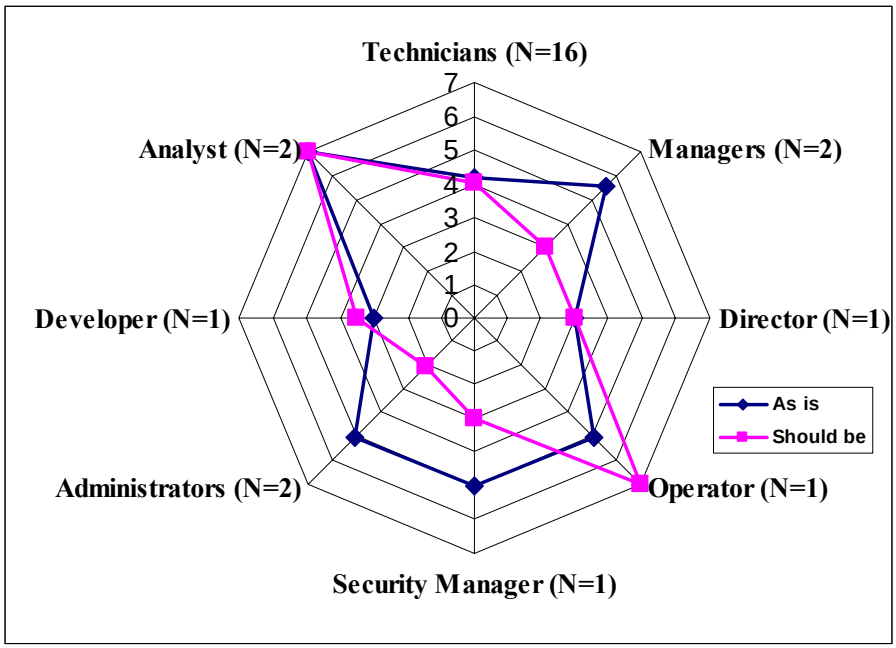


Figure 3-1 Assertive orientation scores for each role

Power distance

Power distance is the degree to which members of the organisation expect power to be distributed equally. Statements that are designated to give a qualitative value for this are: "Followers are expected to obey their leaders without questions" and "Followers should be expected to obey their leaders without questions". If the score is towards 1, it implies that people perceive that power distance is equally distributed in the society and if it tends toward 7 powers is not equally distributed. Figure 3.2 shows that people from all roles perceive that power distance is currently high and they think it should be reduced with the exception of developers who perceive power distance to be low and they suggest it should be increased. The overall picture indicate a gap pattern and this dimension qualifies to be used to analyse IS systems security.

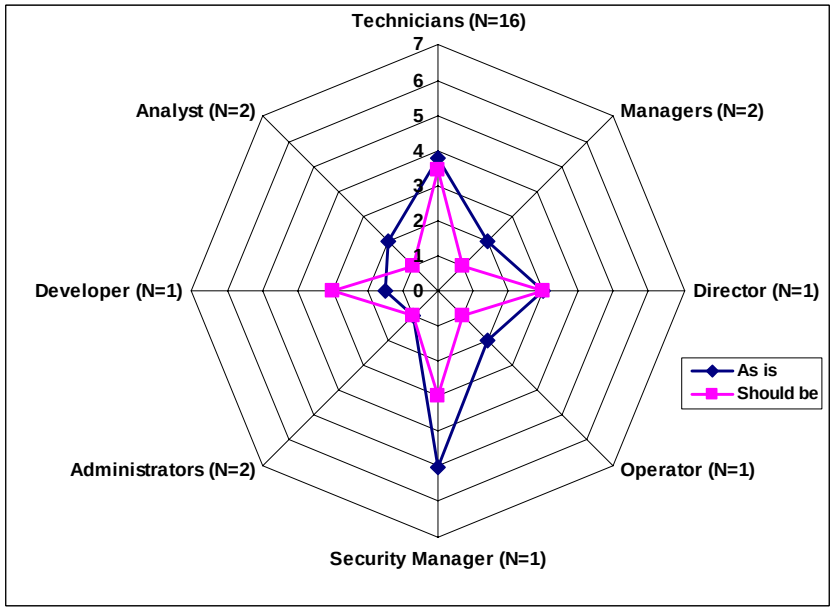


Figure 3-2 Power distance scores for each role

Uncertainty avoidance

Uncertainty avoidance is the extent to which a society, organization, or group of people relies on procedures, social norms, rules and standards to minimise unpredictability of future events. Statements that are used to measure how individuals in a society perceive how their society minimise unpredictability of future events are: “Most people lead highly structured lives with few unexpected events” and “Most people should lead highly structured lives with few unexpected events”.

Figure 3-3 clearly shows the difference between “as is” and “should be” value gap. This implies that uncertainty avoidance is associated with IS insecurities and therefore qualifies to be used in the analysis of IS security.

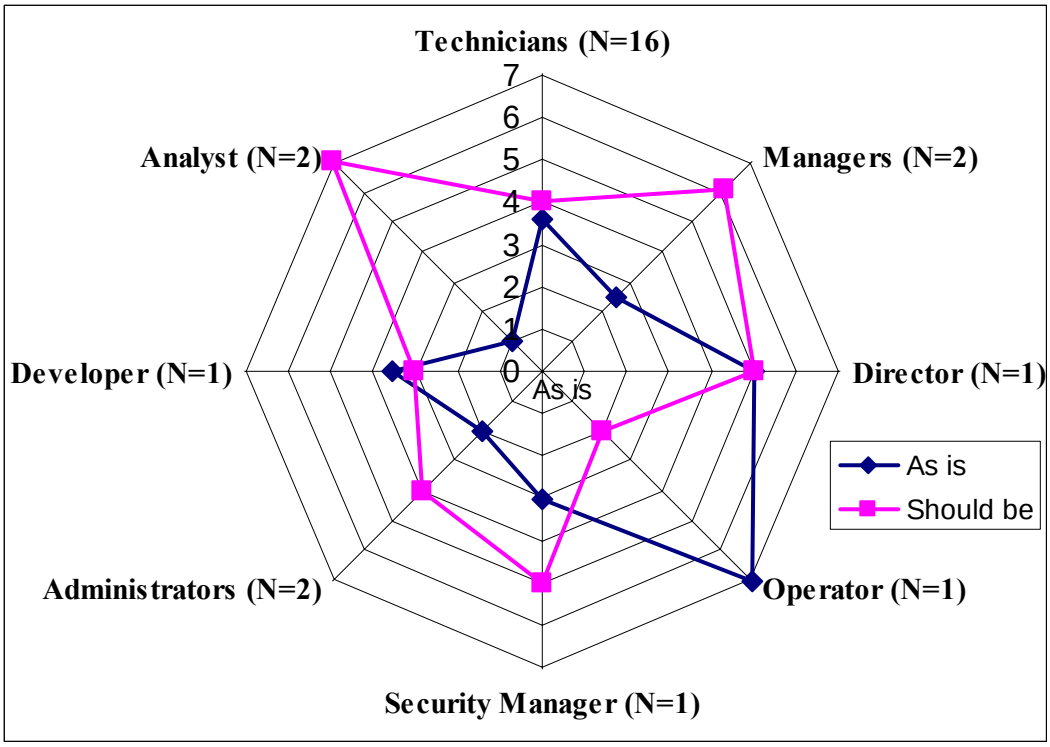


Figure 3-3 Uncertainty avoidance scores for each role

Humane orientation

Humane orientation is the degree to which a society or team encourages and rewards individuals for being fair, altruistic, generous, caring and kind to others. This dimension was measured using the following statements: “People are generally very tolerant of mistakes” and “People should be generally very tolerant of mistakes”. Figure 3.4 show that respondents perceive that the society tolerate mistakes. In addition, they perceived that tolerating mistakes is not right. However the gap is not obvious. This means this dimension is not considered in the analysis of IS security.

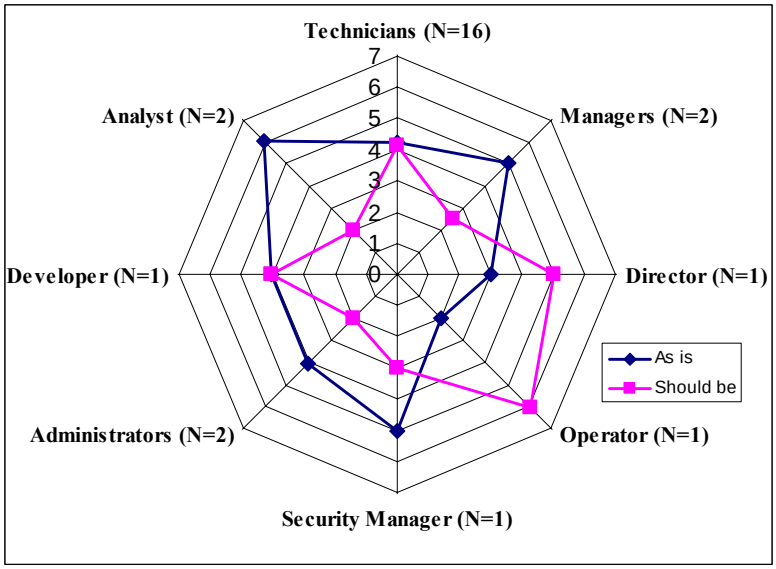


Figure 3-4 Humane orientation scores for each role

Institutional collectivism

Institutional collectivism is the degree to which individuals are integrated into groups within the society. Statements designated to measure this dimension "as is" are: "Leaders encourage group loyalty even if individual goals suffer" and "Leaders should encourage group loyalty even if individual goals suffer". Figure 3-5 depicts that the society tends to be integrated in groups and this dimension is desired. One can see that, from Figure 3-5, there is a difference between "as is" and "should be" graphs. However the gap is not obvious.

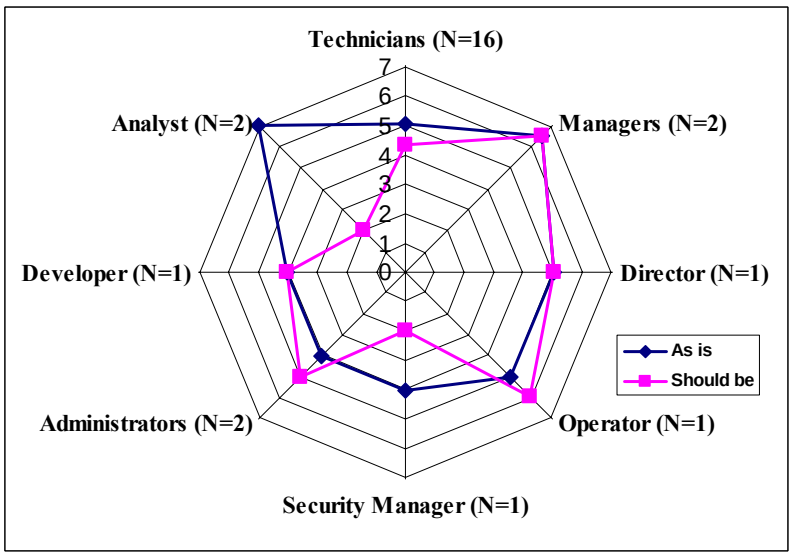


Figure 3-5 Institutional collectivism scores for each role

In-group collectivism

In-group collectivism is the degree to which individuals have strong ties to their small immediate groups. This dimension is measured using the following statements: "In this society, children live with parents until they get married" and "In this society, children should live with parents until they get married". Figure 3-6 show that people tend to agree that in this society there is a strong tie to the immediate small groups like family and

teams at work. The “as is” and “should be” graphs show that in-group collectivism values are distributed such that the gap is not obvious.

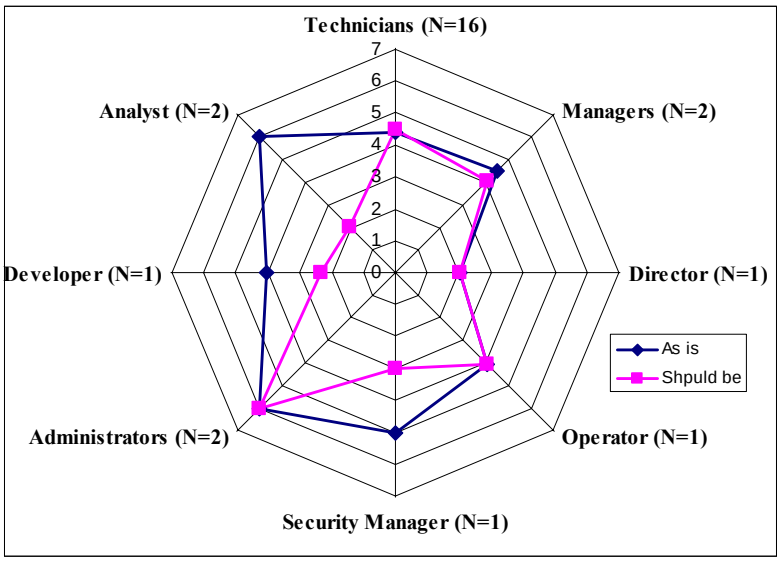


Figure 3-6 In-group collectivism scores for each role

Gender egalitarianism

Gender egalitarianism is the degree to which a society minimizes gender role inequality. Statements that are formulated to measure this dimension are: “Boys are encouraged more than girls to attain a higher education” and “Boys should be encouraged more than girls to attain a higher education”. Figure 3-7 shows that respondents strongly agree that there is a gender imbalance in society and they suggest that this behaviour is not desired. Since the gap for gender egalitarianism is very obvious, this implies that further discussion as to how egalitarianism affects IS security is required.

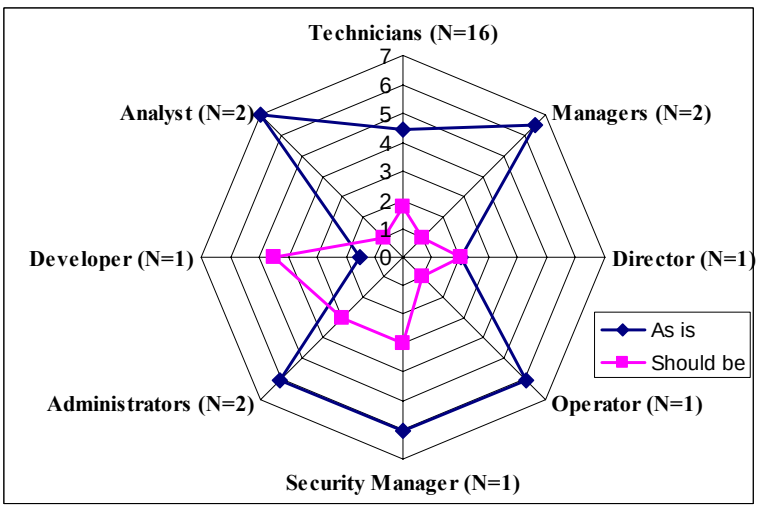


Figure 3-7 Gender egalitarianism scores for each role

Future orientation

Future orientation is the extent to which a collective encourages and rewards future-oriented behaviours such as delaying gratification for the benefit of being able to invest for the future. This dimension is measured using the following statements: “More people live

for the present than for the future" and "More people should live for the present than for the future".

Figure 3.8 show an obvious gap. This implies that it this dimension is relevant for IS security.

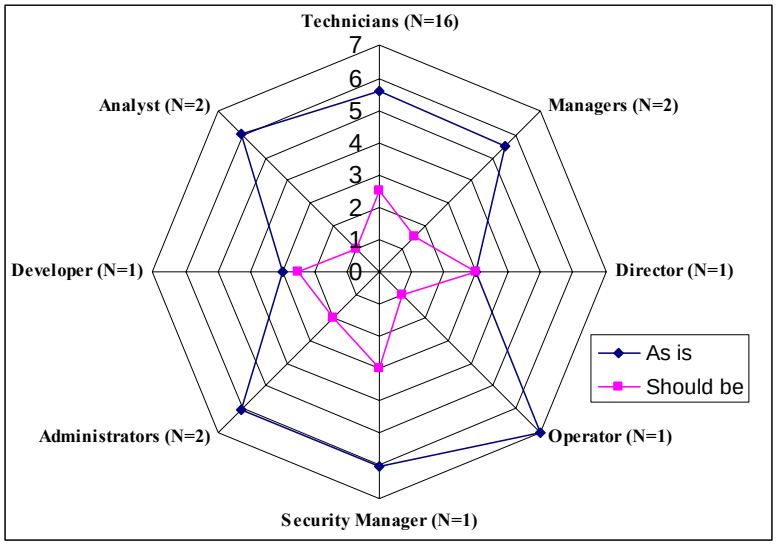


Figure 3-8 Future orientation scores for each role

Performance orientation

Performance orientation is the degree to which a collective encourages and rewards group members for performance improvement and excellence. This is measured using the following statements: "Students are encouraged to strive for continuously improved performance" and "Students should be encouraged to strive for continuously improved performance". Although Figure 3.9 depicts that performance orientation trait prevails in society, there is no obvious gap between the perception of current "as is" values and the "should be" values. Therefore, this dimension is not included in the IS security cultural inventory.

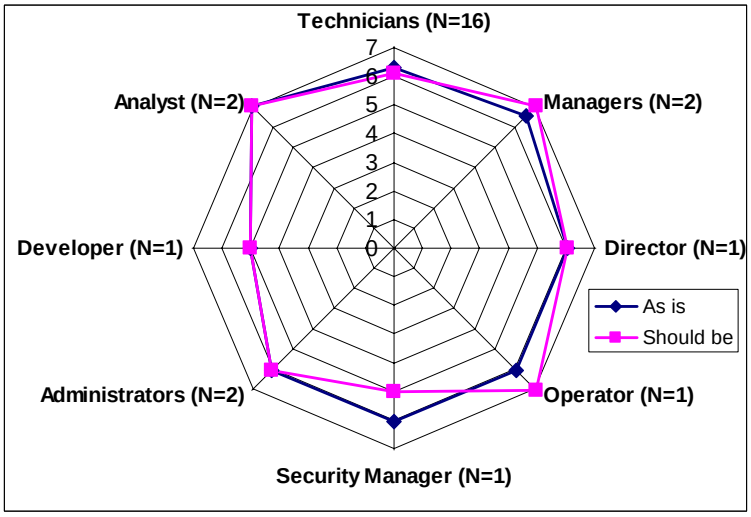


Figure 3-9 Performance orientation scores for each role

3.2.3 Composite visualisation

Figure 3-10 we have a composite visualisation of the all the nine dimensions. Out of the nine dimensions five dimensions are selected based on our heuristic visualisation method. These dimensions are: uncertainty avoidance, Future orientation, Assertive orientation, Power distance and Gender egalitarianism. The choice of these dimensions is based on the fact that there is a gap between "as is" and "should be" values.

In the next section we take these selected dimensions and develop security propositions.

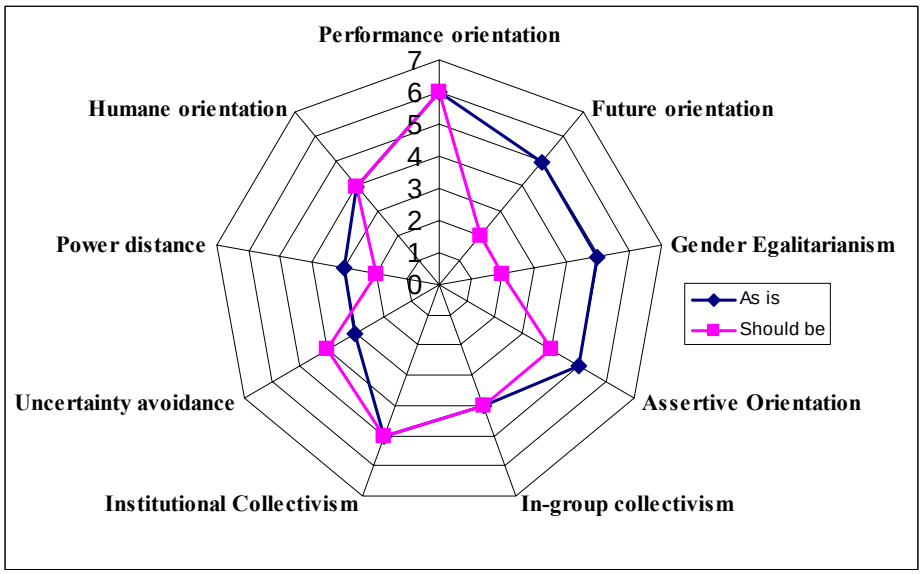


Figure 3-10 Composite visualisation scores of 9 national culture dimensions

3.2.4 Culture Propositions for IS security

The central proposition regarding the relationship between culture and IS security is that culture influence employee's behavior. As a result the way employees carry out their security related work in the organisation depends on their culture. These propositions are formulated as five statements that are based on five national cultural dimensions identified using the visual heuristics method discussed in the previous section. These five propositions are:

1. Low Uncertainty avoidance results in poor in-depth security and lack of holistic approaches to security
2. Low degree of Future orientation results into inefficient contingency planning
3. High Power distance results in poor communication on security issues between management and users i.e. technicians, administrators and computer operators
4. The degree of Assertive orientation influences the way managers manage security related issues
5. High degree of Gender egalitarianism results in system insecurities that are related to economic egalitarianism.

3.3 Implications of National Culture on Information Systems security

In this section we will analyse implications of the findings from the national culture dimension survey. This will be conducted through commenting each of the five propositions from 3.2.4. Some of these comments have emanated from the second survey at TANESCO, but for reasons of clarity, we describe those procedures and results in section 3.4.2:

One of the central issues for security requirements engineering is to understand the environment in which the system will operate. The environment security requirements must define among other requirements assumptions about user behaviour administrators, developers, evaluators and users. In the CC (CCIMB3, 2005, p. 140), the statement of Target Of Evaluation (TOE) security environment has to include the assumptions regarding user behaviour.

> *"**AGD_ADM.1.4C:** The administrator guidance shall describe all assumptions regarding user behaviour that are relevant to secure operation of the TOE".*
>
> *"**AGD_USR.1.4C:** The user guidance shall clearly present all user responsibilities necessary for secure operation of the TOE, including those related to assumptions regarding user behaviour found in the statement of TOE security environment".*

Questioning assumptions related to human behaviour and motivation may be useful to ensure that correct measures are put in place. Correct assumptions about user behaviour may result into better mechanisms to prevent or deter attacks.

We argue that it is difficult to make assumptions about user's behaviour unless we know the culture that influences their behaviour, thus the findings from the national culture surveys can indicate how culture affects handling and managing of information systems security in organisations.

Proposition 1 - Uncertainty avoidance

The proposition that is made about Uncertainty avoidance state that low uncertainty avoidance results into poor in-depth and lack of holistic approaches to security. Behaviour that is related to how people avoid uncertainty could be:

- Lack of security in-depth due
- Lack of attention to details
- Poor risk assessment
- Poor assumptions about motivation, opportunity and methods
- Lack of Information classification
- Poor use of metrics

Proposition 2 - Future orientation

Low degree of Future orientation results into ineffective contingency planning, see. Figure 3-8. Future orientation is necessary for organisations to prevent, deter, detect and recover

from attacks including all proactive security measures. Detection and recovery can be addressed with certainty only if people live for the future. Future oriented behaviour is necessary if contingency plans, training, security budget, detection of new attackers and prediction of motives and re-evaluation of risks are to be considered.

Proposition 3 - Power distance implications

High Power distance can result in poor communication on security issues between management and employees such as technicians, administrators and computer operators. The result of the first survey show that power distance in the society is low. Employees meet with management regularly in meetings. However, the degree to which employees are ready to report unethical conduct by colleagues was low (from second survey, see Appendix C, ethics survey questions). This implies that low power distance does not necessarily mean that people may report illegal or unethical behaviour to the proper authority.

Proposition 4 - Assertive orientation

Behaviour that relate to high degree of Assertive orientation have adverse effect on how security is managed in organisations. Issues such as dealing with unethical conduct or suspicious actions within the organisation need leadership which is strong. If the assertive orientation is low an assumption such as "administrators, operators, officers, auditors, and other users should notify proper authorities" about any security issue that influence their systems in order to minimize the potential for loss or compromise of data may not be correct. Some organisations create a whistle blowing policy where there exit processes and procedures for employees to report suspicious or unethical actions or statements made by other employees. This may not work in cultures where the degree of assertiveness is low. In addition, legal implications as a deterrence force for illegal conduct may as well be ineffective in cultures where assertive orientation is low.

Proposition 5 - Gender egalitarianism

Gender equality and social equality are to some extent related. Social values related to egalitarianism such as economic independence and equal opportunities have serious impact on IS security. In a social system where there is a big gap in economic conditions, it is more likely that the rate of security incidents will be high. For example in many cases fraud is committed motivated by financial gain. Individuals likely to commit dishonest acts include those who are denied the opportunity for economic prosperity.

3.4 IS Security Culture

In this section, we examine some issues from the literature that pertain to information systems security culture. In particular, we examine how culture develops and the rationales of importance to information systems security assurance.

3.4.1 Creating and learning organisational culture

The stakeholders of organisations traditionally play key roles in establishing organisational culture. For instance, founders of organisations have the vision and mission of what the organisation should look like. Therefore, subsequent values that will be developed to fulfil the vision will be passed on to employees through training and experience.

Top management, in organisations, traditionally comprise of people who are loyal to the vision and who may have consistent behaviour. Members of top management set rules and procedures for hiring new employees. Rules will involve how to match employee behaviour to the company culture and how the socialization process will take place. Figure 3-11 depicts how cultural values are passed from the founder to the employees. The socialization process involves familiarization of company operations and internal controls.

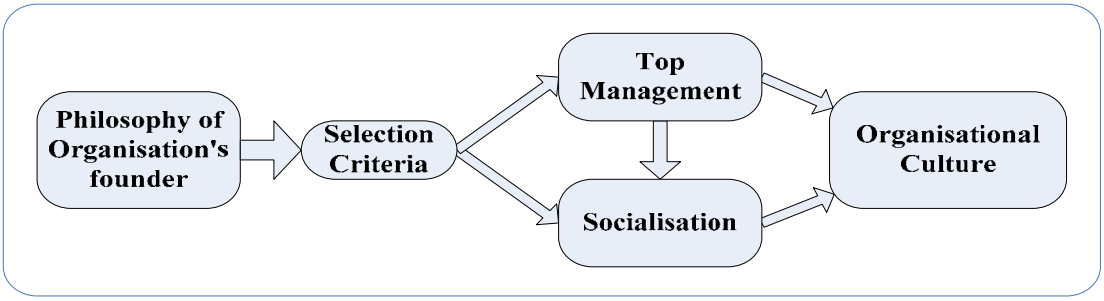


Figure 3-11 How organisation's culture form (Robbins, 2005, p.237)

Security culture can be learnt in a number of ways, the most potent, according to Robbins (2005) are: story telling, rituals practices, material symbols and language. Stories in organisations are developed to provide a narrative explanation of values such as customer care, productivity and risk taking. Similarly, stories could be developed that provide a narrative form of how security is central in the success of an organisation. Rituals are repetitive actions that reinforce values. They are traditionally, in organisations, performed at celebrations or official events by singing or saying words. The language is also important in the process of security culture. Understanding common security terminology can be indications of employees familiarity with IS security.
The benefit of culture to an organisation is that it enhances employee's commitment to the organisations objectives and increases the consistence of employee's behaviour. Results of measurements taken using dimensions from Table 3-6 may differ from organisation to organisation due to objectives, products, services, markets and founders. However, all organisations exhibit similar characteristics in having either a weak or a strong culture. Strong cultures lead to consistent behaviour. Consistent behaviour works well in stable environments but may also be an obstacle to change (Robbins, 2005).

Table 3-6 Organisational culture dimensions and related statements (Robbins, 2005, p. 230)

Dimension	Examples of related ststements
Innovation and risk taking	This organisation rewards (should reward) people who take risks and who are innovative
Attention to details	This organisation encourage (should encourage) employees to exhibit analytical and attention to details abilities
Outcome orientation	In this organisation people are evaluated (should be evaluated) based on the output and not based on the means and processes used to achieve the outcome
People orientation	In the organisation the management involve (should involve) employees in decisions which affect them
Team orientation	In this organisation work is (should be) organised in teams rather than individuals
Aggressiveness	In this organisation employees are (should be) encouraged to strive for continuously improved performance.
Stability	In this organisation activities are (should be) organised with emphasis on the status quo in contrast to growth

3.4.2 Towards Security Culture Dimensions

Developing a security culture within an organisation requires an additional set of dimensions in contrast to the national and organisational cultures. Security culture should include surface and core cultural values. Core cultural values are necessary if we want a people centred security. Figure 3-12 depicts the core and surface cultural values that we must consider when developing or evaluating a security culture within an organisation. The surface values a subject to change as technological and environmental changes take place. Core cultural values represent social subsystem.

Apart from being influenced by Robbins (2005) two specific studies from the literature inspired our choice of security dimensions: the security culture assessment based on ISO 17799 (Martin & Eloff, 2001) and findings related to the need of specifying as well surface as subsurface cultural dimensions in a United Kingdom (UK) health care project (Gaunt, 2000).

Each of these dimensions characteristics exist in a continuum from low to high. Appraising the organisation on these characteristics, then, gives a composite picture of the organisation's security culture.

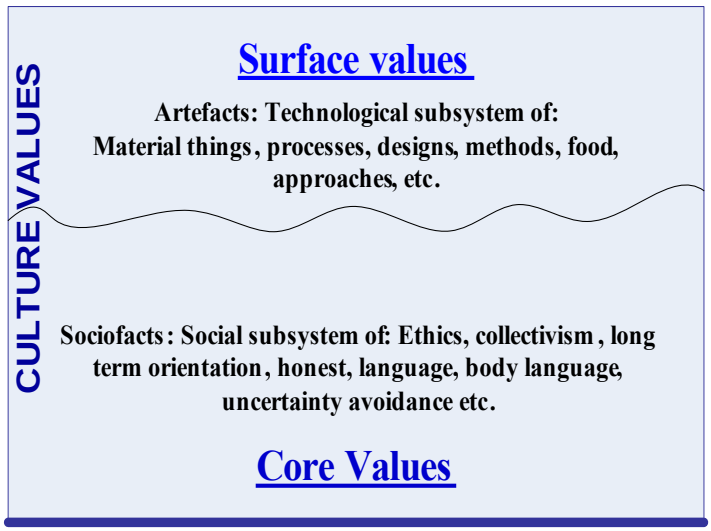


Figure 3-12 Culture core and surface values

The following records our interpretation of each core/surface dimension to be considered as security culture dimensions for the second and third survey at TANESCO.

Awareness orientation: This dimension aims at addressing security awareness related issues such as awareness of security artefacts (technological subsystem-methods, process, products training programs, benchmarks, standards, best practices), menti-facts (ideological subsystem-ideas, knowledge, belief), and socio-facts (Sociological subsystem-economy, power distance, humane, communication to individuals and group, laws and directives), and security readiness.

Management commitment: Clear communication channel between the management and security personnel is an indication that a stronger security culture exist within the

organisation. In addition, the management has to be committed to budget, assignment of roles to security personnel, developing contingency and continuity plans, drafting policies and making sure they are available and reviewed on regular bases.

Ethical orientation: Ethics address issues that are not a crime under civil, criminal and other types of formal laws. Traditionally ethics in security has to do with reporting about security incidences, faulty products and systems, whistle blowing, reporting/disclosing of incompetence, adherence to policies, guidelines and code of conducts.

Risks orientation: It is interesting to measure what people perceive to be a risk than measuring the existence of actual risks. Also as part of the culture evaluation, we can ask questions which will reveal the existence of risk analysis programs, critical assets identification, risk analysis methods, risk prioritisations, risk mappings, and adversary potential analysis.

People orientation: This dimension will be used to measure if the management fosters people centred security through motivation programs, back ground screening, drafting code of ethics, policy guidance and procedures, Role based access control, Socialisation process and security team forming procedures.

Privacy orientation: Can be measured checking whether the organisation maintains a policy on personal data protection, and weather the company fosters territorial confidentiality and integrity, communication linkability, traceability, anonymity and pseudonymity.

Information classification: This can be measured by asking questions such as in this company Information is classification into categories such as top secret, secret, confidential, classified, declassified and public.

Assurance orientation: This dimension aims at measuring availability of evaluation and verification of programs of mechanisms, processes, change management, re-evaluations, auditing, considerations of security in system's life cycle, usability of IS systems and CIA security services including other services such as authentication, access control, non-repudiation, single-sign on etc.

Future orientation: How people value planning for the future in contrast to the present.

Uncertainty avoidance: Preference to structured in contrast to unstructured life. This measure and the future orientation were used to evaluate the national culture.
Attention to details: The extent to which employees are encouraged to exhibit analytical abilities and pay attention to details.

Table 3-7 Security culture dimensions

Security Culture Dimension	Description of Dimension
Management commitment	Clear communication channel between the management and security personnel about security issues, budget, roles assignment, contingents, continuity plans, policies availability and review schedule
Awareness orientation	For security artefacts(technological subsystem: methods, process, products , training programs, benchmarks, standards, best practices), mentifacts (ideological subsystem-ideas, knowledge, belief), and sociofacts(Sociological subsystem-economy, power, humane, communication-individuals, group, including laws and directives), Security redness, security education
Security Culture Dimension	**Description of Dimension**
People orientation	Management towards people centred security, Motivation, back ground screening, code of ethics, policy guidance and procedures, Role based access type of authentication, Socialisation process, team forming procedures
Risks orientation	What people perceive to be a risk, risk analysis program availability, What are critical assets, risk analysis methods, prioritisation and risk mapping, adversary potential analysis
Privacy orientation	Privacy of personal data, Territorial and domestic, Body and communication linkability, traceability, anonymity and pseudonymity
Ethical orientation	Reporting about security incidences, insecure systems, whistle blowing, reporting of incompetence, adherence to policies, guidelines and code of conducts
Information classification	Information classification such as top secret, secret, confidential, classified, and declassified.
Assurance orientation	Evaluation and verification of artefacts: mechanisms, processes, and non-technical aspects, Change management, re-evaluation, auditing, Security in system's life cycle, Easy of use of artefacts for CIA and other security services such as Authentication, access control, non repudiation (Usability)
Future orientation	How people value future plans in contrast to the present in relation to security issues
Uncertainty avoidance	Preference to structured life in contrast to unstructured life in relation to security issues
Attention to details	The extend to which employees are encourage to exhibit analytical and attention to details abilities

3.5 Security Culture Dimensions Surveys

National culture dimensions that were used to evaluate culture in section 3.3 are now integrated with the organisational culture resulting in 11 security dimension. Ethics orientation and management commitment are part of the socio-technical model in which system security takes into account managerial and ethical issues. The assurance orientation dimension is added in order to address questions that are related to the technical assurance as shown in the socio-technical model. Adoption of security standards is added so that questions that points out if an organisation adopts or uses security standards (security assurance and other security standards)

The continuum used in the evaluation is from Strongly-disagree to Strongly-agree as indicated in Table 3-8 and graph's y-axes are labelled using the ratings1 through 5. The numbers represent how strongly the respondent perceived about a particular security dimension.

Table 3-8 Perceived presence of culture continuum

1 Strongly Disagree	2 Disagree	3 Unsure	4 Agree	5 Strongly Agree

Table 3-9 represents average scores of respondents from all the eight roles. Their average is approximately three which indicates that respondents are unsure about the existence of a security culture. The average score represents the average perception of all respondents to whether they strongly agree, agree, unsure, disagree or strongly disagree that a security culture exists in the organisation.

Table 3-9 Average score for each role

S/N	Dimensions	AVERAGE SCORES FOR EACH ROLE (Numbers represent a continuum from strongly disagree to strongly agree)								Average
		Technicians	Sys. analyst	Directors	Operators	Sec. Managers	Administrators	Developers	Manager	
1	Management commitment	3	3	4	4	4	4	4	3	**4**
2	Awareness orientation	4	4	4	5	4	4	4	4	**4**
3	People orientation	3	3	4	4	4	4	4	4	**4**
4	Risk orientation	3	3	4	4	4	4	4	4	**4**
5	Information classification	3	3	3	4	3	3	2	4	**3**
6	Assurance orientation	3	2	3	4	4	3	3	4	**3**
7	Future orientation	2	2	2	3	3	2	2	2	**2**
8	Uncertainty avoidance	2	3	2	1	2	3	1	1	**2**
9	ST Adoption and compliance	3	2	3	4	4	4	4	4	**4**
10	Attention to details	3	3	2	2	2	2	2	3	**2**
11	Ethical orientation	2	3	4	4	3	2	4	2	**3**
	Security Culture	**3.0**	**3.1**	**3.3**	**3.7**	**3.5**	**3.3**	**3.4**	**3.5**	**3**

3.5.1 Management commitment

The management commitment to IS security is evaluated using the following statements:

- In this organisation when recruiting new employees IS security related questions are included in the interviews questions
- This organisation has a written information security policy
- The information security policy reflects the organisation's business objectives
- This organisation has annual budget for information security
- The organisation has an information security contingent plan

- In this organisation, ICT procedures are implemented according to the information systems security policy
- I can easily obtain a copy of the information security policy
- The information security policy is updated regularly as needed
- The organisation ensures that I adhere to the information security policy
- In this organisation, the management perceives information security as important

Management commitment plays central role in ensuring policies are drafted, available to users, enforced and periodically reviewed, security needs are included in the annual budget and IS security is part of the employee selection process. The average score for all respondents is four and it implies that they agree that the management is committed to security. Results in Table 3-7 show that 40.77 % of all the respondents agree that the management is committed, 32.04% are unsure, 11% strongly agree, 15.38% disagree and 0.81% strongly disagree.

Table 3-10 Average for respondents and for each rating and percentage

Dimensions	AVERAGE OF RESPONDENTS FOR EACH METRIC AND PERCENTAGE									
	Strongly Disagree	Strongly Disagree Percentage	Disagree	Disagree percentage	Unsure	Unsure percentage	Agree	Agree percentage	Strongly Agree	Strongly Agree Percentage
Management commitment	0.21	0.81	4.00	15.38	8.33	32.04	10.60	40.77	2.86	11.00
Awareness orientation	0.13	0.50	1.12	4.30	1.75	6.73	14.00	53.85	9.00	34.62
People orientation	0.27	1.04	2.00	7.69	7.43	28.58	12.30	47.31	4.00	15.38
Risk orientation	1.00	3.85	4.00	15.38	9.00	34.62	8.00	30.77	4.00	15.38
Information classification	6.00	23.10	6.00	23.10	3.00	11.50	9.00	34.60	2.00	7.70
Assurance orientation	1.00	3.85	4.50	17.30	14.00	53.85	6.00	23.08	0.50	1.92
Future orientation	7.50	28.85	7.00	26.923	3.00	11.538	6.50	25	2.00	7.6923
Uncertainty avoidance	10.00	38.46	5.00	19.23	5.00	19.23	3.00	11.54	3.00	11.54
ST Adoption and compliance	0.00	0.00	5.00	19.20	9.00	34.60	10.50	40.40	1.50	5.80
Attention to details	0.00	0.00	9.00	34.60	8.00	30.80	7.00	26.90	2.00	7.70
Ethical orientation	2.00	7.69	8.00	30.77	8.00	30.77	5.00	19.23	3.00	11.54
Total	28.11	108.15	55.62	213.87	76.51	294.26	91.90	353.45	33.86	130.27
Security Culture	2.56	9.83	5.06	19.44	6.96	26.75	8.35	32.13	3.08	11.84

3.5.2 Awareness orientation

Awareness orientation is measured by aggregating answers for the following statements:

- I received a formal training in information security
- I think Internet is a secure network
- It is important to determine the organisation's information security needs
- Information security should be regarded as a functional (business) issue
- I know what the term information security implies
- I know what the term information security assurance implies
- I think it is important to implement information security in the organisation
- I am aware of information security responsibilities that are related to my job role

There are other statements that may be used to measure security awareness. The aim of using this dimension is to determine the basic level of respondent's awareness for IS security. It is expected that in environments where security culture if fostered, people will be aware of basic security controls. 11% of all respondents strongly agree that people in the organisation are aware of basic security, 53.85% agree, 6.73% are unsure, 4.3% disagree and 0.5% strongly disagree. The average score for all respondents is 4 and it implies that the overall respondents agree that security awareness is there.

3.5.3 People orientation

People orientation was measured using the following statements:

- In this organisation new employees are briefed about security issues in the socialisation process
- My manager involves me in decisions that affect me
- In this organisation background checking is always part of the selection process for new employees
- Management regards the privacy of information about employees as important
- Management enforce information access security policy for all job levels

The overall score for this dimension, as indicated in Table 3-9 is 4 which imply that respondents agree that security is people centered. Out of 26 respondents, 15.38% strongly agree that security in the organisation is people cantered, 47.31% agree, 28.58% are unsure, 7.69 disagree and 1.04% strongly disagree.

3.5.4 Risk orientation: Orientation to risk analysis

Risk orientation was measured using questions that solicit about present of risk assessment programs and whether there exist people with risk assessment roles in the organisation. Statements include the following:

- In this organisation risk analysis of information systems is performed regularly
- In this organisation assets are classified based on how critical they are
- In this organisation there is ICT risk assessment team

15.38% percent of all the respondents strongly agree to the risk orientation dimension, 30.77% agree, 34.62% are unsure, 15.38% disagree and 3.85% strongly disagree. The overall score for the dimension 4 and it implies that the company is aware of risks.

3.5.5 Information classification

The statement “In this organisation information is classified e.g. into confidential, public, classified, declassified” aimed at measuring if information classification exist in the organisation. If information is not classified protecting it from attacks (which are people cantered) will be harder. Information classification should be encouraged and we believe its presence is an indication of a security culture. In the survey, as indicated in Table 2.4, 7.7% strongly agree that information in the company is classified, 34.6% agree, 11.50%, are unsure, 23.10% disagree and 23.10% strongly disagree. The average score is 3 and it implies that respondents are unsure if information is classified.

3.5.6 Assurance orientation

Information assurance is a one of indicators of an existing security culture. It is expected that any company must be concerned about mechanisms and security functions to work according to requirements in the system’s life cycle. Questions that we developed to solicit assurance orientation are:

- In this organisation, information security is measured by using defined security metrics
- In this organisation, there is a formal procedure shows how I should report information security incidents.

- The organisation use certified information systems products

Out of 26 respondents, as shown in Table 2-5, 1.92% percent strongly agreed that information assurance is exercised in the organisation, 23.08% agreed, 53.85% are unsure, 17.30% disagree and 3.85% strongly disagree. The average score is 3 indicating that respondents are unsure about assurance practices.

3.5.7 Future orientation

Future orientation is one of dimensions which may indicate how the organisation organise issues that require looking into the future such as contingent plans, backups, business continuity plans and assurance process. The statements we use to solicit this are:

- Investing in information security should be seen as a future investment and
- More people in this society live for the present than for the future

The average score for this dimension is 2, indicating that respondents disagree on future orientation. Out of 26 respondents, 7.69% strongly agree on the future orientation, 25% agree, 11.54% are unsure, 26.92% disagree and 28.85% strongly disagree on future orientation. This dimension reflects that the organisation may exhibit poor contingence plans, poor disaster recovery plans, etc.

3.5.8 Uncertainty avoidance

Uncertainty avoidance was measured using the following statements:

- In this organisation security incidence and emergence handling procedures available
- This organisation has a remote backup site
- In this organisation information systems security personnel follow more structured procedures when working with information systems

Respondents overall score is 2 which indicate that respondents disagree that people live structured lives. 11.54% strongly agree that people live structured lives, 11.54% agree, 19.23% are unsure, 19.23% disagree and 38.46% strongly disagree.

3.5.9 Standards adoption and compliance procedure

Internal procedures and standards are good indicators of security culture and we test this using the following statements:

- The organisation’s information security measures comply with international standards
- In this organisation we have internal standards for ICT
- Information systems procedures are audited regularly

Overall score for standards adoption and procedures is 4 which is an indication that people perceive that the company has some level of standardised procedures. 5.8% of respondents strongly agree that standards are adopted and used, 40.40% agree, 34.60% are unsure, 19.20% disagree and 0% strongly disagree.

3.5.10 Attention to details

This organisation encourages employees to exhibit analytical and attention to details abilities statement is used to solicit hoe respondents perceived this dimension. In the overall respondents scored 2 and is an indication that respondents disagree about employees being encouraged to cultivate a culture of exhibiting analytical abilities. 7.7% of all respondents strongly agreed that people are encouraged to be analytical and attend to details, 26.9% agreed, 30.80% are unsure, 34.60% disagree.

3.5.11 Ethical orientation

Statements used to measure ethical orientation can be found in Appendix C. Statements are designed to measure how respondents perceived about issues such as professional conduct and reporting about security misconducts and fraud. 11.54% of all the respondents strongly agree that there is some level of ethical conduct in the organisation, 19.23% agree, 30.77% are unsure, 30.77% disagree and 2.21% strongly disagree. The average overall score is 3 and it implies that respondents are unsure about ethical conduct in the organisation.

3.6 Chapter Summary

This chapter has examined how national culture relates the management of IS security and how to develop a security culture within organisations. The national culture defines how people think about the future, plans, uncertainties and more importantly defines our personalities. As a result, national culture shapes organisational culture in that it affects strategic organisational contingencies and leader's attributes and behaviour

Developing security culture in organisations can be successful if it is made part of the traditional process of developing organisational culture where the management may be committed in making sure new employees adopt the organisation's values and philosophy. In this study, our interest is to analyse and suggest the best ways of analysing security culture. Figure 6-14 shows scores for each job role.

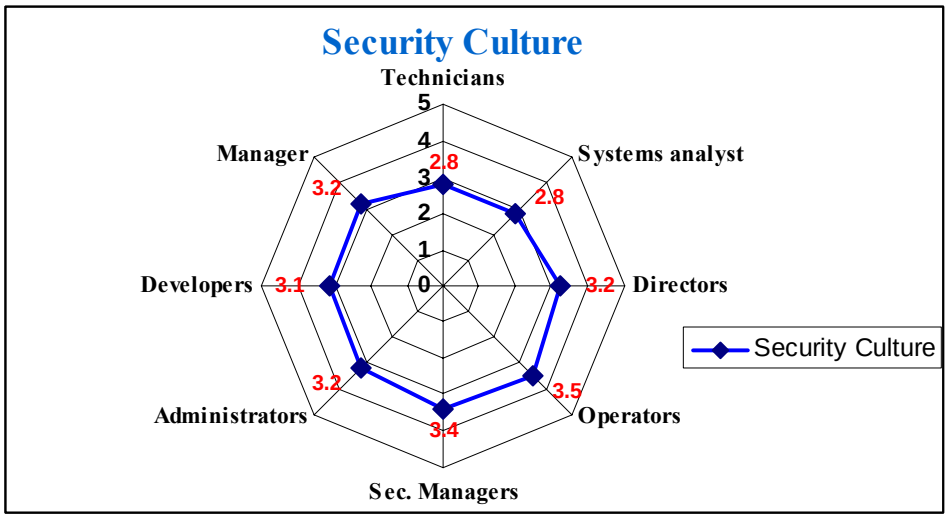


Figure 3-13 Security culture average overall score for each job role

Operators may perceive security culture to be present in the organisation more than other job roles because they are subject to frequent audits and follow strictly working procedures such as banking cash at every end of sales, monthly password expiry, strictly backup or disk transfer procedure etc. Similarly, the security manager perceives that there

is security culture in the company. Figure 3-14 indicate the percentages of respondents. Less than 43% of respondents agree that there is a security culture and 26.75% are unsure about security culture, 19.44% disagree that there is a security culture and 9.83% strongly disagree.

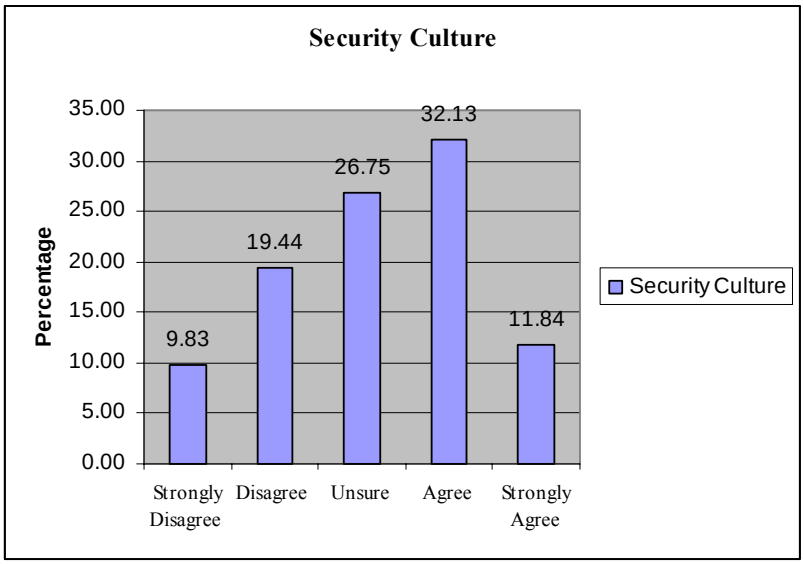


Figure 3-14 Percentage of respondents for each metric

Figure 3-15 shows how respondents from all job roles responded on each the nine dimensions. Out of these, five dimensions are selected to make the propositions. These dimensions are: uncertainty avoidance, future orientation, assertive orientation, power distance and gender egalitarianism. The choice of these dimensions is based on the fact that there is a significant gap between the "as is" and "should be" ratings.

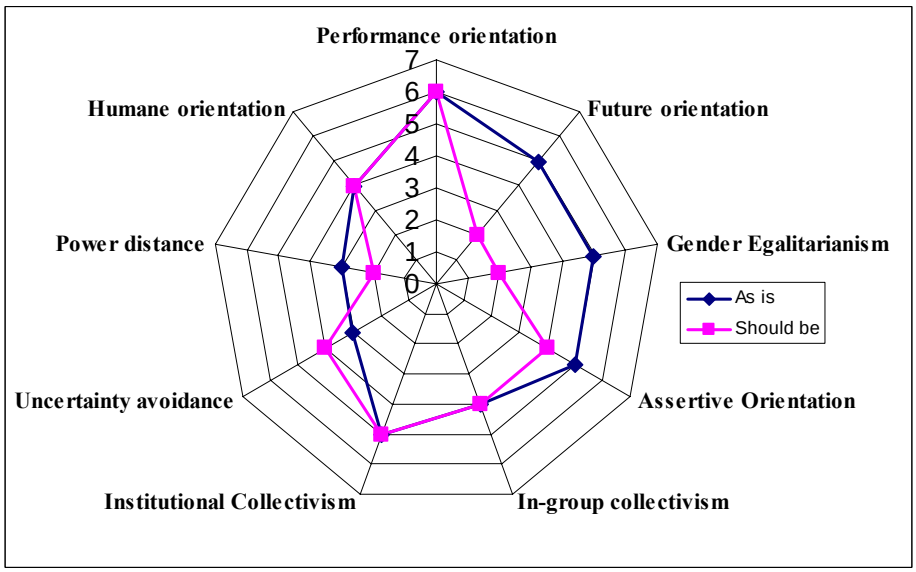


Figure 3-15 Composite scores of 9 national culture dimensions

Chapter 4

4 Security Usability

In this chapter, we present usability aspects of information systems security assurance. The focus is on analysing usability problems that are related to the interaction with the system interface. Usability problems occurring in the Man-Machine-Interface (MMI) may have the potential of rendering a system with well-designed and implemented security assurance policies insecure. We are attempting to address one of the environmental problems that may result in users making mistakes due to poor systems interface design of the security functions.

The purpose is to examine possible causes to insecurities that may result from usability problems occurring in the MMI, as well as usability problems that may occur in the use of the security assurance methodology that security assurance evaluators employ to perform usability analysis in a cost-effective way.

This chapter consist of four sections. The first section describes the usability in general. The second part examines the password problems. The third part deals with usability analysis of the LUKU prepayment system. The fourth part deals with usability conclusions.

Traditionally, systems security assurance focuses on technical issues only. Issues related to usability still need more research (Anderson, 2001; Zurko *et al.,* 1996).

Usability assurance addresses operational security problems that may be related to human error. Users may make mistakes that may result in the systems security being compromised. Usability problems have to be analysed carefully in order to determine if users have enough knowledge to use the system in a secure way or if the reason in due to poor interface design of the (security) interface itself.

4.1 Security Usability Problems

There is a general agreement that usability is not the primary motivation for security assurance (Cranor *et al.,* 2004; Zurko *et al.,* 1996; Norman, 1983; Clear, 2002; Davis *et al.,* 2004; Anderson, 1994). Consequently, human error is the principal cause of accidental system failures. Errors could be caused by:

- Poorly designed interface
- Poor environment
- The human factor

Poorly designed interfaces could result in poor information presentation; for example, lack of unit of measurements as well as lack in distinctiveness of different parameters and commands. The operational environment itself, if not adjusted correctly, is usually the primary source of distraction; factors such as lighting, vibrations, noise and temperature can cause an experienced user to make errors. Human factors include; skill level, mental and physical fatigue, overload, stress, ease of use, clarity of error messages, attitude as well as ease of learning how to use the system.

Security and usability both require a holistic approach that system developers, architects share (Anderson, 2001). Rarely is security and usability successfully added on at the end of development process. Both must be designed and built in from the beginning. Recently, there has been a growing realisation that usability problems are hindering security effort. Some researchers (Cranor *et al.*, 2004; Norman, 1983; Viega, 2002) have dedicated themselves to work at the interface level of security and usability. Some of their important suggestions regarding the design of usable security systems are:

- Design systems that perform security or privacy functions without user intervention
- Design systems that let users intuitively use security and privacy correctly
- Teach users what they need to know how to use effectively security and privacy tools
- Keep it simple

As appealing as these suggestions may be, system users may prefer the first approach, but a system that automatically works with security denies the user all the power to control security. Consequently, there will be security overhead. Since information systems do not work in isolation from other systems such as the legal system, the concept of designing simple and usable system security features is the way forward. Also, education in security awareness is necessary for secure usage of information systems.

4.2 The Password Problem

In daily life, as the general context of use, we all suffer from the overload of too many different passwords, PINs (Personal Identity Number) codes and login names (Clear, 2002). Table 4-1 shows that virtually all users face the most common usability security problem when they try to use the security feature of their cell phone, access their E-mail, use their PIN and user name to access digital libraries, use their magnetic card for accessing buildings, use their identity and PIN to access their bank account, etc. Users are asked to change their passwords often and to use strong passwords (many characters) that are hard to remember and guess. To simply cope with the burden of remembering all different passwords, users simply attempt to use the same PIN or password whereas others jot them down on a piece of paper.

Alternatives to the traditional password and PIN code, such as biometrics and image recognition are proposed to authenticate users to systems (Davis *et al.,* 2004).

Table 4-1 Context of use

Context of use	Service	Security mechanisms
Banking	Internet banking	Digi-pass
	Online transactions	Token and PIN
	Automatic teller machine	Token and PIN
Mobile phone	Phone code and PIN	Phone code, PIN1 and PIN2
Buildings	Access to rooms	Physical key, Token and PIN
	Access to library	Token and PIN
	Elevators	Token and PIN
Networks	Access to digital libraries	Login name and password
	Access to VPN	Login name and password
	Access to Internet & e-mail	Login name and password
	Access to corporate network	Login name and password
	Access to remote network	Login name and password
Travel	Airport services	User name and PIN

The use of physical attributes such as biometrics for user authentication may prove both effective and efficient. However there are challenges associated with using biometrics since it requires specialised equipment. The challenge is how to combine the equipment with remote access in order to satisfy the need for people to access systems remotely (Adams, 1999).

Apart from password problems users suffer from poorly designed interfaces. Users are very much affected if; screen colours are not carefully selected, error messages are displayed in a cryptic manner, the system interacts in an informal language, several different passwords have to be used to access the system, users have to remember what they have done elsewhere in another part of the system, etc. These problems and related ones are referred to, in this paper as usability problems. To address such problems requires through analysis of the user interface.

4.3 Usability Analysis Methods

Usability analysis is a form of operational systems assurance that aims at ensuring that the system is easy to use. That is, the mental effort required by the user to access and use the service is minimal and the occurrence of accidental errors is prevented. There a several usability analysis methods that can be used depending on the needs, such as usability heuristics (Nielsen, 1992). Heuristics draw attention to usability problems often found in single user applications such as how feedback is provided, clarity of information, user control, confirmation option before performing an important action, error prevention, user memory overhead and clarity of error messages. User memory overhead occurs when users are required to remember information from previous dialogue.

Another useful method to capture usability problems is the questionnaire. Several researchers advocate evaluation through questionnaires that are filled in by the people who use the system (SUMI, 2004; Shneiderman, 1997). In this study questionnaires are used after the usability experiment where the respondents completed the questionnaire that had questions designed to clarify what was observed in the experiment. Then respondents

conducted discussions with the author for clarifications of discovered usability errors and possible solutions to the problems.

In this study, the experiment involved technicians (N= 4) and one system administrators (N=1) of the old electricity pre-payment system (LUKU). The respondents were briefly introduced to the experiment and a system walkthrough of the new system was conducted to make sure every respondent was aware of the system functionality. In-depth explanation of the system was not required because all the respondents had more than three years experience of working with similar systems.

LUKU is a Swahili word given to the Tanzania Electricity supply company (TANESCO) prepayment systems as depicted in Figure 4-1. The experiment involved analysing the usability of the interface of the security module of the system. The need for analysis was part of the information systems assurance project at TANESCO, analysing internal controls and security requirements development and re-use of the security requirements. In addition, the analysis was partly required because TANESCO was in the process of replacing a legacy system with a new system. Usability analysis results were to be used to further improve the new system.

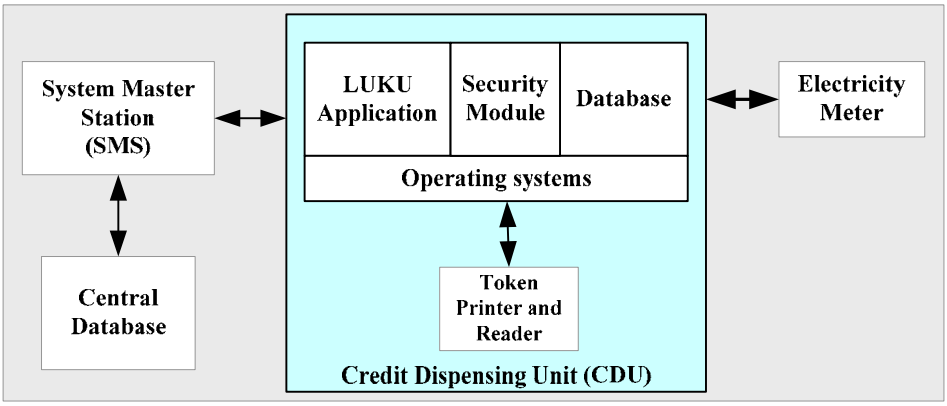


Figure 4-1 LUKU Prepayment system

The respondents were tasked to run the experiment twice. In the experiment, errors were recorded based on the given heuristics. However, respondents were also allowed to record any error they may come across even if not stated in the heuristics.

Some of the security heuristics that were used to conduct the usability analysis are defined below.

- User comfortable with authentication process: Users should be comfortable with the authentication process i.e. systems support for single sign-on, challenge if the password is forgotten and feedback if the password used is not up to standard.
- Clarity of user rights: When system administrator assigns permission to users what they are permitted to do and what they are not permitted to do, must be explained in a clear language.
- Clarity of security module configuration information: Language used must instruct and give feedback related to security module configuration must be clear, such as algorithm used, key type, selection of keys etc.

- Clarity of errors generated due to the security module malfunctioning: Error related to security must be clear and instructions should be given how to solve the problem or a direction where to get more information on how to solve the problem must be included.
- Visibility of system status: Is the information about what is going on in the system understandable.
- Match between the system and the real world: The language used by the system should not be formulated in system-oriented terms; information must appear using natural language.
- User control and freedom: If the user enters an unwanted state there must be a way to get out of the unwanted state.
- Error prevention: Users should be asked for confirmation before committing an action. Critical errors should be prevented by repeating the warning message.
- Minimise the user memory load: Instructions for using the system should be clear. Sufficient information should be available in the interface as how to proceed to the next stage.
- Recovering from system errors: Error messages related to system functionality should be expressed in clear language, explaining the problem and provide alternative solutions.

Figure 4-2 presents a process for developing heuristics for the system under evaluation. The list of heuristics presented therein is not as comprehensive as it can be. It can be expanded to include more heuristics depending on the rigour of the requirements in the evaluation process. Prior to developing heuristics, the process involves identification of the system's security functions and usability threats.

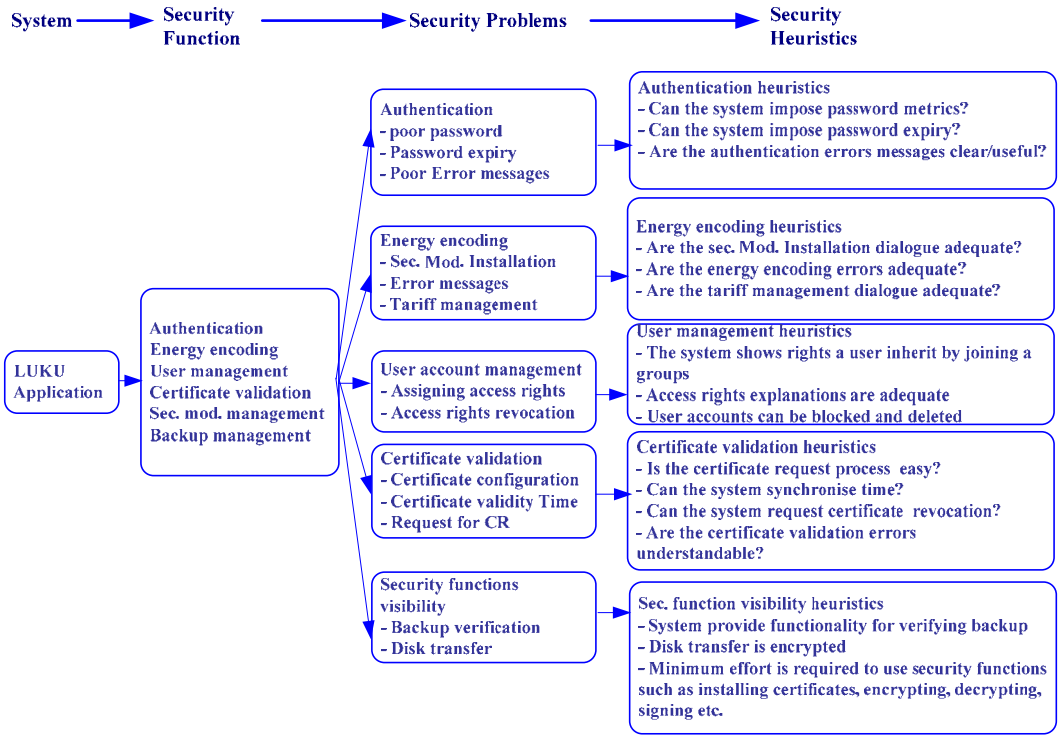


Figure 4-2 Mapping usability threats to security heuristics

The respondents were supposed to run the experiment twice. In the experiment, error recording is based on the observation and the given heuristics that guided the evaluation.

However, respondents were also allowed to record any other error they may come across even if is not stated in the heuristics. Each respondent evaluated the product alone. A follow-up questionnaire was completed by each respondent and results were aggregated and analysed.

4.4 Experimental Process

The experimental process, which is depicted in Figure 4-3, involves developing security heuristics, which are known security features of the interface, and the related questions that can be used during result discussion with respondents. The questions should help to cross-check any misunderstanding or lack of knowledge on security functionality that is lacking in the interface. The selection of participants in the experiments shall be done depending on the nature of the system which is being tested. More respondents may be required if the system can be classified as a critical system. The level of analysis also depends on the respondents' skills. If he is less skilled in using the system or similar types of systems, then, a detailed briefing of how to use the system will be required and if he is skilled, the briefing will mainly focus on how to run the experiment.

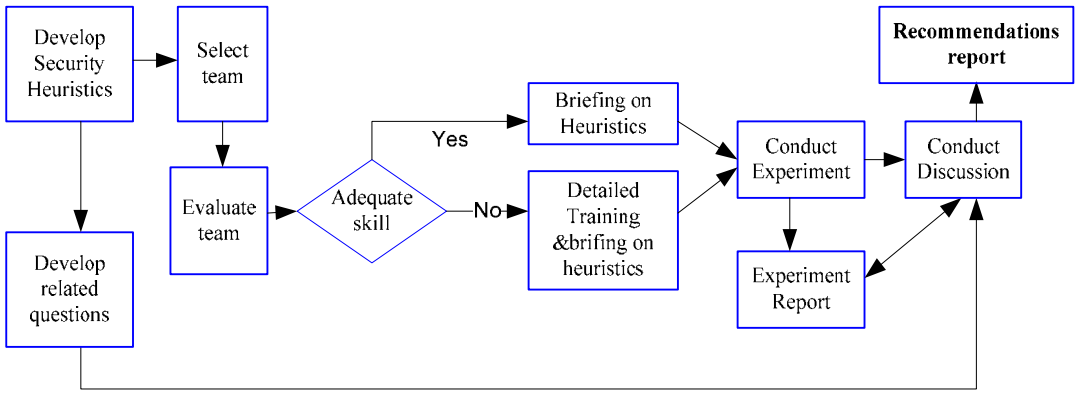


Figure 4-3 Experimental process using security usability heuristics and questionnaire

Table 4-4 contains sample questions for the questionnaire that had to be completed by each respondent participating in the heuristics analysis experiment. All respondents had to rate their choice on each heuristic using a 5-point rating scale with the anchor points Strongly Agree-Strongly Disagree. The questionnaire contained 48 usability heuristics questions in total.

Table 4-2 Sample questions in the usability heuristics questionnaire

No	Question	1 Strongly Agree	2 Agree	3 Unsure	4 Disagree	5 Strongly Disagree
1	The system indicated password metrics such as minimum size					
2	Error messages are clear and indicated where to obtain more information/solution					
3	I struggled to view user permission I had previously assigned					
5	This error message enabled me to know what is wrong with the security module (ISBX Error. No return code. Try again: Yes/No)					

Figure 4-4 is a graphical representation of experimental data. The experiment took into account three interfaces of the LUKU system, namely: the Credit Dispensing Unit application, the systems master station interface and the interface to the database. All these use the security module for security of all the transitions they process.

Data collected from the respondents was then used to calculate the following parameters:

- The proportion of all usability problems found by a single operator.
- Number of different usability problems found by aggregating reports from independent evaluators starting from one up to five.
- The total number of usability problems in the interfaces.

The figure clearly shows that there is a payoff from using more than one evaluator. However, five evaluators could suffice because after the fourth evaluator there are much less new usability problems discovered. More evaluators should be used if the system was a safety-critical systems being evaluated.

The shape of the curve depends on the characteristics of the product and on the total number of usability problems and proportion of usability problems found by a single evaluator.

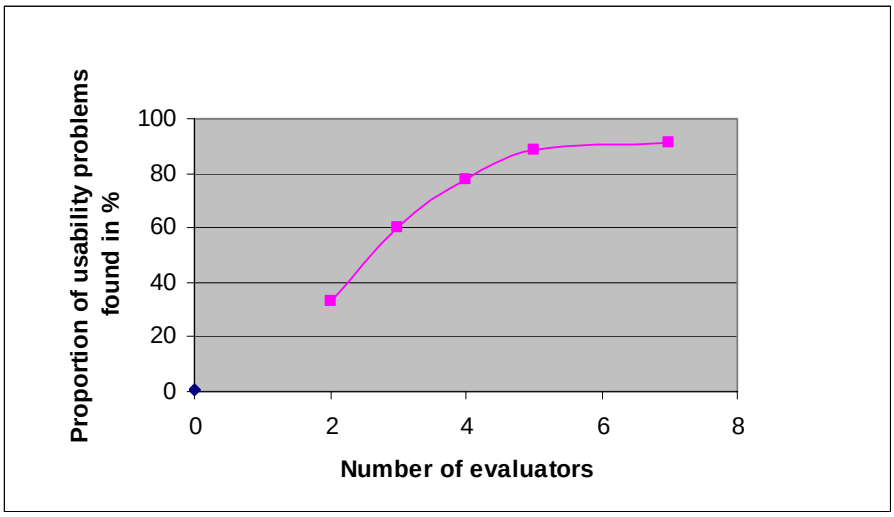


Figure 4-4. Proportion of usability problems for five evaluators

Questionnaires provided useful information as how the respondents judged various usability problems in the interface. Generally, four respondents disagreed on the issue that error messages are not adequate and one strongly disagreed, as depicted in Figure 4-5. This implies that they regard error messages as very adequate. Error messages that are related to the security of the entire system are not elaborative of what is the possible source of error and does not offer suggestions as to what should be done to solve the problem.

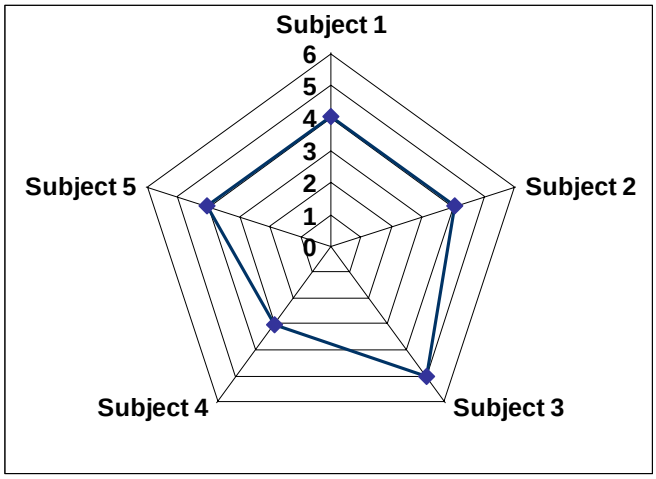


Figure 4-5 Error message adequacy

Figure 4-6 depicts a screen shot of an error message occurring when the security module malfunctions. It was also found that one respondent obtained the same error message when attempting to sell energy while the token was not inserted in the magnetic card printer.

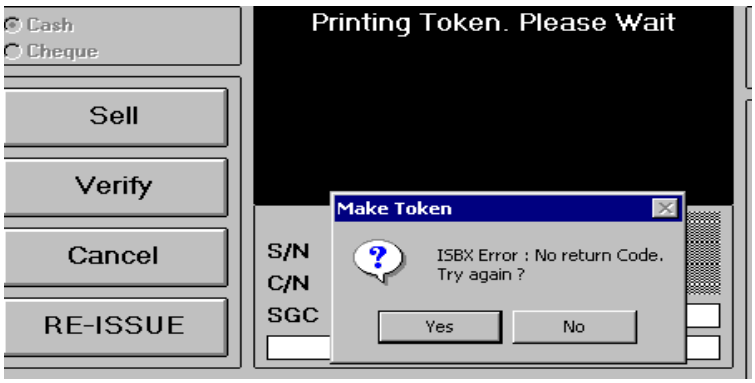


Figure 4-6 Security module error message

Two respondents strongly agreed that the system process status indication design was adequate and three respondents agreed on the same issue (Figure 4-7). A good system designer usually tries to avoid situations where users are not informed of what is going on in the system.

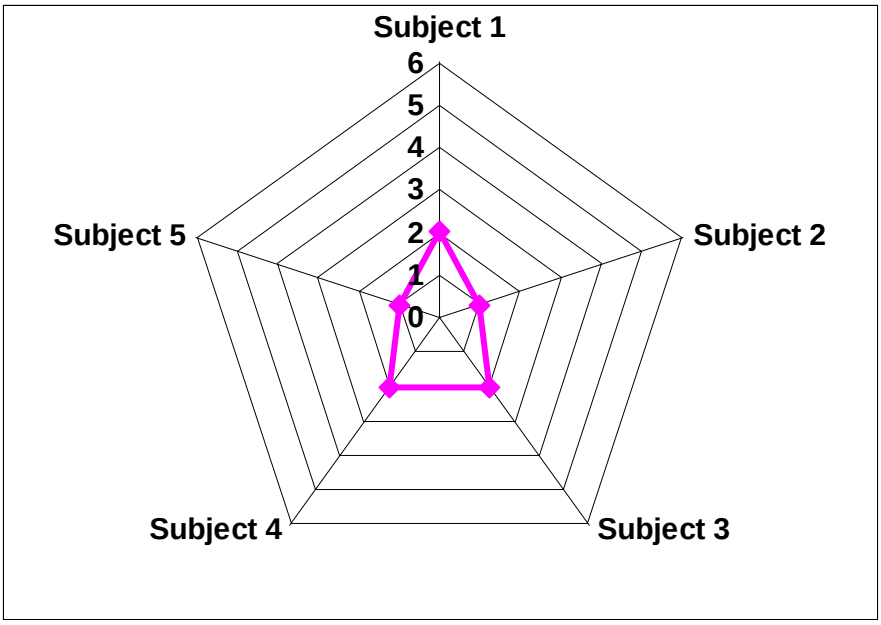


Figure 4-7 System process status indication

The combination of the two methods, i.e. heuristics analysis and questionnaire gave a clearer picture of existing problems in the LUKU interface than if one method only is used. A questionnaire can be tailored to capture the respondent's views. Also in the questionnaire, there is a room for respondents to indicate if they are not sure whether a particular part of the interface is adequately designed.

In the heuristics analysis respondents may not clearly explain their opinion usability problem. For example, the explanation "error messages are inadequate" has the potential to be subjective. We may not understand if the respondent strongly agrees or just agrees that error messages are inadequate.

4.5 Conclusion Regarding Usability Issues

This chapter reports the usability evaluation experiment that was conducted at TANESCO. We examined the method and machine component of the socio-technical model and specifically we examined the usability of the LUKU application security interface module. Respondents generally were unhappy with the login module because the authentication process involved four different passwords that were needed for; login to the system, shutting down the system, making transitions and accessing the configuration settings. This forced them to facilitate the memorization of the different passwords using stickers. The design of using different passwords for different system functionalities was intended to increase security by the system designers. This may not be the case since users ended up writing down passwords on stickers and storing the stickers in open drawers.

The heuristics and questionnaire methods provide quick feedback to developers. The experiment has demonstrated that if the report from the heuristics analysis experiments is combined with results of questionnaires it may be more useful for system developers to improve the system. In addition, results show that these methods have the potential to save time and cost since fewer respondents can be used to evaluate the system and give a rough picture of usability errors that exist in the interface during the whole development process.

In the next chapter, we examine internal security controls and security metrics that relate to the administrative, procedural and the application components of the SBC model.

Chapter 5

5 Internal Controls and Security Metrics

This chapter presents and discusses internal security controls and security metrics that can be used in organisations to indicate the maturity of various security processes. This chapter is related to the previous chapter in that we examine two more levels of the SBC model which are administrative and procedures levels.

The purpose of this chapter is to examine the role of internal security processes in improving the overall systems security and suggesting metrics that can be used to improve awareness and maturity of critical security processes within an organisation.

This chapter is divided in three sections which are internal security controls, metrics for internal security controls and metrics for evaluations of security system.

5.1 Protecting Assets and Services

In organisations, key goals such as productivity, cash flow, customer satisfaction, increased shareholder value, strategic plans, return on investment, etc. are essential organisational success factors that are monitored and controlled carefully. In order to achieve these goals, organisations invest in information systems (IS) that they acquire for information storage, processing and transfer. While the benefits IS brings to business success are increasingly high, there exist serious security problems that may cause information systems to fail. Since IS are extensively used, its failure might result into disruption or failure of the key organisational objectives. Therefore, IS security should be considered as a very important issues that need the management commitment and that must be constantly monitored and controlled.

The concept of security and the organisation's assets is depicted in Figure 5.1. It is necessary for an organisation to have internal security process such as policy evaluation, threat analysis, risk assessment, etc. to be done so that they are properly documented and their severity level established. Levels such as the risks level is high or low is useful when making decision about how these risks should be addressed. Also such metrics can help in selecting the right protection mechanisms.

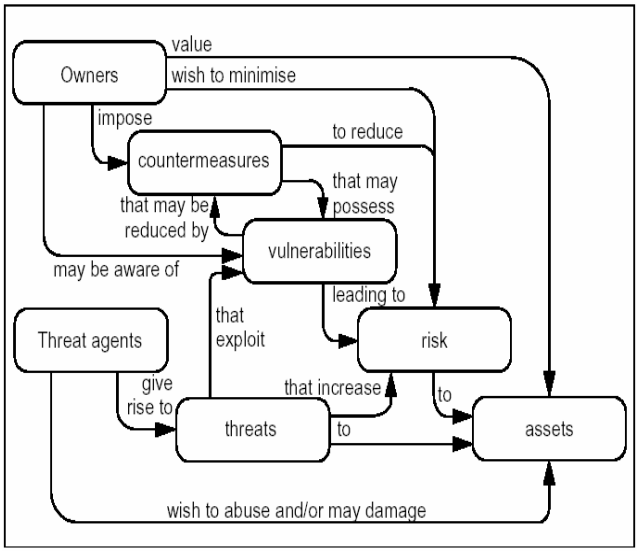


Figure 5-1 Security Concepts (CCIMB1 2005, p. 14)

The use of security metrics is also important when implementing countermeasures, vulnerability assessment, implementation of a security processes and security testing. Security metrics should be helpful to establish the maturity level of a security processes, coverage in the testing process and to indicate the availability of security programs in the organisation.

5.2 Metrics for Quantifiable Information

IT security metrics should be designed to yield quantifiable information (Swanson *et al.*, 2003; Ammann & Black, 1999; FBCA, 2002; SSE-CMM, 2003). The quantifiable information is useful for the following purposes:

- Comparison of security maturity
- Tracking changes using the same point of reference
- Coverage measurements during testing
- Cost justification when insecurities can be clearly shown in metrics
- Indication and determination of critical and non critical security parameters and test cases
- Redirect assets and set proper priorities for most critical security needs
- Security problem isolation and
- Determine the effectiveness of security testing efforts

IT security metrics can be created to guide each aspect of security program including systems evaluation, internal security processes such as training and systems testing and risk assessment. The use of security metrics will allow organisations to determine effectiveness of implemented IS security processes, and control by relating results of IS security activities measurements (SSE-CMM, 2003).

5.3 Metrics and Measurements

There are two types of metrics namely process metrics and security metrics (Jelen, 2001). The process metrics are some measures that can be used as evidence of the maturity of the security engineering process and the security metrics indicate the extent which some security attribute for example confidentiality, integrity, non repudiation, access control

and availability is present in the security engineering process. In metrics units like absolute numbers are sometimes useful, percentage, binary, and averages are most common.

The difference between metrics and measurements is that metrics are function of measurements and time. That is they are obtained by taking measurements over time and metrics should be specific, measurable, comparable, attainable, repeatable, and time dependant. Measurements provide one time view of specific measurable parameters and are represented by numbers, binary statements and weights. Metrics are useful because when two or more measurements are compared with predefined baseline measurements, over a period of time, provides a means for interpretation of collected data.

5.4 Internal Security Controls

Internal organisational controls could significantly improve the overall systems security. In this chapter we do not intend to provide a comprehensive list of controls but we provide a few of them as an example how to go about establishing internal controls and metrics. These controls are not directly related to the systems security functionality they are related to various internal security processes. These may help to minimise security risks that result from poorly monitored security processes. In Table 5-1 we present some of the internal control that must be enforced to ensure a desired level of security is maintained within an organisation. Others security metrics are outlined in SSE-CMM (2003).

Table 5-1 Critical internal security controls

Elements	Controls
Risk management	Percent of systems that had a formal risk assessment performed and documented
Threat management	Percent of systems that had a formal internal threats analysis performed continuously Percent of systems that had a formal external threats analysis performed continuously
Systems evaluation	Percentage of systems whose security controls have been tested last year Percentage of systems that have been patched and whose security control have been retested last year
System certification	Percentage of systems in use that are accredited and certified
Policy	Percentage of systems policy statements which have been updated last year
Security incidents	Procedure for reporting security incidents and response
Response capability	Number of internally reported incidents Number of incidents reported to law enforcement/external regulators Number of components with incidence handling

	capability
Usability	Systems with usability test conducted
Documentation	Systems with security configuration documented
Code of conduct	Is the security code of conduct available to security personnel/engineers
Physical security	Is the IS and physical security jointly managed
Integrity checks	Percentage of critical systems that have integrity checks conducted
Recruitment	Security Background checks for new employees
Socialisation	Security Socialisation process for new employees

5.5 Metrics for Internal Security Controls

Quantifying internal security controls is necessary to address one of the security central issues-the human factor. People tend to forget, misinterprets security patterns, and make errors when performing security functions. It is desirable that when an employee is disgruntled, the access rights to various resources should be immediately stopped. Also it is desired that antivirus and firewalls must be updated regularly according to plans. But lack of security metrics makes it difficulty to control such update. Another area where metrics are important is managing changes in systems security or track systems security updates.

5.6 Metrics and Risk Prioritisation

Metrics that are used to indicate high risk areas are expected to quantify the seriousness of the implication in case such a security mechanism is compromised.

RISK = probability * consequence

An asset with high security risk implies that the probability that the security incident will occur is high and the consequence of a security incident is severe. Such assets security metrics must have high security indicators. By high indicators we mean if a disgruntled staff pose high risk to assets then we expect the percentage of systems that access was terminated immediately upon employees termination to be about a hundred percent. Risk level can be set using quantitative values (low, medium, high and very high). In addition to the risk level in the security program other parameters such as program goal and objective must be defined. Also the metric, purpose, source of data, frequency, implementation evidence and the indicator must be defined.

The program goal states the desired results of the security controls that are measured by the metric. Such goals can be evaluating if systems in use are evaluated. The program objective will list all the necessary questions that will be asked according to the program goal. Answers to questions will be provided along with quantitative measurements such as percentage or number of evaluated systems.

The purpose of the metric could be improving the security process or evaluating the maturity level of the security process. The implementation evidence should give the evidence of system evaluation such that the presence of evaluation certificates.

Frequency indicates the suggested time frame when the security process is to be performed. The formula indicates how the metric is going to be calculated using data that can be obtained from documentation, database or the organisation's repository. The indicator describes the meaning of the metric. If the metric is a percentage then the indicator will describe the implications when the metric approaches to zero percentages and when approaches to hundred percent. Table 5-2 presents an example of how a metrics program can be organised.

Table 5-2 Example of how to take metrics for a security process

Program goal	To test how many systems have been installed with integrity check program
Risk level	Very high
Program evidence	Is integrity check stipulated in the security policy
Metric	Percentage of systems with integrity check tool installed
Purpose	To evaluated maturity of integrity check installation process
Implementation Evidence	• Is integrity program installation history data available? • How often integrity check are performed? • Total number of critical systems
Frequency	Daily/weekly
Formula	Number of systems with integrity check installed divide by Total critical number of systems
Data source	Systems inventory database
Indicator	The target is 100 percent. The higher the percentage the higher the security maturity for this process. If the percentage is law it implies that this area needs attention

Similar metrics as in Table 5-2 may be done for other security processes such as training for staff on security issues, contingence plans, incidence response readiness etc. Such metrics should enhance internal security through easy monitoring and comparison of security maturity for various security processes.

5.7 Metrics for Evaluation of a Security Systems

The process of evaluating security systems and functions that provides security tends to be complex and tedious. Complexity and tediousness is a result of complex systems and test cases that cannot be full covered. As a result, testers and evaluators are often left with the option of choosing some of the test cases that may not, in some cases, represent a good coverage of the entire system. Therefore, it is imperative to indicate the coverage when evaluating systems. Figure 5-2 depicts a system with security functions 1 to n. Test cases for each security function must be generated. Generating all the test cases is often unachievable goal. However, we want using metrics to be able to show the degree of coverage.

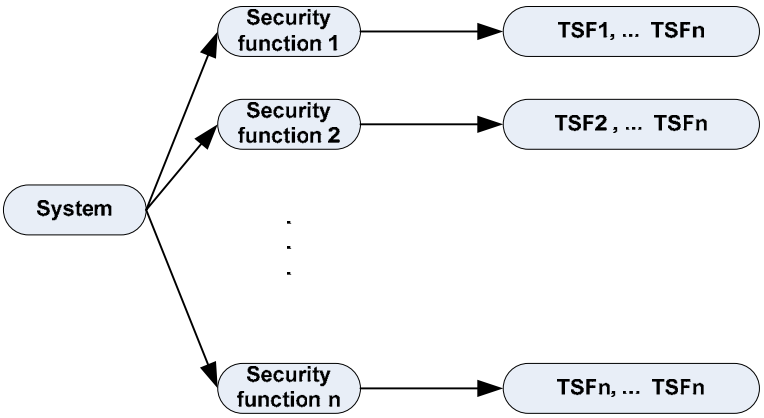


Figure 5-2 Generation of test cases of the Target of Evaluation Security Function (TSF)

The sensitivity of various test cases will varies significantly. The variation will depend on the threats a particular test attempts to mitigate. This section gives the summary of the risk levels associated with each test case. To provide granularity, four levels should be specified: low, medium, high and very high. It can be expected that for high and very high risks security functions the test coverage must be higher than lower risky functions.

Threats to assets may vary depending on the security function that is involved to provide security. In Chapter 6, requirements for a PKI system are presented. Such a system evaluation will focus on security functions that are designated to verify the validity of the certificate. The verification of the certification involves verifying that X509 certificate (RFC3280, 2002) attributes are validated correctly by the system (some of which are shown below).

Certificate serial number: Certificate Revocation Lists (CRL) lists certificate serial numbers. A non-standard certificate serial number other than 20 octets will disrupt interoperability. If CA key is compromised a malicious user may use it to post false CRL lists.
Signature: A malicious user may sign certificates in order to gain access to unauthorized assets
Issuer name: Issuer name testing is to make sure a malicious user does not issue certificate in the chain.
Subject name: Subject name testing is to make sure a malicious user is not issued certificate in the chain. If this is compromised malicious entity may gain access to assets.
Validity period: Threats that are being addressed by this testing include unauthorised access to assets by terminated employees, business partners whose association does no longer exist, accessing resources before or after authorised time, loss of legal rights.
Certificate policy: This test makes sure certificates are used in no transaction other than the stated purpose of use. If this is compromised malicious users may gain access to unauthorized assets.
CRL Distribution point: Threats associated with this test are loss of integrity, loss of confidentiality and non-repudiation
CRL Signature: A malicious CA may sign CRL
CRL Issuer name: Issuer name testing is to make sure a malicious CA does not issue CRL. If this is compromised malicious entity may disrupt business

Revoked Certificate: Threats associated to this test is compromised integrity, non-repudiation, confidentiality, authentication, and availability, and access control.

System evaluation metrics should clearly show the assurance level with respect to the security service it provides. Table 5-3 shows an example of security functions that may be involved in verifying X.509 certificate (RFC3280 2002) and the risks associated if such as functionality is compromised.

Table 5-3 some of the X.509 certificate security functions and the corresponding assurance level

No	Test Case	Security service						Assurance Level (Metrics)			
		Integrity	Confidentiality	Authentication	Availability	Access control	Non-repudiation	Low	Medium	High	Very High
1.	**Certificate serial number**							X			
2.	**Signature**	X	X	X		X	X				X
3.	**Issuer name**	X	X	X		X	X				X
4.	**Subject name**	X	X	X		X	X				X
5.	**Validity period**		X	X		X	X				X
6.	**Certificate policy**		X			X	X			X	
7.	**CRL Distribution point**								X		
8.	**CRL Signature**			X						X	
9.	**CRL Issuer name**			X						X	
10.	**Revoked Certificate**	X	X	X		X	X				X

Security functions with high or very high risk level are expected to have high degree of coverage during evaluation. A function such as signature verification needs through testing because if signature cannot be verified correctly the implications are that fraudulent transactions may be successfully committed.

Table 5-4 Certificate serial number verification testing metric

Program goal	To test how the application verify the certificate serial number
Risk level	Low
Program evidence	Is the certificate serial number type and size specified?
Metric	Percentage of test cases that are executed
Purpose	To evaluated the ability of the application to verify the certificate serial number correctly
Implementation Evidence	• Number of generated test cases? • Number of possible test cases that can be generated • Method of generating test cases • Number of test cases actually executed
Frequency	Every time a system is developed, acquired or upgraded
Formula	Number of test cases executed divided to total number of possible test cases.
Data source	Systems test case documentation or test database
Indicator	The coverage of test cases must approach 100 percent. The higher the percentage of test cases it implies that little chance is there for the system not correctly verify the serial number of certificate

Similar metrics can be collected for other security functions. Security functions with high risk levels are expected to be tested more thoroughly. Therefore, the coverage percentage should approach a hundred percent.

5.7.1 Test cases generation

Generating test cases is one of the most challenging tasks in the testing process because some testing requires extremely large number of test cases. Test cases that require large number are those involving parameters such as time and names because time can be represented in different times and it is hard to know when the system will fail. Also the verification of names requires many test cases because names can be written in many different ways. Test cases can be generated manually or by using tools designed to automate the generation process. These tools may automate the generation of test cases, documentation of testing process and re-use of test cases. Tools for generating functional test cases in different test environment are presented (Passmark, 2006; Tools, 2006).

Test cases must be generated such that when applied should give valid results and invalid results. This is demonstrated in test cases examples for testing PKI application ability to verify certificate that are presented below. In the PKI scenario the chain may include the top CA (TCA), registration CA (RCA) and the end entity (EE).

Table 5-5 Signature validation tests

Test	Description	Expected Result
TS1_Valid EE signature	This is to test the application's ability to validate correctly valid signature on the EE certificate	The path should validate correctly
TS2_Invalid EE signature	This is to test the application's ability to validate correctly invalid signature on the EE certificate	The path should not validate correctly
TS3_Valid RCA signature	This is to test the application's ability to validate correctly valid signature on the intermediate CA certificate	The path should validate correctly
TS4_Invalid RCA signature	This is to test the application's ability to validate correctly invalid signature on the EE certificate	The path should not validate correctly

TS2_Invalid EE signature

The purpose of this test is to verify PKI enabled application ability to validate correctly an end entity certificate with invalid signature.
Certification chain comprises the following objects:

TCA Root Cert, TCA Root CRL; Valid RCA Cert, Valid RCA CRL; Invalid EE Signature TS2 EE

Procedure: Open and verify signed test message Invalid EE Signature TS2 EE.

The expected result is that the path should not validate successfully, as the signature on the end entity certificate is invalid.

TVD Time validity
Validity period is the time interval that the TCA warrants that it will maintain the information status of the certificate (RFC3280, 2002). This interval is marked by validity beginning date (notBefore) and validity end date (notAfter). Both notBefore and notAfter can be encoded in UTC or GeneralizedTime and the certificate using application must support UTC and Generalized Time encoding (RFC3280, 2002). UTCTime specifies the year through the two low order digits and time is specified to the precision of one minute or one second. UTCTime includes either Z (for Zulu, or Greenwich Mean Time) or a time differential.

The validity time test cases shall be generated to include valid and invalid expected results in the following scenarios according to (RFC3280, 2002) specification:

- Entire certificate chain
- UTC time format
- GMT time format
- Not after date

- Not before date
- Before 2000
- After 2050

All these scenarios need different test cases for different encoding of time.

Table 5-6 some examples of time validity test cases

Test	Description	Expected Test Results
TVD1_Valid EE notBefore date	notBefore date on the EE certificate must be earlier than the current date	The path should validate successfully
TVD2_Invalid EE notBefore date	notBefore date on the EE certificate must be later than the current date	The path should not validate successfully
TVD3_Valid CA notAfter date	notBefore date on the intermediate CA certificate must be earlier than the current date	The path should validate successfully
TVD4_Invalid CA notBefore date	notBefore date on the intermediate CA certificate must be later than the current date	The path should not validate successfully
TVD5_Valid EE notAfter date	notAfter date on the EE certificate must be after the current date	The path should validate successfully
TVD6_Invalid EE notAfter date	notAfter date on the EE certificate must be before the current date	The path should not validate successfully

TVD2_Invalid EE notBefore Date

The purpose of this test is to verify the applications ability to process the chain verification id the notBefore date is after the current date.
The certification chains comprise the following objects:

TCA Root Cert, TCA Root CRL; Valid RCA Cert, Valid RCA CRL; Invalid notBefore Date TVD2 EE

Procedure: Open and verify signed test message Invalid EE notBefore Date TVD2 EE.

Expected result: The path should not validate successfully as the notBefore date in the end entity certificate is after the current date.

Generally, generating manually voluminous test cases such as time test cases will be tedious and may result into errors of omission of critical test cases. Using tools to generate such test cases could save time and allow for re-sue of test cases when we need to re-test.

5.8 Chapter Summary

In this chapter we examined the internal process and security metrics that can be used in the administration of internal security controls. Also we examine metrics in testing a system. More details about security metrics, testing and internal controls are presented in Appendix G. The next chapter presents the development of TANESCOS's PKI security requirements and re-use of security requirements.

Chapter 6

6 Security Requirements and Analysis

In this chapter, we examine the concept of developing security requirements using the Common Criteria and re-use of security requirements.

The purpose of this chapter is to examine challenges in developing security requirements such as how we can make, with certainty, assumptions about the environment in which the system will be used. In addition, we examine the re-use of security requirement.

This chapter consists of four sections namely TOE description, security environment, security objective and re-use of security requirements. The TOE is TANESCO's prepayment system namely, LUKU and the security environment includes people and other systems with which the LUKU system will interact.

Re-use is an emerging trend not only in security area but also in software information systems development area. Developing security requirements is an expensive and difficulty process even for experienced developers. This increases the need for approaches that can be applied to develop requirements in a more cost effective way. This process involves developing TANESCO's Public Key Infrastructure Protection Profile that is a product of this research and will be used to evaluate and examine security requirements re-use. A full PP document is in Appendix F. This approach has at least three advantages:

- PP development process enable to researchers to get knowledge how to work with systems security functions
- PP development process used to examine re-use
- PP as a product of research will actually be used by TANESCO when developing ST for the PKI

The Protection Profile for TANESCO is an enabling technology for implementing organisational PKI. It is the baseline for a formal security requirement. It fulfils the requirements of the Common Criteria version 2.2, the ISO standard 15408. The key components of this document are PP Introduction, TOE description, TOE security environment, Security objectives, IS security requirements and Rationale.

This Protection Profile specifies the Information Technology (IT) security requirements for TANESCO's PKI enabled Electricity Prepayment System application. The system comprises the credit-dispensing unit (CDU), System master Station, Electric Dispensers

(meters). The same application software runs on the SMS and CDU. The security requirement in this PP applies only to the application software.

6.1 PKI Protection Profile

The introduction section of the protection profile provides information about the PP version, authors, document management information highlights to enable the reader judge the usefulness of the PP in his environment. TOE provides description that is sufficient to enable the reader understand how the TOE works. The TOE security environment presents the threats to TOE that could result from the environment such as cryptographic algorithms, operating system, users and the network. The security objectives provide security description of the TOE security function. The IT security requirement section details the functional security requirements and the rationale section demonstrates that whether the PP is complete and consistence.

6.1.1 PP Introduction

The introduction section provides document management and overview information necessary to operate a protection profile registry. The introduction provides background information that will enable the reader to gain a high-level understanding of the protection profile.

6.1.2 Protection Profile Overview

The target of evaluation (TOE) is the application software for TANESCO's Electric Prepayment System (EPS). This application is used to vend electricity and as a system master station. In the vending transaction process functions such as operator authentication, security module, Printer/Reader/Writer (PRW), Customer identification by the account number or meter number must be invoked. Other functions include credit limit validation, Swipe card reader, coded token cancellation, transaction abortion, reissuing in case the token is damaged or lost, token verification, sell, Banking, token energy limit, issuing free token, tariff verification and meter tariff key change.

6.1.3 Protection Profile Document Organisation

Table 6-1 Summary of the components of the PP document

No	PP COMPONENT	DETAILS OF THE COMPONENT
1	**PP Introduction**	PP Identification PP Overview Document organisation Related PP and other documents
2	**TOE Description**	Coherent TOE description
3	**System security Environment**	Assumptions regarding threats Assumptions regarding policies Threat addressed by the system Detailed attacks countered by the system Threats addressed by the system with support by Attacks countered by the environment Organisations general security policies of the Organisations detailed policies assigned to the Organisations general security policies for system Organisational detailed security policies assigned
4	**Security Objectives**	Security objectives for the system

		Security objectives for the environment
5	**IT security**	Systems security functional requirements Security assurance requirements
6	**Rationale**	General threat and attack rationale Attack and security objectives correspondence Detailed policy and general policy mapping Detailed policy statement and security objective Security objective security requirements rationale Security requirement dependence analysis

This structure is proposed in CCIMB1, (2005) and CCIMB4 (2005). The rationale section is presented to give the evaluator a clear picture whether the security requirements match the threats.

The System Master Station (SMS) includes the following main functions: Credit Dispensing Unit registration, operator's management, meter registration, Customer registration, disk transfer, reports and end of month processing. This PP does not include network connections, hardware on which the application software is installed, operation system and other software that might be installed in same computer as the TOE. These make up the environment on which the TOE will operate and its security assumptions are listed in the TOE environment.

6.1.4 Related PPs

The re-use analysis is carried out using the Department of Defence Public Key-Enabled Application Family of Protection Profiles, Version 2.5, of October 31, 2002, (USMC, 2002). The major difference between the new PP and the reference PP is that the environment is different. Therefore, it is expected that a PP developed in a different environment cannot be directly used in a different environment because of the following reasons:

- Different threat assumptions
- Different attacker potential
- Different user trust, knowledge and security culture
- Different organisational security policies
- Different laws governing the use and evaluation of ICT systems
- etc.

6.1.5 The CCToolkit

Protection Profiles are complex documents in which security component of a product, group of products or systems are specified. Selecting the security components from CC and organising them into a coherent PP is a complex and tiring work especially for a novice user.

In effort to address this problem, the National Information Assurance Partnership developed the CCToolkit. This is a tool to aids authors of PPs and STs in the process of drafting PPs and STs. It aids inexperienced CC users in identifying relevant CC

components to their system. Figure 6-1 shows some of the necessary toolkit functions The CCToolkit is public and available at (NIAP, 2000).

The CCTookit provides the PP developer with the ability to perform several tasks including conducting environmental and components interview, define the context, allocation, specify the evaluation assurance level and generate reports.

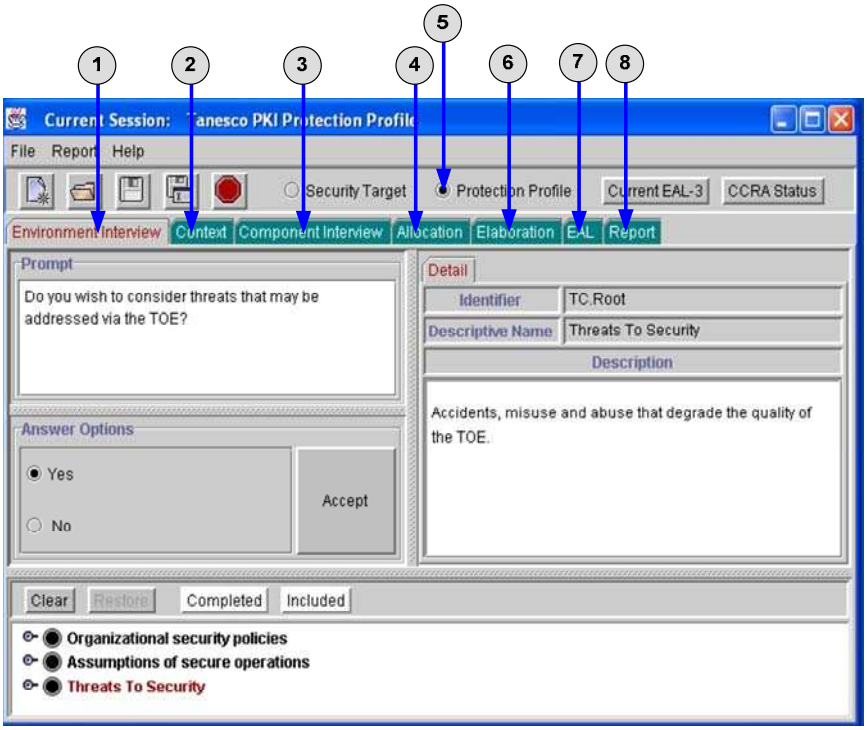


Figure 6-1 CCToolkit interface

Environmental interview: The TOE security environment and related objectives basic elements in this environment include policies, threats and assumptions that constrain the TOE's target environment.

Context: The context interface provides the user with the ability to specify the TOE environment. A TOE environment specification consists of the following: set of selected security objectives, set of policies, set of threats, set of assumptions and mappings relating each of the policies, threats, assumptions to one or more members of the set of selected security objectives.

Component interview: The purpose of this interface is to assist the user, by means of an interview process, in specifying the TOE security environment

Allocation: Allows the developer to iterate, extend, edit and delete a security function or assurance components

Elaboration: This interface allows the developer to elaborate how a user guidance or functional testing should be performed.

EAL (Evaluation Assurance Level): This interface allows you to specify an EAL by selecting one of the buttons corresponding to the seven assurance levels found in the CC.

Report: The report interface will allow the PP developer to generate a draft report which will include the PP introduction, TOE description, TOE security environment, Security objectives, IT security requirements and the rationale.

Limitations of CCTookit

The CC Toolbox is not designed to produce a completely finished PP report. Rather, it is designed to remove as much of the tedium as is feasible from the process of creating reports. Consequently, it attempts to handle only the mechanical and structural aspects of report building. It has limited editing capabilities leaving some additional polishing and refinement to the author in a more suitable application tools and environment.

6.2 TOE Description

The TOE Description is a critical part of the protection profile. It provides a TOE description that enables the reader to:

- Gain an understanding of how the system operates
- Know where the component fits into the system
- Be able to define the TOE operation and its limitations. Be sure to include at least one figure that shows the relationship of the TOE elements or shows the relationship of the TOE to its environment.

6.2.1 Overview

A traditional electronic prepayment metering system operates on three levels. At the lowest level, are the meters, which are installed in the customer's home. The customer interfaces with the Electricity Dispenser (meter) with a token entry method, such as a magnetic card slot, or a keypad. Status indicators show token accept/reject, power available level with analogue bar graph and/or a digit display, and the consumption rate with flashing light. The next level being the vending stations (Credit Dispensing Units (CDU)), which are placed at the utility's office or at appointed agents and are operated by CDU Operators. The CDU issues tokens and provides for first line administrative and financial control.

The communication between the vending stations and meters is in the form of a token which is used to top up the credit in the meter as well as to transfer or download information to the meter, and in some cases upload information (depending on the token choice) back to the vending station. The Token refers to the disposable magnetic stripe card (for Magnetic meters) or push-button twenty-digit number (for Keypad meters) issued to the customer, as shown in Figure 6.2.

The magnetic stripe on the token or the twenty-digit number (Keypad token) carries an encoded number which credits the customer's meter with the credit purchased. After use, the token is discarded. At the top level is the System Master Station (SMS) or Master Client, which is necessary to ensure a common database for reporting as well as to provide for total management, administration, financial and engineering control. The SMS communicates with the CDUs through online or offline means. Information on the consumers, tariff changes, etc. is communicated to the vending station and detailed customer sales are communicated back up to the SMS.

An application is PKI enabled if: (1) securely manages keys, trust anchors, and certificates. (2) Uses one or more of the security services supported by the PKI system by accepting and processing a X.509 digital certificate, (3)It is able to obtain relevant certificates and revocation data, (4) Checks each certificate for validity, using procedures described in the X.509 standard (RFC, 3082), and prior to reliance, including checking for revocation, (5) Has access to accurate and reliable system time in order to verify the dates on certificates, revocation data, and application data, (6) Collects, stores and maintains the data required to support digital signature verification in the future, (7) Is able to automatically select from multiple private decryption keys if it performs public key based decryption and (8) Is able to interoperate with other Government PKI.

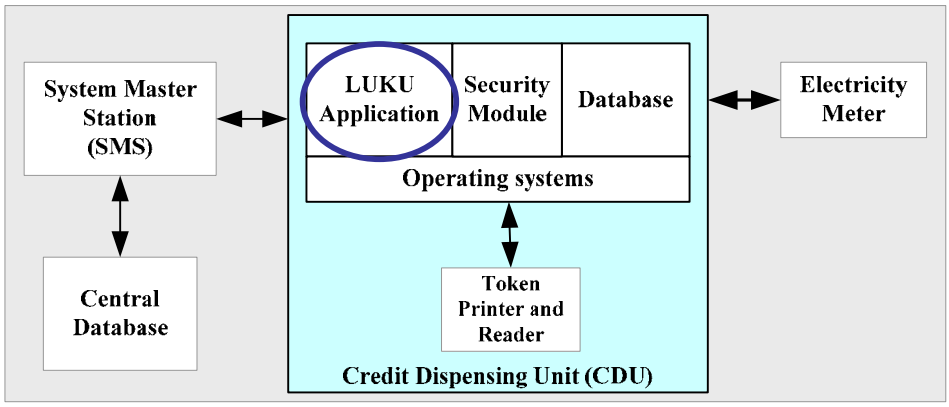


Figure 6-2 Prepayment application (LUKU) environnent

6.2.2 Public Key Infrastructure hierarchy

The certifications hierarchy, in Figure 6-3, includes the TOP certification authority, local certification authority and the end entity. Top CA in this case is the headquarters that will keep control of the entire system by controlling the certificate issuance process and keeping the revocation list. The local certification authority responsibility is to register all users in a particular region who are legible to receive certificates from the Top CA. The local CA plays the role of a registration authority and policy authority; apart from registration of users, local CA will set policy for users. Users include master station operators, CDU operators, and backup operators. Certificate policy defines the role of each user of the PKI system.

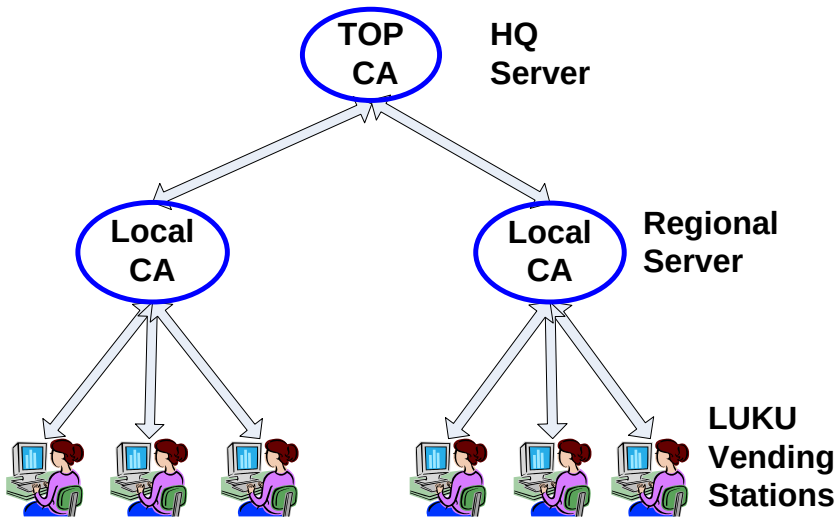


Figure 6-3 PKI Structure showing TOP and Local CAs and the End users

This PP can be applicable to other applications that are PKI enabled. An application is PKI enabled if:

- Securely manages keys, certificates and trusted anchors
- Accept and processes X.509 certificates
- Is able to process certificates revocation data
- Is able to check each certificate for validity
- Can reliably validate certificate; revocation data and application data using time from reliable source
- Support future verification of digital signature
- Bind users correctly to digital certificates
- Is able to interoperate with other PKI enabled application within the organisation
- Is able to handle variety of public key mechanisms such as signature generation, signature verification, encryption, decryption and entity authentication

6.2.3 Approach

This section provides information about the approach used to develop this PP. The underlying assumption is that the reader posses some knowledge about CC, PKI, cryptographic algorithms and security requirements specification.

6.2.4 Approach for PKI requirements

X.509 certificate is defined in ITU-T Recommendation X.509 and further defined in Internet Engineering Task Force (IETF) Request for Comments (RFC, 3280). To ensure secure interoperation of PKI-enabled applications that makes use of X.509 certificates, the path validation must be done in accordance to these specifications.

X.509 is part of X.500 series of ITU-T recommendations that define a directory service. X.509 is based on the use of public key cryptography and digital signature. The X.509 certificate comprises the following attributes:

- **Version:** The default version is 1. If the unique identifier is available then the version must be 2. If one or more extensions are available the version must be 3.
- **Serial number:** Is unambiguous and unique integer value within the issuing CA that is associated with this certificate (RFC, 3280). The default value for serial number must be 20 octets
- **Signature algorithm identifier:** Identifies the algorithm used to sign the certificate
- **Issuer name:** Is the name of the CA who signed this certificate
- **Period of validity:** This is the first and last date on which the certificate is valid. This is the period the CA vouches responsibility of maintaining the certificate.
- **Subject name:** Is the name of the user to whom this certificate refers.
- **Subject's public key information:** This is the public key of the subject, identifier of the algorithm together with any associated parameters.
- **Issuer unique identifier:** is a bit string field used to identify uniquely the issuing CA in the event the X.500 name has been reused for different entities.

- **Subject unique identifier:** is a bit string field used to identify uniquely the issuing CA in the event the X.500 name has been reused for different entities.
- **Signature:** It contains the hash code of the other fields, encrypted with the private key of the CA. This field includes the signature algorithm identifier.
- **Extensions:** A set of one or more extensions fields. These were options added in X.509 version 3 in efforts to address the shortcomings of version 2. The extension fall into three categories namely Key and policy information, subject and issuer attributes, and certificate path constraints. The policy indicates the applicability of the certificate to a particular community. The certificate subject and Issuer attributes extension supports alternative names, alternative name's format for the issuer and subject to increase certificate's user confidence. More added information includes details such as the postal address, position within the corporation or picture image may also be required for identification (RFC, 3280). The certificate path constraints are included in certificates issued for CAs by other CAs. This extension includes the basic constraint that indicates whether the subject can act as a CA, name constraint, and policy constraint.

This PP provides functional requirements for processing all of the certificate path and extensions. It provides the ability to select cryptographic algorithms from a range of available options. Some requirements such as audit data management may be met by the environment, which might include operating systems.

6.2.5 Certificate path validation

The Certificate Path Validation involves the process of validating for validity of all certificates in the chain, that is, all certificates including the Top CA, intermediate CA and the end entity certificates. The path validation process also involves processing the following extensions:

- Validity period
- Certificate policy extension
- Mapping policy
- Name constraints
- Signature generation
- Signature verification
- Key transfer algorithms
- Certificate revocation list and
- Audit data management

Validity period is the time interval that the CA warrants that it will maintain information status of the certificate (RFC3280, 2002). This interval is marked by validity beginning date validity end date. Time can be encoded in UTC or Generalized Time and the certificate using application must support UTC and Generalized Time encoding (RFC3280, 2002).

6.3 TOE Security Environment

In this section, the security environment in which the TOE will be used and the manner the TOE will be employed is presented. The security environment defines the context in which the TOE is intended to be used as presented in Table 6-1. It includes the laws, threats that are present in the environment, organisational security policies, customs,

expertise and knowledge that are relevant according to the CC General model CCIMB1, (2005

6.3.1 Secure Usage Assumptions

Table 6-2 present the secure usage assumption of the IT Environment. The symbol √ marks re-used assumption statements and X is used to marks new security assumptions.

Table 6-2 Secure usage assumptions for the IT Environment

No	Assumption Name	Description	?
1	AE.Authorized_Users	Authorized users are trusted to perform their security assigned functions.	√
2	AE_Culture	Users cultural values that pose as risk to security are identified and associated behaviour are controlled and Users have access to the code of conduct.	x
3	AE.Configuration	The configuration and installation of the TOE is done properly.	√
4	AE.Crypto_Module	The TOE environment is assumed to include one or more cryptographic module(s) that are all validated at FIPS 140 series Level 1 or higher. All cryptographic modules in the TOE shall be validated at FIPS 140 series Level 1	√
5	AE.Medium	The attack potential on the TOE is assumed to be Medium.	X
6	AE.Physical_Protection	Physical protection is assumed to be provided by the environment. The TOE hardware and software is assumed to be protected from unauthorized physical access by outsiders and insiders. Insiders are monitored to prevent misuse of TOE and colluding with outsiders.	X
7	AE.PKI_Info	The certificate and certificate revocation information is available to the TOE.	√
8	AE.Time	Accurate system time with required precision in GMTformat is assumed to be provided by the environment.	√
9	AE.Hacker_Social_Eng	Users are aware about social engineering and mechanisms are in place to prevent social engineering	X

Threat to security for the TOE

This section defines security threats to TOE as presented in Table 6-2. The security threat agents include, but not limited to: 1) Failure of TOE and 2) People with access to TOE who have considerable expertise, poor resources, and uncontrolled cultural traits such as short-term orientation behaviour, poor motivation and disgruntled personnel.

The assumption about attackers is that they have various levels of expertise, resources, and motivation. Attackers can be either insiders or outsiders. Relevant expertise may be in general semiconductor technology, software engineering, hacker techniques, or the specific TOE. Attackers may possess IT resources such as personal computers and inexpensive card reading/coding devices to very expensive and sophisticated engineering test. In addition, they may possess a replica of TANESCO vending devices. They may also include software routines, some of which are readily available on the Internet. Motivation may include economic reward, resentment, or notoriety of defeating high-grade security. The symbol √ in Table 6-3 marks re-used assumption statements and X is used to marks new security assumptions.

Table 6-3 Threats to security for the TOE

No	Threat Name	Threat Description	
1	T.Impersonation	An unauthorized individual may impersonate an authorized user of the TOE and thereby gain access to TOE secure data and functions.	√
2	T.Modification	An attacker may modify data, e.g., stored security attributes or private keys, in order to gain access to the	√

		TOE and its assets.	
3	T.Object_Init	An attacker may gain unauthorized access to an object upon its creation	√
4	T.Attacker	Insiders or outsiders may attempt to perform actions that the individual is not authorized to perform without being detected.	**X**
5	T.Bypass	An unauthorized individual or user may bypass security attributes or other data in order to gain unauthorized access to TOE assets.	√
6	T.Private_key	An unauthorised individual may assume the identity of a user by generating or using the private key of the user.	√
7	T.Role	A user may assume more privileged role than permitted and use the enhanced privilege to take unauthorized actions.	√
8	T.Secure_Attributes	A user may be able to change the security attributes of an object and gain unauthorized access to the object.	√
9	T.Shoulder_Surf	An unauthorized user may read authentication credentials by look over the shoulder of the authorized user while authentication is in progress.	√
10	T.Tries	An unauthorized individual may guess the authentication information using trial and error method.	√
11	T.Component_Fail	An attacker exploits failure of one component resulting in loss of systems critical functionality	X
12	T.Load_Bad_data	An unauthorized individual may load bad data or software that could modify or expose data on the TOE	X
13	T.Clone	An unauthorised individual may clone the TOE to develop further attacks	X

TANESCO PKI TOE is required to counter threats that may be broadly categorized as:

- Threats addressed by the TOE:
  - Threats associated with physical attack on the TOE
  - Threats associated with logical attack on the TOE
  - Threats associated with control of access
  - Threats associated with unanticipated interactions
  - Threats regarding cryptographic functions
  - Threats that monitor information

- Threats that are directly related to the functioning of the TOE such as threats to path validation, signature verification, policy mapping, revocation list verification, validity time verification etc.

6.3.2 Threats to security packages

This section defines security threats to TOE functions namely path validation for basic policy, name constraints, signature generation and verification, encryption and decryptions, online certificate status protocol, certificate revocation and audit data management. Table 6-4 presents additional threats to basic functions and the symbol √ marks re-used assumption statements and X is used to marks new security assumptions.

Table 6-4 Certificate Path validation (CPV) threats to basic functions

No	Threat Name	Threat Description	?
1	T.Certificate_Modi	An untrusted user may modify a certificate resulting in using a wrong public key.	√
2	T.DOS_CPV_Basic	The revocation information or access to revocation information could be made unavailable, resulting in loss of system availability.	√
3	T.Expired_Certificate	An expired certificate could be used for signature verification.	√
4	T.Comp_Prvkey	The private key may be compromised and an attacker may use the certificate before certificate revocation is effected	X
5	T.Masquarade	An untrusted entity (Certification Authority (CA)) may issue certificates to bogus entities, permitting those entities to assume identity of other legitimate users.	√
6	T.No_Crypto	The user public key and related information may not be available to carry out the cryptographic function.	√
7	T.Path_Not_Found	A valid certification path is not found due to lack of system functionality.	√
8	T.Revoked_Certificate	A revoked certificate could be used as valid, resulting in security compromise.	√
9	T.User_CA	A user could act as a CA, issuing unauthorized certificates.	√

6.3.3 Organisational Security Policies

This section identifies and defines organisational security policies. Also identify organisational security policy statements or rules with which the TOE must comply.

- Protection mechanisms shall be applied such that the TOE maintains the appropriate level of confidentiality, integrity, authentication, and non-repudiation based on mission criticality, sensitivity of information handled by the system.
- Digital Signature Standard keys shall use at least 160 bit private key and at least 1024 bit prime modulus. Minimum public key size shall be 1024 bits for Key Exchange Algorithm. Minimum public key size shall be 2048 bits for RSA.
- Applications must be signed by TANESCO approved entity to controls the loading of applications loading

6.4 Security Objectives

This section identifies and defines the security objectives for the TOE and its environment. Security objectives should reflect the stated intent, be suitable to counter all identified threats, and cover all identified organisational security policies and assumptions.

6.4.1 Security Objectives for the TOE

Table 6-5, below, define the Security Objectives for TOE. The environment may meet TOE security objectives. We use O prefix for security objective for TOE representation. The symbol √ marks re-used assumption statements and X is used to marks new security assumptions.

Table 6-5 – Security Objectives for the TOE

No	Objective Name	Objective Description	?
1	O.DAC	The TSF shall control and restrict user access to the TOE assets in accordance with a specified access control policy.	√
2	O.I&A	The TSF shall uniquely identify all users, and shall authenticate the claimed identify before granting a user access to the TOE facilities.	√
3	O.Init_Secure_Attr	The TSF shall provide valid default security attributes when an object is initialized.	√
4	O.Invoke	The TSF shall be invoked for all actions.	√
5	O.Limit_Actions_Auth	The TSF shall restrict the actions a user may perform before the TSF verifies the identity of the user.	√
6	O.Limit_Tries	The TSF shall restrict the number of consecutive unsuccessful authentication attempts.	√
7	O.No_Echo	The TSF shall not echo the authentication information.	√
8	O.Protect_I&A_Data	The TSF shall permit only authorized users to change the I&A data.	√
No	**Objective Name continue**	**Objective Description continue**	**?**
9	O.Secure_Attributes	The TSF shall permit only the authorized users to change the security attributes.	√
10	O.Security_Roles	The TSF shall maintain security-relevant roles and association of users with those roles.	√
11	O.Self_Protect	The TSF shall maintain a domain for its own execution that protects it and its resources from external interference, tampering, or unauthorized disclosure and unsafe fail	X
12	O.Trust_Anchor	The TSF shall permit only authorized users to manage the trust anchors.	√
13	O.TSF_Data	The TSF shall permit only authorized users to modify the TSF data.	√
14	O.Input_Safe	The TSF shall permit only the input of safe data	X
15	O.Clone	The TSF shall permit only authorised users to clone the TOE or part of TOE	X

6.4.2 Security Objectives for the Environment

Table 6-6 lists security objectives for the environment. This section clearly states and traces security objectives for the environment back to aspects of identified threats not

completely countered by the TOE and/or organisational security policies or assumptions not completely met by the TOE. The symbol √ marks re-used assumption statements and X is used to marks new security assumptions.

Table 6-6 – Security Objectives for the Environment

No	Objective Name	Objective Description	?
1	OE.Authorized_Users	Authorized users are trusted to perform their authorized tasks.	√
2	OE.Configuration	The TOE shall be installed and configured properly.	√
3	OE.Crypto	The environment shall include one or more cryptographic (modules) that are all validated at FIPS 140 series Level 1 or higher.	√
4	OE.Medium	The identification and authentication functions in the TOE shall be designed and implemented for a minimum attack potential of medium as validated by the vulnerability assessment and strength of function analyses.	√
5	OE.Physical_Security	The environment shall provide an acceptable level of physical security so that users cannot tamper with the TOE.	√
5	OE.PKI_Info	The IT environment shall provide the TOE certificate and certificate revocation information.	√
No	**Objective Name**	**Objective Description**	**?**
6	OE.Time	The environment shall provide access to accurate current time with required precision, translated to GMT	√
7	OE.Comm	The environment shall provide secure communication for users authentication data	X
8	OE.Social_Eng	The environment shall provide mechanisms that prevent social engineering attack and user awareness about the social engineering risks	X
9	OE.Res_Con	The IT environment shall prevent attacker or user from monopolising resources to deny service to others.	X

6.5 Re-use of Security Requirements

In this section we examine the re-use of security requirements. Specifically we examine the re-use in the development of PPs. Re-use of security requirements have advantages such as saving time and cost. Other advantages include shortened leaning curve of how to develop the security requirements for a particular class of system such as PKI. Most of the evaluated systems include smart card, IDS, PKI and operating systems VPL (2006).

Figure 6-4 shows new and reused security assumptions, threats, policies and the security objectives.

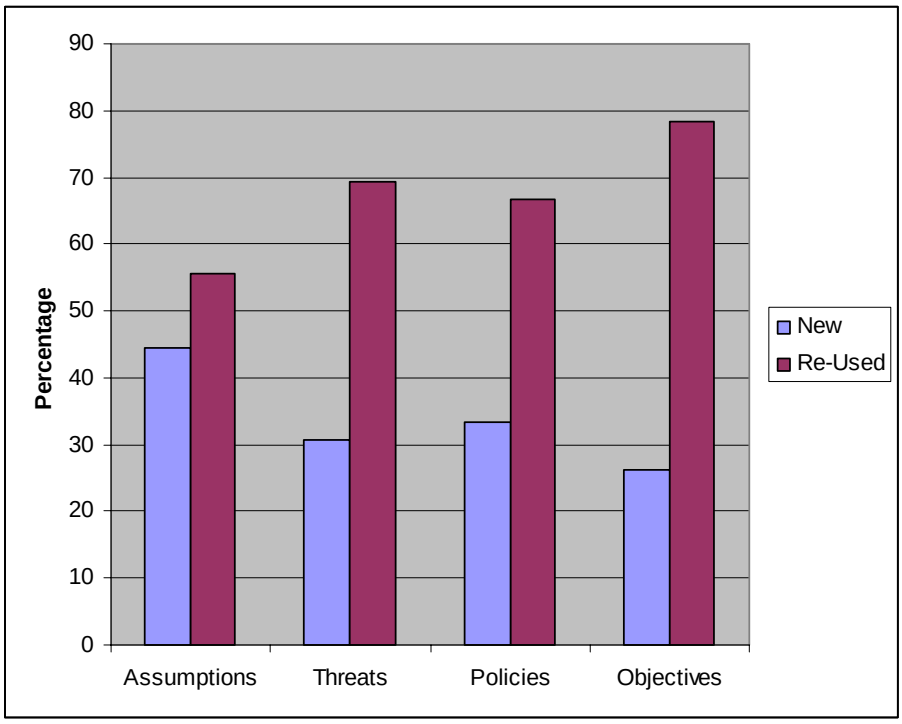


Figure 6-4 Re-Used TANESCO security assumptions, threats, policies and objectives from the source PP

The security assumptions of IT environment are very important and should address issues that are related to the users such as the following:

- **Cultural values:** Users cultural values that pose as risk to security are identified and associated behaviour are controlled and Users have access to the code of conduct.
- **Physical protection:** Physical protection is assumed to be provided by the environment. The TOE hardware and software is assumed to be protected from unauthorized physical access by outsiders and insiders. Insiders are monitored to prevent misuse of TOE and colluding with outsiders.
- **Social engineering**: Users are aware about social engineering and mechanisms are in place to deter, prevent or respond to social engineering attacks.

These assumptions may vary in different environments but forty four percent of all the security assumptions of the environment are new and 55.5% are reused from the reference PP. Table 6.3 defines thirteen threats to TOE, only 30.7% of all the threats are new and 69.2% are reused. Organisational security policy has 33.3% new policy and 66.6% reused policy.

The percentage of new and reused security objectives is a combination of the security objectives for TOE and environment. Only 26% of all the security objectives are new and 78.2% are from reuse of the reference PP.

The average of re-use of security assumptions, threats, policies and objectives is 67.4% and the average percentage of new security statements is 33.66%. This observation indicates that reusing security requirements has the potential minimising requirement development time hence saving cost.

However, the advantage of having high percentage of reuse depends on how the source PP environment, policy and assurance level correlate. If the source PP is meant for a completely different type of system, then the percentage of items that may be reused will be low.

6.6 Chapter Conclusion

This chapter dealt with the issue of developing security requirements and the issue of how to re-use security requirements. Results show that re-use of security requirements has the potential of minimising requirements development time and cost as a result making requirements development cost effective. The next chapter deals with the IS security assurance framework. This chapter combines the findings from chapters 3,4,5 and 6 to provide a coherent explanation how to address IS assurance while taking into consideration both technical and social aspects of IS.

Chapter 7

7 Framework for IS Security Assurance

In this chapter, we present a framework of thinking about IS systems security assurance. The purpose is to present information systems assurance as a support structure that can be useful when thinking and carrying out systems security assurance. The development of this structure has taken into considerations issues and challenges that we examined in previous chapters.

This chapter consists of three main sections, namely system life cycle, non-technical assurance issues and technical assurance issues. The life cycle section describes issues related to policy, design, and implementation and operation. The non-technical section describes non-technical issues and the technical assurance section describes technical aspects of IS security assurance.

A framework is a support structure or extensible structure for describing a set of concepts, methods, technologies, and cultural changes necessary for a complete product design and manufacturing process. Key components of this framework are three assurance aspects, namely: system life cycle, non-technical assurance factors and technical assurance factors. Figure 7-1 shows issues to consider prior to identifying assurance methods. The selection of assurance methods depends on the life cycle stage and whether one wants to deal with non-technical or technical issues.

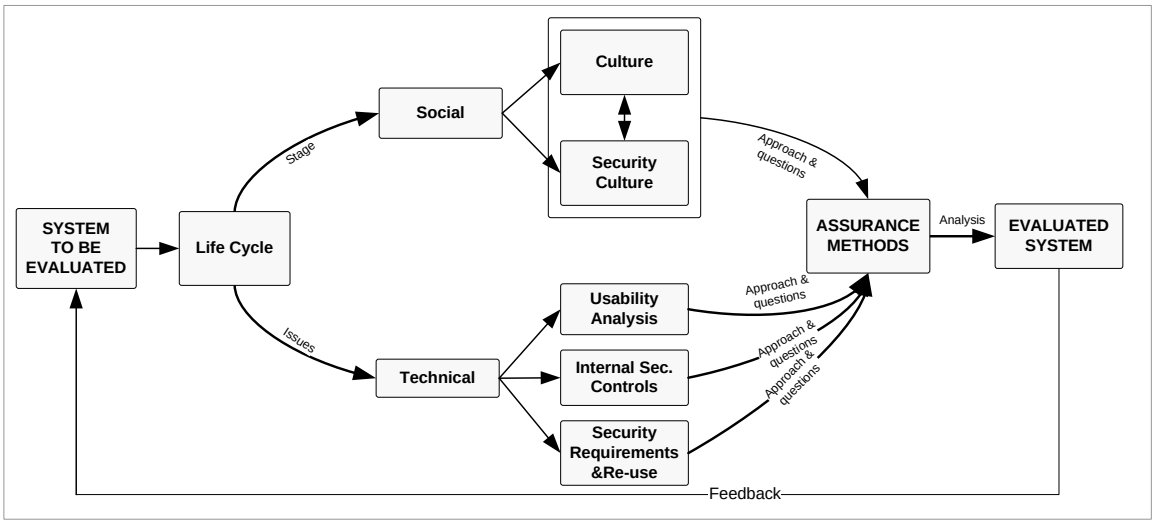


Figure 7-1: Framework for information systems security assurance

7.1 Assurance in the system life cycle

The life cycle of the system starts when the system is conceived for development or is procured and ends when the system is no longer used. A typical life cycle process is defined in stages namely policy, design, implementation and operational assurance, as depicted in Figure 7-2. Assurance in these stages involves establishing the evidence that the security requirements are met.

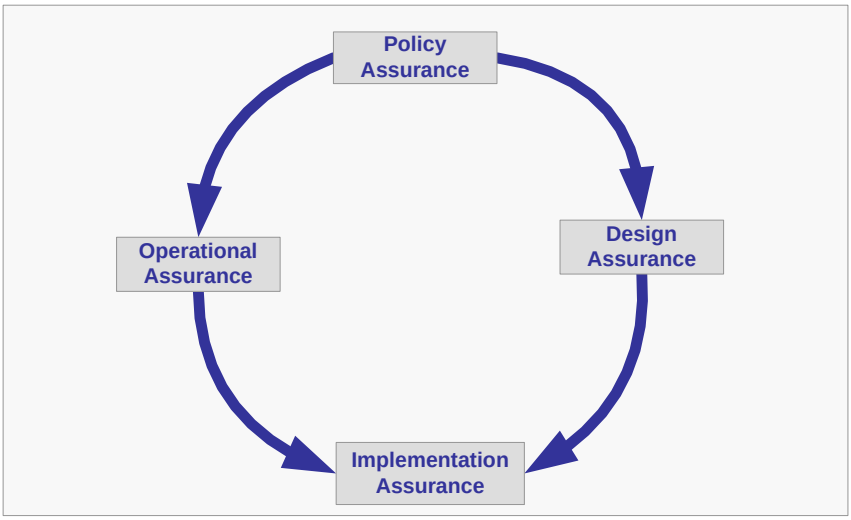


Figure 7-2: Assurance in the system's life cycle

7.1.1 Security policy assurance

A security policy is a high-level specification of the security properties that a given system should possess through out the life cycle. It is a means for designers, evaluators, implementers and auditors to communicate with each other. It is a blueprint to drive a project from design through implementation validation and operations Anderson (2001).

Policy assurance is a process of establishing that security requirements in the policy are complete and comply with the security requirements of the organisation. Successful security design, implementation and operational assurance depends on how policy is developed. Flaws in the policy may likely propagate down to the operational stage.

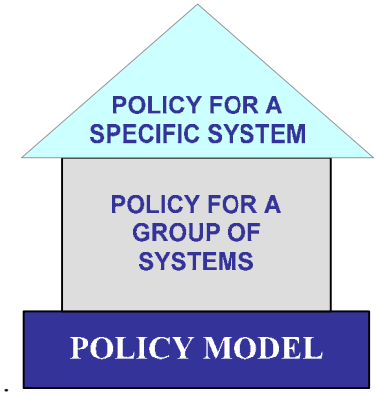


Figure 7-3: System's security policy levels

A good policy must explicitly state the protection mechanisms and how breaches are to be detected. To address this requirement, the Common Criteria standard introduces terms like protection profile and security target. A Protection profile is an implementation-

independent security requirement. It is implementation-independent to allow comparison of evaluated products and their versions. This policy may cover a group of related systems, such as TANESCO's PKI system.

A security target is a detailed description of what a specific implementation must provide. Detailed and implementation-independent security policies must be based on a security policy model. A security policy model usually is a brief description of the security properties that a system must provide.

From an organisational perspective, the policy model will be the organisation's security policy for IS systems. Policy for a group of systems includes systems such as operating systems and policy for a specific system could be policy for a specific type and version of operating system. In addition, security policy assurance involves establishing that the policy addresses issues related to regulations, laws and standards of IS security.

7.1.2 Design assurance

Design assurance is the process of establishing that the design meets the requirements of the security policy. Design stage assurance is necessary to ensure that security requirements are integrated into the system from the beginning. Too often, security is added after that the design is completed. Consequently, this costs more than including security at the design stage.

Assurance at the design stage should involve establishing that the following items are present at the design stage and comply with the security policy that defines the system:

- Security specifications are consistent with the policy and are complete
- Technical and operational security controls are consistent
- Security controls are testable and can be validated
- Security requirements documentation is consistent
- Personnel requirements are consistent

7.1.3 Implementation assurance

Implementation assurance is a process of establishing that the implementation is according to the security requirements of the security and design policy. At this stage, the assurance process involves procedures, tools, techniques and standards that are used to develop the system. Other aspects may include formation of review teams and processes, documentation guidelines, testing metrics, and version control.

Challenges of implementation assurance culminate from fast-track development methods such as prototyping. A prototype is a partially developed product that allows customers and developers to examine some aspects of the proposed system and decide if it is suitable for the end product. In general, prototypes are used to demonstrate concepts and try options (Sommerville, 2000). Rapid development is essential in the prototyping method. This has several disadvantages, namely:

- The quality of documentation is compromised due to the need for rapid development. As a result, the development process will be minimally visible to managers.
- Continuous change as the customer suggests improvements to the prototype has the potential of causing the software structure to be poor. As a result maintenance of poorly structured software tends to be difficult and expensive.

7.1.4 Operational assurance

Operational assurance is a process of establishing evidence that the security requirements of the system policy are not compromised during installation, configuration, patching and daily operations.

Patches must meet the same security requirements as the original product. Third-party extensions must also meet the same security requirements. Unfortunately, this is not always the case. Since systems we use are extensible, security flaws can easily be introduced into the system when third party software is installed. It is crucial to maintain the assurance level of the original development whenever one needs to install third-party extensions.

Daily operations aims at personnel background checks, training, contingency plans for turnover of employees who are responsible for security and proper use of metrics in all processes pertaining security.

7.2 Social (non-technical) Assurance Factors

For the past decades, IS security research placed more emphasis on technical aspects and security education, namely developing methodologies, tools and techniques. In this research non-technical assurance aspects are being addressed. Figure 7-4 depicts some of the non-technical factors namely: social, security culture, legal, economic and organisational processes.

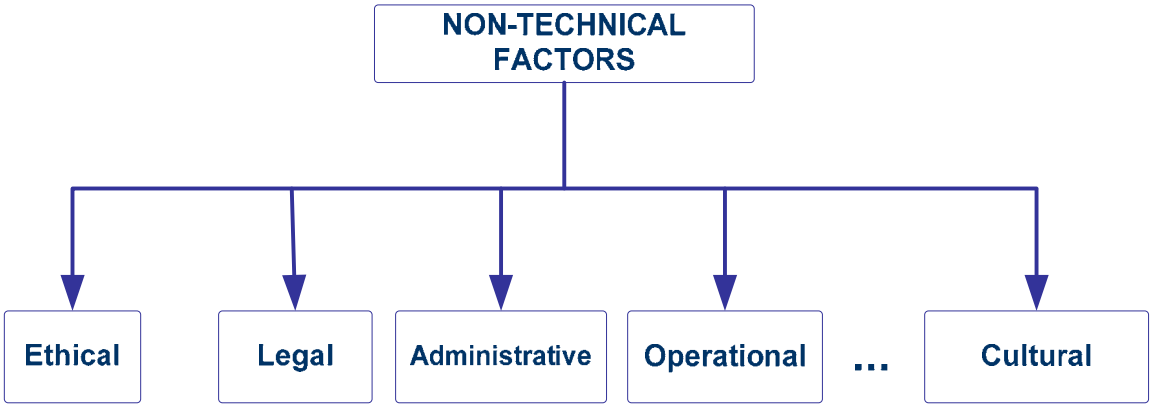


Figure 7-4: Non-technical security assurance factors

7.2.1 Cultural factors

Social factors are discussed in details in Chapter 3. These cultural dimensions define how different cultures are oriented towards behaviours and attitudes that define their culture. It is interesting to observe that back-up recovery plans for critical systems are not well managed. We argue that this is related to the short-term orientation of society.

Effective security management requires that organisations should be able to prevent, deter, detect and recover from attacks. In as much as we need to have long-term power supply alternatives, or plans to preserve the environment, we need future plans for the security of information systems. Steps such as keeping budget for security, contingency plans for recovery and continuous updating of the policy require proactive actions that need future orientation. Changing culture is difficult. However, in Chapter 3 we discussed how to develop a security culture. In this discussion, we argued that a security culture may be

developed within the organisation when developing its working culture. Table 7-1 shows hows the security dimensions that are related to insecurity.

Table 7-1 Security dimensions and their implications for IS security

Culture Dimension	Security Implication
Future orientation	**Significant**: Poor contingency plans such as disaster recovery plans
In-group collectivism	Not significant
Institutional collectivism	Not significant
Performance orientation	Not significant
Uncertainty avoidance	**Significant:** Lack of in-depth defence strategies
Power distance	**Significant:** Poor communication link between management and system operators and technicians or managers strictly controlling security
Humane orientation	Not significant
Assertive orientation	**Significant:**
Gender egalitarianism	**Significant:** Economic and opportunity egalitarianism has the potential of causing insecurities that are related to financial gain, such as fraud

Social culture frames the environment in which systems are used in relation to values, behaviour, freedom, constraints and self-interest.

7.2.2 Security culture

In Chapter 3 we examined security culture. We proposed dimensions that are useful to indicate the maturity level of the security culture in an organisation. We argued that security culture must be developed as an integral part of the organisational culture. According to Robbins (2005), organisational culture is unique in different companies and is developed depending on the company objectives and goals by developing employee behaviour based on the following dimensions: innovation and risk taking, attention to details, outcome orientation, people orientation, team orientation, aggressiveness and stability. These dimensions guide employee selection and socialisation process. Security maturity can be measured using the dimensions we have proposed in Chapter 6. These should help the security manager determine the security culture maturity level of the organisation.

7.2.3 Ethics factors

IS security ethics and legal factors play a major role in addressing security problems. IS ethical codes must clearly state which actions are ethical and which are not. If not properly addressed, ethics and legal factors has the potential of becoming a point of failure. For instance, if an employee commits fraud, there must be a legal framework that addresses IS-related cases. If such a framework is lacking, prosecuting people who commit illegal actions will not be effective.

7.2.4 Administrative and operational security processes

Organisational processes play a major role in the process of planning and implementing security mechanisms. One of the organisational responsibilities is resource allocation. It requires computer power to implement and execute the security mechanisms. It requires training of security personnel and other computer users on security tools and how to interpret and address security incidents, establishing metrics for measuring effectiveness of security mechanisms that are in place, and how to update and implement the non-

technical aspects of the security policy and establishing a clear communication line between management and security staff.

7.3 Technical Factors

Technical factors refer to methods, tools and techniques that can be used to ensure that information systems are secure. The choice of these artefacts depends on the type of the system that we need to secure. Some methods and tools are designed for critical systems and therefore they tend to be expensive and using them requires expert knowledge. In this section we present various methods and tools and discuss the environment in which they can be used. Figure 7-5 shows some of the technical factors that have to be taken into consideration in the security assurance process.

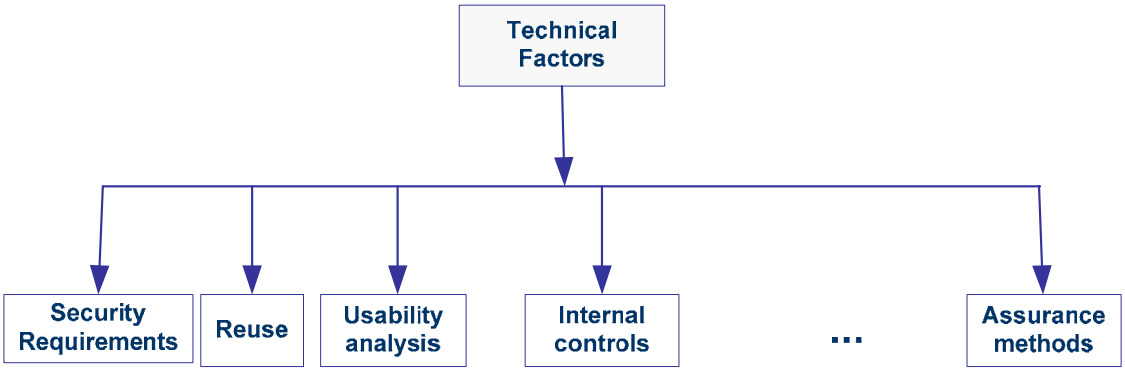


Figure 7-5: Technical security assurance factors

7.3.1 Security requirements

Developers use requirements to create system specifications. Specifications give a picture of how the system should function. In this view, it is necessary that specifications should be as formal as possible. In order to achieve this goal the use of guidelines in the process of requirements engineering is essential. Numerous guidelines are internationally accepted and used, such as the Common Criteria (CCIMB1, 2005). They provide a list of security functions and what security risks are to be considered. Such a list is not complete but it is comprehensive. Table 7-2 shows some of the assurance methods that are helpful in security requirements and specifications.

Table 7-2 Some of the security assurance methods

Standard	Description	Reference
CC	Product security assurance except security protocols. Covers areas such as security functions, assurance level, developing protection profiles, security targets. This standard evolved from the old standards namely: The orange book or TCSEC (1985), Information technology security evaluation criteria ITSEC (1991), the Canadian trusted computer products evaluation criteria CTCPEC (1992) and the US Federal criteria. It is equivalent to ISO/IEC 15408-1	(CCIMB1, 2005) (CCIMB2, 2005) (CCIMB3, 2005) (CCIMB4, 2005)
SSE-CMM	Security process assurance. Cover processes such as people, maturity level of the organisation	(SSE-CMM, 2003)
FIPS PUB 140-2	Security requirements for cryptographic modules.	(FIPS 1402, 2002)
ISO/IEC 15443-2	ISO/IEC 15443-2 - Information technology - Security techniques - A framework for IT security assurance - Part 2: Assurance methods	(ISO 15443-2, 2002)

ISO/IEC 15408-1	ISO/IEC 15408-1 Information technology - Security techniques - Evaluation criteria for IT security - Part 1: Introduction and general model	(ISO 15443-1, 2001)
PRISMA	NIST standard that aims to: assist agencies in improving their information security programs, support Critical Infrastructure Protection (CIP) Planning and facilitate exchange of effective security practices within the federal community.	(PRISMA, 2004)
COBIT	COBIT is an IT governance framework and supporting tool that allows managers to bridge the gap between control requirements, technical issues and business risks. COBIT enables clear policy development and good practice for IT control throughout organizations	(COBIT)
VNRM	The VNRM tool is a graphical tool for applying the Network Rating Methodology (NRM) for constructing comprehensive and convincing assurance arguments that an information system or component satisfies required security properties.	(VNRM, 1999)
OVAL	Open Vulnerability and Assessment Language (OVAL) is an information security community baseline standard for checking the presence of vulnerabilities and configuration problems on computer systems. It can collect system characteristics and configuration information; test the systems for the presence of vulnerabilities; configuration problems, and patches.	(OVAL, 2006)

7.3.2 Re-use of security artefacts

Re-use of security artefacts has the advantage that development cost and time is minimised when components are re-used. The challenge for re-use of security requirements is the change in the environment which system is used. The environment may be completely different and one environment may have fewer threats compared to the other. Therefore, when a product operates securely in its target environment, it does not imply that it will work securely in another. However, it is possible to re-use some of the security artefacts in a different environment than the target environment during development provided the new environment is re-evaluated correctly and its threats identified. This concept is examined in details in Chapter 6 when examining the use of CC to develop Protection Profiles (PP).

7.3.3 Assurance techniques and metrics

The Common Criteria (CCIMB3, 2005) and Software Engineering Capability Maturity Model (SSE-CMM) methodologies provide metrics for systems assurance levels and organisational maturity levels. Metrics in CC are named Assurance Evaluation Levels (AEL). AEL1 represents the lowest AEL7 represents the highest level of security assurance. AEL3+ or above may suffice for products that a government may use. SSE-CMM (SSE-CMM, 2003) provides maturity levels 1 up to level 5 metrics for evaluating organisational security maturity level. In practice, the maturity of various security processes is measured and eventually an estimate of organisation's security maturity level is established.

The use of methods such as the CC is expensive so it is not worth having every country conduct evaluations of the same product. Some of the motivating factors for the CC arrangements between countries is to eliminate or reduce duplicate evaluations of IT products and protection profiles, improve global market opportunities for the IT industry, encourage formal security testing of IT products, increase the availability of evaluated, security-enhanced IT products and protection profiles for national use. CCRA (2000) outlines rules and conditions for countries that seek membership.

7.3.4 Assurance tools

Assurance tools are designed to scan software or networks within a limited amount of time to identify flaws that may cause vulnerability and possibly suggest solutions for fixing the problem. Table 7-3 present some of these tools.

Table 7-3 Assurance tools

TOOL	APPLICATION AREA	HOW IT WORKS
RATS	Source code scanning	The Rough Auditing Tool for Security (RATS) is open source and capable of scanning C, C++, Perl, PHP and Python source code. It identifies flaws related to race conditions and buffer overflow and will suggest fixes (RATS, 2004).
ITS4	Source code scanning	Scans C and C++ source code for common flaws (RATS, 2004).
Flawfinder	Source code scanner	Open source software scanner. It is developed to scan C and C++ source code. Flawfinder works on Unix-like systems today (it's been tested on GNU/Linux) (FFINDER 2004)
SOAPscope	Database tool	Mindreef SOAPscope is a Web services diagnostics system for examining, debugging, testing, tuning and supporting Web services (MINDREEF, 2004)
AppDetective	Database tool	A network-based, vulnerability assessment scanner, appdetective discovers database applications within your infrastructure and assesses their security strength (APPDET, 2003).
Boomerang	Executable files ecompiler	A general, open source, retargetable decompiler of machine code programs, (Boomerang, 2005)
Fakebust	Executable code analyser	This is a one of the programmers tools for analysing executable files (FAKEBUST, 2004).
NESSUS	Network scanner	Public network scanners (NESSUS, 2006)
Enterprise Security Manager	Network scanner	The Enterprise security manager enables organizations to define, measure, and report on the compliance of information systems against security policies, standards and government regulations (SESM, 2005)
STAT	Network scanner	STAT is a security threat avoidance tool that focuses on providing vulnerability management solutions to organizations networks. STAT Scanner will also perform anonymous scans on target systems to which it cannot authenticate. Version 5 of STAT Common Criteria certification (EAL 2+) (STAT, 2003).

Table 7-3 does not present all the tools available for security assurance of information systems. There are many other similar tools, both commercial and non-commercial, that can help to identify flaws and possibly suggest solutions. Some of the weaknesses in assurance tools are:

- Few source code scanners tools are available for other programming languages other than C and C++

- Flaws that can be detected by the tool are limited to those that the tool is programmed to detect. New flaws that can be available in the source code under review may remain undetected
- Tools for network vulnerability assessment collect so much data that it might be difficult to analyse. In addition, false alarms may decrease the chance of successfully using such tools

7.4 Conclusion on Framework Issues

This chapter presents the framework for information systems assurance. The purpose of this chapter was to include all issues discussed in Chapters 3, 4, 5, and 6 into a structure that can be used for IS assurance. The framework includes cultural, technical and life cycle components that should be looked at prior to deciding on suitable methods and tools for assurance. In the next chapter conclusions and reflections on are presented.

Chapter 8

8 Conclusion and Reflections

This chapter presents conclusions and reflections of all the chapters. The purpose is to show how the overall purpose of research objectives have been achieved, what the contributions were and what can be suggested as further work. This chapter is divided into six sections; research purpose, research methods, research results, research quality, research contributions and further research.

8.1 Research Purpose we Attempted to Achieve

The purpose of this research was to examine social and technical aspects of information systems security assurance dimensions to be included into an information systems security assurance framework. In that attempt problems that relate to human behaviour, security culture, security usability, security internal controls, security requirements development and re-use of security requirements were addressed.

8.2 Methods, Techniques and Tools

Our research was mainly qualitative as its aims were to understand the technical and non-technical aspects of information systems security assurance. In this endeavour we chose to conduct a socio-technical analysis including the aspects of security culture, usability testing, specification of security requirements and re-use of such requirements, and the establishment and measuring of internal security controls. All analyses were made on our unit of analysis: the electricity prepayment system LUKU of TANESCO in Tanzania. The Common Criteria was extensively used, either directly or secondarily for this work. They were used to analyse systems security requirements including examining the possibilities of their re-use, and developing heuristics for performing the usability evaluation. They were also together with Systems Security Engineering Capability Maturity Model a large part of background information for the analyses of internal controls and security metrics. For the cultural studies, the GLOBE culture dimensions together with the Common Criteria were used as a starting point to examine the national, organisational and security culture through questionnaires.

8.3 Research Results and how we Addressed Research Questions

Questions we attempted to answer throughout the research were as follows:

1. How does culture affect/relate to IS security aspects?
2. How is security culture evaluated?
3. How do organisations develop a security culture?

4. Can re-use of security requirements save time and money?
5. How can usability problems affect the security of information systems?
6. How can usability be evaluated in a cost-effective way?
7. How can internal security controls effectiveness be measured?.

Figure 8-1 illustrates the social technical model components that were addressed in this thesis.

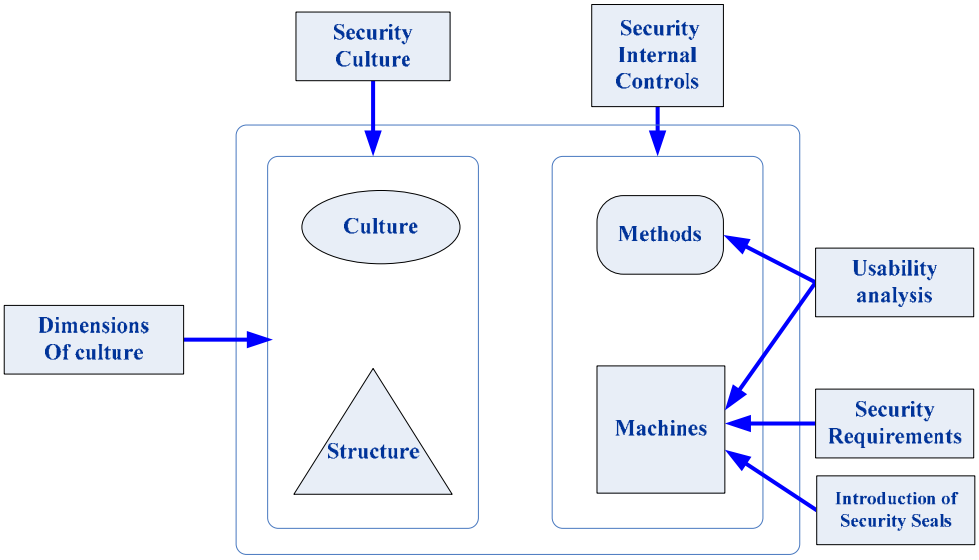


Figure 8-1 Information systems security assurance approach

Research questions 1-3 were answered through studies of the social subsystem "Culture" through performing evaluations of national, organisational and security culture in the Tanzanian company TANESCO. Out of nine national cultural dimensions, five were found to have implications for security. These were Uncertainty avoidance, Future orientation, Assertive orientation, Power distance and Gender egalitarianism. To create a security culture, eleven security dimensions were found; management commitment, awareness orientation, people orientation, risk orientation, information classification, assurance orientation, future orientation, uncertainty avoidance, standards adoption and compliance, attention to details and ethical orientation. As an aid for the analyses a heuristic visualisation technique was developed.

The research question 4 studied the technical subsystem "Machines". The average percentage of re-use of security assumptions, threats, policies and objectives using CC for a specific case in TANESCO was 67.4%. This observation indicates that reusing security requirements may have the potential to minimise requirement development time hence reducing cost. However, the advantage of having high percentage of reuse depends on how the source PP environment, policy and assurance level correlate. If the source PP is for a completely different type of system, then the percentage of items that may be reused will be significantly lower.

In addressing research questions 5-6 for the usability problems the technical subsystems "Methods" and "Machines" were studied from the point of view of the evaluators as users. This was done in order to reach an understanding of how to use the interface so that the evaluation can be cost-effective. Results indicate that usability not only is an interface problem but a systems engineering problem and a business problem, that usability is

dependant on the environment in which the system will be used, and that the number of evaluators needed for a cost and time effective evaluation depends on whether the system is a critical system or not. Finally it was concluded that usability evaluation reports should be requested by the buyer before purchasing a system.

Question number 7 addressed the sub system "Methods" using metrics. The result indicated that the existence of security metrics in a security programme shows that there is some level of maturity in the evaluation process, that there are security in depth practices for internal security processes within the organisation, that points of weaknesses easily can be identified, and that it is easy to justify a security programme in relation to budget and other resource commitments. Security metrics are further useful to show indicators such as increase or decrease in security incidents, coverage in security policy implementations, percentage of patched systems etc.

As a surprise – and practical result for TANESCO from the usability analyses even though some weaknesses in the interfaces such as poor error messages feedback, poor password management and poor security module management functions were found, it was not weaknesses of the security interfaces which were the source of fraud but a weakness in the physical security. As an additional result to questions 4, fraud problems at TANESCO significantly subsided after all computers were sealed so that no cover could be opened unless the seal is broken. As a special comment, this is added to Figure 8-1 as "Introduction of Security Seals" influencing the subsystem "Machines".

8.4 Research Quality

The quality of qualitative research can be judged by looking at credibility, transferability, dependability and confirmability as defined by Hoepfly (1997) and Lincoln & Gubas (1985) and we will here comment these issues specifically since our research mainly was qualitative.

8.4.1 Credibility

The creditability of qualitative research depends more on the analytical ability of the researcher and richness of gathered information than on the sample size of data (Patton, 1990). In this view, credibility of this work is maintained by asking the respondents to corroborate the findings and making raw data available for others to evaluate. The dimensions used also enhance credibility as they are used by the GLOBE project to analyse cultures across the globe.

8.4.2 Transferability

In qualitative research, the researcher cannot specify transferability of findings, but only provide sufficient information to the reader who can subsequently determine whether the findings are applicable to his or her environment (Lincoln & Gubas, 1985). Cultural values are different in different countries or regions of the world. Our focus was to improve assurance effectiveness through including social subsystems to the process of evaluating information system, our study showed how culture is related to security.

8.4.3 Dependability

Dependability and consistency are directly related. The consistence in our research is addressed through reviewing processes where by multiple reviewers reviewed the process

and the product of our research. This also involved reviewing questionnaires and final product of the research.

8.4.4 Confirmability

Confirmability refers to the degree that we can demonstrate the research interpretation neutrality using criteria such as outlined by Lincoln & Gubas (1985 pp. 320-321) where the research must provide audit trail data such as:

- Raw data
- Analysis notes
- Preliminary development and analysis information
- Processes notes, etc

In our research process the raw data, analysis notes and the preliminary research development were made available for review and discussions. As a result, in the preliminary research development questionnaires had to be reviewed. In this review, we had to translate one questionnaire to Swahili language in order to address the problem of understanding security terminology.

8.4.5 Rigor and relevance

We approached the problems by applying the socio-technical approach to analyse social and technical aspects of information systems security assurance. Our study covered issues of culture, security culture, usability analysis, requirements development, re-use of security requirements and analysis of internal security controls and its measurements. Throughout the research we argued that security culture must be an integral part of the assurance process and showed that an effective assurance process requires understanding human behavior and the organisational security culture.

The underlying assumptions about user behaviors when interacting with ISs are fundamental. Systems that are developed in a different environment may not work well in other environments simply because underlying assumptions are different. Assumptions about human behavior play a central role both in developing security policy, security requirements, internal security processes and in the system's life cycle. Other issues in our research included usability evaluations and re-use of security requirements.

8.5 Research Contributions

The contributions of this thesis to the general knowledge on information systems security assurance are mainly in the following areas:

The major contribution of this thesis is that firstly, it explains the implications of culture on IS security. In particular, it provides insights to how IS security assurance should be done in an effective manner using a socio-technical approach. In the Common Criteria, assumptions about the environment include the assumption about human behaviour such as user's trustworthiness and user's knowledge on security issues such as social engineering and password usage. These assumptions may be wrong if the culture of the organization that is using the evaluated product is not well known.

Human behaviour is traditionally defined, predicted and controlled to achieve objectives such as increase productivity, improve ethical conduct, maintain and improve satisfaction. In this research, we examined human behaviour to determine how it affects systems

security. In order to control human behaviour knowledge about culture is necessary. We showed that understanding human behaviour is central in addressing security problems related to technical and non-technical issues. Human factor is a popular term in system design, implementation, operation and training process. In all these aspects, we need to make assumptions about user behaviour. This can be done with a higher degree of certainty if the culture is known.

Secondly, re-use of security requirements as a means to reduce requirement development time and cost. This is an important issue because assurance is seen as being time consuming and costly. The process and methods we suggested (use of heuristics and questionnaires) can be useful for organisations looking at cost effective methods of conducting systems assurance

Thirdly, the framework of thinking is a contribution in that it provides a coherent explanation of how to approach information system security assurance, and how to ask the right questions. Contrary to the traditional way of looking at ISSA which addresses only technical issues, it is suggested in this framework that a more effective ISSA can be achieved by using a socio-technical approach.

8.6 Suggestions for Further Work

This research can be extended further by examining information system security assurance issues such as composition, measuring security culture maturity level, economics of IS assurance and modelling IS adversaries (Figure 8-2).

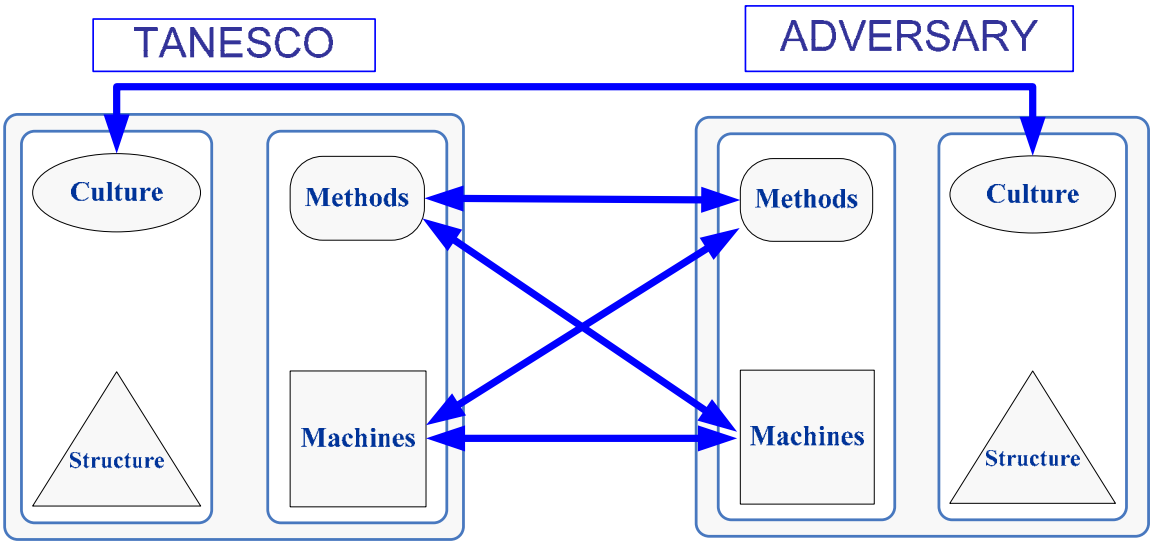


Figure 8-2 Extending effective ISSA through mirroring the Socio-technical approach on adversaries

Composition is problematic from two aspects. Firstly, when attempting to evaluate a system with subsystems having different levels of security requirements. Secondly, similar problems will exist if one organisation has higher security requirements than interoperable systems that belong to another organisation.

Furthermore, establishing security levels for security culture in organisations requires more research. In this thesis, we have addressed the issue of evaluating culture but the security culture maturity is out of scope. Establishing maturity level has the potential of

significantly improving organisational security culture because of the use of metrics which will clearly show maturity and act as a feedback to the ISSA process.

Further work is also required in the area of assurance economics. More specifically issues such as the role of business models in systems insecurities, how to estimate assurance costs and how to estimate loss due to lack/or poor security assurance of systems and processes as shown in Figure 8-2.

Finally, we propose using the socio-technical approach to analyse the interaction between TANESCO socio-technical system and adversaries' socio-technical system. This analysis should give insight in how to effectively deal with the current and future adversaries.

References

Adams, A. & Sasse, M.A. (1999) "Users are not the Enemy: Why Users Compromise Computer Security Mechanisms and how to Take Remedial Measures", *Communications of the ACM,* Vol. 42, No. 12, pp. 41 – 45.

Ammann, P. E. & Black, P. E. (2001) "A Specification-Based Coverage Metric to Evaluate Test Sets", *International Journal of Reliability, Quality and Safety Engineering*, Vol. 8, No. 4, pp. 275 – 300, Singapore: World Scientific Publishing.

Anderson, R. J. (2001) Security Engineering: A Guide to Building Dependable Distributed Systems, New York: John Wiley & Sons.

Anderson, R. J., (1994) "Why Cryptosystems fail", [online] http://www.ftp.cl. cam.ac.uk/ftp/users/rja14/wcf.pdf (May 2005).

APPDET (2003) "Database vulnerability assessment tool", [online] http://www.appsecinc.com/products/appdetective/ (March 2006).

Bell, D. E. & LaPadula, L. J. (1974) "Secure Computer Systems: A Refinement of the Mathematical Model", *Mitre,* TR-2547, Vol. 3, Mitre Corporation, Bedford, MA.

Bertalanffy, Ludwig Von (1968) General System Theory: Foundations, Development, Applications, revised edition, New York: George Braziller.

Biba, K .J. (1977) Integrity Considerations for Secure Computer Systems, ESD-TR-76-372, ESD/AFSC, Hanscom AFB, Bedford, Mass.

Bishop, M. & Armstrong, H. (2005) "Uncovering Assumptions in Information Security", *Proceedings of the Fourth World Conference on Information Security Education,* Moscow, Russia, pp. 223 – 231.

Bishop, M. (2002) Computer Security: Art and Science, Addison-Wesley Professional, 1st edition.

Boomerang (2005) "Program files decompiler", [online] http://boomerang. sourceforge.net/index.php (March, 2006).

Boulding, K. E. (1956) "General Systems Theory – The Skeleton of Science", *Management Science,* Vol. 2, pp. 197 – 208.

CCIMB1 (2005) Common Criteria for Information Technology Security Evaluation: Introduction and General Model, CCIMB-2005-08-001, Version 2.3, [online] http://www.commoncriteriaportal.org/public/files/ccpart1v2.3.pdf (November 2005)

CCIMB2 (2005) Common Criteria for Information Technology Security Evaluation: Security Functional Requirements, CCIMB-2005-08-002, Version 2.3, [online] http://www.commoncriteriaportal.org/public/files/ccpart2v2.3.pdf

(November 2005)

CCIMB3 (2005) Common Criteria for Information Technology Security Evaluation: Security Assurance Requirements, CCIMB-2005-08-003, Version 2.3, [online] http://www.commoncriteriaportal.org/public/files/ccpart3v2.3.pdf (November 2005)

CCIMB4 (2005) Common Evaluation Methodology for Information Technology Security Evaluation, Version 2.3, CCIMB-2005-08-003, [online] http://www.commoncriteriaportal.org/public/files/cemv2.3.pdf (November 2005)

Chaula J. A. (2003) Security Metrics and Public Key Infrastructure Interoperability Testing, Department of Computer and Systems Sciences (DSV), Report No. 2003:021, Stockholm University and Royal Institute of Technology, Sweden. [online] http://www.dsv.su.se/research/seclab/pages/pdf-files/03-021.pdf (January, 2004)

Chaula, J. A, Yngström, L. & Kowalski, S. (2004a) "Security Metrics and Evaluation of Information Systems Security", *Proceedings of the ISSA,* South Africa. [Online] http://icsa.cs.up.ac.za/issa/2004/Proceedings/Research/048.pdf (September 2006)

Chaula, J. A, Yngström, L. & Kowalski, S. (2005) "A Framework for Evaluation of Information Systems", *Proceedings of the ISSA'2005,* Johanesberg, South Africa, [online] http://icsa.cs.up.ac.za/issa/2005/Proceedings/Full/062_Article.pdf (September 2006)

Chaula, J. A, Yngström, L. & Kowalski, S. (2006) "Technology as a Tool for Fighting Poverty: How Culture in the Developing World Affect the Security of Information Systems", *Proceedings of the IEEE/TEDC,* July 10th, 2006, Iringa, Tanzania, pp. 27– 34

Chaula, J. A. & Yngström, L. (2003) "Public Key Infrastructure Security and Interoperability Testing and Evaluation", *Proceedings of the IITC,* Colombo, Sri Lanka, pp. 56 – 68.

Checkland, P. & Holwell, S. (1988) Information, Systems and Information Systems: Making Sense of the Field, Chichester: John Wiley & Sons.

Checkland, P. & Howell, S. (1998) "Software", IEE Proceedings-Software Engineering, *IEE Proceedings*, Vol. 145, Issue 4, pp. 95 – 99.

Checkland, P. (1999) Systems Thinking, Systems Practice, Chichester: John Wiley & Sons.

Clark, David D. and Wilson, David R.A. (1988), Evaluation of Model for Computer Integrity, in the 11th National Computer Security Conference, October 17-20, Baltimore, Maryland.

Clear, T. (2002) "Design and Usability in Security Systems: Daily Life as a Context of Use", *ACM SIGCSE Bulletin,* Vol. 34, Issue 4, pp. 13 – 14.

COBIT (2005) "Control Objectives for Information and Related Technology", [online]http://www.isaca.org/Content/NavigationMenu/Members_and_Leaders/COBIT6/C OBIT_Publications/COBIT_Components.htm (March 2006).

CORBA (2003) "Common Object Request Broker Architecture", [online] http://www.medinfo.rochester.edu/hl7/v3.0/mdf_08.htm (July 2005).

Cranor, L. F. & Garfinkel, S. (2004) "Secure or Usable?", *Security & Privacy Magazine,* IEEE, Vol. 2, Issue 5, Sept.-Oct., pp.16 – 18.

CTCPEC (1992) The Canadian Trusted Computer Product Evaluation Criteria (CTCPEC), Version 3.0, Canadian System Security Establishment, Government of Canada, April 1992, Draft.

Davis, D., Monrose, F. & Reiter, M. K, (2004) "On user Choice in Graphical Password Schemes", *Proceedings of the 13th USENIX Security Symposium,* San Diego, pp. 151-164.

DoD5-3600.1 (1997) Information Assurance: Legal, Regulatory, Policy and Organizational Considerations, Joint Chief of Staff, US department of defense, 3rd edition, September 1997.

Eloff, M. M. & Von Solms, S. H. (2001) "Information Security Management: An Approach to combine Process Certification, and product Certification'," *Computers & Security,* Vol. 19, No. 8, pp. 698 – 709.

eSecretariat (2001) "Proposal for Tanzania's ICT Policy Formulation", [online] http://www.ethinktanktz.org/esecretariat/ArchiveDoc.htm, (November 2003).

ETM (2003) Security Target for the Secure LOGIX Enterprise Telephone Management System Ver. 4.0.1, EWA-Canada, [online] http://www.cse.dnd.ca/en/services/ccs/ETM_v4.0.1_CPL_En.html, (May2005).

FAKEBUST (2004) "Executable Code Analyzer", [online] http://www.thesecure.net/blog/archives/000141.html (March, 2006).

FBCA (2002) "X.509 Certificate Policy, Federal Bridge Certification Authority", [online] http://www.cio.gov/fpkipa/documents/fbca_cp_09-10-02.pdf (November, 2003).

Finkelstein, L., Land, F., Carson, E. R. & Westcott, J. H. (1988) "Systems Theory and Systems Engineering Science", *Measurement & Technology,* IEE Proceedings, Vol. 135, Issue 6, pp. 401 – 406.

FIPS 140-2 (2002) "Security Requirements for Cryptographic Modules", [online] http://csrc.nist.gov/publications/fips/fips140-2/fips1402.pdf (May 2005).

FIPS PUB 180-1 (1993) "Federal Information Processing Standards Publication 180-1", [online] http://www.itl.nist.gov/fipspubs/fip180-1.htm, (October 2002).

Flawfinder (2004) "Source Code Scan Tool for C and C++ Code", [online] http://www.dwheeler.com/flawfinder (March 2006).

Gaunt, N. (2000) "Practical Approaches to Creating a Security Culture", *International Journal of Medical Information,* Vol. 60, No. 2, November, pp.151 – 157.

GLOBE (2003) "The GLOBE Research Program", [online] http://www.haskayne.ucalgary.ca/mg/GLOBE/public (June 2004).

Gollmann, D. (1999) Computer Security, John Wiley & Sons; First edition.

HAPI (2003) "High-Level Public-Key Based Cryptographic Services API", 2003, NIST, [online] http://csrc.nist.gov/pki/pkiapi/api.pdf (July 2003).

Herrmann, D. S. (2001) A Practical Guide to Security Engineering and Information Assurance, New York: Auerbach Publications.

Herrmann, D. S. (2003) Using the Common Criteria for IT Security Evaluation, ISBN 0-8493-1404-6, New York: Auerbach Publications.

Hoepfl, M. C. (1997) "Choosing Qualitative Research: A Primer for Technology Education Researchers", *Journal of Technology Education,* Vol. 9, No. 1, Fall 1997.

Housley, R. & Polk, T. (2001) Planning for PKI: Best Practice Guide for Deploying Public Key Infrastructure, New York: Wiley & Sons, 1st edition.

ISO/IEC DTR 15443-1 (2001) Information Technology - Security Techniques: A Framework for IT Security Assurance - Part 1: Overview and Framework.

ISO/IEC DTR 15443-2 (2002) Information Technology - Security Techniques: A Framework for IT Security Assurance - Part 2: Assurance Methods.

ITSEC (1991) Information Technology Security Evaluation Criteria (ITSEC): Provisional Harmonized Criteria, June 1991, Version 1.2, Luxembourg: Office of the Official Publications of the European Communities, 1991.

ITU-T (1993) Recommendation X.500 "Information Technology – Open Systems Interconnection – The Directory: Overview of Concepts", Models and Services, [online] http://www.dante.net/np/ds/osi.html (May 2002).

Jaafari, A. B. (2004) Reusability of the Smart Card Protection Profile for producing the Mobile Phone Digital Rights Management Protection Profile, Masters thesis, Department of Computer and Systems Sciences, KTH, Kista, Sweden.

Jelen, G. (2001) "SSE-CMM Security Metrics", [online] http://csrc. nist.gov/csspab/june13-15/jelen.pdf (May 2004).

Kahraman, E. (2005) Evaluating IT Security Performance with Quantifiable Metrics, Master's Thesis, Department of Computer and Systems Sciences, Stockholm University, Sweden.

Klein, H. K. & Myers, M. D. (1999) "A Set of Principles for Conducting and Evaluating Interpretive Field Studies in Information Systems," *MIS Quarterly,* Special Issue on Intensive Research, Vol. 23, No. 1, pp. 67 – 93.

Kowalski, S. (1994) IT Insecurity: A Multi-disciplinary Inquiry, PhD Thesis, Report No. 94-0004, ISBN91-7153207-2, Department of Computer and Systems Sciences, KTH, Kista, Sweden.

Liimatainen, S. (2005) "Usability of Decentralized Authorization Systems – A Comparative Study", *Proceedings of the 38th Hawaii International Conference on System Sciences*, pp.186 – 196.

Lincoln, Y. & Gubas, E. (1985) Naturalistic Enquiry, Beverly Hills, California: Sage Publications Inc.

Martins, A. & Eloff, J. (2001) Information Security Culture, Rand Afrikaans University, South Africa.

MINDREEF (2004) "Website Vulnerability Analysis Tool", [online] http://www.webservices.org/vendors/mindreef/mindreef_announces_soapscope_4_0_release, (March 2006).

Myers, M. D. (1997) "Qualitative research in Information Systems, MISQ Discovery", [online] http://www.qual.aucland.ac.nz (November 2005).

NESSUS, (2006) "Network Vulnerability Scanner", [online] http://www.nessus.org/. (March 2006).

NIAP (2000), "CC Toolkit, National Information Assurance Partnership", [online] http://niap.nist.gov/tools/cctool.html (June 2004).

NIAP (2005) "Protection Profiles, National Information Assurance Partnership", [online] http://niap.nist.gov/pp/index.html (June 2005).

NICT (2003) "National Information and Communication Technologies Policy, Ministry of Communication and Transport, Tanzania", [online] http://www.ethinktanktz.org/esecretariat/ArchiveDoc.htm (June 2003).

Nielsen, J. (1992) "Finding Usability Problems Through Heuristic Evaluation", *CHI'92 Proceedings of The ACM,* Monterey, CA, 3-7 May 1992, pp. 373 – 380.

NIST94a, (1994) "Advanced Authentication", [online] http://cs-www.ncsl.nist.gov/publications/nistpubs/800-10/node48.html(January 2004).

Norman, D. A. (1983) "Design Rules Based on Analyses of Human Error", *Communications of the ACM,* Vol. 26, No. 4, pp. 254 – 258.

OVAL (2006) "Open Vulnerability and Assessment Scheme", [online] http://oval.mitre.org/, (April 2006).

Passmark TestLog (2006) "Software tool for integrated management environment for test cases", Passmark TestLog, [online] http://www.testlog. com/products/testlog.htm, (March 2006).

Patton, M. Q., (1990) Qualitative Evaluation and Research Methods, 2nd edition, Newbury Park: Sage Publications.

Payne, D. (2003) "Engineering ethics and business ethics: Commonalities for a comprehensive code of ethics", *IEEE Region 5, 2003 Annual Technical Conference*, pp. 81 – 87.

PKIC (2001) EEMA, "Interoperability between PKI products", [online] https://www.eema.org/pki-challenge/files/description.pdf (August 2002).

PPVID (2004) "U.S Government Family of Protection Profiles for Public Key Enabled Application", Version 2.61, July 31, 2004, [online] http://niap.nist.gov/cc-scheme/pp/PP_VID3004a.pdf: (May 2005).

PRISMA (2003) "The NIST Program Review for Information Security Management Assistance (PRISMA"), [online] http://prisma.nist.gov/ (March 2006).

RATS (2004) "Rough Auditing Tool for Security", [online] http://www.securesoftware.com/resources/tools.html (March 2006).

RFC3280 (2002), X.509 Version 3 Certificates and CRL version 2, IETF. **[Online]** www.ietf.org/rfc/rfc3280.txt (June 2004)

Robbins, S. P. (2005) Essentials of Organisational Behaviour, New York: Prentice Hall, 8th edition.

Schneier, B. (2000) Secrets and Lies: Digital Security in a Networked World, New York: John Wiley & Sons.

SESM (2005) "Enterprise Security Manager", [online] http://www.symantec.com/Products/enterprise?c=prodinfo&refId=855,(March 2006).

Shneiderman, B. (1997) *Designing the User Interface*, New York: Addison-Wesley Pub. Co.

Skyttner, L. (1996) General Systems Theory, New York: Word Scientific Publishing Co., Pte. Ltd.

Sommerville, I. (2000) Software Engineering, 6th edition, New York: Addison-Wesley Pub. Co.

SSE-CMM (2003) "Systems Security Engineering Capability Maturity Model", (SSE-CMM Version 3) [online] http://www.sse-cmm.org/ model/ssecmmv2final.pdf (July 2003).

Stake, R. (1994) "Case studies", in Handbook of Qualitative Research, by Denzin, N. K. & Lincoln, Y. S. [eds.], California: Sage Publications.

Stal, D. (2000) "Security Targets", Entrust Technologies, [online] http://www. commoncriteria.org/stRpt/EntrustRA51.pdf (November 2003).

Stallings, W. (1999) Network Security Essentials – Applications and Standards, New York: Prentice Hall; 1st edition.

STAT (2003) "Network Vulnerability Scanner", [online] http://www. statonline.harris.com/solutions/vuln_assess/dvm.html, (July 2005).

SUMI (2004) "The Software Usability Measurement Inventory", [online] http://sumi.ucc.ie/whatis.html (June 2005).

Swanson, M., Bartol, N., Sabato, J., Hash, J. & Graffo, L. (2003) Security Metrics Guide for Information Technology Systems, [online]: http://csrc.nist.gov/csspab/june13-15/sec-metrics.html (September 2003).

SW-CMM (2003) CMM based Appraisals for Internal Process Improvement (CBA Ibis) and Software Process Assessments (SPAs), [online] http://www.sei.cmu.edu/sema/pdf/SW-CMM/2003apr.pdf (July 2003).

TAN (2005) "Tanzania Electricity Supply Company (TANESCO) website", [online] http://www.tanesco.com/ (November 2005).

TCSEC (1985) Department of Defense, Trusted Computer System Evaluation Criteria, (Orange Book), DOD 5200.28-STD.

Tools (2006) "Software and website test tools", [online] http://www.aptest.com/resources.html (May 2005).

USMC (2002) “Public Key-Enabled Application Family of Protection Profiles, US Marine Corp”, [online] http://niap.nist.gov/cc-scheme, (October 2005).

VNRM (1999) “Visual Network Rating Methodology, User’s Manual”, Moore, Strohmayer, “Visual NRM User’s Manual,” NRL Technical Memorandum 5540, Sep 1999, [online] http://chacs.nrl.navy.mil/Projects/VisualNRM/, (April 2006).

VPL (2006) “CC Scheme Validated products, 2006, Common Criteria Validation and Evaluation Scheme”, [online] http://niap.nist.gov/cc-scheme/ vpl/vpl_type.html#ids (January 2006).

Walsham, G. (1993) Interpreting Information Systems in Organizations, Chichester: John Wiley & Sons.

Webster, (2004) “Dictionary.com, Web based dictionary” [online] http://dictionary.reference.com/search?q=epistemology. (November 2005).

Westin, A. F. (1967) “Intrusions on Privacy: Self-revelation, Curiosity, and Surveillance”, in: Privacy and freedom, by Alan F. Westin [ed.], New York: Atheneum,

Wynekoop, J. L. (1992) “Strategies for Implementation Research: Combining Research Methods”, *13th International conference on Information systems,* pp. 185 – 193.

Wynekoop, J. L. & Russo, N. L. (1993) “System Development Methodologies: Unanswered Questions and the Research-Practice Gap”, *14th International Conference on Information Systems,* ICIS’1993, Orlando, Florida, pp. 181 – 190.

Wynekoop, J. L. & Russo, N. L. (1997) “Studying System Development Methodologies: An Examination of Research Methods” *Information Systems Journal,* Vol. 7, No. 1, pp. 47 – 66.

Yin, R. K. (2003) *Application of Case Study Research,* Second edition, California: Sage Publications Inc.

Yngström, L. (1996) A Systemic-Holistic Approach to Academic Programs in IT Security, *PhD Thesis*, Report No. 96:021, ISBN 91-7153-521-7 Department of Computer and Systems Sciences, KTH, Kista, Sweden.

Zurko, M., E. (1996) “User-Centered Security”, *Proceedings of the 1996 Workshop on New Security Paradigms,* Lake Arrowhead, California, United States, pp. 27-33.

Appendices

Appendix A: Acronym

APF	Application function
CA	Certification Authority
CAC	Common Access Card
CC	Common Criteria
CDU	Credit Dispensing Unit
CDU	Credit Dispensing Unit
CEM	Common Evaluation Methodology
CPV	Certification Path Validation
CRL	Certificate Revocation List
CRLDP	CRL Distribution Point
DH	Diffie Hellman
DISA	Defense Information Systems Agency
DN	Distinguished Name
DSA	Digital Signature Algorithm
EAL	Evaluation Assurance Level
ECDH	Elliptic Curve Diffie Hellman
EFS	Encrypted File System
EKU	Extended Key Usage
ERMS	Electricity Revenue Management System
FIPS	Federal Information Processing Standard
GMT	Greenwich Mean Time
HMAC	Hash based Message Authentication Code
IDS	Intrusion Detection System
IEC	International Electrotechnical Committee
IETF	Internet Engineering Task Force
ISO	International Organisation for Standards
IS	Information systems
ISS	Information systems security
ISSA	Information systems security assurance
IT	Information Technology
LSPKI	Large scale public key infrastructure
LUKU	Lipa Umeme Kadri Unavyotumia (Pay for electricity as you consume)
MMI	Man-Machine-Interface
NC	Name constraint
OCSP	On-line Certificate Status Protocol
OS	Operating System
PIN	Personal Identity Number
PKCS	Public Key Cryptography Standard

PKE	Public Key Enabled
PKI	Public Key Infrastructure
PKIPP	Public Key Infrastructure Protection Profile
PKIX	Public Key Infrastructure Working Group --IETF
PP	Protection Profile
PRW	Print Read Write
PST	Product's security function
RFC	Request for Comment
RSA	Rivest, Shamir, and Adelman
SCVP	Simple Certificate Validation Protocol
SFP	Security Function Policy
SM	Security metrics
SMS	Systems Master Station
SOF	Strength of Function
SSE-CMM	Systems Security Engineering Capability Maturity Model
ST	Security Target
TANESCO	Tanzania Electricity Supply Company
TCA	Top certification authority
TOE	Target Of Evaluation
TSC	TSF Scope of Control
TSF	TOE Security Function

Appendix B: Culture evaluation

1 Strongly Disagree	2 Disagree	3 Tend to Disagree	4 Undecided	5 Tend to Agree	6 Agree	7 Strongly Agree

Number 1-7 represent metrics that represent your answer to a particular statement as indicated below.

Mark your answer with an **X** in the block next to the answer that you select

	1	2	3	4	5	6	7
1. Followers are expected to obey their Leaders without question.							
2. Followers should be expected to obey their Leaders without question.							
3. Most people lead highly structured lives with few unexpected events.							
4. Most people should lead highly structured lives with few unexpected events.							
5. People are generally very tolerant of mistakes.							
6. People should be generally very tolerant of mistakes							
7. Leaders encourage group loyalty even if individual goals suffer.							
8. Leaders should encourage group loyalty even if individual goals suffer.							
9. In this society, children live with parents until they get married.							
10. In this society, children live with parents until they get married.							
11. People are generally dominant.							
12. People should be generally dominant.							
13. Boys are encouraged more than girls to attain a higher education.							
14. Boys should be encouraged more than girls to attain a higher education.							
15. More people live for the present than for the future.							
16. More people should live for the present than for the future.							
17. Students are encouraged to strive for continuously improved performance.							
18. Students should be encouraged to strive for continuously improved performance.							

Appendix C: Developing Security Culture

Instructions for the remainder of the questionnaire: Please mark your answer with an **X** in the block next to the answer that you select

STATEMENT	1	2	3	4	5
	Strongly Agree	Agree	Unsure	Disagree	Strongly Disagree
In this organisation when recruiting new employees ICT security related questions are included in the interviews questions.					
This organisation has a an information security policy.					
The information security policy reflects the organisation's business objectives.					
This organisation has annual budget for information security					
The organisation has an information security contingent plan.					
In this organisation, ICT procedures are implemented according to the information security policy.					
I can easily obtain a copy of the information security policy.					
The information security policy is updated regularly as needed					
The organisation ensures that I adhere to the information security policy.					
In this organisation, the management perceives information security as important.					
I received a formal training in information security.					
I think Internet is a secure network					
It is important to determine the organisation's information security needs.					
Information security should be regarded as a functional (business) issue.					
I know what the term information security implies.					
I know what the term information security assurance implies					
I think it is important to implement information security in the organisation.					
I am aware of information security responsibilities that are related to my job role.					
In this organisation new employees are briefed about security issues in the socialisation process					
My manager involves me in decisions that affect me.					
In this organisation background checking is always part of the selection process for new employees					
Management regards the privacy of information about employees as important					
Management enforce information access security policy for all job levels					
In this organisation risk analysis of information systems is performed regularly					
In this organisation assets are classified based on how critical they are					
In this organisation there is ICT risk assessment team					
In this organisation security incidence and emergence handling procedures available					
This organisation has a remote backup site					
In this organisation information systems security personnel follow more structured procedures when working with information systems.					
In this organisation, information security is measured by using defined security metrics					
In this organisation, there is a formal procedure shows how I should report information security incidents.					
The organisation use certified information systems products					
Investing in information security should be seen as a future investment and					
More people in this society live for the present than for the future.					
The organisation's information security measures comply with international standards.					
In this organisation have internal standards for ICT					
Information systems procedures are audited regularly					
This organisation encourages employees to exhibit analytical and attention to details abilities					
This organisation encourages employees to exhibit analytical and attention to details abilities					

Ethics dimension

Question	*Strongly Agree*	*Agree*	*Unsure*	*Disagree*	*Strongly Disagree*
I shall disclose to appropriate persons or authorities any actual or potential danger to information systems.					
I shall appropriately report any activity related to the profession that I believe to be unlawful, and I shall cooperate with any resulting investigation.					
I am familiar with the policy on Conflict of Interest.					
I am familiar with the policy on Professional Conduct.					
I am familiar with the policy on Code of Conduct for Officers and Administrators.					
I am familiar with the policy on reporting known and suspected fraud					
I am familiar with the policy on Confidential Information.					
In the course of my professional activities, I shall conduct myself in accordance with the highest standards of moral, ethical and legal behaviour.					
I shall not misuse information to which I become party in the course of my duties, and I shall maintain the confidentiality of all information in my possession that is so identified.					
If my project team member may be fired if I disclose unlawful or unethical act he/she committed then it is not fare to disclose it to the authority.					
I shall report any resistance to efforts to promote the understanding and acceptance of prudent information security measures throughout organisation					
I shall support efforts to promote the understanding and acceptance of prudent information security measures throughout organisation					
I approve software only if they have a well-founded belief that it is safe, meets specifications, passes appropriate tests					
I shall appropriately report any well-founded belief that the system is not safe, do not meets specifications, do not passes appropriate tests					
In my department members of staff are fair and avoid deception in all statements, particularly public ones, concerning software or related documents, methods and tools.					
I shall not knowingly use software that is obtained or retained either illegally or unethically.					
In my department limitations of experience and education are always disclosed honestly and forthrightly					
I will appropriately report any use of property of a client or employer in ways which are unauthorized, and without the clients or employer's knowledge and consent.					
I will keep private any confidential information gained in my professional work, where such confidentiality is consistent with the public interest and consistent with the law.					
In my department outside work is not accepted only if it is detrimental to the work for the primary employer					
I always Identify, define and address ethical, cultural, and legal issues related to work projects.					
I ensure that specifications for systems on which I work have been well documented, satisfy the users' requirements and have the appropriate approvals.					

Please mark your answer with an **X** in the block next to the answer that you select

Section 2: Biographical

What is your age?	
24 or younger	
25 to 35 years	
36 to 45 years	
46 to 54 years	
55 years or older	

Academic background

What is your highest qualification?	
Form IV or VI	
Graduate or Diploma	
Post Graduate	

Contacts

E-mail:	
Department:	
Position:	

Appendix D: Results for culture evaluation

This appendix contains data that was collected from TANESCO in the culture survey questionnaire.

Power Distance
Followers are expected to obey their Leaders without question.
Followers should be expected to obey their Leaders without question

Table D.1 Power Distance

Department/Role	Current	Should be
Technicians	3.75	3.44
Managers	2	1
Directors	3	3
Operators	2	1
Managers	5	3
Administrators	1	1
Developers	1.5	3
Analyst	2	1

Uncertainty avoidance
Most people lead highly structured lives with few unexpected events.
Most people should lead highly structured lives with few unexpected events.

Table D.2 **Uncertainty avoidance**

Department/Role	Current	Should
Technicians	3.56	4
Managers	2.5	6
Directors	5	5
Operators	7	2
Managers	3	5
Administrators	2	4
Developers	3.5	3
Analyst	1	7

Humane Orientation
People are generally very tolerant of mistakes
People should be generally very tolerant of mistakes

Table D.3

Department/Role	Current	Should
Technicians	4.19	4.13
Managers	5	2.5
Directors	3	5
Operators	2	6
managers	5	3
Administrators	4	2
Developers	4	4
Analyst	6	2

Institutional collectivism
People are generally very tolerant of mistakes.
People should be generally very tolerant of mistakes

Table D.4 **Institutional collectivism**

	Current	Should
Technicians	4.19	4.13
Managers	5	2.5
Directors	3	5
Operators	2	6
managers	5	3
Administrators	4	2
Developers	4	4
Analyst	6	2

In-group collectivism
In this society, children live with parents until they get married.
In this society, children live with parents until they get married.

Table D.5 **In-group collectivism**

Department/Role	Current	Should
Technicians	4.38	4.44
Managers	4.5	4
Directors	2	2
Operators	4	4
Managers	5	3
Administrators	6	6
Developers	4	2.33
Analysts	6	2

Assertive orientation
People are generally dominant
People should be general dominant

Table D.6 **Assertive orientation**

Department/Role		
	Current	Should
Technicians	4.19	4
Managers	5.5	3
Directors	3	3
Operators	5	7
managers	5	3
Administrators	5	2
Developers	3	3.5
Analysts	7	7

Gender Egalitarianism
Boys are encouraged more than girls to attain a higher education.
Boys should be encouraged more than girls to attain a higher education.

Table D.7 **Gender Egalitarianism**

Department/Role	Current	Should be
Technicians	4.44	1.75
Managers	6.5	1
Directors	2	2
Operators	6	1
Security Managers	6	3
Administrators	6	3
Developers	1.5	4.5
Analyst	7	1

Future Orientation

More people live for the present than for the future.
More people should live for the present than for the future.

Table D.8 **Future Orientation**

Department/Role	Current	Should be
Technicians	5.56	2.5
Managers	5.5	1.5
Directors	3	3
Operators	7	1
Security Managers	6	3
Administrators	6	2
Developers	3	2.5
Analyst	6	1

Performance orientation

Students are encouraged to strive for continuously improved performance.
Students should be encouraged to strive for continuously improved performance.

Table D.9 **Performance orientation**

Department/Role	Current	Should be
Technicians	6.25	6.06
Managers	6.5	7
Directors	6	6
Operators	6	7
Managers	6	5
Administrators	6	6
Developers	5	5
Analyst	7	7

Table D.10 Organisational culture

	As is	Should be
Continuous Improvement	6	6
People live for present than future	5	2
Boys encouraged than girls to attain higher education	5	2
People are generally dominant	5	4
Children live with parents until they are married	4	4
Institutional Collectivism	5	5
Leaders encourage group royalty than individual goals	3	4
Followers are expected to obey their leaders without questioning	3	2

Usability evaluation

Usability heuristics

- System impose password expiry
- Authentication errors messages clarity and usefulness
- Security module Installation dialogue adequate
- Energy encoding errors adequate
- Are the tariff management dialogue adequate
- The system shows rights a user inherits by joining groups
- Access rights dialogues are adequate
- User accounts creation, blocking and deleting
- Certificate request process
- System's time synchronisation
- Certificate revocation process
- Certificate validation errors
- Taking and verifying backup
- Disk transfer processes
- Security module installation and configuration

Usability analysis questionnaire

Question	1 Strongly Agree	2 Agree	3 Unsure	4 Disagr.	5 Strongly Disagree
The system indicated password metrics such as minimum size					
Error messages are clear and indicated where to obtain more information/solution					
I struggled to view user permissions I had assigned previously					
I know the algorithms present in the security module and the policy for certificates usage					
This error message enabled me to know what is wrong with the security module (ISBX Error. No return code. Try again: Yes/No)					
The system gave me option to set password expiry date					
I keep records of my passwords in a safe place so that I can refer in case I forget my password					
LUKU systems require administrators and managers to use more than one password in system's the routine administration.					
The authentication errors messages are understandable and useful					
The security module configuration and installation dialogue are adequate					
The security module installation is easy					
The electricity encoding errors are adequate					
Electricity encoding error provide information about how to solve the problem					
The tariff management dialogues are adequate					

The system shows the rights user inherit when join a group					
LUKU application shows user's access rights clearly					
User accounts can be blocked and deleted easily					
It is easy to obtain a certificate from the server					
My computer user local time does not pick time from the server					
There exists a clear procedure of certificate revocation					
Certificate validation errors are adequate					
LUKU application provide functionality for verifying backup					
The disk transfer data is encrypted					
Minimum effort is required to view LUKU audit data					
The audit data shows the link between events and users					
Audit data is protected					

Adequacy of error messages that are generated by LUKU application

Adequacy of error messages	
	Scores
Subject 1	4
Subject 2	4
Subject 3	5
Subject 4	3
Subject 5	4

Errors found by each evaluator in the usability experiment

S/N	Error found by each evaluator
Subject 1	15
Subject 2	17
Subject 3	19
Subject 4	23
Subject 5	14

LUKU application processes status indication

Process status indication	
	Scores
Subject 1	2
Subject 2	1
Subject 3	2
Subject 4	2
Subject 5	1

The system indicated password metrics such as minimum size

Process status indication	
	Scores
Subject 1	4
Subject 2	5
Subject 3	5
Subject 4	4
Subject 5	4

I struggled to view user permissions I had assigned previously

Process status indication	
	Scores
Subject 1	4
Subject 2	3
Subject 3	5
Subject 4	3
Subject 5	4

I know the algorithms present in the security module and the policy for certificates usage

Process status indication	
	Scores
Subject 1	5
Subject 2	4
Subject 3	5
Subject 4	5
Subject 5	4

This error message enabled me to know what is wrong with the security module
(ISBX Error. No return code. Try again: Yes/No)

Process status indication	
	Scores
Subject 1	4
Subject 2	5
Subject 3	5
Subject 4	4
Subject 5	3

The system gave me option to set password expiry date

Process status indication	
	Scores
Subject 1	1
Subject 2	1
Subject 3	1
Subject 4	1
Subject 5	1

I keep records of my passwords in a safe place so that I can refer in case I forget my password

Process status indication	
	Scores
Subject 1	1
Subject 2	4
Subject 3	4
Subject 4	1
Subject 5	1

LUKU systems require administrators and managers to use more than one password in system's the routine administration.

Process status indication	
	Scores
Subject 1	1
Subject 2	1
Subject 3	1
Subject 4	2
Subject 5	1

The authentication errors messages are understandable and useful

Process status indication	
	Scores
Subject 1	2
Subject 2	1
Subject 3	3
Subject 4	2
Subject 5	1

The security module configuration and installation dialogue are adequate

Process status indication	
	Scores
Subject 1	1
Subject 2	1
Subject 3	3
Subject 4	2
Subject 5	1

The security module installation is easy

Process status indication	
	Scores
Subject 1	1
Subject 2	1
Subject 3	1
Subject 4	1
Subject 5	1

The electricity encoding errors are adequate

Process status indication	
	Scores
Subject 1	1
Subject 2	1
Subject 3	1
Subject 4	2
Subject 5	1

Electricity encoding error provide information about how to solve the problem

Process status indication	
	Scores
Subject 1	2
Subject 2	1
Subject 3	1
Subject 4	2
Subject 5	1

The tariff management dialogues are adequate

Process status indication	
	Scores
Subject 1	3
Subject 2	1
Subject 3	2
Subject 4	2
Subject 5	1

The system shows the rights user inherit when join a group

Process status indication	
	Scores
Subject 1	1
Subject 2	2
Subject 3	2
Subject 4	2
Subject 5	1

LUKU application shows user's access rights clearly

Process status indication	
	Scores
Subject 1	2
Subject 2	1
Subject 3	1
Subject 4	2
Subject 5	1

User accounts can be blocked and deleted easily

Process status indication	
	Scores
Subject 1	1

Subject 2	1
Subject 3	2
Subject 4	2
Subject 5	1

It is easy to obtain a certificate from the server

Process status indication	
	Scores
Subject 1	4
Subject 2	4
Subject 3	3
Subject 4	5
Subject 5	5

My computer user local time does not pick time from the server

Process status indication	
	Scores
Subject 1	2
Subject 2	1
Subject 3	2
Subject 4	1
Subject 5	1

There exists a clear procedure of certificates revocation

Process status indication	
	Scores
Subject 1	4
Subject 2	5
Subject 3	4
Subject 4	5
Subject 5	5

Certificate validation errors are adequate

Process status indication	
	Scores
Subject 1	3
Subject 2	4
Subject 3	3
Subject 4	3
Subject 5	4

LUKU application provide functionality for verifying backup

Process status indication	
	Scores
Subject 1	1
Subject 2	1
Subject 3	2
Subject 4	1
Subject 5	1

The disk transfer data is encrypted

Process status indication	
	Scores
Subject 1	2
Subject 2	4
Subject 3	3
Subject 4	1
Subject 5	1

Minimum effort is required to view LUKU audit data

Process status indication	
	Scores
Subject 1	2
Subject 2	3
Subject 3	1
Subject 4	1
Subject 5	1

The audit data shows the link between events and users

Process status indication	
	Scores
Subject 1	1
Subject 2	1
Subject 3	2
Subject 4	1
Subject 5	1

Audit data is protected

Process status indication	
	Scores
Subject 1	1
Subject 2	2
Subject 3	1
Subject 4	2
Subject 5	2

Appendix E: Security requirement for TANESCO's PKI

Tanzania Electricity Supply Company (TANESCO) Public Key Infrastructure Protection Profile

Version 1.0

Date: September 2005

Prepared By: Job Asheri Chaula
Prepared For: Tanzania Electricity Supply Company

PKI PP

For TANESCO

Foreword

This document is the Protection Profile for TANESCO PKI, which is an enabling technology for implementing organisational PKI. It is the baseline for a formal security requirement. It fulfils the requirements of the Common Criteria version 2.1, the ISO standard 15408. The key components of this document are: PP Introduction, TOE description, TOE security environment, Security objectives, IT security requirements and Rationale.

Job Asheri Chaula (si-jac@dsv.su.se) developed this PP because of PhD project that was carried out at Tanzania Electricity Supply Company as a case study. The project is funded by SIDA and a result of cooperation between the Department of Computer and Systems Sciences-Stockholm University and the University of Dar es salaam. Comments for this PP should be send to si-jac@dsv.su.se

PP for TANESCO's Prepayment system

Table of Contents

Conventions and Terminology

In this section, we present definitions of terms unique to this document and which are frequently used.

Pre-Payment system (LUKU): Is a system which comprise the electricity dispensing unit, system master station for customer database management and the database. The Luku application is the prime interface to all security functionalities of the system including the PKI, energy coding and on line and off line data transfer.

Public Key Infrastructure (PKI): Public Key Infrastructure is a system that is used to communicate securely and with trust among entities in closed or open distributed computing environments.

Certification authority (CA): This is the top certification authority. All certificates for users in various departments will be issues from this central CA.

Registration authority (RA): This is the departments authority responsible with registering users(computers and persons) of the PKI system.

Certificate Revocation List (CRL): This is a list of revoked certificates for expiry, compromise and other reasons like terminated associations. Before use of a certificate, this list is checked to verify that the certificate at hand is still valid.

Public key certificate: A X.509 V3 certificate that binds the user public key to its identity

X.509 directory: A public directory as defined by X.509 standard contains public key certificates.

Security module: Hardware specifically designed to provide transactions security. It stores keys and cryptographic algorithms which are used to encode energy. It encodes energy for a specific customer by combining the key, currency amount, meter number and customer ID.

Introduction

The introduction section provides document management and overview information necessary to operate a protection profile registry. The Introduction provides background information that will enable the reader to gain a high-level understanding of the protection profile.

This protection profile is a result of research work conducted at Tanzania Electric Supply Company (TANESCO) in order to address electricity prepayment systems security requirements. The structure is established through the use of Common Criteria version 2.2

1.1 - Identification

Title: Tanzania Electricity Supply Company (TANESCO) Public Key Infrastructure Protection Profile.
PPAuthors: Job Asheri Chaula

Vetting Status: Draft

CC Version: 2.2 Final

General Status: Request for comments

Registration: <To be completed during registration>

Keywords: Public key infrastructure, certification authority, registration authority, security services, confidentiality, integrity, authentication, on-repudiation, repository, X.500 certificate, and PKI enabled application.

1.2 - Protection Profile Overview

This Protection Profile specifies the Information Technology (IT) security requirements for PKI enabled TANESCO Electricity Prepayment System application. The system comprises the credit-dispensing unit (CDU), System master Station, Electric Dispensers (meters). The same application software runs on the SMS and CDU. The security requirement in this PP applies only to the application software.

The target of evaluation is the application software for TANESCO Electric Prepayment System (EPS). This application is used to vend electricity and as a system master station. In order complete each vending transaction, some of the important functions are invoked operator authentication, security module, Printer/Reader/Writer (PRW), Customer identification by the account number or meter number, credit limit validation, Swipe card reader, coded token cancellation, transaction abortion, reissuing in case the token is damaged or lost, token verification, sell, Banking, token energy limit, issuing free token, tariff verification and meter tariff key change.

The System Master Station (SMS) includes the following main functions: Credit Dispensing Unit registration, operator's management, meter registration, Customer registration, disk transfer, reports and end of month processing

This PP does not include network connections, hardware on which the application software is installed and other software that might be installed together with the EPS.

1.3 – PP Document Organisation

Table 1-1 Summary of the components of the PP document

No	PP COMPONENT	DETAILS OF THE COMPONENT
1	**PP Introduction**	PP Identification PP Overview Document organisation Related PP and other documents
2	**TOE Description**	Coherent TOE description
3	**System security**	Assumptions regarding threats Assumptions regarding policies Threat addressed by the system Detailed attacks countered by the system Threats addressed by the system with support by the system Attacks countered by the environment Organisations general security policies of the system Organisations detailed policies assigned to the system Organisations general security policies for system and Organisational detailed security policies assigned to the
4	**Security Objectives**	Security objectives for the system Security objectives for the environment
5	**IT security**	Systems security functional requirements Security assurance requirements
6	**Rationale**	General threat and attack rationale Attack and security objectives correspondence Detailed policy and general policy mapping Detailed policy statement and security objective mapping Security objective security requirements rationale Security requirement dependence analysis

Section 6 provides a rationale to explicitly demonstrate that the information technology security objectives satisfy the policies and threats. Arguments are provided for the coverage of each policy and threat. The section then explains how the set of requirements are complete relative to the objectives, and that each security objective is addressed by one or more component requirements. Arguments are provided for the coverage of each objective. Next Section 6 provides a set of arguments that address dependency analysis, strength of function issues, and the internal consistency and mutual supportiveness of the protection profile requirements.

An acronym list is provided to define frequently used acronyms and a reference section is provided to identify background material.

1.4 - Related Protection Profiles –Related documents

This section provides a protection profile cross-references.

- International Standard ISO/IEC 15408 Information technology — Security techniques — Evaluation criteria for IT security
- X.509 Internet Public Key Infrastructure Certificate and CRL Profile, RFC 2459, January 1999
- International Organization for Standards/Internet Electro-technical Committee (ISO/IEC) 9594-8:"Information Technology- Open Systems Interconnection- The Directory: Public Key and Attribute Certificate Frameworks" (X.509 Standard)
- Common Methodology for Information Security Evaluation (CEM) Version 2.3, 2005
- Department of Defence Public Key-Enabled Application Family of Protection Profiles, Version 2.5, October 31, 2002
- Protection Profiles Reuse, Case study of reusability of the smart card Protection Profile for producing the Mobile Digital Rights Management Protection Profile, Version 1.0, February 2004

1.5 Common Criteria Conformance

This PP has been built with Common Criteria (CC) Version 2.2 (ISO/IEC 15408 Evaluation Criteria for Information Technology Security. CC is divided into Part 1: Introduction and general model, Part 2: Security functional requirements and Part 3: Security assurance requirements). The PP is at assurance level EAL 3 augmented.

2 - TOE Description

The TOE Description provides an overview of the TOE and details how the TOE functions. It provides a TOE description that enables the reader to:

- gain an understanding of how the system operates,
- know where the component fits into the system, and
- be able to define the TOE operation and its limitations. Be sure to include at least one figure that shows the relationship of the TOE elements or shows the relationship of the TOE to its environment.

Overview

A traditional electronic prepayment metering system operates on three levels. At the lowest level, are the meters, which are installed at the customer homes. The customer interfaces with the Electricity Dispenser (meter) with a token entry method, such as a magnetic card slot, or a keypad. Status indicators show token accept/reject, power available level with analogue bar-graph and/or a digit display, and the consumption rate with flashing light. The next level being the vending stations (Credit Dispensing Units (CDU)), which are placed at the utility's office or at appointed agents and are operated by CDU Operators. The CDU issues tokens and provides for first line administrative and financial control.

The communication between the vending stations and meters is in the form of a token which is used to top up the credit in the meter as well as to transfer or download information to the meter, and in some cases upload information (depending on the token choice) back to the vending station. The Token refers to the disposable magnetic stripe card (for Magnetic meters) or push-button twenty digit number (for Keypad meters) issued to the customer, as shown in figures 1, 2 and 3 above.

The magnetic stripe on the token or the twenty digit number (Keypad token) carries an encoded number which credits the customer's meter with the credit purchased. After use, the token is discarded. At the top level is the System Master Station (SMS) or Master Client, which is necessary to ensure a common database for reporting as well as to provide for total management, administration, financial and engineering control. The SMS communicates with the CDUs through online or offline means as indicated in Figure 5.1. Information on the consumers, tariff changes, etc. are communicated to the vending station and detailed customer sales are communicated back up to the SMS Smart et al (1995).

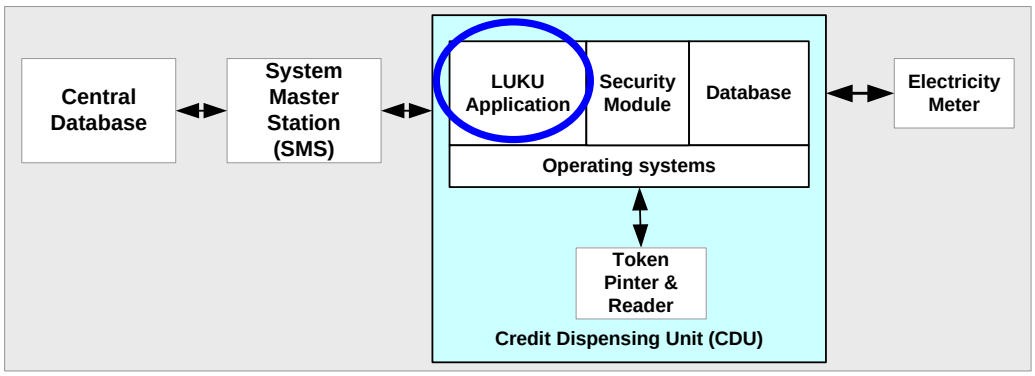


Figure 0-1 LUKU Application environment

LUKU application is PKI enabled if it meets the following requirements:

- Checks each certificate for validity, using procedures described in the X.509 standard RFC 3080 including checking for revocation information
- Has access to accurate and reliable time such as GMT in order to verify the dates on certificates, revocation data, and application data.
- Correctly interoperates with the security module
- Collects stores and maintains the data required to support digital signature verification in the future.
- Is able to automatically select from multiple private decryption keys if it performs public key based decryption.
- Supports secure cryptographic key management, trust anchors, and certificates.
- Uses one or more of the security services supported by the TANESCO PKI by accepting and processing a TANESCO's X.509 digital certificate.
- Is able to obtain relevant certificates and revocation data.

Public Key Infrastructure hierarchy

The certifications hierarchy is depicted in Figure 5.2 and includes the TOP certification authority, local certification authority and the end entity. Top CA in this case is the headquarters that will keep control of the entire system by controlling the certificate issuance process and keeping the revocation list. The local certification authority responsibility is to register all users in a particular region who are legible to receive certificates from the Top CA. The local CA plays the role of a registration authority and policy authority; apart from registration of users local CA will set policy for users. Users include master station operators, CDU operators, and backup operators. The roles for each of the roles will be defined in the certificate policy.

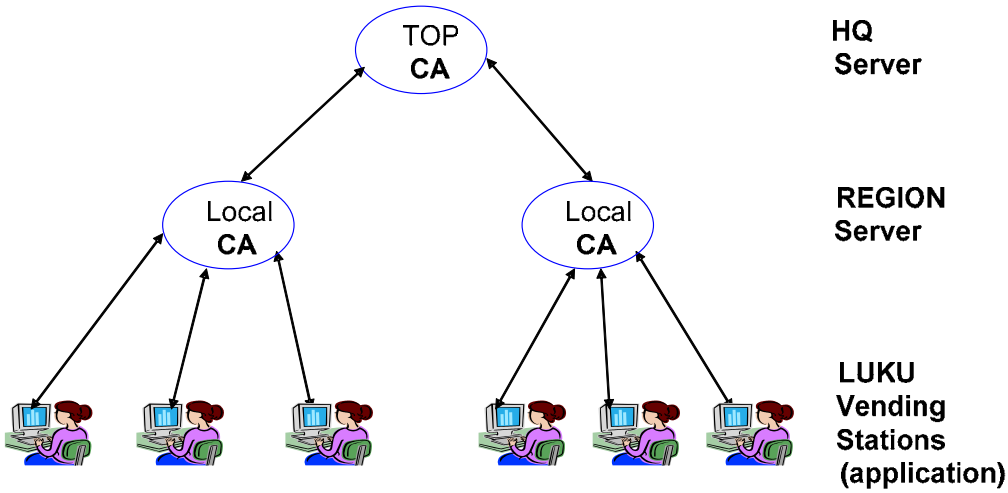


Figure 0-2 PKI Structure showing TOP and Local CAs and the End users

This PP can be applicable to other applications that are PKI enabled. An application is PKI enabled if:

- Securely manages keys, certificates and trusted anchors
- Accept and process X.509 certificates
- Is able to process certificates revocation data
- Is able to check each certificate for validity
- Can reliably validate certificate, revocation data and application data using time from reliable source.
- Support future verification of digital signature
- Bind users correctly to digital certificates
- Is able to interoperate with other PKI enabled application within the organisation
- Is able to handle a variety of public key mechanisms such as signature generation, signature verification, encryption, decryption and entity authentication.

Approach

This section provides information about the approach used to develop this PP. The underlying assumption is that the reader posses some knowledge about Common Criteria, Public Key Infrastructure, cryptographic algorithms and systems security requirements specification.

Approach for PKI requirements

X.509 certificate is defined in ITU-T (1993) Recommendation X.509 and further defined in Internet Engineering Task Force (IETF) Request for Comments RFC 3280. To ensure secure interoperation of PKI-enabled applications that makes use of X.509 certificates, the path validation must be done in accordance to these specifications.

X.509 is part of X.500 series of ITU-T recommendations that define a directory service. X.509 is based on the use of public key cryptography and digital signature. The X.509 certificate comprises the following attributes:

- **Version:** The default version is 1. If the unique identifier is available then the version must be 2. If one or more extensions are available the version must be 3.
- **Serial number:** Is unambiguous and unique integer value within the issuing CA that is associated with this certificate (RFC 3280). The default value for serial number must be 20 octets.
- **Signature algorithm identifier:** Identifies the algorithm used to sign the certificate.
- **Issuer name:** Is the name of the CA who signed this certificate
- **Period of validity:** This is the first and last date on which the certificate is valid. This is the period the CA vouches responsibility of maintaining the certificate.
- **Subject name:** Is the name of the user to whom this certificate refers.
- **Subject's public key information:** This is the public key of the subject, identifier of the algorithm together with any associated parameters.
- **Issuer unique identifier:** is a bit string field used to identify uniquely the issuing CA in the event the X.500 name has been reused for different entities.
- **Subject unique identifier:** is a bit string field used to identify uniquely the issuing CA in the event the X.500 name has been reused for different entities.
- **Signature:** It contains the hash code of the other fields, encrypted with the private key of the CA. This field includes the signature algorithm identifier.
- **Extensions:** A set of one or more extensions fields.

These were options added in X.509 version 3 in efforts to address the shortcomings of version 2. The extension fall into three categories namely Key and policy information, subject and issuer attributes, and certificate path constraints. The policy indicates the applicability of the certificate to a particular community. The certificate subject and issuer attributes extension supports alternative names, alternative name's format for the issuer and subject to increase certificate's user confidence. The additions includes information such as the postal address, respondents position within the corporation or picture image may be required (RFC 3280). The certificate path constraints are included in certificates issued for CAs by other CAs. This extension includes the basic constraint that indicates whether the subject can act as a CA, name constraint, and policy constraint.

This PP provides functional requirements for processing all of the certificate path and extensions. It provides the ability to select cryptographic algorithms from a range of available options. Some requirements such as audit data management may be met by the environment, which might include operating systems.

Certificate path validation and the CC concept of packages

The Certificate Path Validation involves the process of validating for validity of all certificates in the chain, that is, all certificates including the Top CA, intermediate CA and the end entity certificates. The path validation process also involves processing the extensions listed below. These extensions, in this document are also referred to as packages:

- Validity period
- signature generation
- signature verification
- name constraints
- certificate policy extension
- mapping policy
- Key transfer algorithms
- Certificate revocation list and
- Audit data management

Validity period is the time interval that the CA warrants that it will maintain information status of the certificate RFC3280 (2002). This interval is marked by validity beginning date validity end date. Time can be encoded in UTC or Generalized Time and the certificate using application must support UTC and Generalized Time encoding RFC3280.

Package concept

The approach used in writing this PP was to use the concept of packages. A package, as defined by the CC may be thought of as sets of defined functionality requirements. LUKU application is required to perform certain processes such as signing or verifying a signature, verifying names, mapping policy, verifying certificate revocation lists etc. Other processes may or may not be performed, depending upon the needs when performing transactions.

For completeness and clarity, each package that represents a discrete set of threats, objectives, and requirements is named and their corresponding threats, objectives and functional requirements are identified.

Table 2-1 – Summary of identified Packages

Package Name	Functionality	Dependency
Certification Path Validation (CPV) – Basic package	Perform all X.509 validation checks except policy processing and name constraints processing	None –this is the basic package
CPV – Basic Policy	Process certificate Policies extension	CPV – Basic
CPV – Policy Mapping	Process policy mapping related extensions: policyMapping, policyConstraints, and inhibitAnyPolicy	CPV – Basic, CPV – Basic Policy
CPV – Name	Process nameConstraints extension	CPV – Basic
PKI Signature Generation	Use private key for signature generation Generate the signature information (e.g., Public Key Cryptography Standard (PKCS 7))	None
PKI Signature Verification	Process the signature information (e.g., PKCS 7 blob) Use public key to verify signature	CPV – Basic
PKI Encryption using Key Transfer Algorithms	Generate the encryption envelope information (e.g., PKCS 7 blob) Use public key for encryption	CPV – Basic
PKI Encryption using Key Agreement Algorithms	Generate the key agreement envelope information (e.g., PKCS 7 blob) Use decryptor public key for key agreement Use encryptor private key for key agreement	CPV – Basic
PKI Decryption using Key Transfer Algorithms	Process encryption envelope information Use private key for decryption	None
PKI Decryption using Key Agreement Algorithms	Process the key agreement envelope information Use encryptor public key for key agreement Use decryptor private key for key agreement	CPV – Basic
PKI Based Entity Authentication	Carry out the assigned authentication protocol Use public key for authentication	CPV – Basic
Online Certificate Status Protocol Client	Generate OCSP request in accordance with RFC 2560 Process OCSP response	None
Certificate Revocation List (CRL) Validation	Obtain CRL Process CRL	None
Audit Management	Generate Audit Log Protect Audit Log Generate human readable audit reports	None
Continuous Authentication	Perform Continuous Authentication	PKI Based Entity Authentication, CPV - Basic

Certification Path Validation – Basic Package

The CPV-Basic –Basic Package provides validation checks for all X.509 except policy processing and name constraints processing. This package addresses the validation of the certification path. Certification path validation generally consists of validating certificates starting with the one certified by a trust anchor and ending with the one issued to the end user (i.e. the LUKU vending station or CDU). However, in order to be implementation neutral, this package

Trust anchors: These are self-signed certificates by the TOP CA that do not require any validation. The trust anchor certificate is generally included to obtain Top CAs Distinguished Name (DN), public key, algorithm identifier, and the public key parameters. End users certificate: This is the last certificate in the certification path and is issued to the credit dispensing machines,

Certification Path Validation – Basic Policy Package

The functionality in The Certification Path Validation – Basic Policy package is the processing of certificatePolicies extension. This package is dependent on the CPV –Basic package.

Certification Path Validation – Policy Mapping Package

The functionality in this package is the processing of validating certificate policies related extension namely policyMapping,inhibitAnyPolicy, and policyConstraints. The Certification Path Validation – Policy Mapping package is dependent on the CPV Basic Policy and the CPV – Basic packages.

Certification Path Validation – Name Constraints Package

The Certification Path Validation – Name Constraints is dependent on the Certificate Path Vvalidation – Basic package. The functionality in this package is the processing of nameConstraints extension.

Signature Generation Package

The PKI Signature Generation Package provides functionality to use the private key for signature generation and to generate the signature information. Signature information includes details such as e-mail address, name, location, date and time of signing.

Signature Verification Package

The PKI Signature Verification Package is dependent on the CPV – Basic package. This package provides functionality for processing the signature information and using the public key to verify a signature.

Encryption using Key Transfer Algorithms Package

The PKI Encryption using Key Transfer Algorithms package is dependent on the CPV – Basic package. The package provides functionality for performing public key encryption using key transfer algorithms.

Encryption using Key Agreement Algorithms Package

The Encryption using Key Agreement Algorithms Package is dependent on the CPV – Basic package and this package provides functionality to perform key encryption using key agreement algorithms.

Decryption using Key Transfer Algorithms Package

The PKI Decryption using Key Transfer Algorithms Package provides functionality to perform public key decryption using key transfer algorithms such as RSA. Since only the decrypting party's private key is used, this package is not dependant on the CVP-Basic package.

Decryption using Key Agreement Algorithms Package

The Decryption using Key Agreement Algorithms package is dependent on the CPV – Basic package. This process needs the processing of the certificate path verification.

Based Entity Authentication Package

The PKI Based Entity Authentication is dependent on the CPV – Basic package and allows PKI to be used for an entity authentication service and this package allows the select a PKI based entity authentication standard for identification and authentication of a remote entity such as the CDU or SMS.

Online Certificate Status Protocol Client Package

The Online Certificate Status Protocol (OCSP) allows clients such as CDUs to make online check of the certificate status in the CRL repository. The client package allows the TOE to make Online Certificate Status Protocol (OSCP) requests. and to validate OCSP responses. This package permits validates the OCSP responses and use the OCSP Responder as a trust anchor, as intermediate CA, or an end entity given authorisation to sign OCSP responses.

Certificate Revocation List (CRL) Validation Package

The Certificate Revocation List Validation Package allows the TOE to validate a CRL. This package permits the use of the same public key for CRL signature verification as the one used for verifying the signature on the certificate or can use a compliant implementation in which case a different certification path is developed. In such a case where a compliant implementation is used then the CPV – Basic package may be used in combination with this package.

Audit Management Package

The Audit Management package generates and protects audit events relevant to the TOE. Examples of audit events are: Management of trust anchors (addition, deletion), Identification and Authentication, Signature verification success, date and time, and policies under which signatures were valid, Signature verification failure, cause of failure (signature on the object failed, certification path failure, policy failure, etc.), time of failure, User override events (current CRL availability, accept policy failure, accept null policy, etc.).

Continuous Authentication Package

The Continuous Authentication Package is dependent on the CPV – Basic packages and PKI Based Entity authentication. This package is used for continuous authentication of the protocol, packets etc.

Assurance Requirements

The assurance level of this PP is EAL 3 with augmentation. The EAL 3 with augmentation PPs is selected because the TOE requires a moderate level of independently assured security and requires a thorough investigation of the TOE and its development without substantial re-engineering.

3. TOE Security Environment

In this section the security environment in which the TOE will be used and the manner the TOE will be employed is presented. The security environment defines the context in which the TOE is intended to be used as presented in Table 3-1. It includes the laws, threats that are present in the environment, organisational security policies, customs, expertise and knowledge that are relevant according to the CC General model. (CCIMB1, 2005)

3.1 - Secure Usage Assumptions

Table 3-1 Secure usage assumptions for the IT Environment

No	Assumption Name	Description	?
1	AE.Authorized_Users	Authorized users are trusted to perform their security assigned functions.	√
2	AE_Culture	Users cultural values that pose as risk to security are identified and associated behaviour are controlled and Users have access to the code of conduct.	x
3	AE.Configuration	The configuration and installation of the TOE is done properly.	√
4	AE.Crypto_Module	The TOE environment is assumed to include one or more cryptographic module(s) that are all validated at FIPS 140 series Level 1 or higher. All cryptographic modules in the TOE shall be validated at FIPS 140 series Level 1	√
5	AE.Medium	The attack potential on the TOE is assumed to be Medium.	X
6	AE.Physical_Protection	Physical protection is assumed to be provided by the environment. The TOE hardware and software is assumed to be protected from unauthorized physical access by outsiders and insiders. Insiders are monitored to prevent misuse of TOE and colluding with outsiders.	X
7	AE.PKI_Info	The certificate and certificate revocation information is available to the TOE.	√
8	AE.Time	Accurate system time with required precision in GMTformat is assumed to be provided by the environment.	√

9	AE.Hacker_Social_Eng	Users are aware about social engineering and mechanisms are in place to prevent social engineering	X

Threat to security for the TOE

This section defines security threats to TOE as presented in Table 3-2. The security threat agents include, but not limited to: 1) Failure of TOE and 2) People with access to TOE who have considerable expertise, poor resources, and uncontrolled cultural traits such as short term orientation, poor motivation and disgruntled personnel.

Attackers are assumed to have various levels of expertise, resources, and motivation. Attackers can either be insiders or outsiders. Relevant expertise may be in general semiconductor technology, software engineering, hacker techniques, or the specific TOE. Resources may range from personal computers and inexpensive card reading/coding devices to very expensive and sophisticated engineering test, measurement devices, and replica of TANESCO vending devices. They may also include software routines, some of which are readily available on the Internet. Motivation may include economic reward, resentment, or notoriety of defeating high-grade security. Given sufficient time and expertise, the prepaid electricity vending application software might be compromised.

Table 3-2 Threats to security for the TOE

No	Threat Name	Threat Description	?
1	T.Impersonation	An unauthorized individual may impersonate an authorized user of the TOE and thereby gain access to TOE secure data and functions.	√
2	T.Modification	An attacker may modify data, e.g., stored security attributes or private keys, in order to gain access to the TOE and its assets.	√
3	T.Object_Init	An attacker may gain unauthorized access to an object upon its creation	√
4	T.Attacker	Insiders or outsiders may attempt to perform actions that the individual is not authorized to perform without being detected.	X
5	T.Bypass	An unauthorized individual or user may bypass security attributes or other data in order to gain unauthorized access to TOE assets.	√
6	T.Private_key	An unauthorised individual may assume the identity of a user by generating or using the private key of the user.	√
7	T.Role	A user may assume more privileged role than permitted and use the enhanced privilege to take unauthorized actions.	√
8	T.Secure_Attributes	A user may be able to change the security attributes of an object and gain unauthorized access to the object.	√
9	T.Shoulder_Surf	An unauthorized user may read authentication credentials by look over the shoulder of the authorized user while authentication is in progress.	√
10	T.Tries	An unauthorized individual may guess the authentication information using trial and error method.	√
11	T.Component_Fail	An attacker exploits failure of one component resulting in loss of systems critical functionality	X
12	T.Load_Bad_data	An unauthorized individual may load bad data or software that could modify or expose data on the TOE	X
13	T.Clone	An unauthorised individual may clone the TOE to develop further attacks	X

TANESCO PKI TOE is required to counter threats that may be broadly categorized as threats addressed by TOE and threats that are directly related to the functioning of the TOE.
Threats addressed by the TOE:

- Threats associated with physical attack on the TOE
- Threats associated with logical attack on the TOE
- Threats associated with control of access
- Threats associated with unanticipated interactions
- Threats regarding cryptographic functions in the security module
- Threats that monitor information

Threats that are directly related to the functioning of the TOE such as threats to path validation, signature verification, policy mapping, revocation list verification, validity time verification etc.

Threats to security packages

This section defines security threats to TOE functions namely path validation for basic policy, name constraints, signature generation and verification, encryption and decryptions, online certificate status protocol, certificate revocation and audit data management. Table 3-3 presents additional threats to the basic package

Table 3-3 **Certificate Path validation (CPV) threats to basic packages**

No	Threat Name	Threat Description	?
1	T.Certificate_Modi	An untrusted user may modify a certificate resulting in using a wrong public key.	√
2	T.DOS_CPV_Basic	The revocation information or access to revocation information could be made unavailable, resulting in loss of system availability.	√
3	T.Expired_Certificate	An expired certificate could be used for signature verification.	√
4	T.Comp_Prvkey	The private key may be compromised and an attacker may use the certificate before certificate revocation is effected	X
5	T.Masquarade	An untrusted entity (Certification Authority (CA)) may issue certificates to bogus entities, permitting those entities to assume identity of other legitimate users.	√
6	T.No_Crypto	The user public key and related information may not be available to carry out the cryptographic function.	√
7	T.Path_Not_Found	A valid certification path is not found due to lack of system functionality.	√
8	T.Revoked_Certificate	A revoked certificate could be used as	√

		valid, resulting in security compromise.	
9	T.User_CA	A user could act as a CA, issuing unauthorized certificates.	√

Threats to certificate path validation (CPV) of basic policy

Threats to certificate path validation that result due to unknown policies are shown in Table 3-4

Table 3.4 – Threats for the CPV – Basic Policy

No	Threat Name	Threat Description	
1	T.Unknown_Policies	The user may not know the policies under which a certificate was issued.	
2	T.Cert_Policy	An attacker may change the certificate policy as a result the certificate can be used to perform prohibited functions	X

Threats to the policy mapping function

This section presents threats to policy mapping to users of certificate. These threats are presented in Table 3.5

Table 3.5 – Threats for the CPV – Policy Mapping Package

No	Threat Name	Threat Description	
1	T.Mapping	The user may accept unacceptable certificates or reject acceptable certificates due to improper certificate policy mapping.	
2	T.Wrong_Policy	The user may accept certificates that were not generated with the diligence and security acceptable to the user. The user may reject certificates that were generated with the diligence and security acceptable to the user.	

Threts for the CPV of name constraints

Threats for the certificate name constraint path validation are presented in Table 3-6

Table 3.6 – Threats for the CPV – Name Constraints

No	Threat Name	Threat Description	
1	T.Name_Collision	The user may accept certificates from CA where the CA's understanding and the user's understanding of the names differ, i.e., user and CA associate different identity with the same name.	
2	T.Name_Encoding	The application may fail to validate certificate if the name is encoded in a non compliant format	X

Threats to signature generation function

The following threats are defined for the PKI application Signature generation function.

Table 3.7 – Threats for the PKI Signature Generation Package

No	Threat Name	Threat Description	
1	T.Clueless_PKI_Sig	The user may try only inappropriate certificates for signature in absence of hint.	

Threat to signature verification function

The following threats are defined for the PKI application Signature verification function.

Table 3.8 – Threats for the PKI application Signature Verification

No	Threat Name	Threat Description	
1	T.Assumed_Identity_PKI_Ver	A user may assume the identity of another user in order to verify a PKI signature.	
2	T.Clueless_PKI_Ver	The user may try only inappropriate certificates for verification in absence of hint.	

Threat for encryption function using key transfer algorithms

The following threats are defined for the PKI application Encryption function using Key Transfer Algorithms

Table 3.9 – Threats for the PKI Encryption using key transfer algorithms

No	Threat Name	Threat Description	
1	T.Assumed_Identity_WO_En	A user may impersonate another user in order to perform encryption using Key Transfer algorithms.	
2	T.Clueless_WO_En	The user may try only inappropriate certificates for encryption using Key Transfer algorithms in absence of hint.	

Threat for encryption function using key agreement algorithms

The following threats are defined for the PKI application Encryption function using key agreement algorithms

Table 3.10 – Threats for the PKI Encryption using key agreement algorithms

No	Threat Name	Threat Description	
1	T.Assumed_Identity_With_En	A user may impersonate another user in order to perform encryption using Key Agreement algorithms.	
2	T.Clueless_With_En	The user may try only inappropriate certificates for encryption using Key Agreement algorithms in absence of hint.	

Threat for decryption function using key transfer algorithms

The following threats are defined for the PKI application decryption function using key agreement algorithms

Table 3.11 – Threats for the PKI Decryption using Key Transfer Algorithms

No	Threat Name	Threat Description	
1	T.Garble_WO_De	The user may not apply the correct key transfer algorithm or private key, resulting in garbled data.	

Threat for decryption function using key agreement algorithms

The following threats are defined for the PKI application decryption function using key agreement algorithms

Table 3.12 – Threats for the PKI Decryption using Key Agreement Algorithms

No	Threat Name	Threat Description	
1	T.Assumed_Identity_With_De	A user may assume the identity of another user decrypting using Key Agreement algorithms.	
2	T.Clueless_With_De	The user may try only inappropriate certificates for decryption using Key agreement algorithms in absence of hint.	
3	T.Garble_With_De	The user may not apply the correct key agreement algorithm or private key, resulting in garbled data.	

Threat for the PKI entity authentication

Table 3-13 shows threats for PKI entity authentication process

Table 3.13 – Threats for the PKI application Entity Authentication package

No	Threat Name	Threat Description	
1	T.Assumed_Identity_Auth	A user may assume the identity of another user to perform entity based authentication.	
2	T.Replay_Entity	An unauthorized user may replay valid entity authentication data.	

Threats for the OCSP

Threats for the online certificate status protocol client are presented in Table 3-14

Table 3.14 – Threats for the OCSP Client function

No	Threat Name	Threat Description	
1	T.DOS_OCSP	The OCSP response or access to the OCSP response could be made unavailable, resulting in loss of system availability. This may result due to adversary act or protocol overloading.	
2	T.Replay_OCSP_Info	The user may accept an old OCSP response resulting in accepting a currently revoked certificate.	

3	T.Wrong_OCSP_Info	The user may accept a revoked certificate or reject a valid certificate due to a wrong OCSP response.	

Threats for CRL

Threats to the certificate revocation list validation are presented in Table 3-15.

Table 3.15 – Threats for the Certificate Revocation List Validation

No	Threat Name	Threat Description	
1	T.DOS_CRL	The CRL or access to CRL could be made unavailable, resulting in loss of system availability.	
2	T.Replay_Revoc_Info_CRL	The user may accept an old CRL resulting in accepting a currently revoked certificate.	
3	T.Wrong_Revoc_Info_CRL	The user may accept a revoked certificate or reject a valid certificate due to a wrong CRL.	
4	T.Malicious_CA	A malicious CA may sign CRL	X
5	T.Issuer_Alternativename	The user may accept a wrong certificate due to certificate issuer alternative name	X

Threats for audit management

Audit data management threats are presented in Table 3-16

Table 3.16 – Threats for the Audit Management functions

No	Threat Name	Threat Description	
1	T.Accountability	The security relevant audit events cannot be linked to individual	
2	T.Audit_Excess	The security audit log has excessive data for analysis.	
3	T.Audit_Fill	The security audit log gets filled too fast to be of practical use.	
4	T.Audit_Modify	The accuracy of the security audit log cannot be trusted since unauthorized modification may have been made.	
5	T.Audit_Unreadable	The audit log cannot be read and interpreted by human beings and hence security relevant events cannot be investigated.	
6	T.No_Audit	There is no audit log to investigate security relevant events.	

Threats to continuous authentication

Threats to continuous authentication of entities are outlined in Table 3-17

Table 3.17 – Threats for the Continuous Authentication functions

No	Threat Name	Threat Description	
1	T.Hijack	An unauthorized user may hijack an authenticated session.	

Organisational Security Policies

This section identifies and defines organisational security policies. Also identify organisational security policy statements or rules with which the TOE must comply.

- Protection mechanisms shall be applied such that the TOE maintains the appropriate level of confidentiality, integrity, authentication, and non-repudiation based on mission criticality, sensitivity of information handled by the system.

- Digital Signature Standard keys shall use at least 160 bit private key and at least 1024 bit prime modulus. Minimum public key size shall be 1024 bits for Key Exchange Algorithm. Minimum public key size shall be 2048 bits for RSA.

- All security algorithms must be evaluated using FIPS140-2 at least level 1.

4 - Security Objectives

This section identifies and defines the security objectives for the TOE and its environment. Security objectives should reflect the stated intent, be suitable to counter all identified threats, and cover all identified organisational security policies and assumptions.

4.1 - Security Objectives for the TOE

Security Objectives for TOE are defined in Table 4-1, below. TOE security objectives may be met by the environment; in that case (OE) prefix is used. O prefix is used for security objective for TOE.

Table 4.1 – Security Objectives for the TOE

No	Objective Name	Objective Description	
1	O.DAC	The TSF shall control and restrict user access to the TOE assets in accordance with a specified access control policy.	
2	O.I&A	The TSF shall uniquely identify all users, and shall authenticate the claimed identify before granting a user access to the TOE facilities.	
3	O.Init_Secure_Attr	The TSF shall provide valid default security attributes when an object is initialized.	
4	O.Invoke	The TSF shall be invoked for all actions.	
5	O.Limit_Actions_Auth	The TSF shall restrict the actions a user may perform before the TSF verifies the identity of the user.	
6	O.Limit_Tries	The TSF shall restrict the number of consecutive unsuccessful authentication attempts.	
7	O.No_Covert	The TSF shall not permit any covert channel that may disclose authentication information.	X
8	O.Protect_I&A_Data	The TSF shall permit only authorized users to change the I&A data.	

9	O.Secure_Attributes	The TSF shall permit only the authorized users to change the security attributes.	
10	O.Security_Roles	The TSF shall maintain security-relevant roles and association of users with those roles.	
11	O.Self_Protect	The TSF shall maintain a domain for its own execution that protects it and its resources from external interference, tampering, or unauthorized disclosure.	
12	O.Trust_Anchor	The TSF shall permit only authorized users to manage the trust anchors.	
13	O.TSF_Data	The TSF shall permit only authorized users to modify the TSF data.	
14	O.Audit	TOE must provide means of recording selected security-relevant information	X

4.2 - Security Objectives for the Environment

Table 4-2 lists security objectives for the environment. This section clearly states and traces security objectives for the environment back to aspects of identified threats not completely countered by the TOE and/or organisational security policies or assumptions not completely met by the TOE

Table 4-2 – Security Objectives for the Environment

No	Objective Name	Objective Description	
1	OE.Authorized_Users	Authorized users are undergo training, background check and trusted to perform their authorized tasks.	X
2	OE.Configuration	The TOE shall be installed and configured in secure manner.	
3	OE.Crypto	The environment shall include one or more cryptographic (modules) that are all validated at FIPS 140 series Level 1 or higher.	
4	OE.Low	The identification and authentication functions in the TOE shall be designed and implemented for a minimum attack potential of low as validated by the vulnerability assessment and strength of function analysis.	
5	OE.Physical_Security	The environment shall provide an acceptable level of physical security so that the TOE cannot be physically tampered with.	
5	OE.PKI_Info	The IT environment shall provide the TOE certificate and certificate revocation information.	
6	OE.Time	The environment shall provide access to accurate current time with required precision, translated to GMT	
7	OE. ulture	Users cultural values that pose as risk to security are identified and associated behaviour are controlled and Users have	X

		access to the code of conduct.	
8	OE.Store	The environment shall provide secure storage area of the private keys	X

4.3 Security objective for the packages

The following subsection defines the security objectives for each security package**.**

4.3.1 Certification Path Validation – Basic Package

This section defines security objectives for the Certification Path Validation – Basic package

Table 4.3 – Security Objectives for CPV – Basic Package

No	**Objective Name**	**Objective Description**	
1	O.Availability	The TSF shall continue to provide security services even if revocation information is not available.	
2	O.Correct_Time	The TSF shall provide accurate temporal validation results.	
3	O.Current_Certificate	The TSF shall only accept certificates that are used within validity period.	
4	O.Get_KeyInfo	The TSF shall provide the user public key and related information in order to carry out cryptographic functions.	
5	O.Path_Find	The TSF shall be able to find a certification path from a trust anchor to the subscriber.	
6	O.Trusted_Keys	The TSF shall use trusted public keys in certification path validation.	
7	O.User	The TSF shall only accept certificates issued by a CA.	
8	O.Verified_Certificate	The TSF shall only accept certificates with verifiable signatures.	
9	O.Valid_Certificate	The TSF shall use certificates that are valid.	
10	T.Comp_Prvkey	The TSF shall use secure store of the private key	X

4.3.2 Certification Path Validation – Basic Policy Package

This section presents security objective for the Certification Path Validation – Basic Policy package.

Table 4.4 – Security Objectives for CPV – Basic Policy Package

No	**Objective Name**	**Objective Description**	
1	O.Provide_Policy_Info	The TSF shall provide certificate policies for which the certification path is valid.	
2	T.ProtectCert_Policy	The TSF shall restrict unauthorised users from modifying certificate policy	X

4.3.3 Certification Path Validation – Policy Mapping Package

The following security objectives are defined for the Certification Path Validation – Policy Mapping package.

Table 4.5 – Security Objectives for CPV – Policy Mapping Package

#	Objective Name	Objective Description	
1	O.Map_Policies	The TSF shall map certificate policies in accordance with user and CA constraints.	
2	O.Policy_Enforce	The TSF shall validate a certification path in accordance with certificate policies acceptable to the user.	

4.3.4 Certification Path Validation – Name Constraints Package

The following security objective is defined for the Certification Path Validation – Name Constraints package.

Table 4.6 – Security Objectives for CPV – Name Constraints Package

#	Objective Name	Objective Description
1	O.Authorised_Names	The TSF shall validate a certificate only if the CA is authorized to issue a certificate to the subject.

4.3.5 PKI Signature Generation Package

The following security objective is defined for the PKI Signature Generation package.

Table 4.7 – Security Objectives for PKI Signature Generation Package

#	Objective Name	Objective Description
1	O.Give_Sig_Hints	The TSF shall provide hints for selecting correct certificates for signature verification.

4.3.6 PKI Signature Verification Package

The following security objectives are defined for the PKI Signature Verification package.

Table 4.8 – Security Objectives for PKI Signature Verification Package

#	Objective Name	Objective Description
1	O.Use_Sig_Hints	The TSF shall use hints for selecting correct certificates for signature verification.
2	O.Linkage_Sig_Ver	The TSF shall use the correct user public key for signature verification.

4.3.7 PKI Encryption using Key Transfer Algorithms Package

The following security objectives are defined for the PKI Encryption using Key Transfer Algorithms package.

Table 4.9 – Security Objectives for PKI Encryption using Key Transfer Algorithms Package

#	Objective Name	Objective Description
1	O.Hints_Enc_WO	The TSF shall provide hints for selecting correct certificates or keys for PKI Encryption using Key Transfer Algorithms.
2	O.Linkage_Enc_WO	The TSF shall use the correct user public key for key transfer.

4.3.8 PKI Encryption using Key Agreement Algorithms Package

The following security objectives are defined for the PKI Encryption using Key Agreement Algorithms package.

Table 4.10 – Security Objectives for PKI Encryption using Key Agreement Algorithms Package

#	Objective Name	Objective Description
1	O.Hints_Enc_W	The TSF shall provide hints for selecting correct certificates or keys for PKI encryption using Key Agreement algorithms.
2	O.Linkage_Enc_W	The TSF shall use the correct user public key for key agreement during encryption.

4.3.9 PKI Decryption using Key Transfer Algorithms Package

The following security objectives are defined for the PKI Decryption using Key Transfer Algorithms package.

Table 4.11 – Security Objectives for PKI Decryption using Key Transfer Algorithms Package

#	Objective Name	Objective Description
1	O.Correct_KT	The TSF shall use appropriate private key and key transfer algorithm.

4.3.10 PKI Decryption using Key Agreement Algorithms Package

The following security objectives are defined for the PKI Decryption using Key Agreement Algorithms package.

Table 4.12 – Security Objectives for PKI Decryption using Key Agreement Algorithms Package

#	Objective Name	Objective Description
1	O.Hints_Dec_W	The TSF shall provide hints for selecting correct certificates or keys for PKI decryption using Key Agreement algorithms.
2	O.Linkage_Dec_W	The TSF shall use the correct user public key for key agreement during decryption.
3	O.Correct_KA	The TSF shall use appropriate private key and key agreement algorithm.

4.3.11 PKI Based Entity Authentication Package

The following security objectives are defined for the PKI Based Entity Authentication package.

Table 4.13 – Security Objectives for PKI Based Entity Authentication Package

#	Objective Name	Objective Description
1	O.I&A_Remote	The TSF shall uniquely identify all remote entities, and shall authenticate the claimed identify before granting a remote entity access to the TOE facilities.

2	O.Limit_Actions_Auth_Remote	The TSF shall restrict the actions a remote entity may perform before the TSF verifies the identity of the remote entity.
3	O.Linkage	The TSF shall use the correct user public key for authentication.
4	O.Single_Use_I&A	The TSF shall use the I&A mechanism that requires unique authentication information for each I&A.

4.3.12 Online Certificate Status Protocol Client Package

The following security objectives are defined for the Online Certificate Status Protocol Client package.

Table 4.14 – Security Objectives for Online Certificate Status Protocol Client Package

#	**Objective Name**	**Objective Description**
1	O.Accurate_OCSP_Info	The TSF shall accept only accurate OCSP responses.
2	O.Auth_OCSP_Info	The TSF shall accept the revocation information from an authorized source for OCSP transactions.
3	O.Fresh_OCSP_Info	The TSF accept only reasonably current revocation information for OCSP transactions.
4	O.User_Override_Fresh_OCSP	The TSF shall permit the user to override the freshness requirement for the OCSP response.

4.3.13 Certificate Revocation List (CRL) Validation Package

The following security objectives are defined for the Certificate Revocation List Validation Package.

Table 4.15 – Security Objectives for Certificate Revocation List (CRL) Validation Package

#	Objective Name	Objective Description
1	O.Accurate_Rev_Info	The TSF shall accept only accurate revocation information.
2	O.Auth_Rev_Info	The TSF shall accept the revocation information from an authorized source for CRL.
3	O.Fresh_Rev_Info	The TSF shall accept only reasonably current CRL .
4	O.User_Override_Fresh_CRL	The TSF shall permit the user to override the freshness requirement for CRL.

4.3.14 Audit Management Package

The following security objectives are defined for the Audit Management Package.

Table 4.16 – Security Objectives for Audit Management Package

#	Objective Name	Objective Description
1	O.Audit	The TSF shall audit security relevant events.
2	O.Audit_Protect	The TSF shall protect the security audit log from unauthorized modifications.
3	O.Audit_Readable	The TSF shall be able to generate human readable reports from the audit log.
4	O.Audit_Select	The TSF shall permit authorized users to select auditable events.
5	O.Audit_User	The TSF shall be capable of associating audit events with individual users.

4.3.15 Continuous Authentication Package

The following security objective is defined for the Continuous Authentication package.

Table 4.17 – Security Objectives for Continuous Authentication Package

#	Objective Name	Objective Description
1	O.Continuous_I&A	The TSF shall continuously authenticate the entity.

5 - IT Security Requirements: Functional and environmental requirements

This includes an overall summary of the functional and environmental security requirements for the TOE, and environment. TOE security functional requirements are defined in Table 5.1 and the assurance security requirements are defined in Table 5.2. A functional requirement can be present in Part2 or Part3 of the CC documents. A PP or TOE is Part 2 extended if the functional requirements include functional components not in Part 2 but found in Part3. The definition of Part 2 extended is found in the CC Part 3, section 5.4. Security requirements are drawn from CC Part2 and Par3 as shown in Table 5.1 below.

Table 5.1 – Part 2 Functional Requirements

	Requirement	Part 2 or extended from Part 3
	FAU_GEN.1	Part 2
	FAU_GEN.2	Part 2
	FAU_SAR.1	Part 2
	FAU_SEL.1	Part 2
	FAU_STG.1	Part 2
	FDP_ACC.1	Part 2
	FDP_ACF.1	Part 2
	FIA_AFL.1	Part 2
	FIA_ATD.1	Part 2
	FIA_UAU.1	Part 2
	FIA_UAU.4	Part 2
	FIA_UAU.6	Part 2
	FIA_UAU.7	Part 2
	FIA_UID.1	Part 2
	FMT_MSA.1	Part 2
	FMT_MSA.3	Part 2
	FMT_MTD.1	Part 2
	FMT_SMF.1	Part 2
	FMT_SMR.2	Part 2
	FPT_RVM.1	Part 2
	FPT_SEP.1	Part 2
	FPT_STM.1	Part 2
	FCS_CRM_FPS.1	Part 2 Extended
	FDP_CPD.1	Part 2 Extended
	FDP_DAU_CPV_CER.1	Part 2 Extended
	FDP_DAU_CPV_CER.2	Part 2 Extended

	FDP_DAU_CPV_CER.3	Part 2 Extended
	FDP_DAU_CPV_CER.4	Part 2 Extended
	FDP_DAU_CPV_CER.5	Part 2 Extended
	FDP_DAU_CPV_INI.1	Part 2 Extended
	FDP_DAU_CPV_INI.2	Part 2 Extended
	FDP_DAU_CPV_INI.3	Part 2 Extended
	FDP_DAU_CPV_INI.4	Part 2 Extended
	FDP_DAU_CPV_OUT.1	Part 2 Extended
	FDP_DAU_CPV_OUT.2	Part 2 Extended
	FDP_DAU_CPV_OUT.3	Part 2 Extended
	FDP_DAU_CRL.1	Part 2 Extended
	FDP_DAU_ENC.1	Part 2 Extended
	FDP_DAU_ENC.2	Part 2 Extended
	FDP_DAU_ENC.3	Part 2 Extended
	FDP_DAU_OCS.1	Part 2 Extended
	FDP_DAU_SIG.1	Part 2 Extended
	FDP_ETC_ENC.1	Part 2 Extended
	FDP_ETC_ENC.2	Part 2 Extended
	FDP_ETC_SIG.1	Part 2 Extended
	FDP_ITC_ENC.1	Part 2 Extended
	FDP_ITC_ENC.2	Part 2 Extended
	FDP_ITC_PKI_INF.1	Part 2 Extended
	FDP_ITC_SIG.1	Part 2 Extended
	FIA_UAU_SIG.1	Part 2 Extended

5.1 - TOE Base security Functional Requirements

Table 5.2 provides a list of the base security functional requirements that specify the ability of TOE to manage multiple private keys, associated certificates, and identifying data and associations among them. The term manage, means the ability to do one or more of the following functions: generate, delete, use, import, export, modify, destroy, store, etc. Some or all of the base requirements may be met by the environment such as a trusted hardware, operating system and cryptographic modules such as FIPS 140 series validated cryptographic module.

Table 5.2 – TOE Base Security Functional Requirements included in all PPs in this PP Family

#	Functional Requirement	Title
1	FDP_ACC.1	Subset Access Control – PKI Credential Management
2	FDP_ACF.1	Security attribute based access control – PKI Credential Management
3	FIA_AFL.1	Authentication failure handling
4	FIA_ATD.1	User attribute definition
5	FIA_UAU.1	Timing of authentication
6	FIA_UAU.7	Protected authentication feedback
7	FIA_UID.1	Timing of identification
8	FMT_MSA.1	Management of security attributes
9	FMT_MSA.3	Static attribute initialisation
10	FMT_MTD.1	Management of TSF data
11	FMT_SMF.1	Specification of management functions
12	FMT_SMR.2	Restrictions on security roles
13	FPT_RVM.1	Non-bypassability of the TSP
14	FPT_SEP.1	TSF domain separation

5.1.1 Class FDP – User Data Protection

FDP_ACC.1 Subset access control – PKI Credential Management

FDP_ACC.1.1 The TSF shall enforce the PKI credential management SFP on subjects, objects, and operations among subjects and objects covered by the SFP

Application Note: *The terms object and subject refer to generic elements in the TOE. For a policy to be implemented, these entities must be clearly identified. For most systems there is only one type of subject, usually called a process or task, which needs to be specified in the ST. The ST author should specify the list of subjects, objects, and operations among subjects and objects covered by the SFP.*

Dependencies: FDP_ACF.1 Security attribute based access control

FDP_ACF.1 Security attribute based access control – PKI Credential Management

FDP_ACF.1.1 The TSF shall enforce the PKI credential management SFP to objects based on the identity of the subject and the set of roles that the subject is authorized to assume.

FDP_ACF.1.2	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed [selection of one or more by the ST author: a) Private keys may be generated, imported, exported, destroyed, used by owner or administrator as defined by the ST author b) Public key certificates may be imported, exported, deleted by owner, or administrator or other roles as assigned by the ST author c) Public key certificates may be used by anyone. d) other rule(s) as assigned by the ST author
FDP_ACF.1.3	The TSF shall explicitly authorize access of subjects to objects based on the following additional rules based on security attributes that explicitly authorize access of subjects to objects as assigned by the ST author.
FDP_ACF.1.4	The TSF shall explicitly deny access of subjects to objects based on the rules on security attributes that explicitly deny access of subjects to objects as assigned by the ST author.
Dependencies:	FDP_ACC.1 Subset access control, FMT_MSA.3 Static attribute initialisation

5.1.2 Class FIA – Identification and Authentication

FIA_AFL.1 Authentication failure handling

FIA_AFL.1.1	The TSF shall detect when a number of unsuccessful authentication attempts occur related to authentication events that the ST author will assign. The author of ST will assign the number of unsuccessful authentication attempts and the authentication events.
FIA_AFL.1.2	When the defined number of unsuccessful authentication attempts has been met or surpassed, the TSF shall [assignment by the ST author: list of actions].
Dependencies:	FIA_UAU.1 Timing of authentication

FIA_ATD.1 User attributes definition

FIA_ATD.1.1	The TSF shall maintain the following list of security attributes belonging to individual users: *list of role*.

FIA_UAU.1 Timing of authentication

FIA_UAU.1.1	The TSF shall allow [assignment by the ST author: list of TSF mediated actions] on behalf of the user to be performed before the user is authenticated.
FIA_UAU.1.2	The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.
Dependencies:	FIA_UID.1 Timing of identification

FIA_UAU.7 Protected authentication feedback

FIA_UAU.7.1	The TSF shall provide only [assignment by the ST author: list of feedback] to the user while the authentication is in progress.
Dependencies:	FIA_UAU.1 Timing of authentication

FIA_UID.1 Timing of identification

FIA_UID.1.1	The TSF shall allow [assignment by the ST author: list of TSF-mediated actions] on behalf of the user to be performed before the user is identified.
FIA_UID.1.2	The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.
Dependencies:	None.
Application Note:	*Identification and authentication rules may vary between TOEs; those rules need to be specified in the ST.*

5.1.3 Class FMT – Security Management

FMT_MSA.1 Management of security attributes

FMT_MSA.1.1	The TSF shall enforce the PKI credential management SFP to restrict the ability to change_default, query, modify, delete, and other specified operations, the security attributes such as user role, key identifier, association between private key and public key certificate, and other security attributes that the author of ST will define, to owner, user, administrator and other roles as defined by the author of ST.
Dependencies:	FMT_SMF.1 Specification of Management Functions, FMT_SMR.1 Security roles, FDP_ACC.1 Subset access control

FMT_MSA.3 Static attribute initialisation

FMT_MSA.3.1	The TSF shall enforce the PKI credential management SFP to provide specific default values for security attributes that are used to enforce the SFP.
FMT_MSA.3.2	The TSF shall allow the [selection of one or more by the ST **author: owner, user, administrator, [assignment by the ST author: other role(s) defined]] to specify alternative initial values** to override the default values when an object or information is created.
Dependencies:	FMT_SMR.1 Security roles, FMT_MSA.1 Management of security attributes

FMT_MTD.1 Management of TSF data

FMT_MTD.1.1	The TSF shall restrict the ability to [selection of one or more by the ST author: change_default, modify, delete, clear, import, add, [assignment by the ST author: other operations]] the [selection of one or more by the ST author: trust anchors, identification data, authentication data, number of unsuccessful authentication attempts [assignment by the ST author: other TSF data]] to [selection of one or more by the ST author: owner, user, administrator, [assignment by the ST author: other role(s) defined]].
Dependencies:	FMT_SMF.1 Specification of Management Functions, FMT_SMR.1 Security roles
Application Note:	*The ST author may iterate the requirement as necessary. The ST author must select identification data and authentication data in order to meet the security objective O.Protect_I&A_Data. The ST author must select trust anchors in order to meet the security objective O.Trust_Anchor.*

FMT_SMF.1 Specification of Management Functions

FMT_SMF.1.1	The TSF shall be capable of performing the following security management functions: [assignment by ST author: list of security management functions to be provided by the TSF].

FMT_SMR.2 Restrictions on security roles

Hierarchical to: FMT_SMR.1

FMT_SMR.2.1	The TSF shall maintain the roles [selection of one or more by the ST author: user, owner, administrator, remote entity assignment by the ST author: other role(s) defined]].
FMT_SMR.2.2	The TSF shall be able to associate users with roles.
FMT_SMR.2.3	The TSF shall ensure that the conditions [assignment by the ST author: conditions for the different roles] are satisfied.
Dependencies:	FIA_UID.1 Timing of identification

5.1.4 Class FPT – Protection of the TOE Security Functions

FPT_RVM.1 Non-bypassability of the TSP

FPT_RVM.1.1 The TSF shall ensure that TSP enforcement functions are invoked and succeed before each function within the TSF Scope of **Control (TSC) is allowed to proceed.**

FPT_SEP.1 TSF domain separation

FPT_SEP.1.1	The TSF shall maintain a security domain for its own execution that protects it from interference and tampering by untrusted subjects.
FPT_SEP.1.2	The TSF shall enforce separation between the security domain

5.2 - TOE Security Assurance Requirements for the IT Environment

The functions in this section address the security functional requirements for the IT environment in which TOE will operate. The environment will address functions such as providing cryptographic algorithms.

5.2.1 Class FCS – Cryptographic Support

FCS_CRM_FPS.1 FIPS compliant cryptographic module

FCS_CRM_FPS.1.1 The IT environment shall provide all cryptographic modules necessary for the TSF.

FCS_CRM_FPS.1.2 Each cryptographic module shall be FIPS 140 series Level 1 validated.

5.2.2 Class FDP – User Data Protection

FDP_ITC_PKI_INF.1 Import of PKI information from outside the TSF

FDP_ITC_PKI_INF.1.1 The IT environment shall ensure the availability of [selection of one or more by the ST author: certificates, CRLs, OCSP responses, [assignment by the ST author: other PKI information]], to the TOE [assignment: a defined availability metric] given the following conditions [selection of one or more by the ST author: availability of network connection, availability of information server, availability of information in the application protocol, availability of information to the IT environment, [assignment by the ST author: other conditions to ensure availability]].

5.2.3 Class FPT – Protection of the TSF

FPT_STM.1 Reliable time stamps

FPT_STM.1.1	The IT environment shall be able to provide reliable time stamps for TSF use.

5.3 Security Functional Requirements for Packages

The following section defines security requirements for each package. Table 5.3 shows security functional requirements for each package and their dependencies.
A summary of package and dependencies is as follows:

- Certification Path Validation – Basic Package is a dependency of the following
- Certification Path Validation – Basic Policy Package
- Certification Path Validation – Policy Mapping Package
- Certification Path Validation – Name Constraints Package
- PKI Encryption using Key Transfer Algorithms
- PKI Encryption using Key Agreement Algorithms
- PKI Decryption using Key Agreement Algorithms
- PKI Signature Verification
- PKI Based Entity Authentication
- Continuous Authentication
- Certification Path Validation – Basic Policy is a dependency of Certification Path

Validation – Policy Mapping Package

- PKI Based Entity Authentication is a dependency of Continuous Authentication Package

5.3.1 Certification Path Validation – Basic Package

This package is provides security function of validating certificate path. The path includes certificate named: trusted anchor, intermediate certificates and end entity certificates. Trusted anchor are self signed certificate by the top CA, intermediate certificates are issued to the intermediate CA and the end entity certificate are issues to the end entity.

Verifying the path is a process which among other verifying the validity of signature, date, and names. This means the validation process is ment to prove the binding of the certificate to entities that claims to own certificate. Validity date is verifies to ensure that the certificate is used within its specified time of use.

5.3.1.1 Class FDP – User Data Protection

FDP_CPD.1 Certification path development

FDP_CPD.1.1	The TSF shall develop a certification path from a trust anchor provided by user/administrator to the subscriber using matching rules for the following subscriber certificate fields or extensions: distinguished name, subject alternative names, subject key identifier, subject public key algorithm, certificate policies and other certificate fields or extensions as defined in X.509.
FDP_CPD.1.2	The TSF shall develop the certification path using the following additional matching rule:
	a) none,
	b) keyUsage extension has nonRepudiation bit set,
	c) keyUsage extension has digitalSignature bit set,

	d) keyUsage extension has keyEncipherment bit set,
	e) key Usage extension has keyAgreement bit set.

Table 5.3 – Summary of Security Functional Requirements in Packages

Package Name	Functional Requirement	Dependency Package
Certification Path Validation – Basic	FDP_CPD.1	none
	FDP_DAU_CPV_INI.1	
	FDP_DAU_CPV_CER.1	
	FDP_DAU_CPV_CER.2	
	FDP_DAU_CPV_OUT.1	
Certification Path Validation – Basic Policy	FDP_DAU_CPV_INI.2	Certification Path Validation – Basic
	FDP_DAU_CPV_OUT.2	
Certification Path Validation – Policy Mapping	FDP_DAU_CPV_INI.3	Certification Path Validation – Basic, Certification Path Validation – Basic Policy
	FDP_DAU_CPV_CER.3	
	FDP_DAU_CPV_OUT.3	
Certification Path Validation – Name Constraints	FDP_DAU_CPV_INI.4	Certification Path Validation Basic
Constraints	FDP_DAU_CPV_CER.4	Validation – Basic
	FDP_DAU_CPV_CER.5	
PKI Signature Generation	FDP_ETC_SIG.1	none
PKI Signature Verification	FDP_ITC_SIG.1	Certification Path Validation – Basic
	FDP_DAU_SIG.1	
PKI Encryption using Key Transfer Algorithms	FDP_ETC_ENC.1	Certification Path Validation – Basic
	FDP_DAU_ENC.1	
PKI Encryption using Key Agreement Algorithms	FDP_ETC_ENC.2	Certification Path Validation – Basic
	FDP_DAU_ENC.2	
PKI Decryption using Key Transfer Algorithms	FDP_ITC_ENC.1	None
PKI Decryption using Key Agreement Algorithms	FDP_ITC_ENC.2	Certification Path Validation – Basic
	FDP_DAU_ENC.3	
PKI Based Entity Authentication	FIA_UAU.1;1	Certification Path Validation – Basic
	FIA_UAU.4	
	FIA_UAU_SIG.1	
	FIA_UID.1;1	
Online Certificate Status Protocol Client	FDP_DAU_OCS.1	None
Certificate Revocation List Validation	FDP_DAU_CRL.1	None
Audit Management	FAU_GEN.1	None
	FAU_GEN.2	
	FAU_SAR.1	
	FAU_SEL.1	
	FAU_STG.1	
Continuous Authentication	FIA_UAU.6:1	PKI Based Entity

	FIA_UAU.6:2	Authentication, Certification Path Validation – Basic

Example of Certification Path Validation – Basic FDP_CPD.1

FDP_CPD.1.3	The TSF shall develop the certification path using the following additional matching rule [selection of one by the ST author:
	a) none,
	b) extendedKeyUsage extension contains EFS or anyExtendedKeyUsage OID,
	c) extendedKeyUsage extension contains SCL or anyExtendedKeyUsage OID,
	d) extendedKeyUsage extension contains code signing or anyExtendedKeyUsage OID,
	e) extendedKeyUsage extension contains OCSP signing or anyExtendedKeyUsage OID,
	f) [assignment by the ST author: other extended key usage OID related matching rules]].
FDP_CPD.1.4	The TSF shall bypass any matching rules except [selection of one or more by the ST author: distinguished name, subject alternative names, subject key identifier, subject public key algorithm, certificate policies, [assignment by the ST author: other certificate fields or extensions]] if additional certification paths are required.
Application Note:	*In FDP_CPD.1.2, the assignment nonRepudiation should be used if the path is being developed for signature verification; the assignment digitalSignature should be used if the path is being developed for entity authentication; the assignment keyEncipherment, should be used if the path is being developed* *for encryption certificate using a key transfer algorithm anyExtendedKeyUsage is a match for any application.*

5.4 PPs With EAL 3 With Augmentation

This PP is assurance Evaluation Assurance Level 3 (EAL3) augmented by ALC_FLR.1. The assurance components are listed in Table 5.4.

Table 5.4 – EAL3 with Augmentation Assurance Requirements

Assurance Component Identifier	Assurance Component Title
ACM_CAP.3	Authorisation controls
ACM_SCP.1	TOE CM coverage
ADO_DEL.1	Delivery procedures
ADO_IGS.1	Installation, generation, and start-up procedures
ADV_FSP.1	Informal functional specification
ADV_HLD.2	Security enforcing high-level design
ADV_RCR.1	Informal correspondence demonstration
AGD_ADM.1	Administrator guidance
AGD_USR.1	User guidance
ALC_DVS.1	Identification of security measures
ALC_FLR.1	Basic flaw remediation
ATE_COV.2	Analysis of coverage
ATE_DPT.1	Testing: high-level design
ATE_FUN.1	Functional testing
ATE_IND.2	Independent testing - sample
AVA_MSU.1	Examination of guidance
AVA_SOF.1	Strength of TOE security function evaluation
AVA_VLA.1	Developer vulnerability analysis

ACM_SCP.1 TOE CM Coverage

Dependencies: ACM_CAP.3 Authorisation controls

Developer action elements:

ACM_SCP.1.1D The developer shall provide a list of configuration items for the TOE.

Content and presentation of evidence elements:

ACM_SCP.1.1C The list of configuration items shall include the following: implementation representation and the evaluation evidence required by the assurance components in the ST.

Evaluator action elements:

ACM_SCP.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ACM_CAP.3 Authorisation controls

Dependencies: ALC_DVS.1 Identification of security measures

Developer action elements:

ACM_CAP.3.1D	The developer shall provide a reference for the TOE.
ACM_CAP.3.2D	The developer shall use a CM system.
ACM_CAP.3.3D	The developer shall provide CM documentation.

Content and presentation of evidence elements:

ACM_CAP.3.1C	The reference for the TOE shall be unique to each version of the TOE.
ACM_CAP.3.2C	The TOE shall be labeled with its reference.
ACM_CAP.3.3C	The CM documentation shall include a configuration list and a CM plan.
ACM_CAP.3.NEWC	The configuration list shall uniquely identify all configuration items that comprise the TOE.
ACM_CAP.3.4C	The configuration list shall describe the configuration items that comprise the TOE.
ACM_CAP.3.5C	The CM documentation shall describe the method used to uniquely identify the configuration items.
ACM_CAP.3.6C	The CM system shall uniquely identify all configuration items.
ACM_CAP.3.7C	The CM plan shall describe how the CM system is used.
ACM_CAP.3.8C	The evidence shall demonstrate that the CM system is operating in accordance with the CM plan.
ACM_CAP.3.9C	The CM documentation shall provide evidence that all configuration items have been and are being effectively maintained under the CM system.
ACM_CAP.3.10C	The CM system shall provide measures such that only authorised changes are made to the configuration items.

Evaluator action elements:

ACM_CAP.3.1E	The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

6 Rationale

This section provides the rationale or further evidence and explanation to support the certification of this PP. It includes rationale for the environment security objectives.

6.1 Security Objectives Rationale

6.1.1 Base and Environmental Security Objectives Rationale

Table 6.1 – Mapping the TOE Base Assumptions and Threats to Objectives

Assumption/Threat	Objectives
AE.Authorized_Users	OE.Authorized_Users
AE.Configuration	OE.Configuration
AE.Crypto_Module	OE.Crypto
AE.Low	OE.Low
AE.PKI_Info	OE.PKI_Info
AE.Physical_Protection	OE.Physical_Security
AE.Time	OE.Time
T.Attack	O.DAC
T.Bypass	O.Invoke
T.Imperson	O.I&A, O.Limit_Actions_Auth
T.Modify	O.Self_Protect, O.DAC, O.Protect_I&A_Data, O.Trust_Anchor, O.TSF_Data
T.Object_Init	O.Init_Secure_Attr
T.Private_key	O.DAC
T.Role	O.Security_Roles
T.Secure_Attributes	O.Secure_Attributes
T.Shoulder_Surf	O.No_Echo
T.Tries	O.Limit_Tries

Table 6.2 – Mapping the Base TOE and Environmental Objectives to Threats and Assumptions

Objective	Threats
OE.Authorized_Users	AE.Authorized_Users
OE.Configuration	AE.Configuration
OE.Crypto	AE.Crypto_Module
OE.Low	AE.Low
OE.Physical_Security	AE.Physical_Protection
OE.Time	AE.Time
O.DAC	T.Attack, T.Modify, T.Private_key
O.I&A	T.Imperson
O.Init_Secure_Attr	T.Object_Init
O.Invoke	T.Bypass
O.Limit_Actions_Auth	T.Imperson
O.Limit_Tries	T.Tries
O.No_Echo	T.Shoulder_Surf
O.Protect_I&A_Data	T.Modify
O.Secure_Attributes	T.Secure_Attributes
O.Security_Roles	T.Role
O.Self_Protect	T.Modify
O.Trust_Anchor	T.Modify
O.TSF_Data	T.Modify

6.1.2 Security Objectives Rationale for Packages

The following subsections provide the mapping and rationale for the security objectives and threats associated with each individual package.

6.1.2.1 CPV – Basic Package Security Objectives Rationale

The following tables demonstrate the mapping of threats to objectives and objectives to threats for the CPV – Basic package.

Table 6.3 – Mapping of Threats to Objectives for CPV – Basic Package

#	Threat	Objectives
1	T.Certificate_Modi	O.Verified_Certificate
2	T.DOS_CPV_Basic	O.Availability
3	T.Expired_Certificate	O.Correct_Time O.Current_Certificate
4	T.Masquarade	O.Trusted_Keys
5	T.No_Crypto	O.Get_KeyInfo
6	T.Path_Not_Found	O.Path_Find
7	T.Revoked_Certificate	O.Valid_Certificate
8	T.User_CA	O.User

6.2 Security Requirements Rationale

In this section, the objectives are mapped to the functional requirements and rationale is provided for the selected EAL and its components and augmentation.

6.2.1 Functional Security Requirements Rationale

The mapping of all security objectives to functional requirements (components) or to assumptions is provided in Table 6.33.

Table 6.33 – Security Objective to Functional Component Mapping

#	Objective	Functional Components
Mapping for Objectives for the TOE		
1	O.DAC	FDP_ACC.1, FDP_ACF.1
2	O.Invoke	FPT_RVM.1
3	O.I&A	FIA_ATD.1, FIA_UAU.1, FIA_UID.1
4	O.Init_Secure_Attr	FMT_MSA.3
5	O.Limit_Actions_Auth	FIA_UAU.1, FIA_UID.1
6	O.Limit_Tries	FIA_AFL.1
7	O.No_Echo	FIA_UAU.7
8	O.Protect_I&A_Data	FMT_MTD.1, FMT_SMF.1
9	O.Secure_Attributes	FMT_MSA.1, FMT_SMF.1
10	O.Security_Roles	FMT_SMR.2
11	O.Self_Protect	FPT_SEP.1
12	O.Trust_Anchor	FMT_MTD.1, FMT_SMF.1
13	O.TSF_Data	FMT_MTD.1, FMT_SMF.1

List of Acronyms for TANESCO PP

CA	Certification Authority
CAC	Common Access Card
CC	Common Criteria
CDU	Credit Dispensing Unit
CEM	Common Evaluation Methodology
CPV	Certification Path Validation
CRL	Certificate Revocation List
CRLDP	CRL Distribution Point
DH	Diffie Hellman
DISA	Defense Information Systems Agency
DN	Distinguished Name
DSA	Digital Signature Algorithm
EAL	Evaluation Assurance Level
ECDH	Elliptic Curve Diffie Hellman
EFS	Encrypted File System
EKU	Extended Key Usage
ERMS	Electricity Revenue Management System
FIPS	Federal Information Processing Standard
GMT	Greenwich Mean Time
HMAC	Hash based Message Authentication Code
IDP	Issuing Distribution Point
IDS	Intrusion Detection System
IEC	International Electrotechnical Committee
IETF	Internet Engineering Task Force
ISO	International Organisation for Standards
IT	Information Technology
LUKU	Lipa Umeme Kadri Unavyotumia(Pay for power as you use)
OCSP	On-line Certificate Status Protocol
OS	Operating System
PKCS	Public Key Cryptography Standard
PKE	Public Key Enabled
PKI	Public Key Infrastructure
PKIX	Public Key Infrastructure Working Group --IETF
PKIPP	Public Key Infrastructure Protection Profile
PP	Protection Profile
PRW	Print Read Write
RFC	Request for Comment
RSA	Rivest, Shamir, and Adelman
SCVP	Simple Certificate Validation Protocol

SFP	Security Function Policy
SMS	Systems Master Station
SOF	Strength of Function
ST	Security Target
TOE	Target of Evaluation
TSC	TSF Scope of Control
TSF	TOE Security Function

Appendix F: Related Publications

In this section, we present summaries of publications of this research. In the course of this research, we published at least four-refereed papers, one poster and made one presentation to different international conferences where these papers went through a double blind review processes.

Summaries of Refereed Papers

Paper 1 Public Key Infrastructure Security and Interoperability Testing and Evaluation

This paper was presented at the international information technology conference, IITC 2003 - Towards an ICT Enabled Society, Chaula and Yngström (2003) in Colombo, Srilanka. The paper presents and describes challenges faces when developing and deploying public key infrastructure and testing and evaluation as a means to address interoperability problems.

Public Key Infrastructures (PKIs) are currently being fielded in increasing sizes, numbers, fast changing technologies, and varying environments but our operational experience to date has been limited to a relatively small scale and small number of environments. Consequently, some open technical and environmental interoperability problems about the ways in which PKIs will be organised and operated in large-scale applications need to be addressed. For instance, (1) Non interoperable proprietary vendor-provided public key infrastructures (2) the distribution of revocation information which has serious security implications and the disadvantage to be very costly when running large scale PKI. This paper introduces the concept of security testing and evaluation as a basis for PKI systems interoperability.

Paper 2 Security Metrics and Evaluation of Information Systems Security

A paper on security metrics and evaluation of information systems security Chaula, Yngström and Kowalski (2004) was published at the Information Security South Africa, ISSA2004, and conference. The paper highlights how the evaluation of information systems security process is carried out. It details ways in which evidence for assurance is identified, gathered, and analysed against criteria for security functionality and assurance level. This can result in a measure of trust that indicates how well the system meets particular security target. Also the paper shows how as the information systems complexity increases, it becomes increasingly hard to address security targets.

This paper attempts to examine the use of security metrics in the information systems evaluation process to yield quantifiable information that can be used to improve the evaluation process especially risk assessment, penetration testing, vulnerability assessment, protection profiles, and test coverage. This work is based on the Common Criteria (CC) and the Systems Security Engineering Capability maturity Model. These are useful established methods for security functions identification, assurance levels classification and security processes and organisations maturity levels classification.

The paper further presents advantages of security metrics such as security metrics facilitate improved understanding of various security processes, performance, and informed decision making of various security mechanisms and procedures implementation. Moreover, security metrics are useful for indication and determination of critical and non critical security parameters when evaluating a system and security processes, measuring the effectiveness of a security process in an organisation, security implementation cost justification, security problem isolation, system testing coverage, redirecting of security assets and efforts, and tracking changes to security processes in an organisation.

A full paper is available in the Peer-reviewed Proceedings HS Venter, JHP Eloff, L Labuschagne and MM Eloff of the ISSA 2004 enabling tomorrow Conference 30 June 2 July 2004, ISBN 1-86854-522-9, Gallagher Estate, Midrand.

Paper 3 **A Framework for Evaluation of Information Systems Security**

Another paper is on a framework for evaluation of information systems security Chaula, Yngström, and Kowalski (2005) was presented at the Information Security South Africa conference, ISSA2005, in Johannesburg, South Africa.

The paper describes challenges of systems assurance such as the challenges of cost and time constraints. These factors are partly attributed to non technical assurance factors, the choice of assurance technique and choice of assurance tools, lack of reuse, and lack of metrics which are essential for cost and effort estimation. Assurance for complex systems like electronic commerce is still abstract because when the systems complexity increased, it becomes harder to examine whether security requirements has been met and therefore the concept of perfect security proves to be unachievable goal for both computer systems vendors and consumers.

The paper describes research work which based on the Common Criteria (CC) which is an established method for security functions identification, assurance levels classification and development of Protection Profiles. In this research an Information Security Assurance Framework is proposed. This can be used to address the Information Security Assurance problem taking into consideration non-technical assurance factors, re-use of Protection Profiles and use of security metrics in the process of information assurance. A Protection Profile defines an implementation-independent set of IT security requirements for a category of IT products. Such products are intended to meet common consumer needs for IT security. Consumers can therefore construct or cite a PP to express their IT security needs without reference to any specific product.

A full paper is available in the Peer-reviewed Proceedings of HS Venter, JHP Eloff, L Labuschagne and MM Eloff, ISSA 2005 New Knowledge Today Conference, 29 June – 1 July 2005, ISBN 1-86854-625-X , Balalaika Hotel, Sandton, South Africa

Paper 4 Technology as a Tool for Fighting Poverty: How Culture in the Developing World Affect the Security of Information Systems

This paper was presented at the IEEE 4th International Workshop on Technology for Education in Developing Countries Chaula, Yngstrom, Kowalski (2006) Iringa, Tanzania. In this paper, we argue that many developing nations strive to automate various processes in anticipation to improve production and quality of service to meet millennium development goals and cope with globalization needs. This has led to the automation of critical systems. It is therefore imperative that the security of such critical systems is one of the central issues that developing nations must look at when plan, acquire and use information systems.

The purpose of this paper is to examine the role of culture in information systems insecurities. We argue that insecure systems undermine economic growth and that culture defines how people plan, acquire and use information systems in a secure way.
We also present some findings of culture evaluation case study that we carried out in Tanzania to determine the role of culture in the process of securing electricity power utility systems.

Summary of a Poster and a presentation

Poster 1 **Information Systems Security Usability Assurance: Evaluation in Search for User Centred Security**

This poster we have presented at the Information Security South Africa conference, ISSA2006, in Johannesburg, South Africa and co-authored by Chaula, Yngstrom and Kowaski. This poster describes the concept of usability evaluation and we argue that information systems security usability is an important aspect of information systems security assurance process. In this poster, we present our research on how security usability of information systems can be evaluated using cost effective methods.
The method involved developing security heuristics, security heuristics questions, choosing evaluators and conducting evaluation experiment. The heuristics evaluation and questioning methods were developed and used to evaluate the security module interface of electricity prepayment system. Evaluators, who had to participate in the security usability experiments, were from the ICT directorate and comprised of systems administrators and technicians.
Results of the experiment indicated that usability problems have the potential to cause serious system security insecurities. In addition, the number of new errors discovered by additional evaluators that could not be discovered by the previous evaluators decreased. This led to the conclusion that one does not gain more errors by using larger number of evaluators. These findings indicate that the method has the potential of being cost effective and suitable for prototyping and rapid development methods.

Presentation1 **IT Security Assurance For the Developing World**

This presentation, IT security assurance for the developing world Chaula, Yngstrom, Kowalski (2004a) was presented at the 5th international common criteria conference in Berlin German. The paper describes the concept that many developing nations are looking to IT infrastructure investments as means to reach sustainable economics growth. They are also hoping to be able avoid some of the major mistakes that developed nations have made in the deployment of their IT infrastructure. Once such mistake that developed nations have made that developing nations are trying to avoid is to retrofit security assurance methodologies and metrics.

In this presentation we present our research to develop IT security assurance methodologies and metrics for developing nations. In particular, we presented our research on the application of the Common Criteria methodology to develop a protection profiles for PKI systems to be used in Tanzania.

Section summary

In this section, we have presented summaries of refereed papers a poster and a presentation. Papers were presented in international conferences over the years since we began this research. In the process of publishing and participation in international conferences, we had useful comments from reviewers and other conference participants.

Appendix G: Licentiate

Department
Computer and Systems
Sciences

Security Metrics and Public key Infrastructure Interoperability Testing

Job Asheri Chaula

12 December 2003

Licentiate Thesis

Stockholm University and Royal Institute of Technology

Submitted to Stockholm University in partial fulfillment of requirements for the degree of Licentiate of Philosophy

Abstract

Public Key Infrastructures (PKIs) are currently being deployed in increasing sizes, numbers, fast changing technologies, and varying environments. However, our operational experience to date has been limited to a relatively small scale and small number of environments.

While research and development has mainly focussed on new trust models, robust certificate revocation status protocols, cryptographic algorithms, and PKI architecture, little attention has been given to interoperability testing and security metrics. Consequently, some open interoperability problems about the ways in which PKIs will be organized and operated in large-scale applications need to be addressed. For instance, (1) Non interoperable proprietary vendor-provided public key infrastructures (2) the distribution of revocation information which has serious security implications and the disadvantage to be very costly when running large scale PKI (3) Legal, policy, and privacy issues which affect inter domain operability.

This thesis introduces the concept of security testing and evaluation to maximize PKI application assurance as a basis for PKI systems interoperability. The security metrics and testing are useful approaches when examining PKI interoperability problems because they can be used to measure and evaluate both PKI technical and environmental problems. Testing is important for minimizing risks, and security metrics are useful for measuring testing coverage, effectiveness, and impact of various security processes in PKI environments.

.

Acknowledgement

My gratitude first goes to my advisor István Orci for the counsel he provided at the early stage of this work. Thanks to my advisor Louise Yngström for the mental support, guidance, suggestions, and timely comments that helped me to write this thesis. Thanks to Stewart Kowalski for his valuable comments and seminars.

SIDA/SAREC is entirely funding my research. Thanks to Love Ekenberg, Lars Askar and Beda Mutagahywa for coordinating the cooperation between SIDA/SAREC and the University of Dar s salaam, Tanzania. Thanks to Dr. Mtalo of the University College of Land and Architectural Studies for his support. Thanks to Birgita Ohlsson and Rodolfo Candia for assisting through the administrative maze at DSV.

Thanks to my friends at DSV especially Jeffy Mwakalinga for the valuable comments on X.509 certificates which have been very useful over the entire period of this work. Thanks to Elia Chaula, Pudenciana Mlanji, Nicas Yabu, Sara Nicas, Rev. Geoffrey Massawe, Rev. Paul Mulokozi and Ubungo TAG for your support and prayers.

Finally, but not least, thanks to my supportive family, my wife Jennifer for your endurance and my sons Jotham and Jeftah who at early ages of their lives missed my presence while I had to stay in Sweden to carry out this work.

Thank you all!

Dedication

This thesis is dedicated to my wife Jennifer and my sons Jotham and Jeftah.

Table of Content

List of Figures

List of Tables

1. Introduction

1.1 Information Technology security overview

The thesis examines the concept of IT security testing, verification, and security metrics as part of research efforts to address interoperability and scaling problems of Public Key Infrastructure systems (PKIs). IT security terminology, in this thesis, means computer and communication security. Computer and communication security deals with the prevention and detection of unauthorized actions by entities of a computer system [Gollmann 2000] my interpretation of this definition is that IT security involves procedural and administrative processes to protect IT resources.

Schneier's [Schneier 2000] assertion that most IT security products on the market are not secure because of lack of testing is true because it is one thing to model the trust models, the threat, design the security policy, and build the counter measures mechanisms like the secure e-mail systems, firewalls, antivirus, VPN, biometric systems, smart cards, digital cash systems, crypto products, digital certificates, Public key infrastructure and mobile agents, but the answer to the question are these systems secure? Is, obviously, not trivial.

Today Information Technology (IT) plays a central role in organizations, academia institutions, and governments. Each of these own modern computer networks which are a complex assembly of databases, web and application servers and various network devices that often span across borders of countries and even continents. In most cases the convenient solution to achieve this kind of connectivity is connection via open distributed network, the Internet.

The Internet is one of the fastest growing systems in terms of the number of users and technology. These factors make the Internet not only attractive to organisations which seek connectivity but it is also attractive to criminals who seek to penetrate different security mechanisms in order to steal information, modify information, spread viruses and worms, send sperm mails and performing illegal actions like child pornography. Therefore, the need for reliable security tools for protecting information technology resources becomes evident. One way to achieve this is for systems development engineers to treat security engineering as part of whole system's engineering process [SSE-CMM 2003].

The security requirement and services seems to be straightforward and summarized with a few worlds: Confidentiality, Authentication, Integrity, Nonrepudiation, Access control and Availability. However, the mechanisms used to address those requirements and services can be quite expensive, complex and understanding them may require rather reasonable expertise [Stallings 1999].

1.2 Organisations and assets

Security services are designated to protect an organisation' assets against various threats. Organisations seek security systems that provide one or more security services. However, understanding the assurance level of the security system requires system testing that is based on established standards like the Common Criteria [CCIMB-99-031]. The concept of security organisation and the organisation's assets is depicted in figure 1.1. Understanding the concept of system's security testing is so important for organisations where an IT infrastructure is central to the organisation's well being.

It is worthwhile for an organisation to have a security policy in place. A Security policy is a statement outlining the organisation's commitment to securing its assets. The countermeasures, vulnerability assessment, implementation of a procedure, systems testing, accreditation and certification will be performed in accordance with the organisations policy. Procedures and mechanisms have to be verified simply because a faulty procedure or mechanism leaves residual vulnerability that can be exploited to cause residual risk to the assets [CCIMB-99-031].

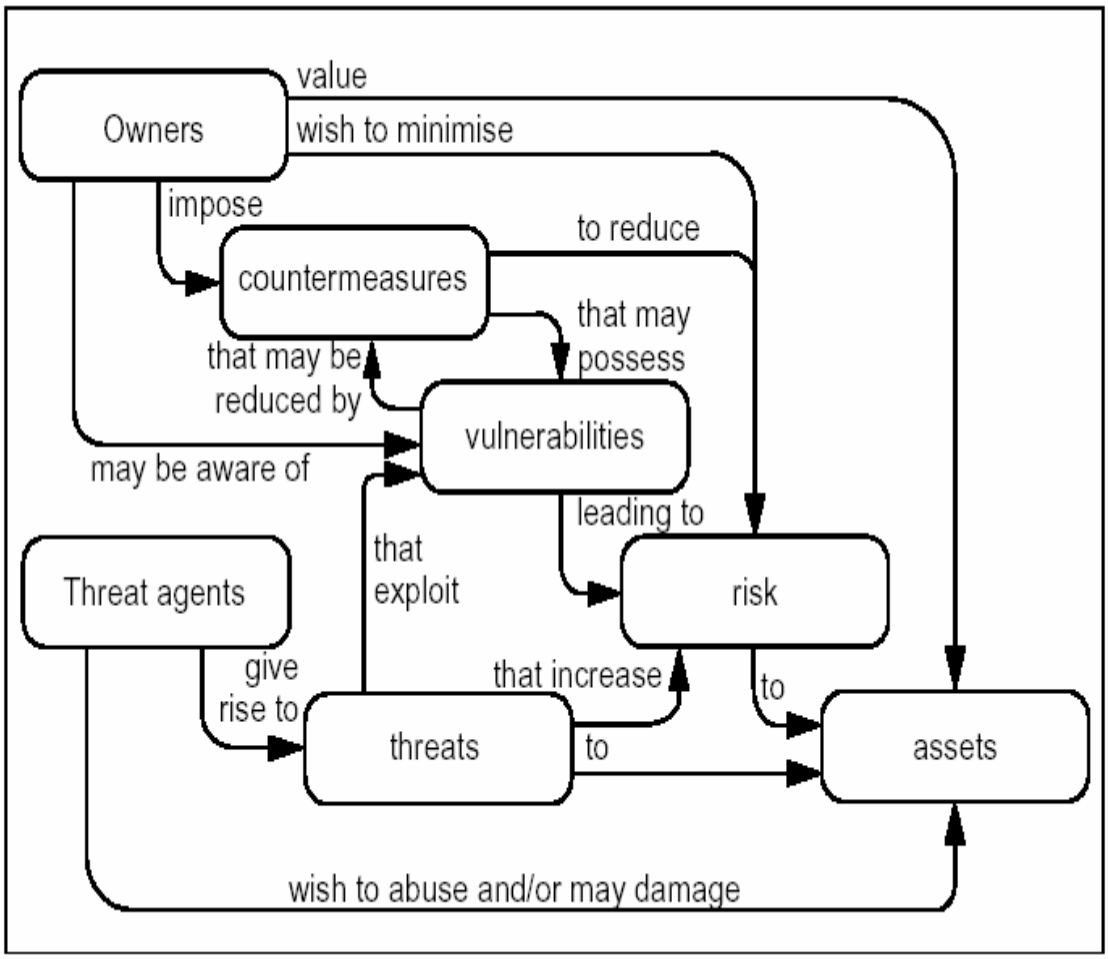


Figure 1.1 Security Concepts
[CCIMB-99-031 Page14]

A threat to an IT asset is a potential violation of a security mechanism [Bishop 2002]. Matt Bishop outlines the following threats: Snooping, modification or alteration, masquerading or spoofing, repudiation of origin, denial of receipt, delay, and denial of service. To stop different attacks appropriate security mechanisms have to be put in place to provide different security services.

Threats can cause damage to the IT assets which organizations have placed values. Threats can cause the IT assets to permanently or temporary stop performing its function or perform at a lower standard or capacity. This is referred to as Loss of availability [CCIMB-99-031], unauthorized entities gaining access to the IT asset is refereed to as loss of confidentiality [CCIMB-99-031], an entity having unauthorized modification of an IT asset is refereed to as loss of integrity [CCIMB-99-031], an entity successfully denying having acted on IT asset is refereed to as loss nonrepudiation and an entity having gained access to unauthorized IT assets is referred as to as loss of access control. Understanding of these terminologies is one of the central issues in IT security because when we examine IT systems testing, security metrics, process metrics, organisations IT security policy, and implementation of security mechanisms the ultimate purpose is to have effective security services in place.

1.3 Security services

In this section we present the definition of security services. These are directly related to our research efforts because when we are examining security functional testing processes, security metrics and process metrics we are basically attempting to address issues related to improved understanding, use and effectiveness of security services. However, their definitions are presented here for completeness.

Confidentiality: Confidentiality is the concealment of information and resources [Bishop 2002]. Confidentiality provides protection against passive attack. This service provides protection against traffic analysis and release of message content. Usually this can be achieved by applying cryptographic functions to all the data we want protected.

Authentication: Authentication is a security service that addresses the concept of authentic communication between entities [Stalling 1999]. This security service provides proof of origin authentication between the sender and the responder. This is achieved through the use of credentials like the user name and password or in PKI environment public key certificates are used.

Integrity: Integrity addresses the concept of trustworthiness of IT assets especially the data, message or a stream of data [Bishop 2002].

Access control: Access control addresses the issue of controlling which person, process or machine have access to which IT assets [Anderson 2001]. Access control deals with who should access what and when. Authentication and authorization are aspects of access control. A variant of public key certificate called attribute certificate can be used in a distributed environment to provide access control service. However in our work we will only examine x.509 version 3 certificates.

Availability: Availability security service address the concept that an attacker can aim at disrupting the normal operation of an IT asset for the aim of denying access to legitimate

users when they desire to use the asset.[Bishop 2002]. Resource unavailability is critical even if a single user uses the service. A variety of attacks can cause loss or reduction of availability. These can be caused by virus clogging the network like the BigF virus, a communication cable unplugged by a staff whose employment is terminated, natural calamity (referred to as act of God), and traffic jamming, and shutting down the system.

Non-repudiation: This service addresses the concept of binding communication entities to the actions they perform on the assets so that the sender or responder cannot later falsely deny having participated in a transaction [Gollmann 1999]

PKI is meant to provide the following security services [Stal 2000 page 3]:

- Integrity
- Authentication
- Confidentiality
- Access control and
- Nonrepudiation.

These are provided by PKI through use of public key certificates and cryptographic services.

1.4 Security Models

The security services we have presented in section 1.3 above have been addressed in several security models. In this section we briefly present four security models stating how the models have helped the author thinking about security services.

Security models help us reason correctly about IT security because they define the philosophy of IT security. They are the basis for the design, evaluation, and implementation of IT security. Although none of these models in practice can be covering every aspect of IT security, there are those that have become more widely known and used than others. These are namely BLP model, Biba model, The Clark-Wilson model and the Systemic-Holistic Model. These are presented in this section for completeness and clarity of the security services whose understanding is central when we examine security metrics and PKI interoperability testing.

Bell La Podla (BLP) model is focussed on the assumption that security policies prevent information flow downwards from a higher security level to a lower security level. It is a model addressing the confidentiality aspects of access control [Bell 1974]. The Biba model addresses integrity in terms of how users access objects. In these model users, processors, and data are classified based on the principle of integrity. In integrity lattice, information may flow downward. [Biba 1977]. The Clark-Wilson model focuses the security requirements of commercial application [Clark 1988]. Clark and Wilson attempted to address integrity and confidentiality in respect to the differences between military and commercial security requirements. This model defines the concept of the relationship between the system's internal state and the real word. This is referred to as external consistency and is enforced by means outside the computing system for instance policy. The Systemic-Holistic approach is a security model developed by [Yngström 1996]. It is based on the General System's Theory, Cybernetics and General Living Theory. Using this model one can know where details fit into the systems. The model is applicable over security testing and evaluation, security education and over many other IT security work.

For more than a decade researchers have put so much effort in developing security methodologies, models and standard definitions of security services [Housley 2001]. However, we still experience systems insecurity and probably we shall continue to experience the same. Systems insecurity alleviation requires thoroughly testing and evaluation. The use of metrics, which we are examining in this thesis, and the use of established method like the Common Criteria can result into improved security systems and mechanisms. Using these methods, when vendors claim their products to be secure, users should be able to enquire for independent security verification or require vendors to supply products with pre defined assurance levels.

1.5 Security testing and verification methodologies

In this thesis we will use the Common Criteria and the Systems Security Engineering Capability Maturity Models as guidelines for metrics and security functional testing.

The Common Criteria is a product of efforts that began in the early 1990's by the International Organisation for Standardisation with the aim to develop international standard evaluation criteria that is generic [CC 1998]. CC had inputs from the Trusted Computer System Evaluation Criteria (TCSEC), Information Technology Security Evaluation Criteria (ITSEC), and Trusted Computer Product Evaluation Criteria (CTCPEC) [CC 1998]. CC version 2.1 is a revision that is in line with the International Standard ISO/IEC 15408:1999. The IT systems that are compliant with CC are considered to be ISO/IEC 15408:1999 compliant as well. CC and the SSE-CMM are discussed in depth in Appendix B.

1.6 Systems and security

For an entity to complete a single transaction several systems may be involved. The transacting processes can be affected by a faulty that happens in one of the system. Complex systems like the PKI systems are made of many components that interact with each other and components are made of thousands of lines of codes. A fault in a single line of code may cause fault in the whole system. It is one thing to theoretically design a secure system but practically the implementation engineering process faces the reality of design tradeoff and imperfect configuration in the implementation process [Schneier 2000]. More secure systems can be developed if the security engineering is part of product life cycle [SSE-CMM 2003].

1.7 General evaluation model

Generally evaluations have two evaluator tasks [CEM-99/045]: the input task and the output task as depicted in Figure 1.2. These two tasks are related to management of evaluation evidence and evaluation report generation [CEM-99/045]. The evaluation evidence is basically a document that describes the security features of TOE.

The objectives of input and output tasks are to make sure that the evaluator has available the correct version of the evaluation evidence necessary for the evaluation and that it is adequately protected against any type of modification [CEM-99/045].

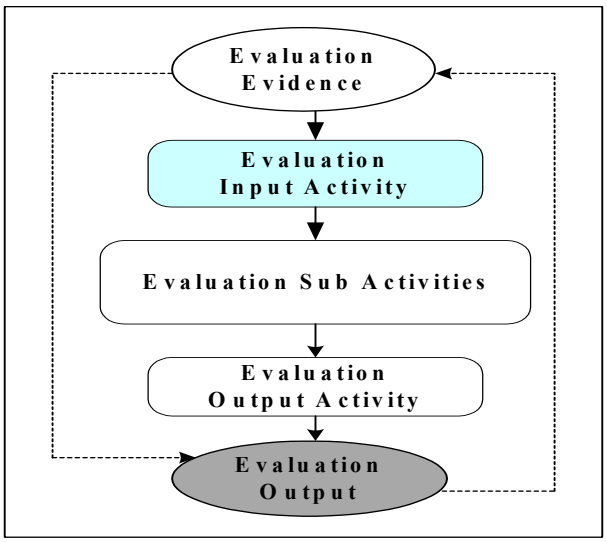


Figure 1.2 General Evaluations Model
[CEM-99/045 Page 10]

This is essential to make sure that the technical accuracy of the evaluation can be assured, and that the evaluation is being conducted in a way to provide repeatable, reproducible, and comparable results [CEM-99/045]. In this view our work is limited to the evaluation input activities.

The main evaluation evidence will be RFC 3280. Other documents that contribute to X.509 version 3 certificates specifications can also be used as evidence of evaluation [CEM-99/045].

Our work is limited to the evaluation input activities as depicted in Figure 1.4. The evaluation input activity involves generating test cases in accordance to the RFC 3280 evaluation evidence. The test cases are eventually analyzed to determine the testing coverage that is directly related to the amount of risk an organisation will accept to take in a particular application.

The dotted lines are included in Figure 1.2 to indicate the relationship of the evaluation output and the evaluation input activity. The feedback dotted line indicates that the evaluation output must be compliant to the evaluation evidence and the forward dotted line indicates that the evaluator can predict or pre define the output of desired test result. If the pre defined test result was fail and the result is pass this is an indication of a faulty application and if the predefined test result was pass and result is fail this is an indication of a faulty application as well.

1.8 The concept of the security flaw and security testing

When conducting security testing it is paramount to keep in mind that security testing is different from normal software testing practice in many ways. This is because security

flaw can happen anywhere in the system [Schneier 2002]. It can be in the design, implementation, source code, platform, interface, protocol or even the cryptographic algorithm. Security is a chain and only as secure as the weakest components [Schneier 2002].

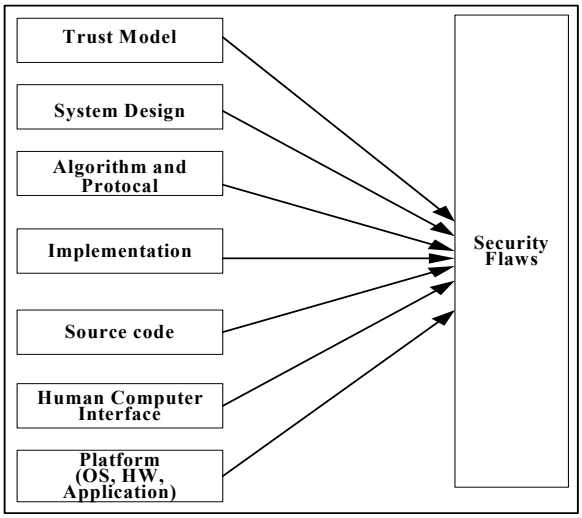


Figure 1.3 Security Flaw Context

Figure 1.3 shows multiple possible sources of security flaws. The only way to have confidence over any system security is to overtime have an expert evaluate it. This is possible if the details of the system are public like the IETF's RFCs.

1.9 Public Key Infrastructure (PKI) Systems

Public Key Infrastructure is a system that is used to communicate securely and with trust among entities in closed or open distributed computing environments. Trust amongst users is achieved through certificate exchange and authentication. PKI particularly provides information protection for systems services on the Internet [Housley 2001]. However, currently, so many individual organizations - small, medium and large are inclined to use PKI. There are numerous operational problems regarding large-scale PKI which collectively are termed interoperability problems.

Interoperability between PKI implementations forms the basis of security infrastructure in large-scale PKI environments [pkiC 2001]. PKI infrastructure, just like telephone infrastructure, transportation system, water, power lines or gas supply system should therefore recognize the similar benefits to users. In order to achieve interoperability public key certificates, PKI applications, and High-level PKI application programming interfaces must be interoperable [HAPI 2003]. Moreover, the repository scalability, robustness of repository access protocols, user awareness, policies and legal environments, as depicted in Figure 1.2, are important factors to achieve large scale PKI interoperability [pkiC 2001].

In order for a certificate to be valid, the certificate user must trust the issuing party

utilizing any of the revocation status verification mechanisms [RFC 3280]. In order to

maintain this environment of trust it is important that the revocation process is well-designed, implemented, maintained, and enforced without ambiguities as to the status of a certificate. Large scale PKI will involve voluminous users accessing the repository at the same time; in these scenario issues of trust, request rate, reliability traffic volume, and timeliness are of great importance to sustainability of the system.

Currently, at the center of efforts to improve security is a group of security protocols such as Secure Electronic Transaction [SET], S/MIME, IPSec and TLS [Housley 2001]. All these protocols on public key cryptography to provide security services such as confidentiality, integrity, authentication, non-repudiation and Access control. PKI is responsible for binding public keys into certificates and managing those certificates in their life cycle. The part of PKI that is responsible with generation, issuance, and revocation of certificates is referred to as Certificate Issuing and Management System or CIMS. The basic components of a PKI system are [RFC 3280]:

- The certification authority (CA): Issues and revokes certificates
- Registration authority (RA): Vouches for the binding between public keys and certificate holder identity and other attributes
- Certificate holder: Users who make use of certificates to encrypt/decrypt information in this thesis we also refer to this as End Entity (EE).
- Clients: Validate digital signature or encrypted messages and their certificate chains from the top CA
- Repositories: Stores and makes available to users certificates and Certificate Revocation Lists.

1.10 Background and motivation of the research

Tanzania has recently embarked in significant e-Government initiatives such as firstly, the implementation of an HR & Payroll system, the implementation of an Integrated Financial Management System (IFMS), the Tax Identification Number (TIN), the Tanzania Social and Economic Database (TSED) [eSecretariat 2001]. Secondly, Infrastructure initiatives such as the rollout plan of the newly privatised Tanzania Telecommunication Company (TTCL) that included computerizing the service order and billing system, the Tanzania Electric System pre paid system (LUKU), and the rollout plans of mobile cellular operators [eSecretariat 2001]. Lastly, is the initiative for partnerships such as the networks among Tanzania's academic institutions [eSecretariat 2001].

More over Tanzania has recently developed its ICT policy which is based on The Tanzania Development Vision 2025 policy [NICT 2003]. This Policy has outlined ten main focal areas in making effective use of ICT in Tanzania which include strategic ICT leadership; Service Sectors; Public Service; Local Content; ICT infrastructure; ICT Industry; Human Capital; Legal and Regulatory Framework; Productive Sectors; and Universal Access [NICT 2003]. While Tanzania is making good progress in the process

of adopting ICT for sustainable development, most of the achievements listed above have been before the National ICT policy was developed. Consequently, the lack of an overall ICT policy, the ban of computers importation since 1974 to 1993, poor harmonization of initiatives, have led to degraded basic user computer skills, random adoption of different systems, adoption of many standards, unnecessary duplication of effort, and waste of scarce financial resources through maintenance and training [eSecretariat 2001].

1.10.1 Security Implication

The Tanzania Government is the largest market of the entire ICT products supplied in Tanzania by local and foreign vendors [eSecretariat 2001]. Local vendors, in most cases, act as partners for foreign vendors so in reality most hardware and software systems are supplied from outside the country. Significant effort in the implementation process is used to train systems users and support staff. Greater emphasis is on the adoption of ICT as quickly as possible. Consequently, there are serious system's insecurity and high maintenance cost. Systems insecurities are largely attributed to:

- Lack of testing and evaluation
- Lack of harmonised security and systems standards [NICT 2003].
- Lack of User, support personnel and stake holders security awareness
- Lack of legal environment that addresses issues related to computer fraud, crime and misuse, and privacy issues.
- Lack of ICT security policy [NICT 2003].
- Lack of security experts

The result is that the government may loose revenue due to undetected fraudulent and faulty systems, individual privacy infringement by Government, and loss of revenue due to maintenance costs caused by sub standard systems, and exposure of government data that is indented to be secret. To address all these problems is not a simple. However, testing and evaluation mechanisms may alleviate the systems insecurities we have tried to point out in this section. The Government of Tanzania for example could have a policy defining the assurance level required for every system sold to the Government or it may require independent testing to be performed by a trusted security lab.

1.11 Research Problem formulation

Research on applications of Public Key Infrastructures (PKIs) has been underway for more than a decade [Housley 2001]. Efforts have been directed to addressing problems that are related to how to improve trust models, public key certificates, certificate revocation mechanisms, directory access protocols and public key cryptography. However, the security benefits of a large scale PKI application to e-commerce, corporate, academic institutions and governments has so far not been widely realized [Housley 2001]. This is in part caused by technical problems and in part by environmental problems as depicted in Figure 1.4. The problems are enumerated as follows:

- Non interoperable proprietary vendor-provided public key infrastructures (PKIs)
- Non interoperable proprietary vendor-provided application programming interface (APIs)
- The distribution of revocation information that has the disadvantage to be very costly when running large-scale PKI systems

- Scaling of directory access protocols
- The emergent of mobile computing technology pose some new problems that we need to address

These problems are related and have direct effects in ways PKI is operated and maintained in large-scale PKI applications environments.

Proprietary applications and APIs course serious interoperability problem. This problem can be addressed through testing and use of security metrics in the user' environments [CCIMB-99-031].

The scenario depicted in Figure 1.4 is too complex to be understood by the end users. In our work we are proposing security metrics, which we believe, shall help users understand better different security processes, policy and legal environments that are in their organization.

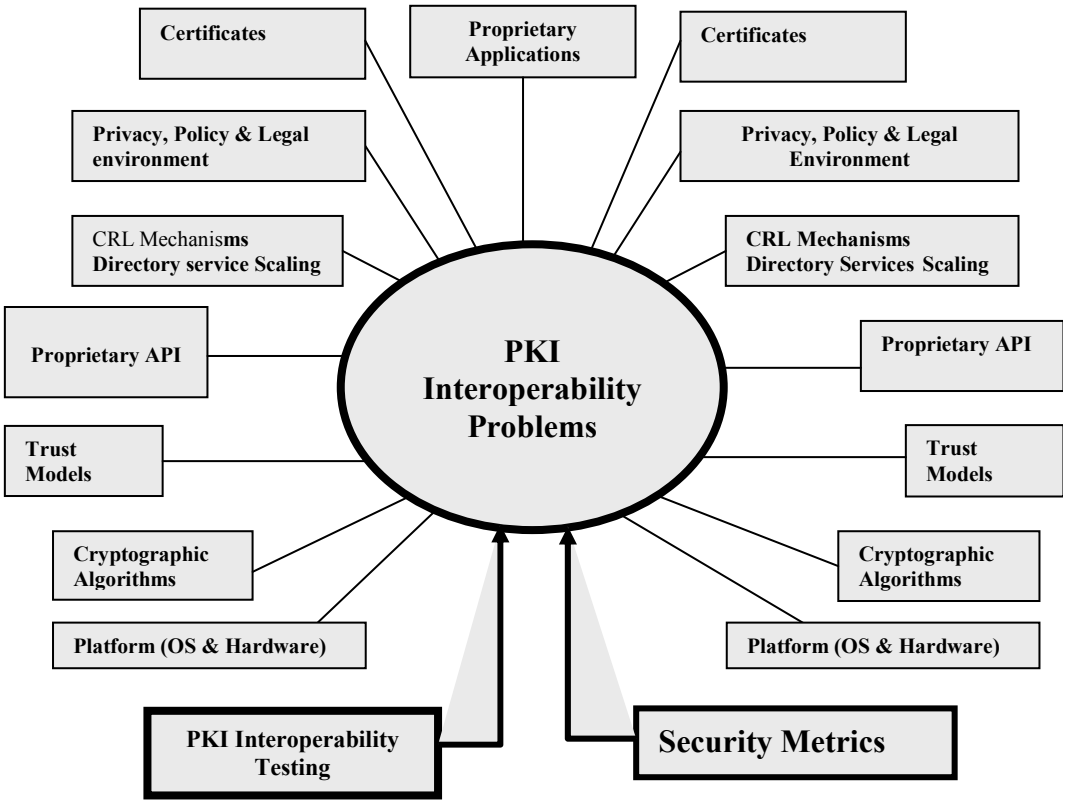


Figure 1.4 PKI Interoperability Problem

The repository is used by PKI applications to store user certificates. This has the potential of being misused especially by governments that for some reasons can be interested in knowing who are the users and what type of transactions they perform. This is a serious infringement of privacy. PKI application operation is also dependant on the trust model in use. There exists various trust models in our work we will use the hierarchical trust model because this is proposed in the evaluation evidence [RFC 3280].

1.12 Research Purpose

The purpose of our research work is to attempt to develop security and process metrics for PKI environments and to develop security functional testing assertions for PKI enabled applications. In my view security and process metrics and security functional testing are very important factors in addressing PKI interoperability problems that are depicted in Figure 1.2 and detailed in chapter 4. If PKI enabled applications are properly tested and security metrics are applied in the testing process and user environments then applications in PKI environments should be able to verify X.509 certificates correctly and continuously. Our work will be guided by the Common Criteria (CC) as a basis for security functional testing [CEM-99/045] and the Systems Security Engineering Capability maturity Model [SSE-CMM 2003] as a guide for developing security and process metrics for process and security improvement.

Process metrics could be used as quantitative or qualitative evidence of the level of maturity of a security functional testing process [SSE-CMM 2003]. Security metrics are measurable attributes of the result of a security engineering process that could serve as evidence of the process effectiveness. Security metrics may be quantitative or qualitative [SSE-CMM 2003]

1.12.1 Security Metrics development form

The metrics development guidelines that are presented in chapter 2 are based on the metrics development guidelines from NIST Security Metrics Guide for Information Technology Systems [Swanson 2003]. This form will be used to document all the test assertions in the implementation evidence field that can eventually be used to calculate the test coverage.

Table 1-1 Metrics form
[Swanson 2003 Page 20]

Performance goal	
Performance objective	
Metrics	
Purpose	
Implementation Evidence	
Frequency	
Formula	
Data Source	
Indicator	

Table 1.1 contains information that defines the goal, objective and purpose of the security metric. Multiple performance objectives can correspond to a single performance goal. In such a case a different table shall be used to document each performance objective. The implementation evidence serves for validating performance of security activities and pinpointing causation factors [Swanson 2003]. The performance of security activities in a

PKI environment involves testing PKI components, User training, certification tracking, Privacy verification, system documentation and system maintenance.

The PKI application testing metrics are central in our work. Other metrics listed above are presented for completeness and they are based on the NIST Security Metrics Guide for Information Technology Systems where they are presented in a more generic way to cover other environments apart from PKI [Swanson 2003].

Performance goal	In the metrics we have developed in chapter 2 we have used Testing goal instead of Performance goal. This is because the purpose of metrics development effort in our work is to provide guideline how to measure security functional testing coverage and process effectiveness and impact. This field states the desired results of testing one or several PKI system security control objectives that are measured by the metric.
Performance objective	This item will list one or more test questions. Multiple testing objectives can correspond to a single testing goal.
Metrics	This field defines the metric by describing the quantitative measurements provided by the metric.
Purpose	The purpose describes the reason of collecting the metrics. This can be process improvement or testing coverage.
Implementation Evidence	This field lists the test assertions that are performed as implementation evidence.
Frequency	The frequency is a suggested time frame when the security function testing is done. For testing there is no fixed period because retesting has to be done whenever there is application change. This can happen when the application is upgraded.
Formula	The implementation evidence listing serves as an input in the formula to calculate the metric.
Data Source	Data source in our case is mostly the evaluation evidence that is RFC 3280 document. However, for the other metrics like user training and application documentation the date source can be the personnel training database or the organisation's repository.
Indicator	The indicator describes the meaning of the metric. If the metric is a percentage then the indicator will describe the implications when the metric is very low and when it tends to 100 percent.

The choice of this form was based on the fact that this standard provides sufficient coverage of the required description of the metric. This is useful for security functional testing of an application as it provides clear way for trucking application's changes and improving the whole testing process.

The SSE-CMM metric workgroup is currently developing security metrics. Their effort is also based on the NIST 800-55 standard therefore they have adopted the same metric form as the one described above [SSE-CMM 2003].

1.12.2 The security functional testing assertions of a PKI enabled application

Matt Bishop defines a security function testing as a "functional testing that is specific to security issues described in the relevant specification" [Bishop 2003 P534]. In our work the functional test assertions we present are based on the RFC 3280 evaluation evidence document. However CC asserts that more than one evaluation evidence can be used to evaluate one TOE [CC].

Table 1-1 Test assertions for PKI enabled applications

TESTS	**SUB-TESTS**	**REFERENCE RFC3280**
TSN	TSN1 Testing integer = 20 octets	4.1.2.2
Certificate's	TSN2 Testing integer less than 20 Octets	
Serial number testing	TSN3 Testing integer greater 20 Octets	
	TSN4 Testing integer = negative number	
	TSN5 Testing integer = zero	
TSINC	TSINC Series of Issuer name tests	4.1.2.4
Name chaining tests	TSINC Series of Subject name tests	
TS	TS1 Signature tests	4.1.2.3
Signature testing		
TVD	TVD Series of UTCTime validity tests series	4.1.2.5.1
Validity tests	TVD Series of GeneralizedTime tests	
TUID	TUID Series of tests	4.1.2.8
Unique identifiers tests		
TCE	TCE Series of subject key identifier tests	4.1.2.9
Certificate extension tests	TCE Series of Authority key identifier tests	
	TCE Series Key usage test	
	TCE Series Private key usage period tests	
	TCE Series Certificate policies tests	
	TCE Series Policy mappings tests	
	TCE Series Subject alternative name tests	
	TCE Series Issuer alternative name tests	
	TCE Series Subject directory attribute tests	
	TCE Series Basic constraints tests	
	TCE Series Name constraints tests	
	TCE Series Policy constraints tests	
	TCE Series Extended key usage tests	

TESTS	SUB-TESTS	REFERENCE RFC3280
	TCE Series CRL Distribution point tests	
	TCE Series Inhibit any policy tests	
	TCE Series Freshest CRL tests	

Table 1.2 is a representation of a summary of security functional tests that we will present in chapter 5. The application's ability to verify each of the above listed items correctly and continuously beyond reasonable doubt is necessary for the application to meet its security targets.

Recently researchers have started focusing their efforts to interoperability testing. In these efforts they try to involve vendors, testers and users so that communities of users can realize the benefits of using PKI. In Europe, the EC funded PKI interoperability project called pki Challenge involves universities, vendors and other organizations that use PKI to test PKI interoperability. One of its objectives is to disseminate to European communities the advantage of using PKI in e-commerce [pki C].

Figure 1.2 depicts a complex scenario where many components are linked. Users in such environment may face difficulties to understand processes that are important for the daily functioning of the system that requires attention in terms of improvements. [SSE-CMM 2003] describes the essential stages through which processes progress as they are specified, implemented, and maintained. The model provides the essential guide for selecting process to be improved and improvement strategies by determining the current base practice and capabilities of specific processes and identifying the issues most critical to quality and process improvement within a particular process. SSE-CMM also covers the following areas:

- Development and use of human resources
- Insertion of appropriate technology into products and
- Insertion of appropriate technology into tools used to produce various products.

SSE-CMM is still a new model and development efforts are still required especially in the area of metrics and human resource.

Testing and metrics are key factors for users to understand the following:

- The level of security the applications provide,
- Privacy, insurance and legal responsibility
- Conformance to standards
- Areas which require improvement
- Cost effectiveness of various testing methodologies
- Make decisions based on empirical data collected using metrics projects and
- Truck and manage systems changes

1.13 Related security testing and metrics research work

The Federal Bridge Certification Authority (FBCA) has developed five certificate policies for use by FBCA to support PKI interoperability with other PKIs [FBCA 2002]. In this project they are not examining any test assertions but they have developed certificate

policy that represents assurance levels for public key certificates. The assurance levels are: Rudimentary, Basic, Medium and High. These levels means how well relying parties in the interoperating PKIs can be certain of the identity of the certificate holder. Their work may be used to decide or classify critical and non-critical tests.

The pkiC project research work has taken into consideration the whole PKI environment [pkiC 2001]. They are testing the application, repository, communication, the certificate management systems and the environment. However, they could not combine the application testing and metrics the way like we have approached this problem [pkiC 2001]. Our method is likely to give improved results as the test results can be easily evaluated and compared. Ammann and Black, in their famous paper "A specification based coverage metrics to evaluate test sets"[Ammann 1999] they developed a methodology to test the coverage of test sets using metrics during testing high assurance applications using formal methods. Their objective was to compare test generation methods, evaluating the coverage of systems tests and minimizing the test sets [Ammann 1999]. Further related work has been done by [Stal 2000] who developed security targets for Entrust, and the security metrics development in the X.509 Certificate Policy project for the Federal Bridge Certificate Authority [FBCA 2002].

1.14 Limitations in the Thesis

Due to the broad nature of this topic we are examining, we need to define the limitations. There are many areas of interest for testing and evaluation and metrics of PKI systems. Our work is limited to testing PKI applications ability to correctly verify X.509 certificate and CRL version 3. This involves the analysis and verification of X.509 Version 3 public key certificate, Certificate Revocation Lists CRL version2 evaluation evidence as depicted in Figure 1.5 label number 1 and 5. Specification documents of the target of evaluation (TOE) are referred to as evaluation evidence in [CCIMB-99-013]. Label 5, indicates a hierarchical trust model. The hierarchical trust model comprises of the top CA and a registration CA. However, other CAs can be included between the RCA and the TCA, as it will deem necessary in different tests assertions [RFC 3280].

The choice of the above mentioned component is based on the fact that public key certificate and CRL validity is central and critical in any PKI system. The measure of how much trust should be placed in a particular PKI system is based on the validity of the certificates in use. Therefore, every application of PKI must be capable of verifying public key certificates chain beyond reasonable doubt.

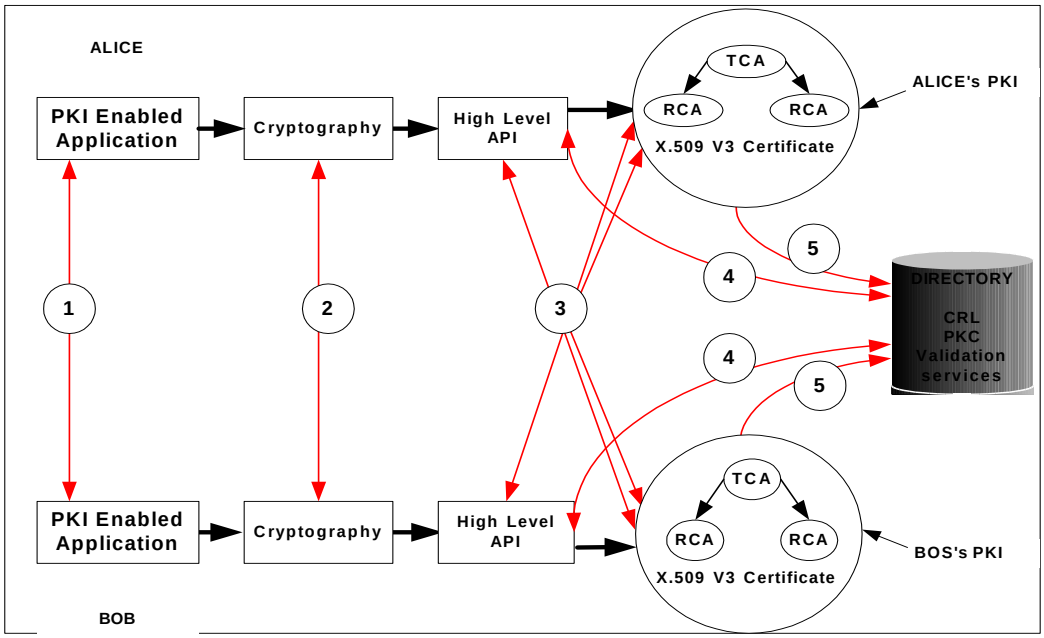


Figure 1.5 PKI Interoperability

Figure 1.5 depicts two interoperating PKIs. The labels and arrows are interpreted below:

Label 1: The double-sided arrow indicates operational transaction between the two PKI enabled applications exchanging information securely using the underlying cryptographic algorithms and the PKI. The application has one critical function of verifying the chain of the public key certificate to prove whether it can be trusted or not be trusted. The main purpose of our work in this area is to develop test assertion that can be used by users, developers and testers to verify the applications capability to validate public key certificates.

Label 2: The double-sided arrow indicates that the cryptographic algorithms will be negotiated by the communicating applications. The scope of this thesis does not include the cryptographic algorithms testing. However, CC recommends that if the cryptographic algorithms are part of the TOE, then the testing scheme should provide

Label 3: The three double-sided arrows indicate that the two PKI will have to exchange cross certificates in order to be able to interoperate. The cross certificate will eventually be used by a user from one PKI to communicate with users from another PKI.

Label 4: The double-sided arrow is meant to indicate that there are operational transactions as well as management transactions that end entities perform in the repository. This involves verifying the status of the certificate and updating the repository.

Label 5: The one sided arrow indicates that the Registration CA and the Top CA publish CRL and certificates in the repository Certificate path processing and includes determining that the certificate has been issued by a recognized trust anchor or its trusted subordinate, the digital signature of the certificate is valid, the certificate is within its stated validity period, the certificate has not been revoked, the certificate is being used in

a manner which is consistent with its policy constraints, name constraints, and intended usage restrictions.

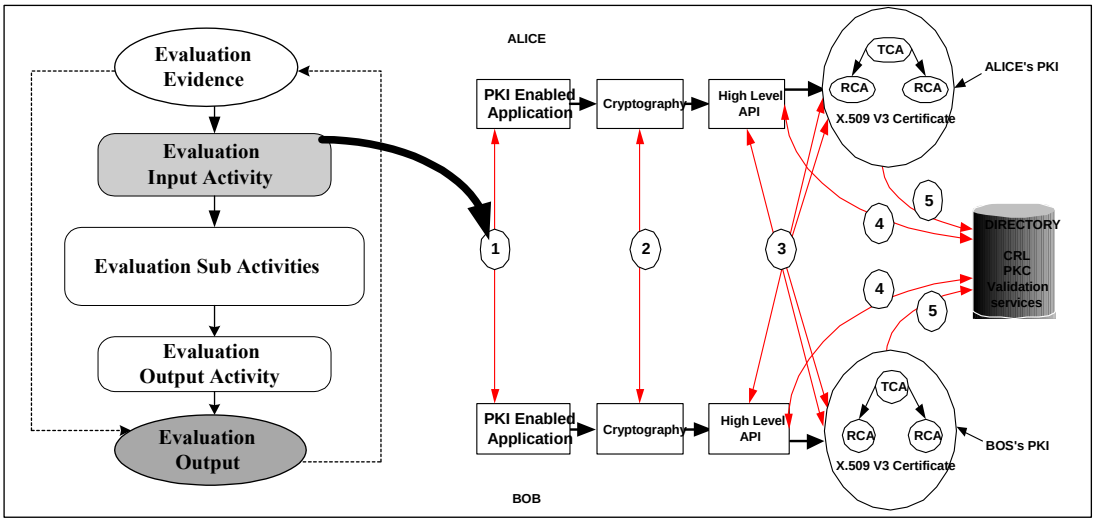


Figure 1.6 Relationship of the general evaluation model to the research limitation

Figure 1.6 illustrates the relationship between the general evaluation model and the limitations in our research. The evaluation input activity is expanded into subsections that are labeled 1 through 5. From labeled 1 through 5 our work is further limited only to the PKI enabled applications ability to validate the X.509 certificates in its life cycle time. The PKI applications testing assertions with its associated metrics are presented in chapter 4.

1.15 Thesis outline

The structure of this report is as follows:

Chapter 1.
In this chapter we present IT security overview, PKI concept, the purpose of our research, security testing, security metrics, general evaluation model, Common Criteria overview, Systems Security Engineering Capability Maturity Model (SSE-CMM), the research background information, and the limitations of this research.

Chapter 2.
In this chapter we present the security metrics importance argument to sustainability of large scale PKI and a guide how to develop security metrics for PKI applications testing.

Chapter 3
In this chapter we present the guideline for developing test assertions for a PKI enabled application ability to correctly verify X.509 certificates Version3 and CRL Version 2 certificates testing.

Chapter4
In chapter 4 Presents the thesis Conclusion, reflections, remarks, and further work

References

Appendix
Appendix A: lists acronyms.
Appendix B: PKI Cryptography Technologies.
Appendix C: Common Criteria (CC) and SSE-CMM.
Appendix D: PKI large-scale Implementation problems.
Appendix E: PKI Interoperability based on various approaches.

2. Chapter 2

2.1 Security Testing Metrics

This chapter is one of the core chapters that present the PKI application security metrics. These metrics can be used in evaluating PKI enabled applications and the PKI application environment. More generic security metrics are presented in [Swanson 2003] and [SSE-CMM 2003].

IT Security metrics are tool that facilitate improved understanding, performance, Coverage, and decision making of various security processes, mechanisms and procedures [Swanson 2003] [Jelen 2000]. This can be achieved through collection of data, data analysis, and reporting of performance related data [Jelen 2000]. IT security testing metrics data must be based on the TOE evidence, survey forms, repositories, and organisation's security staffs database, incidents and incidents responses logs and interviews [Swamnson 2003]. The TOE evidence is a requirement specification document that states the desired security result of the system security program implementation [Swanson 2003], [Ammann 1999].

2.2 Importance of metrics

IT security metrics should be designed to yield quantifiable information for the following [Swanson 2003], [Ammann 1999], [FBCA 2002], [SSE-CMM 2003]:

- Comparison purposes
- Tracking changes using the same point of reference and apply mathematical formulas for analysis.
- Coverage measurements
- Cost justification
- Indication and determination of critical and non critical security parameters and test cases
- Redirect assets securing efforts
- Security problem isolation and
- Determine the effectiveness of security testing efforts

Based on above listed facts IT security metrics can be created to guide each aspect of testing efforts including PKI systems interoperability testing and evaluation, risk assessment, penetration testing, and security testing and evaluation. The use of IT metrics will allow organisations to determine effectiveness of implemented IT security processes, and control by relating results of IT security activities measurements [SSE-CMM 2003].

2.3 Related work

The International Systems Security Engineering Association (ISSEA) comprises the Metrics Work Group that is tasked to develop metrics for [SSE_CMM 2003]. The work group has adapted the methodology for metrics development that is defined in [Swanson 2003]. The work group has proposed 22 Process Areas (PA) for metrics development [SSE-CMM 2003]. The PAs are as follows:

Table 2-1 Security process and security metrics areas as defined in SSE-CMM [SSE-CMM 2003]

Process Areas	Process Areas Description
PA01	Administer Security Control
PA02	Assess Impact
PA03	Assess Security Risk
PA04	Assess Threat
PA05	Assess Vulnerability
PA06	Build Assurance Argument
PA07	Coordinate Security
PA08	Monitor Security Posture
PA09	Provide Security Input
PA10	Specify Security Needs
PA11	Verify and Validate Security
PA12	Ensure quality
PA13	Manage configurations
PA14	Manage Project Risks
PA15	Monitor and Control Technical Efforts
PA16	Plan Technical Efforts
PA17	Define Organisation's Systems Eng. Process
PA18	Improve Organisation's Systems Eng. Process
PA19	Manage product line evaluation
PA20	Manage Systems Eng. Support Environment
PA21	Provide ongoing skills and knowledge
PA22	Coordinate with suppliers

In our work the PA01 to PA12 will be used. These are important metrics to ensure quality-testing process as we have explained in section 2.2. However, the development of these metrics is an ongoing work by the SSE-CMM Security Metrics Work Group. Our metrics are specific for testing the ability of a large scale PKI application to validate the X.509 certificate in its lifetime. Our work is also related to the NIST 800-55 standard [Swanson 2003]. The relationship between NIST 800-55 standard, and the SSE-CMM is that SSE-CMM has adopted the NIST 800-55 methodology of developing security and process metrics. More related work explanation is given in section 1.13.

2.4 Metrics and measurements

There are two types of metrics namely process metrics and security metrics [Jelen 2001], [SSE-CMM 2003]. The relationship between the two types of metrics is depicted in figure 2.1. The process metrics are some measures that can be used as evidence of the maturity of the security engineering process area and the security metrics indicate the extent which some security attribute for example confidentiality, integrity, non repudiation, access control and availability is present in the security engineering process. In metrics units like absolute numbers are sometimes useful, percentage, binary, and averages are most common [SSE-CMM 2003].

The difference between metrics and measurements is that metrics are function of measurements and time. That is they are obtained by taking measurements over time.

Other attributes are: metrics should be specific, measurable, comparable, attainable, repeatable, and time dependant. Measurements provide one time view of specific measurable parameters and are represented by numbers, binary statements and weights [Jelen 2000]. Metrics are useful because when two or more measurements are compared with predefined baseline measurements, over a period of time, provides a means for interpretation of collected data [Jalen 2000].

2.5 Relationship between processes and security metrics

A process is a sequence of steps performed for a given purpose [SSE-CMM 2003]. Security engineers have to follow the processes and procedures to configure and test the system correctness for security enforcement that the system is designed for. As a result of testing process the system's security criteria will be quantified using security metrics which eventually can be used to determine whether the security risk is tolerable or not.

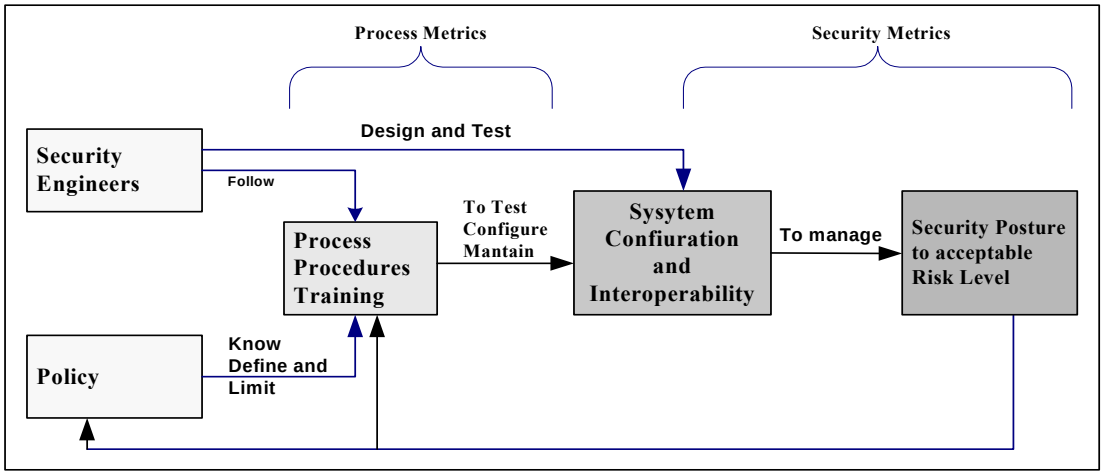


Figure 2.1 Relationships between Process and Security Metrics

2.6 Mapping metrics to testing process area

IT Security metrics development process consists of identification and definition of IT security testing process and selection of metrics to measure test coverage, implementation, efficiency, effectiveness, and the impact of the security control. This procedure involved identification of security threats and classifying the threats to determine what are critical test and non-critical tests [FBCA 2002]. Mapping process area to metrics process area ensures that all threats, threats agents, assumptions, breaches, or mistakes that could potentially lead to breach of security are identified and documented.

The testing processes can be testing an application security features, security incident handling, security training, systems documentation, security system's configuration, certification, protection profiling etc. In the case of testing, the test cases must be identified and documented as accurate as possible so that coverage analysis becomes reliable.

The coverage analysis will be reliable if different standards and encoding are taken into consideration during test cases selection.

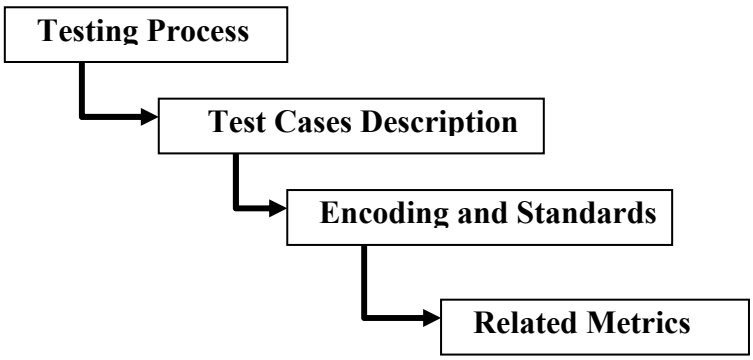


Figure 2.2 Mapping process areas to metrics

2.7 Comparing Security metrics models

Methods to develop security metrics have been under development for several [SSE-CMM 2003], [Swanson 2003], [Jalen]. Currently we have several metrics development methods of which the Department of Defence (DoD) IT Performance assessment method, Stakeholder-Based model and the Information Assurance Technology Analysis Centre (AITAC) capabilities-based model are widely discussed [Jelen].

The IATAC capabilities based model is a product of [SSE-CMM 2003] international metric project. It addresses the functional capabilities: Protect, Detect, and Respond. IATAC defines required performance of the best practices to generate specific results.

The DoD IT performance assessment methodology has three components namely: capabilities, Attribute Level, and specific metrics [Jalen 2000]. The attribute level addresses the requirement that support that mission and the Specific metrics component addresses specific measurable activities that support those mission requirements. The Stakeholder Based Model views metrics from an organisational role perspective: Stockholders, Stockholders responsibility, stockholders interest and actions [Jalen 2000].

2.8 Data collection methods and security metrics in PKI environment

Data collection for metrics development can be accomplished through systematic review of the evaluation evidence of TOE and policy documents, automated data gathering tools, repositories, survey and interview, system configuration verification and observation [Swamnson 2003]. In the case of PKI application the data sources includes the X.509 Certificate standard documents.

The domain of metrics that should be generated from existing evidence of TOE and policies and specification will be quite large. Therefore, the selection of metrics should base on the criteria of threats that are being mitigated, assumptions made, security services, and criticality of the test cases. However, the classification of a test case to be critical or non-critical depends on the threats to mitigate and the security service the object under test provides.

2.9 Use of metrics

In this section we present metrics development assertions that are suitable for PKI environments. The metrics development assertions we present here can be used by vendors, evaluators and users to generate quantifiable results that can be used to determine the effectiveness, coverage and identification of areas that require improvement. The metric we are present in Table 2.5 can be used to quantify the test cases coverage when testing a PKI enabled application ability to validate correctly x.509 certificates. However, this metric can be customized for use in security functional testing of any other application.

Other metrics development assertions are useful in PKI operational environments. This metrics includes certificate revocation, privacy, PKI documentation, PKI user and support personnel training and incident recording and response [Swanson 2003].

2.10 Metric Development

2.10.1 PA04 Threats assessment

This section examines the threats that can face interoperating PKI applications due to a PKI application that cannot verify certificates correctly in its life cycle. The threats are listed in Table 2.

Table 2-2 Security threats addressed in each test case

No	Test Case	Threat to be mitigated
1	**Certificate serial number**	A non-standard certificate serial number other than 20 octets will disrupt interoperability. If CA key is compromised a malicious user may use it to post false CRL lists. CRL lists certificate serial numbers.
2	**Signature**	A malicious user may sign certificates in order to gain access to unauthorized assets or enable unauthorized users to gain access to unauthorized assets
3	**Issuer name**	Issuer name testing is to make sure a malicious user does not issue certificate in the chain. If this is compromised malicious entity may gain access to assets
4	**Subject name**	Subject name testing is to make sure a malicious user does be issued certificate in the chain. If this is compromised malicious entity may gain access to assets
5	**Validity period**	Threats that are being addressed by this testing include unauthorised access to assets by terminated employees, business partners whose association does no longer exist, accessing resources before or after authorised time, loss of legal rights.
6	**Issuer uniqueID**	Malicious user/issuer does not issue certificate in the chain. If this is compromised malicious entity may gain access to assets

No	Test Case	Threat to be mitigated
7	**Subject uniqueID**	Malicious user is not issued certificate in the chain. If this is compromised malicious entity may gain access to assets
8	**Authority key Identifier**	Issuers with multiple signing keys need key identifiers for identifying public key which correspond to a particular private key. NC
9	**Subject key Identifiers**	This is for identifying certificates that contain a particular public key. NC
10	**Key usage**	An end entity can deny having used the certificate to commit a transaction like
11	**Certificate policy**	This test makes sure certificates are used in no transaction other than the stated purpose of use. If this is compromised malicious users may gain access to unauthorized assets.
12	**Policy mapping**	This mapped subject policy to the issuer CA policy. A malicious user may use a certificate in a way that the CA policy does not allow. This may cause access to unauthorized assets.
13	**Subject alternative name**	Threats associated with alternative names are: Access to unauthorized assets and loss of privacy
14	**Issuer alternative name**	Threats associated with alternative names are: Access to unauthorized assets, loss of privacy
15	**Subject directory attributes**	This is involved with identity like nationality. It is presented for interoperability purposes. NC
16	**Basic Constraints**	Indicated the maximum depth of valid certificate path. Threats associated with basic constraints are if the CA key is compromised or malicious CA may issuer certificate that can appear valid in the path. Eventually there will be loss of integrity and confidentiality and nonrepudiation. C
17	**Name constraints**	Is used in CA certificates to indicate name space will be located in all certificates. The name constraint can be applied for url or a particular mail address. Threats associated with this tests are unauthorized access of assets
18	**Policy constraint**	Prohibits policy mapping .It is useful if policy mapping policy is to be applied for a limited number of certificates in the path. Threats associated with this are use of certificate to access unauthorized assets. C
19	**Extended key usage**	Extended key usage purpose must be used for a purpose consistent with other extensions. Threats associated with this test are unauthorized access of assets, loss of confidentiality.
20	**CRL Distribution point**	Threats associated with this test are loss of integrity, loss of confidentiality and nonrepudiation
21	**Inhabit any policy**	This allows a number of CA s certificates to inhibit any policy with no explicit match for other certificates policy an OID with a number 2, 5, 29, 32 or 0 will indicate the number of certificate in the path on which this policy apply. Threats associated with this test may be an entity may use a certificate to access un-authorized assets.

No	Test Case	Threat to be mitigated
22	**Freshest CRL**	This test is intended to test the ability of an application to obtain Freshest CRL. It is presented here for completeness and interoperability purpose.
23	**TbsCertList**	Threats associated with this test is that if the date and time and serial number of the revoked certificate cannot be verified correctly there is a possibility for committing transactions using certificates that are compromised. Hence loss of access control, integrity, authentication, confidentiality and nonrepudiation
24	**Certificate list to be signed**	Threats associated with this test is that if the date and time and serial number of the revoked certificate cannot be verified correctly there is a possibility for committing transactions using certificates that are compromised. Hence loss of access control, integrity, authentication, confidentiality and nonrepudiation
No	**Test Case**	**Threat to be mitigated**
25	**CRL Signature**	A malicious CA may sign CRL in order to disrupt business
26	**CRL Issuer name**	Issuer name testing is to make sure a malicious CA does not issue CRL. If this is compromised malicious entity may disrupt business
27	**CRL This update**	Threats associated to this test is loss non repudiation security service
28	**CRL Next update**	Threats associated to this is the nonrepudiation security service and loss of integrity and confidentiality security services if the next update time is violated
29	**Revoked Certificate**	Threat associated to this test are all security services: Integrity, nonrepudiation, confidentiality, authentication, and availability, and access control
30	**CRL Issuer alternative name**	Threats associated with alternative names are: Access to unauthorized assets
31	**Delta CRL Indicator**	Threats associated to this test are loss of availability. This indicate delta CRL that are useful to reduce network traffic.
32	**CRL reason code**	Threats associated with this may be loss of integrity, confidentiality, non repudiation, and authentications
33	**CRL Invalid date**	Threats associated with this testing are that users with no valid CA association may gain access to assets.

2.10.2 Test cases criticality

The sensitivity of various test cases will vary significantly. The variation will depend on the threats a particular test attempts to mitigate. This section gives the summary of the level of risks associated with each test case. To provide granularity, [FBCA 2002] specifies four levels of security requirements: Rudimentary, Basic, Medium, and High. The summary of these levels is presented in Table 2.2. Table 2.3 presents the classification of test cases into assurance levels.

Table 2-3 Test cases criticality metrics classification
[FBCA 2002]

No.	Assurance Level	Applicability in test cases
1	***Test***	This test item is presented for interoperability testing. It does not mitigate any security threat.
2	**Rudimentary**	Tests that are classified in this level provide data integrity security service and interoperability.
3	**Basic**	Tests classified in this level are those involving objects that mitigate threats associated with data compromise and interoperability.
4	**Medium**	Tests classified in this level are those involving objects that mitigate threats that may cause data compromise, fraudulent data access and malicious user and interoperability.
5	**High**	This level applies to tests whose failure cause high consequences and interoperability.

Table 2-4 Security threats and loss of security service

No	Test Case	Threat Vs Security service						Assurance Level			
		Integrity	Confidentiality	Authentication	Availability	Access control	Nonrepudiation	Rudimentary	Basic	Medium	High
1	**Certificate serial number**							X			
2	**Signature**	X	X	X		X	X	X	X	X	X
3	**Issuer name**	X	X	X		X	X	X	X	X	X
4	**Subject name**	X	X	X		X	X	X	X	X	X
5	**Validity period**		X	X		X	X	X	X	X	X
6	**Issuer uniqueID**		X	X		X	X		X	X	X
7	**Subject uniqueID**		X	X		X	X		X	X	X
8	**Authority key Identifier**					X		X	X	X	X
9	**Subject key Identifiers**					X		X	X	X	X
10	**Key usage**		X			X	X	X	X	X	X
11	**Certificate policy**		X			X	X	X	X	X	X
12	**Policy mapping**		X			X	X		X	X	X
13	**Subject alternative name**			X		X	X		X	X	X
14	**Issuer alternative name**			X		X	X		X	X	X
15	**Subject directory attributes**			X				X			
16	**Basic Constraints**		X	X		X	X		X	X	X
17	**Name constraints**		X				X		X	X	X
18	**Policy constraint**		X	X		X	X		X	X	X
19	**Extended key usage**		X						X	X	X
20	**CRL Distribution point**							X	X	X	
21	**Inhabit any policy**		X	X		X	X	X	X	X	X
22	**Freshest CRL**								X	X	X

23	**TbsCertList**							**X**			
24	**Certificate list to be signed**		**X**	**X**		**X**	**X**		**X**	**X**	**X**
25	**CRL Signature**			**X**					**X**	**X**	**X**
26	**CRL Issuer name**			**X**					**X**	**X**	**X**
27	**CRL This update**		**X**	**X**			**X**		**X**	**X**	**X**
28	**CRL Next update**		**X**	**X**		**X**	**X**		**X**	**X**	**X**
29	**Revoked Certificate**	**X**	**X**	**X**		**X**	**X**	**X**	**X**	**X**	**X**
30	**CRL Issuer alternative name**			**X**			**X**		**X**	**X**	**X**
31	**Delta CRL Indicator**							**X**			
32	**CRL reason code**							**X**			
33	**CRL Invalid date**		**X**				**X**		**X**	**X**	**X**

Table 2.3 presents the security services that can be provided by PKI [Stal 2000], [RFC3280] and security metrics that are useful for measuring the impact and effectiveness of a security process. These metrics can also be useful to determine critical test assertions and non-critical test assertions when developing test cases in chapter 3.

2.10.3 Application functional testing coverage metrics

This table will be used to list all the test cases and sub test cases. It is intended to provide details of test cases coverage. Test cases coverage analysis is useful to reveal accidental mistakes that may eventually lead to serious flaws in the application.

Table 2-5 PA11 PKI Application functional testing coverage metrics

Testing Goal	To determine the percentage of coverage of a functional test.
Associated question	Is the application's function implicitly defined in the TOE?
Metric	Percentage of the functional test cases selected out of all test cases.
Purpose	To ensure that maximum coverage of critical and non-critical test cases is attained.
Implementation Evidence	• Is the functional test clearly defined in the TOE? Yes or No • If the answer to the question above is yes list all possible test cases 1. _____ 2. _____ 3. _____ • _____ • _____ • _____ n. _____ • Total number of test cases defined_____
Frequency	Annually/During testing/During retesting
Formula	Number of test cases performed divide by Total number of test cases
Data source	TOEs
Indicator	The target is 100 percent. Increasing percent is an indication of

	higher coverage of test cases. The higher the coverage the slower the probability of skipping critical test cases.

Remarks: The average of the aggregate percentage of individual functional test will be the coverage percentage of the overall application functional testing. This metric can be used to measure the quality of testing process and in case the application is upgraded or patched, the metric can be used to make comparison between existing results and the new retesting results.

2.11 Metrics for testing processes other than PKI application

In any PKI environment there are many other process which in one way or the other affects interoperability. These processes are related to certificates revocations, user education, application upgrading, security incident handling and the documentation. In this section we present metrics that can be used by users/support personnel/management to measure these processes for improvement and proper planning.

2.11.1 Certificate revocation incident handling metrics

This metric is useful for tracking certificates revocations in the CA's and user's organisation. Revoked certificates tracking is useful to make sure that the CA includes the revoked certificates in the CRL in timely manner based on the existing certificate revocation policy. Delayed revocation execution after the certificate user requests it has serious consequences that the malicious user who gains access to the private key of the user may gain access to organisations assets.

This metric is useful in PKI environment. It is not related to testing done in chapter 5. However, it is presented here for completeness and interoperability improvement.

Table 2-6 SM-CR1 Certificate revocation incident handling metrics [Swanson2003]

Testing Goal	To test if there is a capability to provide support to user who request certificate revocation
Associated question	Is certificate revocation incident response procedure available and well defined?
Metric	Percentage of certificates revoked within defined time
Purpose	To ensure that there is certificates revocation process capability
Implementation Evidence	• Is there a formal process or document that defines incident and describes how to report key compromise incident? • Are Certificates revocation incidents monitored and tracked until the certificate is revoked? • Number of certificate revocation request during a given period_____
Frequency	Monthly, Quarterly, Semiannually, annually
Formula	Number of certificates whose private keys has been

	compromised but the revocation could not be done timely divide by Total number of certificate revocation request
Data source	Repository/Database
Indicator	The target is 100 percent. Increasing percent is an indication of maturity. Certificate revocation request incident handling capability is important for adequate security. When certificates have to be revoked gracefully certificate revocation requests must be processed within the grace period.

Remarks: Remarks: Other associated questions could cover incident handling for penetration attempts, hardware failure, applications failure and virus attacks. These can be used to estimate the organisation's overall incident handling capability.

2.11.2 Systems Accreditation metrics

Tracking systems accreditation is useful to foster interoperability in the user environment. This metric should reveal systems that are not accredited as an indication of lack of procedure, testing, and policy.

Table 2-7 SM-SA1 PKI system accreditation metric
[Swanson 2003]

Testing Goal	To test if PKI system's components are accredited
Associated question	Is accreditation policy available? Are formal procedures for interoperability in place?
Metric	Percentage of total PKI system component that have been accredited
Purpose	To bring up awareness and prevent use of un accredited systems
Implementation Evidence	• Is accreditation in the organisation's policy • Are all interoperability connections accredited • Is there a systems inventory? • Is the inventory maintained and updated? • Total number of systems ___
Frequency	Quarterly, Semiannually, annually
Formula	Number of systems that are accredited divide by Total number of systems
Data source	Systems inventory database
Indicator	The target is 100 percent. Increasing percent is an indication of organisation's policy implementation level and the management involvement in ensuring systems interoperability capability.

Remarks: Accreditation shows that a system has been tested as required and users accept legal responsibility. Internal management accreditation policy is useful to make sure PKI systems are sustainable.

2.11.3 Maintenance of hardware and software metric

This metrics can be used to measure PKI application hardware conformance and approval. This metrics is important because the PKI application platform malfunction can affect interoperability.

Table 2-8 SM-HS1 Hardware and system software maintenance
[Swanson 2003]

Testing Goal	To determine if new and revised hardware and software are authorized, tested, and approved before implementation.
Associated Question	Are software change request forms used to document requests and related approval?
Metric	Percentage of software changes documented and approved through change request
Purpose	To determine the level of software configuration changes that are documented and approved
Implementation Evidence	• Do you have a policy in place for requesting and tracking software changes on systems and obtain approval for each change? Yes? No? If yes do you have automated system track change history and approval? • Number of software changes or updates that occurred during reporting period _____ • Number of changes that have a corresponding documented software change request form_____
Frequency	Quarterly, Semiannually, annually
Formula	Number of documents approved software changes with form divide by Total number of software change
Data source	Configuration management database or software change request form documentation
Indicator	The target is 100 percent. Software change should be controlled, tested and approved. Lack of this increases the complexity of PKI system, version control and security updates that must be applied to the system.

Remarks: One of the identified interoperability problem of LSPKI is unmanaged system change. This metric can be useful for organizations to make sure application's change is managed and therefore minimize eventual interoperability problems

2.11.4 **PKI application documentation**

This metric should reveal either presence or lack of documentation on PKI systems available in the user's environment. Lack of documentation may affect interoperability simply because users cannot refer to product's documents when they face situations they don't understand.

SM-D1 PKI Systems documentation metric

Table 2-6 PKI applications documentation metric
[Swanson 2003]

Testing Goal	To determine if there sufficient documentation explaining how PKI application has been tested or retested
Associated question	Is there PKI application testing documentation?
Metric	Percentage of applications with documentation on file
Purpose	To make sure that testing process can be repeated, compared and hence improved
Implementation Evidence	• How many PKI applications are in the inventory? • How many applications have test results maintained on the file?
Frequency	Semiannually, annually
Formula	Number of application with test documentation on file to Total number of PKI applications
Data source	Documentation repository
Indicator	The target is to 100 percent. As the percentage approach 100 it is an indication good best practice and the application test coverage can be compared in case the application is retested.

Remarks: The documentation metric can be applied further to determine the percentage of user and technical manuals available for users.

2.11.5 Privacy metric

The Common Criteria defines the privacy as anonymity, pseudonym, unobservability and unlinkability [CC]. The metric we provide in this section is limited to testing if the private information contained in X.509 certificates is protected. Users are concerned about disclosure of their e-mail addresses and names when their certificates are posted in the PKI repository.

Table 2-7 SM-P1 Privacy metric

Testing Goal	To test if there is a capability to protect the disclosure of private information on certificates
Associated question	Is formal incident response capability available?
Metric	Percentage of certificates in the repository with subject names unencrypted
Purpose	To ensure that private information on certificates is not disclosed
Implementation Evidence	• Is there a formal procedure to make sure private information of individuals and the organisation is protected before certificates are placed in the repository? • Number of certificates in the repository_____
Frequency	Quarterly, Semiannually, annually

Formula	Number of certificates with unprotected private information divide by Total number certificates in the repository
Data source	Repository
Indicator	The target is 100 percent. Increasing percent is an indication of user awareness on privacy.

2.11.6 Personnel training metrics

PKI Security personnel training are essential for interoperable PKI system. Training metrics should reveal whether users and support personnel have adequate training on use and maintenance of PKI system.

Table 2-8 SM-PT1 PKI Personnel training metrics
[Swamnson 2003]

Testing Goal	To test if users and support personnel have received adequate training on PKI
Associated question	Are the users and support personnel trained, examined and certified on the use and support of PKI?
Metric	Percentage of personnel who have been trained, examined and certified how to use and support PKI
Purpose	To measure the level of expertise among the personnel who use and support PKI
Implementation Evidence	• Are security qualifications criteria defined? • Are training records maintained? (Type of training and certificates) • How many staffs who use and support PKI have received training_____
Frequency	Semiannually, annually
Formula	Number of users and support staffs who have received PKI training divide by Total number of PKI users and support staff
Data source	Training records
Indicator	The target is 100 percent. If users and support staffs have no adequate training on how to use and support PKI the organisation will lack the necessary advantage PKI brings to its overall security.

Remarks: Advantages of adequately training staffs includes reduced maintenance costs, easy interoperability, reduced incident response time and understanding of legal environment.

2.12 Chapter 2 summary

The lesson learnt from the metrics development assertions is that testers should be able to define the coverage of test process. A hundred percent coverage is difficult to achieve. However, testers should be able to quantify using metrics the coverage of the test cases. This can be achieved using metrics that can be developed prier to the actual testing. This is useful not only for improving the whole test process but also for costing justification, user understanding of the test process, and for comparison of results whenever the product has to be reevaluated.

3. Chapter 3

3.1 PKI application ability to Validate Certificates testing

This chapter presents the PKI application functional testing assertions that are essential for verifying the application's ability to validate X.509 certificates in its entire life cycle. The application's ability to validate certificates correctly is central for large scale PKIs interoperability. As stated in section 1.5, the CC will be used as a guide in this process. The PKI interoperability problems and CC are detailed in Appendices C and D.

The CC presents security functional requirements that are the current state of the art in requirements specification and evaluation [CCIMB-99-031]. The security functional components express security requirements intended to address threats in the assumed operating environment of the TOE [CCIMB-99-031]. The CC security functions are outlined in Table 3.1.

Table 3-1 Security function
[CCIMB-99-031]

No	Security Function Classes	Security Function Family	Remarks
1	**FAU: Security audit**	Security audit automatic response (FAU_ARP) Security audit data generation (FAU_GEN) Security audit analysis (FAU_SAA) Security audit review (FAU_SAR) Security audit event selection (FAU_SEL) Security audit event storage (FAU_STG)	
2	**FCO: Communication**	Non-repudiation of origin (FCO_NRO) Non-repudiation of receipt (FCO_NRR)	
3	**FCS: Cryptographic support**	Cryptographic key management (FCS_CKM) Cryptographic operation (FCS_COP)	
4	**FDP: User data protection**	Access control policy (FDP_ACC) Access control functions (FDP_ACF) Data authentication (FDP_DAU) Export to outside TSF control (FDP_ETC) Information flow control policy (FDP_IFC) Information flow control functions (FDP_IFF) Import from outside TSF control (FDP_ITC) Internal TOE transfer (FDP_ITT) Residual information protection (FDP_RIP) Rollback (FDP_ROL) Stored data integrity (FDP_SDI) Inter-TSF user data confidentiality transfer protection (FDP_UCT) Inter-TSF user data integrity transfer protection (FDP_UIT)	
5	**FIA: Identification and authentication**	Authentication failures (FIA_AFL) User attribute definition (FIA_ATD) Specification of secrets (FIA_SOS) User authentication (FIA_UAU) User identification (FIA_UID) User-subject binding (FIA_USB)	

No	Security Function Classes	Security Function Family	Remarks
6	**FMT: Security management**	Management of functions in TSF (FMT_MOF) Management of security attributes (FMT_MSA) Management of TSF data (FMT_MTD) Revocation (FMT_REV) Security attribute expiration (FMT_SAE) Security management roles (FMT_SMR)	
7	**FPR: Privacy**	Anonymity (FPR_ANO) Pseudonymity (FPR_PSE) Unlinkability (FPR_UNL) Unobservability (FPR_UNO)	
8	**FPT: Protection of the TSF**	Underlying abstract machine test (FPT_AMT) Fail secure (FPT_FLS) Availability of exported TSF data (FPT_ITA) Confidentiality of exported TSF data (FPT_ITC) Integrity of exported TSF data (FPT_ITI) Internal TOE TSF data transfer (FPT_ITT) TSF physical protection (FPT_PHP) Trusted recovery (FPT_RCV) Replay detection (FPT_RPL) Reference mediation (FPT_RVM) Domain separation (FPT_SEP) State synchrony protocol (FPT_SSP) Time stamps (FPT_STM) Inter-TSF TSF data consistency (FPT_TDC) Internal TOE TSF data replication consistency (FPT_TRC) TSF self test (FPT_TST)	
9	**FRU: Resource utilization**	Fault tolerance (FRU_FLT) Priority of service (FRU_PRS) Resource allocation (FRU_RSA)	
10	**FTA: TOE access**	Limitation on scope of selectable attributes (FTA_LSA) Limitation on multiple concurrent sessions (FTA_MCS) Session locking (FTA_SSL) TOE access banners (FTA_TAB) TOE access history (FTA_TAH) TOE session establishment (FTA_TSE)	
11	**FTP: Trusted path/channels**	Inter-TSF trusted channel (FTP_ITC) Trusted path (FTP_TRP)	

Table 3.1 summarises the eleven security functional classes as defined in the Common Criteria [CCIMB-99-031]. They are presented here for understanding and mapping purposes with the security functions we are attempting to test in PKI application.

3.2 Security family and component structures

Table 3.2 Summarises the CC functional class' Family mapping to security functional component test cases. All test cases emanating from the list of security function component structures listed above will be grouped based on the security functional family mapping presented in this table. The security threats addresses by these security components are presented in table 2.3.

Table 3-2 Functional Family mapping to security functional component test cases

Test Case: Security function Component structure	CC Security Function Family
Certificate serial number	Revocation (FMT_REV)
Signature	User authentication (FIA_UAU) Data authentication (FDP_DAU)
Issuer name	User identification (FIA_UID)
Subject name	User identification (FIA_UID)
Validity period	Security attribute expiration (FMT_SAE)
Issuer uniqueID	User identification (FIA_UID)
Subject uniqueID	User identification (FIA_UID)
Authority key Identifier	User identification (FIA_UID)
Subject key Identifiers	User identification (FIA_UID)
Key usage	User-subject binding (FIA_USB)
Certificate policy	Access control policy (FDP_ACC)
Policy mapping	Access control policy (FDP_ACC)
Subject alternative name	User identification (FIA_UID)
Issuer alternative name	User identification (FIA_UID)
Subject directory attributes	User identification (FIA_UID)
Basic Constraints	Access control policy (FDP_ACC)
Name constraints	User identification (FIA_UID)
Policy constraint	Access control policy (FDP_ACC)
Extended key usage	Access control policy (FDP_ACC)
Inhabit any policy	Access control policy (FDP_ACC)
CRL Distribution point	Revocation (FMT_REV)
Freshest CRL	Revocation (FMT_REV)
TbsCertList	Revocation (FMT_REV)
Certificate list to be signed	User authentication (FIA_UAU)
CRL Signature	User authentication (FIA_UAU)
CRL Issuer name	User identification (FIA_UID)
CRL This update	Security attribute expiration (FMT_SAE)
CRL Next update	Security attribute expiration (FMT_SAE)
Revoked Certificate	Revocation (FMT_REV)
CRL Issuer alternative name	User identification (FIA_UID)
Delta CRL Indicator	Revocation (FMT_REV)
CRL reason code	Revocation (FMT_REV)
CRL Invalid date	Security attribute expiration (FMT_SAE)

3.3 X.509 Certificate

X.509 certificate is defined in ITU-T Recommendation X.509 and further defined in Internet Engineering Task Force (IETF) Request for Comments [RFC 3280]. To ensure secure interoperation of PKI-enabled applications that makes use of X.509 certificates, the path validation must be done in accordance to these specifications.

X.509 is part of X.500 series of ITU-T recommendations that define a directory service. X.509 is based on the use of public key cryptography and digital signature. The directory is a server or distributed set of servers that maintains a database of information about users. It provides authentication services by the X.500 directory to its users. The authentication protocols defined in X.509 are used in S/MIME, IP security, SSL/TLS, and SET [RFC 3280], [RFC 2408], [RFC 2412].

The user certificates are created by third party Certification Authority (CA) and placed in the directory by the CA or the user [RFC 3280]. The repository provides an easily accessible location for the users to obtain certificates.

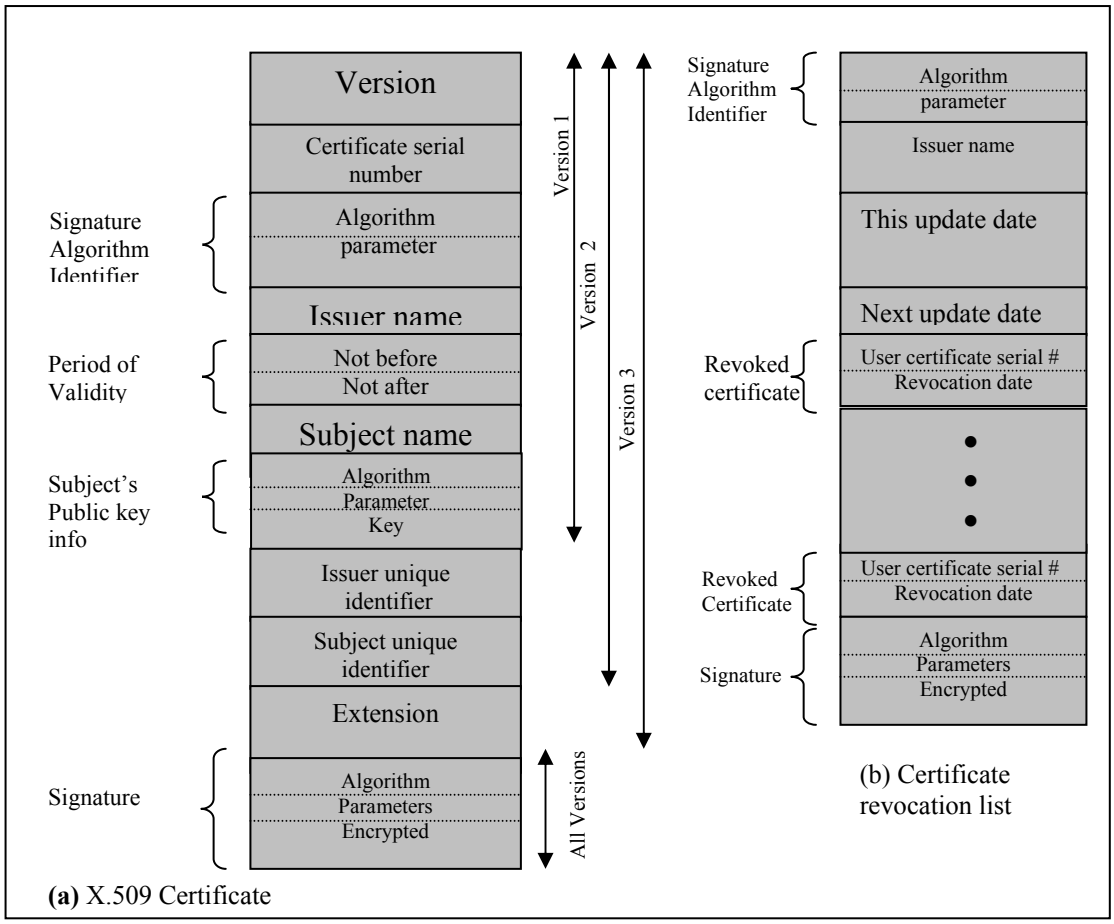


Figure 3.1 X.509 certificate format
[Stallings 2000 p102]

Figure 3.1 depicts the X.509 versions 1,2, and 3certificates content and its explanation are given below.

- **Version:** The default version is 1. If the unique identifier is available then the version must be 2. If one or more extensions are available the version must be 3.
- **Serial number:** Is unambiguous and unique integer value within the issuing CA that is associated with this certificate [RFC 3280]. The default value for serial number must be 20 octets.
- **Signature algorithm identifier:** Identifies the algorithm used to sign the certificate.
- **Issuer name:** Is the name of the CA who signed this certificate
- **Period of validity:** This is the first and last date on which the certificate is valid. This is the period the CA vouches responsibility of maintaining the certificate.
- **Subject name:** Is the name of the user to whom this certificate refers.
- **Subject's public key information:** This is the public key of the subject, identifier of the algorithm together with any associated parameters.
- **Issuer unique identifier:** is a bit string field used to identify uniquely the issuing CA in the event the X.500 name has been reused for different entities.
- **Subject unique identifier:** is a bit string field used to identify uniquely the issuing CA in the event the X.500 name has been reused for different entities.
- **Extensions:** A set of one or more extensions fields. These were options added in X.509 version 3 in efforts to address the shortcomings of version 2. The extension fall into three categories: Key and policy information, subject and issuer attributes, and certificate path constraints. The policy indicates the applicability of the certificate to a particular community. The certificate subject and Issuer attributes extension supports alternative names, alternative name's format for the issuer and subject to increase certificate's user confidence. The additions includes like the postal address, position within the corporation or picture image may be required [RFC 3280]. The certificate path constraints are included in certificates issued for CAs by other CAs. This extension includes the basic constraint that indicates whether the subject can act as a CA, name constraint, and policy constraint.
- **Signature:** It contains the hash code of the other fields, encrypted with the private key of the CA. This field includes the signature algorithm identifier.

3.4 Certificate revocation list

The Certificate Revocation List (CRL) is a list of certificates that a revoked due to one of the following reasons:

- Compromised user's private key
- The CA's certificate is or assumed to be compromised
- Change of association between CA the subject and the CA
- A user name is changed

Every CA must maintain a list consisting of all revoked but not expired certificates [RFC 3280]. This CRL is signed and posted to the directory by the CA. The distribution can be done using the X.500/LDAP directories, Web, and file. The certificate contains a pointer to these locations where the CA has published the CRL. When the user receives a certificate he must verify whether this certificate is valid or revoked. This significantly increases the overall PKI operational and implementation cost. To minimize this cost there have been several CRL schemes developed to curb bandwidth utilization problem.

3.5 Certification path

Certification path is also referred to certification chain. The certification chain is more likely to occur in large community of users where the use of a single CA is not practical. Because it is the CA who signs certificates, every user must have the CA's public key to verify signatures [RFC 3280]. The public key must be provided to each user in a secure manner so that the user has confidence in the associated certificates.

Any user with access to the public key of the CA can recover the user public key that was certified and no party other than the certification authority can modify the certificate [Stallings 2000]. Certification path processing verifies the binding between the subject distinguished name and/or subjects alternative names and subject public key. The binding is limited by the constraints that are specified in the certificate which comprise the path. The basic constraints and policy constraints extensions allow the certification path processing logic to automate the decision making process. Path validation requires obtaining a sequence of certificates that support the binding between the subject's distinguished name and/or alternative name and the subject public key. X.509 standard suggests that CAs be arranged in a hierarchy so that navigation is straightforward.

Certificate path processing includes determining that the certificate has been issued by a recognised trust anchor or its trusted subordinate, the digital signature of the certificate is valid, the certificate is within its stated validity period, the certificate has not been revoked, the certificate is being used in a manner which is consistent with its policy constraints, name constraints, and intended usage restrictions.

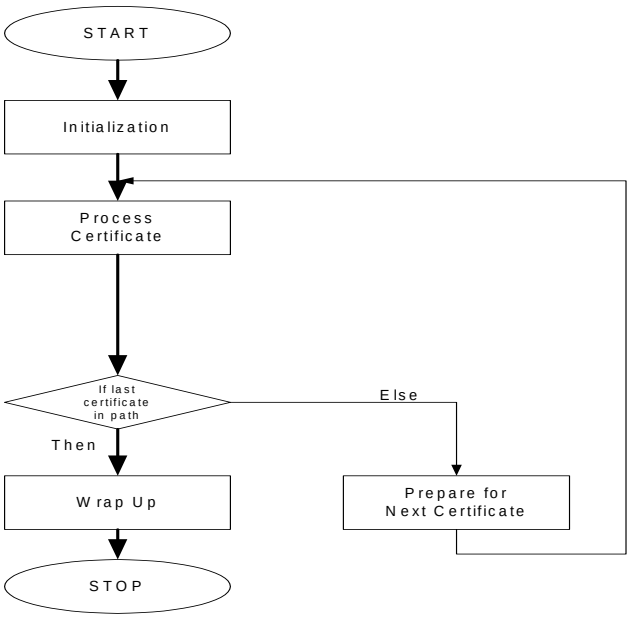


Figure 3.2 X.509 Certificate path validation sequence

Path processing initialization require that the following inputs are provided [RFC 3280]:

- The current time
- The certification path length n where certificate n is the certificate to be validated and the following must be always true if k is a certificate in the certification path.

Let {1, …, n-1} be a set of certificates present in the certification path and k represent a particular certificate in the set then:

For all k element of {1, …, n-1}, the subject of certificate k is the issuer of certificate k + 1

Certificate 1 is the first in the certification chain and is issues by the TCA who is the trusted anchor.

For all k element of {1, …, n-1}, the certificate is valid at the time in question

For all k element of {1, …, n-1}, k does not include self-issued certificates. This implies that self-issued certificates are not counted in the certification path. A certificate is a self-issued certificate if the Distinguished Name (DN) that appears in the subject and issuer field is identical and not empty.

- Policies that are acceptable to the certificate user
- Trusted anchor name, public key algorithm, trusted public key
- Initial policy mapping. This indicates if policy mapping is allowed
- Initial explicitly policy. This indicates if the policy must be valid for at least one of the policies in the initial-policy-set
- Initial-any-policy-inhibit. This indicates whether the anyPolicy object identifier, OID, should be processed if it is included in a certificate.

3.6 Assumptions

The following assumptions are made prior to testing:

- Users are knowledgeable and recognize the need for security
- Physical environment is secured
- The platform is secured
- The cryptographic algorithms are performed on a FIPS 140-1 Validated standard
- Authorized users are trustworthy to perform in accordance to security policy
- The trust model is hierarchical as depicted in figure 3.2

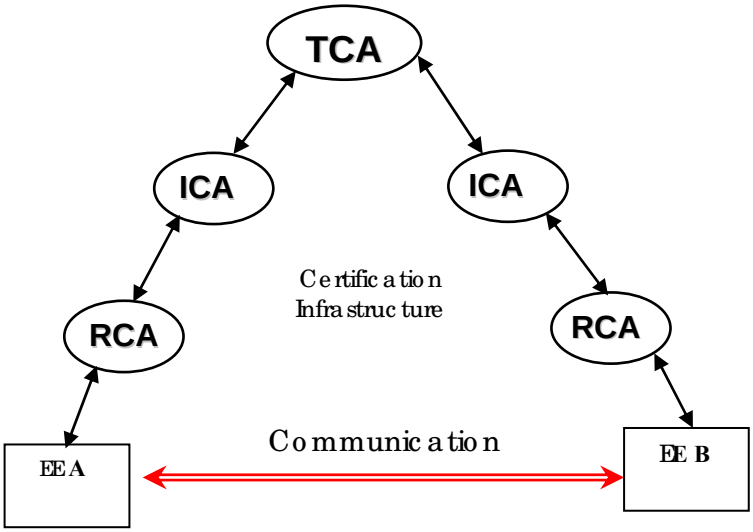


Figure 3.3 Hierarchical trust model

Figure 3.2 depicts a hierarchical trust model that will be used in different test cases that will be presented in this chapter. The Top CA (TCA) provides interoperability between two end entities A and B from different PKI domains. The Registration authority RCA is a CA that interacts directly with the EE and is responsible for EE registration. Intermediate CA (ICA) can either be a policy CA (PCA) or hierarchical CAs (HCA). For simplicity we will use Intermediate CA (ICA) through out this chapter.

3.7 PKI enabled application chain verification testing

3.7.1 Signature Test

Signature and names must initially chain, dates must be correct, intermediate certificate include a basic constraint extension that asserts CA is TRUE, and the key usage extension asserts both keyCertSign and cRLSign. In this test, the PKI enabled applications ability to successfully verify the EE and RCA certificate signature is tested. The assumption made here is that if the application validates successfully all the RCA testing then it should do for other intermediate CAs ICA).

Table 3.3 lists the entire possible test cases from which the tests to be conducted are selected. The selection will be made to provide the best coverage in relation to the testing cost and the risk the organisation is accepting to take. Table 3.3 interpretation and details about security metrics are given in section 2.10.

Table 3-3 Signature testing coverage metrics

Testing Goal	To determine the percentage of coverage of a security functional test.
Associated question	Is the application's security function implicitly defined in the TOE?
Metric	Percentage of the functional test cases selected out of all test cases.
Purpose	To ensure that maximum coverage of critical and non-critical test cases is attained.

Implementation Evidence	• Is the functional test clearly defined in the TOE? Yes • List of all possible test cases 1. TS1_EE invalid signature testing 2. TS2_EE valid signature testing 3. TS3_RCA valid signature testing 4. TS4_RCA invalid signature testing 5. TS5_Intermediate CA valid signature testing 6. TS6_Intermediate CA invalid signature testing • Total number of test cases defined is 6
Frequency	During testing/During retesting
Formula	4/6 = 0.66
Data source	TOE evidence
Indicator	66 Percent

Remarks: This metric can be used to measure the quality of testing process and in case the application is upgraded or patched, the metric can be used to make comparison between the coverage of previous results and the new retesting results.

Table 3.4 lists the four test cases that are selected for completing the certificate signature testing. Separate signed documents which Invalid EE signature, Valid EE signature, Valid RCA signature, and Invalid RCA signature are applied.

Table 3-4 TS Signature Tests

Test	Description	Expected Result
TS1_Valid EE signature	This is to test the application's ability to validate correctly valid signature on the EE certificate	The path should validate correctly
TS2_Invalid EE signature	This is to test the application's ability to validate correctly invalid signature on the EE certificate	The path should not validate correctly
TS3_Valid RCA signature	This is to test the application's ability to validate correctly valid signature on the intermediate CA certificate	The path should validate correctly
TS4_Invalid RCA signature	This is to test the application's ability to validate correctly invalid signature on the EE certificate	The path should not validate correctly

Example of signature Tests

TS2_Invalid EE signature

The purpose of this test is to verify PKI enabled application ability to validate correctly an end entity certificate with invalid signature.

Certification chain comprise the following objects:

TCA Root Cert, TCA Root CRL; Valid RCA Cert, Valid RCA CRL; Invalid EE Signature TS2 EE

Procedure: Open and verify signed test message Invalid EE Signature TS2 EE.

Expected Result: The path should not validate successfully, as the signature on the end entity certificate is invalid.

3.7.2 Validity period tests

Validity period is the time interval that the CA warrants that it will maintain information status of the certificate [RFC3280]. This interval is marked by validity beginning date (notBefore) and validity end date (notAfter). Both notBefore and notAfter can be encoded in UTC or GeneralizedTime and the certificate using application must support UTC and Generalized Time encoding [RFC 3280 4.1.2.5]. UTCTime specifies the year through the two low order digits and time is specified to the precision of one minute or one second. UTCTime includes either Z (for Zulu, or Greenwich Mean Time) or a time differential.

For the purposes X.509 profile, UTCTime values MUST be expressed in Greenwich Mean Time (Zulu) and MUST include seconds (i.e., times are YYMMDDHHMMSSZ), even where the number of seconds is zero [RFC 3280]. Conforming systems MUST interpret the year field (YY) as follows:

Where YY is greater than or equal to 50, the year SHALL be interpreted as 19YY; and Where YY is less than 50, the year SHALL be interpreted as 20YY.

The generalized time type, GeneralizedTime, is a standard ASN.1 type for variable precision representation of time. Optionally, the GeneralizedTime field can include a representation of the time differential between local and Greenwich Mean Time. For the purposes of this X.509 certificate profile, GeneralizedTime values MUST be expressed Greenwich Mean Time (Zulu) and MUST include seconds (i.e., times are YYYYMMDDHHMMSSZ), even where the number of seconds is zero. GeneralizedTime values MUST NOT include fractional seconds.

The PKI enabled application must ensure that the notBefore time of each certificate in the certification path is earlier than or equal to the current time and that the notAfter time of each certificate in the certification path is later than or equal to the current time.

The following test cases involve validating the notBefore time and notAfter time in the certificates found in a certification path.

Table 3-5 Validity period testing coverage metrics

Testing Goal	To determine the percentage of coverage of validity period testing coverage.
Associated question	Is the application's security function implicitly defined in the TOE?
Metric	Percentage of the functional test cases selected out of all test cases.
Purpose	To ensure that maximum coverage of critical and non-critical test cases is attained.
Implementation Evidence	• Is the functional test clearly defined in the TOE? Yes • List of all possible test cases

	1. TVD1_Valid EE notBefore date 2. TVD2_Invalid EE notBefore date 3. TVD3_Valid CA notAfter date 4. TVD4_Invalid CA notBefore date 5. TVD5_Valid EE notAfter date 6. TVD6_Invalid EE notAfter date 7. TVD7_Valid CA notAfter date 8. TVD8_Invalid CA notAfter date 9. TVD9_Valid Pre2000 UTC EE notBefore date 10. TVD10_Invalid UTC EE notBefore date 11. TVD11_Valid UTC EE notAfter date 12. TVD13_Valid GT EE notBefore date 13. TVD13_Valid GT EE notBefore date 14. TVD15_Valid GT EE notAfter date 15. TVD15_Valid GT EE notAfter date 16. TVD16_Invalid GT EE notAfter date 17. TVD17_Invalid GT 2050 EE notAfter date 18. TVD18_Invalid GT 50 EE notAfter date 19. TVD17_Invalid GT 2050 ICA notAfter date 20. TVD17_Invalid GT 50 ICA notAfter date • Total number of test cases is 20
Frequency	During testing/During retesting
Formula	16/20= 0.8
Data source	TOE evidence
Indicator	80 Percent

Remarks: This metric should be covered as much as possible to make sure a malicious user does not use a revoked certificate.

Table 3-6 TVD Certificate Validity date tests

Test	Description	Expected Test Results
TVD1_Valid EE notBefore date	notBefore date on the EE certificate must be earlier than the current date	The path should validate successfully
TVD2_Invalid EE notBefore date	notBefore date on the EE certificate must be later than the current date	The path should not validate successfully
TVD3_Valid CA notAfter date	notBefore date on the intermediate CA certificate must be earlier than the current date	The path should validate successfully
TVD4_Invalid CA notBefore date	notBefore date on the intermediate CA certificate must be later than the current date	The path should not validate successfully
TVD5_Valid EE notAfter date	notAfter date on the EE certificate must be after the current date	The path should validate successfully
TVD6_Invalid EE notAfter date	notAfter date on the EE certificate must be before the current date	The path should not validate successfully
TVD7_Valid CA notAfter date	notAfter date on the intermediate CA certificate must be after the current date	The path should validate successfully
TVD8_Invalid CA notAfter date	notAfter date on the intermediate CA certificate must be before the current date	The path should not validate successfully
TVD9_Valid Pre2000 UTC EE notBefore date	notBefore date on the EE certificate may be set to 1998 or any date pre2000 and UTC encoded	The path should validate successfully
TVD10_Invalid UTC EE notBefore date	notAfter date on the EE certificate must be after the current date and UTC encoded	The path should not validate successfully
TVD11_Valid UTC EE notAfter date	notAfter date on the EE certificate must be after the current date and UTC encoded	The path should validate successfully
TVD12_Invalid UTC EE notAfter date	notAfter date on the EE certificate must be before the current date and UTC encoded	The path should validate successfully
TVD13_Valid GT EE notBefore date	notBefore date on the EE certificate must be before the current date and UTC encoded	The path should validate successfully
TVD14_Invalid GT EE notBefore date	notBefore date on the EE certificate must be after the current date and UTC encoded	The path should not validate successfully
TVD15_Valid GT EE notAfter date	notAfter date on the EE certificate is 2050 and is encoded in GT	The path should validate successfully
TVD16_Invalid GT EE notAfter date	notAfter date on the EE certificate is before the current date and is encoded in GT	The path should not validate successfully

Certificate validity date examples

TVD2_Invalid EE notBefore Date

The purpose of this test is to verify the applications ability to process the chain verification id the notBefore date is after the current date.

The certification chains comprise the following objects:

TCA Root Cert, TCA Root CRL; Valid RCA Cert, Valid RCA CRL; Invalid notBefore Date TVD2 EE

Procedure: Open and verify signed test message Invalid EE notBefore Date TVD2 EE.

Expected result: The path should not validate successfully as the notBefore data in the end entity certificate is after the current date.

TVD7_Invalid CA notAfter Date

The purpose of this test is to test the applications ability to process certification path correctly if the intermediate certificate notAfter date is before the current date.

The certification chain comprises the following objects: TCA Root Cert, TCA Root CRL; Invalid notAfter Date RCA Cert, Invalid notAfter Date RCA CRL; Invalid RCA notAfter Date TVD7 EE

Procedure: Open and verify signed test message Invalid RCA notAfter Date TVD7 EE

Expected results: The path should not validate successfully as the notAfter date in the intermediate certificate is before the current date.

TVD13_Valid GeneralizedTime notBefore Date

The purpose of this test is to test the applications ability to process the certification chain if notBefore date is earlier than the current date and encoded in GT.

Certification chain comprise the following objects:

TCA Root Cert, TCA Root CRL; Valid RCA Cert, Valid RCA CRL; Valid GeneralizedTime notBefore Date TVD13 EE

Procedure: Open and verify signed test message Valid GeneralizedTime notBefore Date TVD13 EE

Expected result: The path should not validate successfully.

3.7.3 Subject and Issuer Name chaining test

PKI enabled application must, In X.509 version 3 certificates, check if the issuer name in each certificate in the certification path matches the subject name of the preceding certificate in the path. Also the following parameters must be checked: Chaining order, Capitalization, unique identifiers, Mandatory attributes and Optional attributes [RFC 3280]

Table 3-7 TSINC Subject and issuer name chaining coverage metrics

Testing Goal	To determine the percentage of coverage of subject issuer name chaining coverage.
Associated question	Is the application's security function implicitly defined in the TOE?
Metric	Percentage of the functional test cases selected out of all test cases.
Purpose	To ensure that maximum coverage of critical and non-critical test cases is attained.
Implementation Evidence	Is the functional test clearly defined in the TOE? Yes List of all possible test cases 1. TSINC1_Valid name chaining EE 2. TSINC2_Invalid name chaining order EE 3. TSINC3_Invalid name chaining EE 4. TSINC4_Valid name chain capitalization 5. TSINC5_Valid name chaining UIDs 6. TSINC6_Valid mandatory attribute type 7. TSINC7_Valid optional attribute types 8. TSINC8_Valid UTF8String Encoded Names 9. TSINC9_Valid Backward Compatibility_univerString 10. TSINC9_Valid Backward Compatibility_ teletexString 11. TSINC9_Valid Backward Compatibility_ printableString 12. TSINC9_Valid Backward Compatibility_ bmpString • Total number of test cases is 12
Frequency	During testing/During retesting
Formula	9/12 = 0.75
Data source	TOE evidence
Indicator	75 Percent

Remarks: This metric ensures that the binding of a user to a certificate is done correctly to avoid malicious users.

Table 3-8 Name chaining test

Test	**Description**	**Expected results**
TSINC1_Valid name chaining EE	The common name portions of the issuer's name in the end entity certificate match the common name portion of the subject's name in the preceding intermediate certificate.	The path should validate Successful
TSINC2_Invalid name chaining order EE	In this test, the issuer's name in the end entity certificate and the subject's name in the preceding intermediate certificate contain the same relative distinguished names but their ordering is different.	The path should not validate Successful
TSINC3_Invalid name chaining EE	In this test, the common name portion of the issuer's name in the end entity certificate does not match the common name portion of the subject's name in the preceding intermediate	The path should not validate successfully

TSINC4_Valid name chain capitalization	In this test, the issuer's name in the end entity certificate and the subject's name in the preceding intermediate certificate differ in capitalization, but match when a case insensitive match is performed.	The path should validate successfully
TSINC5_Valid name chaining UIDs	In this test, the intermediate certificate includes a subject Unique ID and the end entity certificate includes a matching issuer Unique ID.	The path should validate successfully
TSINC6_Valid mandatory attribute type	In this test, this intermediate certificate includes a subject name that includes the attribute types distinguished name qualifier, state or province name, serial number, domain component, organization, and country.	The path should validate successfully
TSINC7_Valid optional attribute types	In this test, this intermediate certificate includes a subject name that includes the attribute types locality, title, surname, given name, initials, pseudonym, generation qualifier, organization, and country.	The path should validate successfully
TSINC8_Valid UTF8String Encoded Names	[RFC3280 4.1.2.4] Points out that by December 2003 certificate issuers must use UTF8String encoding of directory string. In this test, the subject and issuer fields of the end entity and intermediate certificate are UTF8String encoded	The path should validate successfully
TSINC9_Valid Backward Compatibility_universString	In this test, the subject and issuer attribute of the EE certificate and subject of intermediate RCA are UTF8String encoded. However, the issuer field of the intermediate certificate is univerString encoded.	The path should validate successfully

3.8 Subject and Issuer Name chaining tests examples

TSINC9_Valid Backward compatibility from UTF8String to universalString

In this test, the subject and issuer attributes of the EE certificate and the subject field of the RCA are UTF8String encoded. However, the issuer field of the intermediate certificate is universalString encoded.

Certification chain comprise the following objects:

TCA Root Cert, TCA Root CRL, Backward Compatibility from UTF8String to universalString RCA Cert, Backward Compatibility from UTF8String to universalString RCA CRL, and Valid Backward Compatibility UTF8String to universalString encoding TSINC9EE

Procedure: Validate Valid Backward Compatibility UTF8String to universalString encoding TSINC9 EE

Expected results: The path should validate successfully.

TNC2_Invalid Name Chaining EE

In this test, the issuer common name in the end entity certificate does not match the common name of the subject's name in the preceding intermediate certificate.

The certification chain comprise the following objects:

TCA Root Cert, TCA Root CRL, Valid RCA Cert, Valid RCA CRL, and Invalid Name Chaining TSINC2 EE

Procedure: Validate Invalid Name Chaining TSINC2 EE

Expected result: The path should not validate successfully as the name does not chain correctly.

3.8.1 TKU Key Usage Tests

Key usage extension defines the purpose of the key contained in the certificate. This includes: digitalSignature, nonRepudiation, keyEncipherment, dataEncipherment, keyAgreement, keyCertSign, cRLSign, encipherOnly and decipherOnly [RFC3280 4.2.1.3]. The purpose of these tests is to determine the ability of PKI enabled application to process the key usage extension in the certificate. If the extension is labeled true (e.g. certSign true) it implies that the key can be used to sign certificates.

Table 3-9 Key usage testing coverage metrics

Testing Goal	To determine the percentage of coverage of key usage testing coverage.
Associated question	Is the application's security function implicitly defined in the TOE?
Metric	Percentage of the functional test cases selected out of all test cases.
Purpose	To ensure that maximum coverage of critical and non-critical test cases is attained.
Implementation Evidence	Is the functional test clearly defined in the TOE? Yes List of all possible test cases 1. TKU1_Valid keyUsage nonrepudiation extension EE 2. TKU2_Valid keyUsage keyencipherment extension ICA 3. TKU3_Valid keyUsage dataencipherment extension EE 4. TKU4_Valid keyUsage key agreement extension ICA 5. TKU5_Valid keyUsage encypherOnly extension EE 6. TKU6_Valid keyUsage decypherOnly extension EE 7. TKU7_Invalid keyUsage Non Critical keyCertSign False 8. TKU8_Invalid keyUsage Critical keyCertSign False 9. TKU9_Invalid keyUsage Critical cRLSign False 10. TKU1_Valid keyUsage nonrepudiation extension ICA 11. TKU2_Valid keyUsage keyencipherment extension EE 12. TKU3_Valid keyUsage dataencipherment extension ICA 13. TKU4_Valid keyUsage key agreement extension EE 14. TKU5_Valid keyUsage encypherOnly extension ICA 15. TKU6_Valid keyUsage decypherOnly extension ICA • Total number of test cases is 15
Frequency	During testing/During retesting
Formula	9/15 = 0.6
Data source	TOE evidence
Indicator	60 Percent

Remarks: This metric can be used to measure the quality of testing process and in case the application is upgraded or patched, the metric can be used to make comparison between the coverage of previous results and the new retesting results.

Table3-10 TKU Certificate Key usage tests

Test	Description	Expected Result
TKU1_Valid keyUsage extension EE	This test is intended to test the applications ability to validate certificates where the EE assert nonRepudiation	The path should validate successfully
TKU2_Valid keyUsage extension ICA	This test is intended to test the applications ability to validate certificates where the intermediate CA assert keyEncipherment	The path should validate successfully
TKU3_Valid keyUsage extension EE	This test is intended to test the applications ability to validate certificates where the EE assert dataEncipherment	The path should validate successfully
TKU4_Valid keyUsage extension ICA	This test is intended to test the applications ability to validate certificates where the intermediate CA assert keyAgreement	The path should validate successfully
TKU5_Valid keyUsage extension EE	This test is intended to test the applications ability to validate certificates where the EE assert encypherOnly	The path should validate successfully
TKU6_Valid keyUsage extension EE	This test is intended to test the applications ability to validate certificates where the EE assert decipherOnly	The path should validate successfully
TKU7_Invalid keyUsage Non Critical keyCertSign False	In this test, the intermediate certificate includes a Non Critical key usage extension in which keyCertSign is False	The path should not validate successfully
TKU8_Invalid keyUsage Critical keyCertSign False	In this test, the intermediate certificate includes a Critical key usage extension in which keyCertSign is False	The path should not validate successfully
TKU9_Invalid keyUsage Critical cRLSign False	In this test, the intermediate certificate includes a Critical key usage extension in which cRLSign is False	The path should not validate successfully

Key Usage Tests Example:

TKU9_Invalid keyUsage Crtical cRLSign False In this test, the intermediate certificate includes a critical keyUsage extension in which cRLSign is false.
The certification chain comprise the following objects:

TCA Root Cert, TCA Root CRL, keyUSage Critical cRLSign False RCA cert, keyUSage Critical cRLSign False RCA CRL, Invalid keyUsage Critical cRLSign False TKU9 EE

Procedure: Validate Invalid keyUsage Critical cRLSign False TKU9 EE

Expected results: The path should not validate successfully since the subjects public key may not be used to verify signatures on CRLs.

3.8.2 Name constraint test

The tests in this section are designed to verify PKI enabled application ability to process the name constraints extension. The tests in this section include certification paths in which one or more certificates include a name constraints extension with the following: DNS name, and uniform resource identifier (URI), rfc822 name and directory name (DN). In these case tests can be extended to EE certificates that include subject alternative name that fall within the subtree and outside the subtree.

Table 3-11 TNC Name Constraint testing coverage metrics

Testing Goal	To determine the percentage of coverage of name constraint testing coverage.
Associated question	Is the application's security function implicitly defined in the TOE?
Metric	Percentage of the functional test cases selected out of all test cases.
Purpose	To ensure that maximum coverage of critical and non-critical test cases is attained.
Implementation Evidence	Is the functional test clearly defined in the TOE? Yes List of all possible test cases 1. TNC1_Valid DN name constraint 2. TNC2_Invalid DN name constraint 3. TNC3_Invalid Self Issued Certificate DN name constraint 4. TNC4_Invalid Self Issued Certificate DN name constraint 5. TNC5_Valid RFC 822 6. TNC6_Invalid RFC 822 name constraint 7. TNC6_Invalid DNS name Constraint 8. TNC8_Valid URI name constraint 9. TNC9_Invalid URI name constraint • Total number of test cases is
Frequency	During testing/During retesting
Formula	9/9-1
Data source	TOE evidence
Indicator	100 Percent

Remarks: This metric is important to measure the coverage how the constraints can be correctly verifies by the application

Table 3-12 TNC Name constraint test

Test	Description	Expected results
TNC1_Valid DN name constraint	The intermediate CA certificate includes a name constraints extension that specifies a single permitted subtree. The end entity certificate includes a subject name that falls within that subtree.	The path should validate successfully
TNC2_Invalid DN name constraint	The intermediate CA certificate includes a name Constraints extension that specifies a single permitted subtree. The end entity certificate includes a subject name that falls outside that subtree.	The path should not validate successfully
TNC3_Invalid Self Issued Certificate DN name constraint	In this test, the intermediate certificate includes a name constraints extension that specifies a single permitted subtree. The end entity certificate is a self-issued certificate. The subject name in the self-issued certificate does not fall within the permitted subtree specified in the intermediate	The path should not validate successfully
TNC4_Invalid Self Issued Certificate DN name constraint	The EE certificate includes a subject name that falls within the permitted subtree specified in the first intermediate certificate. In the first intermediate certificate the name constraints extension that specifies a single permitted subtree. The second intermediate certificate is a self-issued certificate where subject name does not fall within the permitted subtree specified in the first intermediate certificate.	The path should not validate successfully
TNC5_Valid RFC 822 name constraint	The end entity certificate includes a subject alternative name extension with an e-mail address that falls within that subtree.	The path should validate successfully
TNC6_Invalid RFC 822 name constraint	The end entity certificate includes a subject alternative name extension with an e-mail address that falls outside the subtree.	Unsuccessful
TNC6_Invalid DNS name Constraint	In this test, the intermediate certificate includes a name Constraints extension that specifies a single permitted subtree. The end entity certificate includes a subject alternative name extension with a DNS name that falls outside that subtree.	The path should not validate successfully
TNC7_Valid DNS name Constraint	In this test, the intermediate certificate includes a name constraints extension that specifies a single excluded subtree. The end entity certificate includes a subject alternative name extension with a DNS name that falls outside that subtree.	The path should validate successfully

TNC8_Valid URI name constraint	In this test, the intermediate certificate includes a name constraints extension that specifies a single permitted subtree. The end entity certificate includes a subject alternative name extension with a uniform resource identifier that falls within that subtree.	The path should validate successfully
TNC9_Invalid URI name constraint	In this test, the intermediate certificate includes a name constraints extension that specifies a single excluded subtree. The end entity certificate includes a subject alternative name extension with a uniform resource identifier that falls within that subtree.	The path should not validate successfully

3.8.3 TCP Certificate policy extension testing

This testing is intended to verify the applications ability to verify certificate policy throughout the chain. The result of these tests depends on the certificate policy that is initially imposed on the certificate. A certificate can have the following initial settings [RFC3280]: Initial policy set (inpolset), initial explicitly policy (inexpol), and initial inhibit policy (ininhpol). Assuming that there is a policy called policy n with which anchor and the end entity are constrained the testing will be as follows:

Table 3-13 TCP Certificate policy testing coverage metrics

Testing Goal	To determine the percentage of coverage of certificate policy testing coverage.
Associated question	Is the application's security function implicitly defined in the TOE?
Metric	Percentage of the functional test cases selected out of all test cases.
Purpose	To ensure that maximum coverage of critical and non-critical test cases is attained.
Implementation Evidence	• Is the functional test clearly defined in the TOE? Yes • List of all possible test cases for same policy in all certificates in the chain • CAs and EE are constrained by policy n TCP1_explicity is true and no policy set except the EE TCP2_explicity is false and no policy set except the EE TCP3_explicity is true and no policy set except the RCA TCP4_explicity is false and no policy set except the RCA TCP5_explicity is true and inpolset is any policy TCP6_explicity is false and inpolset is any policy TCP7_explicity is true and inpolset is policy n TCP8_explicity is false and inpolset is policy n TCP9_explicity is true and inpolset set policy n is missing TCP10_explicity is false and inpolset set policy n is missing TCP11_explicity is false and inpolset set policy n is present TCP12_explicity is true and inpolset set policy n is present TCP13_explicity is true and no policy set for both CA and EE TCP14_explicity is false and no policy set for both CA and EE

	TCP15_explicity is true and no policy set except the ICA TCP16_explicity is false and no policy set except the ICA • Total number of test cases is 16
Frequency	During testing/During retesting
Formula	14/16= 0.875
Data source	TOE evidence
Indicator	87.5 Percent

Remarks: The number of test cases will depend on the path length and the number of policies that applies to different certificates in the certification path.

Table 3-14 TCP Certificate policy tests

Test	Description	Expected Test Results
TCP1_explicity is true and no policy set except the EE	Policy n is set on the end entity and the initial explicitly policy is set true	The path should not validate successfully
TCP2_explicity is false and no policy set except the EE	Policy n is set on the end entity and the initial explicit policy is set false	The path should validate successfully
TCP3_explicity is true and no policy set except the RCA	Policy n is only set on the RCA and the initial explicit policy is true	The path should not validate successfully
TCP4_explicity is false and no policy set except the RCA	Policy n is only set on the RCA and the initial explicit policy is false	The path should validate successfully
TCP5_explicity is true and inpolset is any policy	The EE certificate is ha initial policy set to any policy and the explicit policy is set to true	The path should validate successfully
TCP6_explicity is false and inpolset is any policy	The EE certificate is ha initial policy set to any policy and the explicit policy is set to false	The path should validate successfully
TCP7_explicity is true and inpolset is policy n	The EE certificate is has initial policy set to policy n and the explicit policy is set to true	The path should validate successfully
TCP8_explicity is false and inpolset is policy n	The EE certificate is has initial policy set to policy n and the explicit policy is set to false	The path should validate successfully
TCP9_explicity is true and inpolset set policy n is missing	The intermediate certificate has explicit policy set to true and the initial policy n is missing	The path should not validate successfully
TCP13_explicity is true and no policy set for both ICA and EE	No policy is set for both the ICA and the EE but the explicit policy is set true	The path should not validate successfully
TCP14_explicity is false and no policy set for both CA and EE	No policy is set for both the ICA and the EE but the explicit policy is set false	The path should validate successfully
TCP15_explicity is true and no policy set except the ICA	In this test there is no policy set except on the ICA and the initial explicit policy is set to true	The path should not validate successfully

TCP16_explicity is false and no policy set except the ICA	In this test there is no policy set except on the ICA and the initial explicit policy is set to true	The path should validate successfully

3.8.4 TCRL Certificate revocation mechanisms testing

The Certificate Revocation Mechanisms testing, TCRL, are designed for testing the PKI application ability to process CRLs correctly during the certificate path processing. CRL processing is critical because CRL indicates whether the certificate can be used at that particular time.

Table 3-15 TCRL Certificate Revocation List testing coverage metrics

Testing Goal	To determine the percentage of coverage of CRL testing coverage.
Associated question	Is the application's security function implicitly defined in the TOE?
Metric	Percentage of the functional test cases selected out of all test cases.
Purpose	To ensure that maximum coverage of critical and non-critical test cases is attained.
Implementation Evidence	• Is the functional test clearly defined in the TOE? Yes TCRL1_Invalid CRL signature TCRL2_Serial number greater than 20 Octet TCRL3_Zero Serial number TCRL4_Less than 20 octet serial number TCRL5_Negative serial number TCRL6_CRL issuer name TCRL7_Invalid issuer name TCRL8_Revoked certificate EE TCRL9_Revoked certificate ICA TCRL10_Revoked certificate RCA TCRL11_nextUpdate later than the current time TCRL12_nextUpdate is earlier than the current time TCRL13_Generalised time 2050 TCRL14_Generalised time 50 TCRL15_UTC time 1950 TCRL16_UTC time 50 • Total number of test cases is 16
Frequency	During testing/During retesting
Formula	14/16= 0.875
Data source	TOE evidence
Indicator	87.5 Percent

Remarks:

Test cases in this table are derived from the table 3.15.

Table 3-16 TCP Certificate policy tests

Test	Description	Expected Test Results
TCRL1_Invalid CRL signature	The same key used to sign the revoked certificate is used to verify the signature on the CRL	The path should not validate successfully
TCRL2_Serial number greater than 20 Octet	RFC 3280 recommends the serial number to be 20 octet but should gracefully accept serial number which are less than 20 octets or zero [RFC 3280]	The path should not validate successfully
TCRL4_Less than 20 octet serial number	Applications should gracefully accept this kind of certificate from non confirming vendors	The path should validate successfully
TCRL6_CRL issuer name	The issuer name on the certificate must match the issuer name of the CRL	The path should validate successfully
TCRL7_Invalid issuer name	The issuer name on the certificate must match the issuer name of the CRL	The path should not validate successfully
TCRL8_Revoked certificate EE	Revoked certificate is identified by the serial number which should appear in the CRL	The path should not validate successfully
TCRL9_Revoked certificate ICA	Revoked certificate is identified by the serial number which should appear in the CRL	The path should validate successfully
TCRL10_Revoked certificate RCA	Revoked certificate is identified by the serial number which should appear in the CRL	The path should validate successfully
TCRL11_nextUpdate later than the current time	The nextUpdate date indicates the expected date when the CRL will be updated. This must always be later than the current time	The path should validate successfully
TCRL12_nextUpdate is earlier than the current time	The nextUpdate date indicates the expected date when the CRL will be updated. This must always be later than the current time	The path should not validate successfully
TCRL13_Generalised time 2050	GeneralisedTime encoding should be used for certificate valid up to 2050[RFC 3280]	The path should validate successfully
TCRL14_Generalised time 50	GeneralisedTime encoding should be used for certificate valid up to 2050[RFC 3280]	The path should not validate successfully
TCRL15_UTC time 1950	This time encoding is for certificate whose validity time includes year less than 2000	The path should validate successfully
TCRL16_UTC time 50	This time encoding is for certificate whose validity time includes year less than 2000	The path should validate successfully

3.8.5 TBC Basic Constraint testing

The basic constraint testing is designed to test the applications ability to process the basic constraint correctly during the certificate path processing. This is to verify the presence of CA is enforced and in every ICA certificate the field with CA is set true [RFC 3280 10.5.1]

Table 3-17 TBC Basic Constraint testing coverage metrics

Testing Goal	To determine the percentage of coverage of basic constraint testing coverage.
Associated question	Is the application's security function implicitly defined in the TOE?
Metric	Percentage of the functional test cases selected out of all test cases.
Purpose	To ensure that maximum coverage of critical and non-critical test cases is attained.
Implementation Evidence	• Is the functional test clearly defined in the TOE? Yes TBC1_Non_critical flag ICA TBC2_Basic constraint not present ICA TBC3_Critical extension ICA True TBC4_Non_critical flag RCA TBC5_Basic constraint not present RCA TBC6_Critical extension RCA True TBC7_ Critical extension ICA False TBC8_ Critical extension RCA False TBC9_ Non Critical extension ICA True TBC10_ Non Critical extension RCA True • Total number of test cases is
Frequency	During testing/During retesting
Formula	8/10= 0.8
Data source	TOE evidence
Indicator	80 Percent

Remarks:

Table 3-18 TBC Basic constraints tests

Test	Description	Expected Test Results
TBC1_Non_critical flag ICA	In this test the ICA has a basic constraint and is flagged noncritical	The path should not validate successfully
TBC2_Basic constraint not present ICA	In this test the ICA has no basic constraint	The path should not validate successfully
TBC3_Critical extension ICA True	In this test the ICA has a basic constraint and is flagged noncritical	The path should validate successfully
TBC5_Basic constraint not present RCA	The basic constraint is not present in the RCA certificate	The path should not validate successfully
TBC6_Critical extension RCA True	In this test the RCA has a critical extension and CA is marked true	The path should validate successfully
TBC8_ Critical extension RCA False	In this test the RCA has a critical extension and the CA is marked false	The path should not validate successfully
TBC9_ Non Critical extension ICA True	In this test the ICA extension is marked non critical and CA is true	The path should validate successfully
TBC10_ Non Critical extension RCA True	In this test the RCA extension is marked non critical and CA is true	The path should validate successfully

4. Chapter 4

4.1 Concluding remarks

Our research work has attempted to address using established methods and structures as a guide how applications in PKI environments can verify X.509 certificates correctly and continuously. This is the basis for investigating hindrances and obstacles for trustworthiness in large scale PKI interoperability and to suggest a structure for measurements and metrics for benchmarking security. Our work makes use of the Common Criteria as a guideline for functional testing and the Systems Security Engineering Capability maturity Model (SSE-CMM) as a guideline for developing security metrics for process and security improvement. We have used SSE-CMM metrics as part of the testing process input design. This has the advantage that helps to ensure that security and process improvement is an integral part of the larger process, and not a separate and distinct activity.

4.2 Metrics and protection of assets and mitigating threats

In chapter one we pointed out how organisations incline to using IT to foster their wellbeing. In so doing they place values on IT assets which they have to protect in their whole life cycle. However understanding whether the protection mechanisms are effective or how they should be updated without affecting the security posture is not easy to understand for non-experts. This accentuates the need for security metrics that we have attempted to develop in chapter 2.

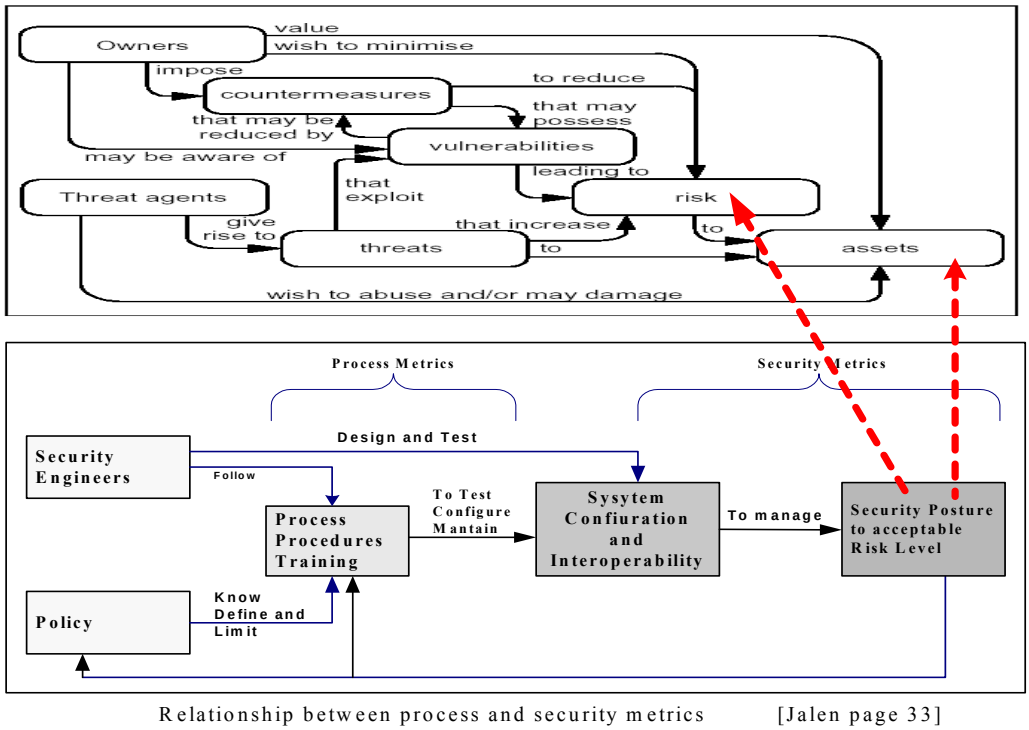


Figure 4.1 Security posture to accept asset's risk level

Table 4.1 lists numerous security processes that require security metrics for measuring the impact and effectiveness of various security controls, threats, vulnerabilities, quality and risks. Amongst these, in our approach, we have addresses metrics that can be used to assess threats in PKI application that is designed to interoperate with other PKIs. Table 2.4 lists the following metrics that can be used to assess different levels of threats:

- ***Test***
- **Rudimentary**
- **Basic**
- **Medium**
- **High**

In my view these metrics should be useful for users, supports personnel, and the management in the process of assessing various threats to the assets and eventually take the right and economical approach in mitigating the threats.

Table 4-1 Areas for Security metrics development
[SSE-CMM 2003]

• Administer Security Control
• Assess Impact
• Assess Security Risk
• Assess Threat
• Assess Vulnerability
• Build Assurance Argument
• Coordinate Security
• Monitor Security Posture
• Provide Security Input
• Specify Security Needs
• Verify and Validate Security
• Ensure quality
• Manage configurations
• Manage Project Risks
• Monitor and Control Technical Efforts
• Plan Technical Efforts
• Define Organisation's Systems Eng. Process
• Improve Organisation's Systems Eng. Process
• Manage product line evaluation
• Manage Systems Eng. Support Environment
• Provide ongoing skills and knowledge
• Coordinate with suppliers

Other metrics we have examined in Tables2.5 to Table 2.9 are metrics that are useful for use in PKI environments. They are useful for measuring various security incidences, system's documentation, system's maintenance, and systems accreditation. The use of these metrics should also improve user awareness and understanding of various security processes which in one way or another may result into non-interoperable PKI.

4.3 Testing in relation to protection of assets and mitigating threats

Figure 4.1 depicts how threats increase risks to assets. There is a direct relationship between risks, security incidence consequence, and the likelihood of an incidence to occur [Bishop 2002 Page 17]. In this relationship risk is the function of consequence and security incidence.

PKI application testing is essential to minimize security incidences that are related to the security functions that are examined in Table 2.4. In so doing the risks to the asset are minimized to an acceptable level.

Table 4-2 Subject and issuer name chaining coverage metrics

Testing Goal	To determine the percentage of coverage of subject issuer name chaining coverage.
Associated question	Is the application's security function implicitly defined in the TOE?
Metric	Percentage of the functional test cases selected out of all test cases.
Purpose	To ensure that maximum coverage of critical and non-critical test cases is attained.
Implementation Evidence	Is the functional test clearly defined in the TOE? Yes List of all possible test cases 1. TNC1_Valid name chaining EE 2. TNC2_Invalid name chaining order EE 3. TNC3_Invalid name chaining EE 4. TNC4_Valid name chain capitalization 5. TNC5_Valid name chaining UIDs 6. TNC6_Valid mandatory attribute type 7. TNC7_Valid optional attribute types 8. TNC8_Valid UTF8String Encoded Names 9. TNC9_Valid Backward Compatibility_univerString 10. TNC9_Valid Backward Compatibility_ teletexString 11. TNC9_Valid Backward Compatibility_ printableString 12. TNC9_Valid Backward Compatibility_ bmpString • Total number of test cases is 12
Frequency	During testing/During retesting
Formula	9/12 = 0.75
Data source	TOE evidence
Indicator	75 Percent

Table 4.2 lists the test cases for issuer and subject name chaining. These tests are important to make sure PKI application is able to reject a certificate from a malicious entity prior to committing a transaction. One would wish to make a hundred percent coverage of all the test cases but due to cost implications this is not practical in most environments due to financial constraints. Therefore, a good enough testing will depend on the assurance level required. Different assurance levels requirements will result into different set of test cases and test coverage.

In our case there are 12 test cases but we have provided only 9 test cases assertions. This implies among all the test cases we are covering only 75 percent. In effect this means the risks of 25 percent is tolerable.

Testing coverage is not the only advantage of testing metrics. Other advantages are testing process repeatability, understanding the testing process and understanding the legal implications of testing in case the product system under testing does not produce the required results.

4.4 The thesis contribution

The thesis has attempted to examine how to develop IT security metrics and test assertions that can be used to test PKI enabled application ability to validate successfully X.509 Version 3 certificates. Metrics are useful for security vendors, testers, organisation's management and users. These can be used to guide each aspect of testing effort, for example systems interoperability testing and evaluation, risk assessment, penetration testing, security testing and evaluation. Security testers can use metrics to isolate problems, redirect efforts, and process improvement.

The use of IT metrics will allow organisations to determine effectiveness of implemented IT security processes, and control by relating results of IT security activities. The implementation, efficiency, effectiveness, and impact metrics types can be calculated for use in performance improvement effeorts.

4.5 Reflections

There are many difficulties related to understanding and describing security testing and metrics. The metrics we have attempted to develop in this work are suitable for partly analyzing threats and test coverage measurements in PKI environments. However, the list of metrics in Table 4.1 indicates that more work is required in the area of developing security metrics that are standard and generic. Currently different environments require different metrics, in my view this is a problem that has to be addressed.

The application testing alone cannot provide the desired systems security neither provide assurance since security flaw can happen due to other factors like improper system implementation, the platform, algorithm, and protocol. Therefore, a good enough testing should take into consideration of the environment.

4.6 Further work

I have several suggestions on further work within this area. More work need to be done regarding developing countries PKI and its environments Protection Profiles (PP).

The protection profile shall cover the following aspects:

- Retention period of revoked certificates and data for archive. This will be done in accordance to the base practice and legal framework. This will involve retention period of achieve, protection of achieve, archive backup procedure, time stamping, archive collection, and archive verification

- Certificate profiles
- Community and applicability. This encompasses the PKI certification authorities, end entities and applicability
- Obligations: CA obligations, RA obligations, End Entities obligations, repository obligations and interoperating with other countries
- Repositories: CA publications, Access control, Frequency of publication
- Identification and authentication: Types of names, meaning of names, uniqueness of names, trademarks, possession of key, authentication of computers, organisations and individuals.
- Certificate revocation mechanisms: Certificate revocation requests, renewal, update, Revocation lists
- Security audit procedures: Protection of security audit data, vulnerability assessment, events recorded, and audit data collection system and backup
- Private key generation and protection and
- Life cycle controls

The case study will be Tanzania. Tanzania being a developing country have problems relating to procuring and maintaining various IT systems (see section 1.10). PP can be useful for third world government's use in controlling systems suppliers and maintenance. For example the government can include a clause in its ITC policy that the banks must use assurance level 3 for all government revenue and expenditure management systems. While this metric means a lot of work has to be done to test the products at the same time it may make sense to government officials who may not be knowledgeable enough on the product.

References

[Ammann1999] P. Ammann and P. Black, 1999, A specification-Based Coverage Metrics to Evaluate Test Sets, "Proceeding of the 4th IEEE International Symposium on High-Assurance Systems Engineering".

[Ann 2001] Ann Frisinger 2001 "A generic Security Evaluation Method for Open Distributed system", PhD thesis, Dept. Of Computer and Systems Science, Royal Institute of Technology

[Bell 1974] Bell, D.E., la Padula L.J, 1974, Secure computer systems: Mathematical foundation and model.

[Biba 1977] Biba, K.J., 1977, Integrity considerations for secure computer systems, ESD-TR-76-372, ESD/AFSC, Hanscom AFB, Bedford, Mass.

[Bishop 2002] Matt Bishop, 2002, Computer Security Art and Science

[CCIMB-99-031] CCIMB-99-031, 1999, Common Criteria for Information Technology Security Evaluation: Introduction and General

[CCIMB-99-033] CCIMB-99-033, 1999, Common Criteria for Information Technology Security Evaluation: Security assurance requirements Version

[CEM-99/045] 1999, Common Methodology for Information Technology Security Evaluation Methodology Version 2

[Clark 1988] Clark, David D., Wilson, David R.A., 1998, Evaluation of model for computer integrity.

[CMM 1995] Systems Engineering Capability Maturity Model, (SE-CMM Version 1.1). 1995, [Online] Available at: http://www.sei.cmu.edu/cmm/cmms/cmms.html (Accessed in August 2003

[Corcoran 1999] David Corcoran, David Sims and Bob Hillhouse, 1999, Smart Cards and Biometrics: Your Key to PKI [Online] Available at: http://www.linuxjournal.com/article.php?sid=3013 (Accessed in 3 August 2002)

[CSRC 2003] Development of High level PKI Service API , 2003, NIST, [Online] Available at: http://csrc.nist.gov/pki/pkiapi/welcome.htm (Read in June 2003)

[CTCPEC 1993] CTCPEC, 1993, Canadian Trusted Computer Product Evaluation Criteria (CTCPEC) version 3.0

[eSecretariat 2001] eSecretariat, 2001, Proposal for Tanzania's ICT Policy Formulation Final version Available at: http://www.ethinktanktz.org/esecretariat/ArchiveDoc.htm (Read in Nov 2003)

[FBCA 2002] X.509 Certificate Policy, 2002, Federal Bridge Certification Authority. Available at: http://www.cio.gov/fpkipa/documents/fbca_cp_09-10-02.pdf (Read in November 4, 2003)

[FIPS PUB 180-1] Federal Information Processing Standards Publication 180-1, 1993, Available at: http://www.itl.nist.gov/fipspubs/fip180-1.htm, (Read October 2002)

[FIPS PUB 46-2] Data Encryption Standard (DES), 1993, Available at: http://www.itl.nist.gov/fipspubs/fip46-2.htm, (Rear June 2002)

[FIPS197] "Advanced Encryption Standard (AES)", 2002, Available at: http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf (Read 2003)

[Gollmann 2000] Dieter Gollmann, 2000, Computer Security

[HAPI 2003] High-level Public-Key Based Cryptographic services API, 2003, NIST. Available at: http://csrc.nist.gov/pki/pkiapi/api.pdf (Read July 2003)

[Housley 2001] Russ Housley, 2001, Planning for PKI: Best Practices guide for deploying public key Infrastructure

[Hubner 2001] Simone Hubner, 2001, Privacy in the Global Information Society [Online] Available at: http://springerlink.metapress.com/app/home/content.asp?Wasp=87exf5a92g1trh9aaw7m&referrer=contribution&format=2&page=1 (Accessed in Sept. 2003)

[IPV6 2002] IPV6, 2002, "IP address services", IANA, Available at http://www.iana.org/ipaddress/ip-addresses.htm (Read in August 2002)

[ITSEC 1991] ITSEC, 1991, Information Technology Security Evaluation Criteria

[Jelen 2000] George Jelen, 2000, SSE-CMM security metrics. [Online] Available at: http://csrc.nist.gov/csspab/june13-15/jelen.pdf

[Kerberos] Kerberos, 1993, The Network Authentication Protocol [Online] Available at: http://web.mit.edu/kerberos/#what_is: Kerberos

[Knudsen 1998] Jonathan Knudsen, 1998, "Java Cryptography"

[Kowalski 1994] Stewart Kowalski, March 1994, "IT Insecurity: A Mult-disciplinary Inquiry" ISSN 1101-8526, ISRN SU-KTH/DSV/R—94/4—SE

[LAMTRAC 2001] LAMTRACK AB, 2001, Survey of the need for a vocational training program for ICT professionals in Tanzania. Available at: http://www.ethinktanktz.org/esecretariat/ArchiveDoc.htm (Read in Nov 2003)

[Linden 2002] Mikael Linden, 2002, "Lesson Learnt in Implementation of PKI in Higher education" The 8th International Conference of European Universities Information Systems, "Lesson Learnt in implementation of PKI in Higher education" [Online] Available at: https://hstya.funet.fi/docs/FEIDHE_lessons_learned.pdf (Read in September 2002)

[McClure 2001] Stuart McClure, Joel Scambray, George Kurtz, 2001, Hacking Exposed: Network Security Secrets & Solutions, Third Edition

[NICT 2003] NICT, 2003, National Information and Communication Technologies Policy. Ministry of Communication and Transport, Tanzania. Available at: http://www.ethinktanktz.org/esecretariat/ArchiveDoc.htm (Read in June 2003)

[NIST94a 1994] NIST94a, 1994, Advanced Authentication,[Online] Available at: http://cs-www.ncsl.nist.gov/publications/nistpubs/800-10/node48.html (Read in July 2003)

[pkiC 2001] EEMA, 2001, "Interoperability between PKI products" Available at: https://www.eema.org/pki-challenge/files/description.pdf (Read in August 2002)

[PKIX 2002] IETF, 2002, "Public Key Infrastructure" [Online] Available at: http://www.ietf.org/html.charters/pkix-charter.html (Accessed in September 2002)

[RFC 1321] "The MD5 Message-Digest Algorithm", R. Rivest, MIT Laboratory for Computer Science and RSA Data Security, Inc., April 1992, Available at: http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc1321.html (Read in October 2002)

[RFC 3280] RFC 3280, 2002, X.509 Version 3 Certificates and CRL version 2, IETF.

[RFC 3281] RFC 3281, 2002, "Attribute certificates" [Online] Available at: http://www.ietf.org/mail-archive/ietf-announce/Current/msg18344.html (Accessed in Sept. 2003)

[RFC2459] RFC 2459, 2001, "Internet X.509 Public Key Infrastructure Certificate and CRL Profile", Request for Comments, [Online] Available at: *ftp://ftp.ietf.org/rfc/rfc2459.txt* (Read in September 2002)

[Ross Anderson 2001] Ross Anderson, 2001, Security Engineering: A guide to building dependable distributed systems

[Russ 2001] Russ Housley and Tim Polk, 2001, Planning for PKI: Best practice Guide for Deploying Public Key Infrastructure"

[Schneier 1993] Bruce Schneier, 1993, "Description of a New Variable-Length Key, 64-Bit Block Cipher (Blowfish)" Available at: http://www.counterpane.com/bfsverlag.html (Read in November 2002)

[Schneier 2000] Bruce Schneir, 2000, Secrets and lies: Digital security in a networked world

[Skott 1999] Skott Oaks (1999), Java security

[Skoudes. 2002] Ed. Skoudes, 2002, Counter Hack: A step by Step Guide to computer Attacks and effective defenses

[SmartGov 2002] "SmartGov", 2002, The GSA Smart Card Initiatives, United States Federal Government Web site [Online] Available at: *http://smart.gov/* (Read in September 2002)

[SSE-CMM 2003] 2003, Systems Security Engineering Capability maturity Model, (SSE-CMM Version 3) [Online] Available at: http://www.sse-cmm.org/model/ssecmmv2final.pdf (Accessed in July 2003)

[Stal 2000] Darryl Stal, 2000, Security Targets. Entrust Technologies. Available at: http://www.commoncriteria.org/stRpt/EntrustRA51.pdf (Read November in 2003)

[Stallings 2000] William Stallings (2000), Network security essential applications and standard

[Steiner 1988] Jennifer Steiner, Clifford Neuman, and Jeffrey I. Schiller (1988) "Kerberos: An Authentication Service for Open Network Systems", [Online] Available at: *http://www.pdc.kth.se/kth-krb/* (Read in June 2002)

[Swanson 2003] Marianne Swanson, Nady Bartol, John Sabato, Joan Hash, and Laurie Graffo, 2003. Security Metrics guide for Information Technology Systems. Available at: http://csrc.nist.gov/csspab/june13-15/sec-metrics.html (Read September 2003)

[SW-CMM 2003] 2003, CBA IPI and SPA Appraisal Results 2002 Year End Update (CMM based Appraisals for Internal Process Improvement (CBA Ibis) and Software Process Assessments (SPAs) [Online] Available at: http://www.sei.cmu.edu/sema/pdf/SW-CMM/2003apr.pdf (Accessed in July 2003)

[TCSEC 1985] TCSEC (1985), Trusted Computer System Evaluation Criteria

[TF-AACE 2002] TERENA, 2002, "Authentication, Authorization Coordination for Europe"Available at http://www.terena.nl/tech/projects/pki/ (Read in September 2002)

[Wasley 2001] David L. Wasley (2001) "The need for strong PKI in a University" Available at: http://middleware.internet2.edu/draft-wasley-pkiapps-00.pdf (Read in September 2002)

[Westin 1967] Alan F. Westin, 1967, Intrusions on Privacy: Self-revelation, Curiosity, and Surveillance in: Alan F. Westin, *Privacy and freedom Atheneum, New York, 1967 pp. 52-63*

[Williamson 2003] Robert L.Williamson, Jr.,Tammy S. Compton, James L. Arnold, Jr., and J. Mark Braga Science Applications International Corporation (SAIC) Common Criteria Testing Laboratory (CCTL) Technical Directorate, SAIC CCTL. Available at: http://www.saic.com/infosec/pdf/CCTL-ITEA.pdf. (Read in March 2003)

[X500 1993] "ITU-T Recommendation X.500, 1993, - Information technology - Open Systems Interconnection - The Directory: Overview of concepts models and services", [Online] Available at: *http://www.dante.net/np/ds/osi.html* (Read in May 2002)

[Yngström 1996] Louise Yngström, (1996), A systemic-holistic approach to academic programmes in IT security

Appendix A: Acronym

PKI	Public Key Infrastructure
CA	Certification Authority
CRL	Certificate Revocation List
FIPS	Federal Information Processing Standard
CC	Common Criteria
PST	Product's security function
TOE	Target of evaluation
PP	Protection profile
TCA	Top certification authority
LSPKI	Large scale public key infrastructure
SSE-CMM	Systems Security Engineering Capability Maturity Model
SM	Security metrics
APF	Application function
NC	Name constraint

Appendix B: PKI Cryptography Technologies

Cryptography

Cryptography is the science of secret writing [[Stallings 2000]. Cryptographic systems are generically classified along three independent dimensions namely 1. The type of operations used for transforming plaintext to cipher text, the number of keys used, and the way in which plaintext is processed. All encryption algorithms are based on two general principals: substitution, in which each element in the plain text (bit, letter, group of bits or letters) is mapped into another element, and transposition, in which elements in the plaintext are rearranged. The fundamental requirement is that no information be lost (that is all operations be reversible). 2. The number of keys used. If both sender and receiver use the same key, the system is referred to as symmetric, single key, or conventional encryption. 3. The way in which plaintext is processed. A block cipher processes the input one block of the elements at a time, producing an output block for each input block. A stream cipher processes the input elements continuously, producing output one element at a time, as it goes along.

Cryptanalysis

The process of attempting to discover the plaintext or the key is termed as cryptanalysis. The success of cryptanalyst attack depends on the amount of information available to the cryptanalyst. The most important information is the algorithm used for encryption and the type of plaintext that is concealed, such as French text, Java source listing, an accounting file and so on. If the opponent knows the algorithm, one possible attack is the brute force approach of trying all possible keys until an intelligible translation of the ciphertext into plaintext is obtained. If the key space is very large, this becomes impractical. If the opponent does not know the algorithm used he must relay on the analysis of the ciphertext itself. Cipher text only attack is easier to defend because the opponent has list of information to work with.

However, in many cases the opponent has more information. The analyst may be able to capture one or more plaintext messages as well as their encryptions. Or he may know that certain plaintext patterns will appear in a message. For example, a file that is enclosed in a postscript format always begins with the same pattern, or there may be standardized header or banner to an electronic funds transfer message, and so on. All these are examples of known plaintext. With this knowledge, the analyst may be able to deduce the key on the basis of the way in which the known plaintext is transformed. Other attacks the opponent may be able to launch are probable-word attack. This happens hen the opponent has some knowledge of what is in the message. For example if the entire accounting file is being transmitted, the opponent may know the placement of certain key words in the header of the file.

Conventional Encryption algorithm

The mostly commonly used conventional encryption algorithms are block ciphers [Gollmann 2000]. In this section three important conventional algorithms are looked into. These are the data encryption standard (DES), the triple data encryption algorithm (TDEA), and the new advanced encryption standard (ASE). Also we provide a brief overview of other popular encryption algorithms.

Feistel Cipher Structure

Essentially all conventional block encryption algorithms, including DES, have derived their structure from a structure first described by Horst Feistel [Gollmann 2000]. The Feistel Principal is depicted in Figure 3.1. Feistel ciphers iterate the same basic step in a number of rounds. The input to round *i* is divided to two halves, L_i *and* R_i and the output can then be computed.

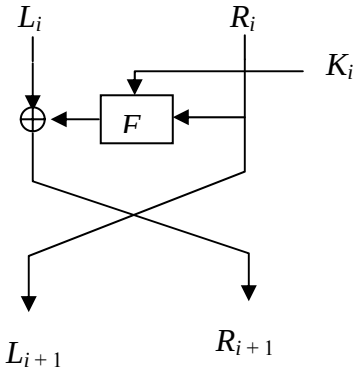


Figure 3.1 The Feistel principal
Source: Gollmann [7]

The realization of Feistel principal depends on the following parameters:

Block size: Larger block size means increased security but reduced encryption/decryption speed.
Key size: Larger key size means greater security but decreased encryption/decryption. The common key length in modern algorithm is 128 bits.
Number of rounds: According to Feistel principle, a single round offers less inadequate security but that multiple rounds offer increasing security.
Subkey generation algorithm: Greater complexity of this algorithm leads to difficulty cryptanalysis.
Round function: As well, greater complexity generally means greater difficulty for cryptanalyst.

Data encryption standard

The National Bureau of Standard, now The Institute of Standard and Technology (NIST) [FIBS PUB 142] adopted the Data Encryption Standard (DES) in 1977 as a Federal Information Processing Standard (FIPS). This is the most widely used encryption scheme

that is defined in the encryption standard (DES). The algorithm itself is referred to as the data encryption algorithm (DEA). In this scheme the plaintext is 64 bits in length and the key is 56 bits in length. DES is based on the Feistel Principal [Gollmann 2000] as depicted in Figure 3.1

The Strength of DES

Over years there have been attempts to find and exploit weakness in DES algorithm, making it the most studied encryption algorithm in existence. Despite all the efforts no one has publicly declared discovering fatal weakness in DES [FIBS PUB 142]. A more serious concern is the key length. As processor speed has risen and the Personal computer price has fallen, it is a simple matter to break DES quickly using brute force attack. DES was broken in 1998 using a special DES Cracker machine [FIBS PUB 142] The attack took less than three days.

Triple DEA

Triple DEA **(TDEA)** was incorporated as part of the data encryption standard in 1999 [FIPS PUB 180]. DEA uses three keys and three execution of DES algorithm. With three distinct key DEA has the key length of 168 bits. The use of two keys is also allowed, with $K_1 = K_3$; this provides for a key length of 112 bits. Since the underlying cryptographic algorithm is DEA, TDEA can claim the same cryptographic strength. With 168-bit key length, brute force attack is practically impossible.

However, TDEA has some principle drawbacks. First, the algorithm is relatively slow in software. The original DEA was designated for mid 1970s hardware implementation and does not produce efficient software code. Secondly both DEA and TDEA use 64-bit block size. For reasons of efficiency and security a larger block cipher is desirable. Because of all these reasons TDEA is not desirable for the next decade use. As a remedy NIST has been working on a new Advanced Encryption Standard (AES) since 1997.

Advanced encryption Standard

The Advanced encryption standard (AES) specifies the Rejndael algorithm, a symmetric block cipher with a block length of 128 bits and support key lengths of 128, 192, and 256 bits. Through out the AES standard the Rejndael algorithm is referred to "AES algorithm". The evaluation criteria for selecting this algorithm out of the rest included security, computational efficiency, and memory requirements, software and hardware suitability, and flexibility [FIPS197], [3]. The final standard was picked and announced in November 2001. Marketplace acceptance may take several years after the announcement of AES.

IDEA

The International Data Encryption Algorithm (DEA) is one of the symmetric block ciphers developed in 1991 by James Massey and Xuejin Lai of the Swiss Federal Institute of Technology. IDEA has the following properties [1]:

- Uses 128-bit key
- Input passes through 8 rounds

- Relies on three mathematical functions namely XOR, binary addition of 16-bit integers and binary multiplication of 16-bit integers.
- The sub-key generation algorithm relies on the use of circular shifts that are used in a complex way to generate a total of six sub-keys for each of the 8 rounds.

IDEA so far appears to be resistant to cryptanalysis. It is one of the early 128-bit proposed replacements of DES. It has been used as one of the alternative algorithms in PGP.

CAST-128

CAST-128 is one of the algorithms developed as part of the CAST project. It is defined in RFC 2144. CAST-128 uses 12 or 16 round Feistel cipher that has a block size of 64-bits and a key size of 40 to 128-bits in the increment of 8-bits. It uses rotational to provide immunity to linear and differential attacks. It uses a mixture of XOR, additional and subtraction in the round function and it uses three variations of the round function itself throughout the cipher. The s-boxes used, ones that are considered to be larger than those used in DES, each have a minimum nonlinearity of 74 and maximum entry of 2 in the difference distribution table. Its decryption/encryption performance is greater than 3.3MB.Sec on any computer higher than 150 MHz Pentium processor.

This algorithm is available world wide on a royalty-free basis for commercial and non-commercial uses. It is beginning to be used in a number of products including PGP.

Blowfish

Blowfish was developed by B. Schneier in 1993. It was designed to be easy to implement and to have high execution speed. It can run with less than 5K of memory. The key length is variable and can be up to 448-bits. In practice 128-bits keys are used and, like DES, Blowfish uses 16 rounds, S-boxes, and function but also uses binary addition. Unlike DES the S-boxes in Blowfish are dynamic and are generated as a function of the key. Blowfish is not suitable for applications in which the secret key changes frequently because the algorithm requires 512 executions to generate the subkeys and S-boxes. In this process the algorithm mangles the bits very thoroughly and as a result cryptanalysis is made very difficulty. So far there is no practical weakness have been found in Blowfish.

RC5

RC5 is defined in RFC 2040. It is used in a number of products from RSA. Ron Rivest one of the inventors of RSA designed RC5. It was designed to be suitable for hardware and software implementation and fast. This was achieved because RC5 use primitive computational operations usually found in microprocessors. RC5 is a simple algorithm and is word oriented and the basic operation work on full words of data at a time. It uses variable length of key and variable number of rounds. These parameters allow tradeoffs between speed and security. RC5 law memory requirement allows RC5 to be suitable for devices with restricted memory like smart cards. Also RC5 has the characteristic of incorporating rotations (circular bit shifts) whose amount is data dependent. This appears to strengthen the algorithm against cryptanalysis.

The conventional encryption algorithms are summarizes together in **Table 3.1**

Conventional Encryption Algorithm (Source: Stallings [Stallings 2000])

Algorithm	**Key Size**	**Number of Rounds**	**Mathematical Operations**	**Applications**
DES –Data Encryption Standard	56 bits	16	XOR, fixed S-boxes	SET, Kerberos
TDES -Triple DES	112 or 168 bits	48	XOR, fixed S-boxes	Financial Key management, PGP, S/MIME
IDEA- International Data Encryption Algorithm	128 bits	8	XOR, additional, multiplication	PGP
Blow fish	Variable to 448 bits	16	XOR, Variable S-boxes, additional	
RC5	Variable to 20408 bits	Variable to 255	Additional, subtraction, XOR, rotation	
CAST-128	40 to 128 bits	16	Addition, subtraction, XOR, rotation, fixed S-boxes	

Public-Key Cryptography and Message authentication

Message authentication is a procedure that communicating entities verify that received messages are authentic. The message is said to be authentic if it has not been modified and that the source is authentic. Another aspects we may verify are timeliness and the sequence. The timeliness can reveal that the message has been artificially delayed and replayed. Generally, all these kind attacks (falsification of data and transaction) are termed as active attack while passive attack can be stopped using encryption.

Convention encryption

Authentication using conventional encryption can be performed if the sender and receiver share a key; then only the sending entity will be able to encrypt messages successfully for the receiving entity. For the receiving entity to be sure that the sequence is proper, no alterations has been made and that the message has not been delayed and replayed the message has to be time stamped and must include error detection code and a sequence number.

Message authentication code

Message authentication code (MAC) is a small block of data which is generated by the use of secret key shared among communicating entities. If A and B are the communicating entities and they share a secret key $K_{AB,}$ when A wants to send a message to B, it calculates the message authentication code as a function of the message and the key:

$$MAC_M = F(K_{AB}, M)$$

Then the authentication code is appended to the message. The receiving entity performs the same calculation on the received message using the same secret key. The calculated code is compared to the received code. If the secret key has not been compromised, then the receiver is sure that the message has not been altered and that the message is from alleged sender.

One-Way Hash Function

Secure hash function is important for both authentication and digital signature. The hash function produces a fingerprint of a block of data. The hash function H, is useful if it has the following properties:

- H can be applied to a block of any size
- H produces a an output of fixed length
- Hardware and software implementation must be easy
- If H(x) = h where h is the code and x is the block of data, then given h, it is computationally infeasible to find x. This is a one-way property. It prevents the attacker from discovering the secret value
- For any given block of data x, it is computationally infeasible to find $y \neq x$ with H(x) = H(y). This property prevents forgery, as it is not possible to generate the hash code from a different block of data.
- It is computationally infeasible to find any pair of x and y such that H(x) = H(y). This property protects against birthday attack.

SHA-1

SHA-1 is a revision of secure hash algorithm (SHA) in 1995[Stallings 2000]. SHA was developed by the National Institute of Standard and Technology and was published as FIPS in 1993.

SHA-1 takes as an input a message with maximum length not greater than 2^{64} bits and produces an output of 160 bits message digest. The input is processed in 512-bits blocks. This algorithm has one peculiar property that every bit of the hash code is a function of every bit of the input. The compression function or module function is repeated such that the result is well mixed. As a result is very unlikely that two messages that are randomly chosen can produce the same hash code. So far there is no weakness in SHA-1 that has been published. SHA-1 provides no more than 80 bits of security against collision attacks.

The National Institute of Standards and Technology (NIST) has made available three other hash algorithms that are more resistant to collision attacks. The, new algorithms are SHA-256, SHA-384 and SHA-512. SHA-256, is a 512-hash function that is designed to provide 128 bits of security against collision attack and the SHA-512 is indented to provide 256 bits of security against collision attacks. The 384 may be obtained by truncating the SHA-512 output.

MD5

MD5 algorithm takes as an input a message of arbitrary length and produces an output of 128 bits message digest. The input is processed in 512-bits blocks. This algorithm is defined in RFC 1341. Currently due to increased processor computing power, the security

of MD5 is questionable. It can be verified that the difficulty of coming out with two message having the same message digest is on the order of 2^{64} operations, while the difficulty of finding a message with a given digest is in the order of 2^{128} operations. The 2^{64} operations are too small for security.

RIPEND-160

RIPEND-160 takes an input of arbitrary length message and the product is an output of 160-bits message digest. Similar to MD5, the input of MD5 is processed in 512-bits blocks. This algorithm was developed under the European RACE Integrity Primitive Evaluation (RIPE) project. It was developed after MD4 and MD5 was successfully attacked.

HMAC

HMAC is defined in RFC 2104. Cryptographic hash code have advantages like fast execution speed in software compared to conventional encryption algorithm like DES, widely available code, and there is no export restrictions. Due to these factors, there has been increased interest in developing MAC that is derived from cryptographic hash code. For example, SHA-1 does not relay on a secret key, therefore it cannot be directly used for MAC. HMAC is a product of efforts to incorporate the secret key in existing hash algorithms and has been chosen as a mandatory to implement MAC for IP security. It is also used in other Internet protocols like TLS and SET.

The design objectives of HMAC as outlined in RFC 2104 are to use without modification freely available hash functions, to allow easy replaceability in case faster algorithms are required, to handle keys in a simple way, to have well understood cryptographic analysis of the strength of the authentication mechanism and to preserve the original performance of the hash function. The performance of HMAC for long messages is approximately the same as the embedded hash function.

Secure Hash Functions comparison (Stallings [Stallings 2000])

	MD5	**SHA-1**	**RIPEND-160**
Digest Length	128 bits	160 bits	160 bits
Basic un it of processing	512 bits	512 bits	512 bits
Number of steps	64(4 rounds of 16)	80(rounds of 20)	160 (5 paired rounds of 16)
Maximum message size	∞	$2^{64}-1$ bits	∞
Primitive logical functions	4	4	5
Additive constants Used	64	4	9

Public Key Cryptography

Public key encryption is used in message authentication and key distribution. Public key cryptography is asymmetric, involving the use of two separate keys namely private key and public key. In public key cryptography the algorithms are based on mathematical functions rather than simple operations on bit patterns. The use of two keys introduces profound effects in the security services like non-repudiation, confidentiality, key distribution and authentication

The public key pair of the pair is made publicly available to other users, while the private key is made secret known only to its owner. Generally a public key algorithm relies on one key for encryption and a different but related key for decryption.

The use of public key crypto systems can be classified into the following categories: 1) Encryption and decryption: the sender encrypts the message with the recipient's public key. The recipient will decrypt the message from the sender using his private key. 2) Digital signature: The sender signs a message with it's private key. The signing is accomplished by applying cryptographic function to the message or a small block of data that is a function of the message. 3) Key exchange: Two sides cooperate to exchange a session key. Some cryptographic algorithm like RSA, are suitable for all these function while other are not.

Table 3.3 Applications for Public-Key Cryptosystems

Algorithm	Encryption/Decryption	Digital Signature	Key Exchange
RSA	Yes	Yes	Yes
Diffie Hellman	No	No	Yes
DSS	No	Yes	No
Elliptic Curve	Yes	Yes	Yes

The RSA Public-Key Encryption Algorithm

RSA is one of the most widely used public-key algorithms. It is a block cipher in which the plaintext and ciphertext are integers between 0 and n-1 for some n.

The strength of RAS can be challenged by the brute force attack. However, the larger the numbers e and d the more secure is the algorithm. Since the private key and public key are both involved in the process of generation, decryption and encryption the lager the keys becomes the slower the system will be.

Diffie-Hellman Key Exchange

Diffie-Hellman is meant for key exchange only. Unlike RAS it does not support encryption and decryption. Its strength depends on the difficulty of computing discrete algorithms. Discrete algorithm can be defined as follows:

PKI architecture

A PKI is defined as the set of hardware, software, people, policies and procedures needed to create, manage store, distribute, and revoke public key certificates (PKCs) based on public-key cryptography [RFC 3280]

A PKI consists of the following components [RFC 3280]:

- Certification Authorities (CA) that issue and revoke PKCs;

- Registration Authorities (RAs) that vouch for the binding between public keys and certificate holder identities and other attributes;
- PKC holders are issued certificate and can sign digital documents and decrypt documents using private key;
- Clients that validate digital signatures and their certification paths from a known public of a trusted CA and that encrypt document using public key from certificates of PKC holders;
- Repositories that store and make available PKCs and Certificate Revocation Lists (CRLs).

Key generation

The CA can either generate by the user or the private-public key pair. This depends on the CA's policy. If the CA generates the key pair, the key material may be distributed to the user on a token or in an encrypted file.

Key compromise

In the event that the key is compromised, the transition from old key to new key will not be graceful in that there will be no planned switching from PKCs and keys. The PKI must support the ability to declare that the previous PKC is now invalid and shall not be used and to announce validity and availability of the new PKC. If the root CA's private key is compromised, that CA's PKC must be revoked and all PKCs subordinate to it must also be revoked until the root CA has been issued a new Pica's and the root CA issues PKCs to users relying upon it. Additionally, once the Root CA is has got the new key, it will be necessary to re-issue PKCs that are signed with new key to all subordinate users since their current PKCs would be signed by the now revoked key. To tell the users to make the switch, some secure out of bound mechanism will have to be used.

Key Expiry

Users will know in advance when the user's or CA's key will expire, therefore, KPI needs to provide a facility to gracefully transition from a PKC with an existing key to a new PKC with a new key. This is particularly important if the expiring key belongs to a CA.

PKI Trust Models

There are several trust models that have been developed and the choice of the model will depend on a particular environment. The following provides some background on the models.

Hierarchical model

Hierarchical model is one of the initial trust models proposed [32]. In this model the root CAs is the top most CAs in the entire domain. The root CA issues certificates to subordinate CAs, and the subordinate CAs offers certificates to the users. In this model control is imposed from top down. The name constraint can be included in the subordinate CA to limit the name space in which they are allowed to issue certificate. Also the root CAs ensures domain wide policies on cross-certification. Cross certificates allows interoperability and can be issued by the root CA or the subordinate CAs.

Local/Federation trust model

In this model the users trust the CA that issued the certificate to them. The idea is since the CA is local and known to users there is more trust rather than with unknown distant CA. In order for the EE s from different CA to communicate the CAs issue each other certificates thereby creating a certification path from one EE to another to form some kind of a federation. The main benefit for the local model is flexibility.

Root Repository

This model uses a file to store the PKCs of many CAs. The RP then trusts any PKC stored in the file. The PKC included in the root repository may be a root CA for some other domain or subordinate CA, but when included in the trust file whatever PKC it is in the other domain, it becomes a root CA for the RP The main advantage for this model is that cross certificate is not required because the RP can just choose to trust any PKC.

Root Repository's Perspective

This model has recently received attention. In this model instead of the CA imposing restraints on the PKC, the RP instead makes the determination as to which certificate to trust. The RP decides which domain it will receive it's certificate, which key usage it will accept, etc. In this model cross certificate is also not required because the RP can decide to trust a particular PKC or domain of PKCs.

Other PKI architectures are Mesh, cross certification and Bridge. The Mesh PKI is the primary alternative to hierarchical. This architecture is also referred to network PKI or web of trust. In this architecture multiple CAs provide services and are related through peer-to-peer relationships. Each user trusts a single CA but the trusted CA is not the same for all users. Connects users from different PKIs.

Certificate revocation list

The Certificate Revocation List (CRL) is a list of certificates that a revoked due to one of the following reasons:

- Compromised user's private key
- The CA's certificate is or assumed to be compromised
- Change of association between the subject and the CA
- A user name is changed

Every CA must maintain a list consisting of all revoked but not expired certificates. This CRL is signed and posted to the directory by the CA. The distribution can be done using the X.500/LDAP directories, Web, and file. The certificate contains a pointer to these locations where the CA has published the CRL. When the user receives a certificate he must verify whether this certificate is valid or revoked. This significantly increases the overall PKI operational and implementation cost. To minimize this cost there have been several CRL schemes which have been developed to curb bandwidth utilization problem.

Certification path

Certification path is also referred to certification chain. Any user with access to the public key of the CA can recover the user public key that was certified and no party other than the certification authority can modify the certificate [Stallings 2000]. Certification path processing verifies the binding between the subject distinguished name and/or subjects alternative names and subject public key. The binding is limited by the constraints that are specified in the certificate which comprise the path. The basic constraints and policy constraints extensions allow the certification path processing logic to automate the decision making process.

Path validation requires obtaining a sequence of certificates that support the binding between the subject's distinguished name and/or alternative name and the subject public key.

X.509 suggests that CAs be arranged in a hierarchy so that navigation is straightforward. For example Fiqure 3.4 is an example of such a hierarchy. The X.509 CAs directory entry includes two types of certificates. 1.) Forward certificate: These are CA's certificates generated by other CAs and 2.) Reverse certificate: These are certificates generated by CA that are certificates of other CAs

Light Directory Application Protocol

The Light Directory Application Protocol (LDAP) was developed by Michigan University and it was further developed and standardised in the IETF as RFC 1777. LDAP was specifically designed to provide access to X.500 directory while not incurring the resource requirement of the Directory Access Protocol (DAP). This protocol specifically targeted at simple management application and browser applications that provide simple read/write interactive access to the X.500 directory.

PKI management protocol

Key management protocol (KMP) is required to support on-line interactions between PKI user and management entities. For instance the management protocol might be used between a CA and a client system with which a key pair is associated, or between two CAs which cross-certify each other. The KMP can support the following functions:

Registration: This is a process where the subject makes itself known to the CA. This can be done directly or through a Registration Authority prior to the CA issuing a PKC or PKCs for that subject. This involves the subject providing credentials (for example its name, domain name, IP address) and other attributes that are verifiable by the CA.

Certification: This is a process in which a CA issues a PKC for a subject's public key, and returns that PKC to the subject or posts that PKC in a repository.

Initialization: Initialization can involve providing the client system with the public key or PKC of a CA, or generating the client system's own key pair.

Key pair recovery: Key pair recovery is useful for implementations that include those who store private key on a hardware token that can be broken or lost or when a private key is protected by a password that can be forgotten. In these cases user's private key can be backed up by the CA or by a separate key backup system without providing an unacceptable risk of compromise of the private key.

Key update: Key update happens when a key has passed its maximum usable lifetime and when the key has been compromised and must be replaced.

Revocation Request: The subject or authorized person advises a CA of abnormal situation requiring certificate revocation.

Appendix B: Common Criteria (CC) and SSE-CMM

Security testing and evaluation

In this chapter the historical background of security evaluation, and an overview of Common Criteria (CC) and Systems Security Engineering Capability Maturity Model (SSE-CMM) formal security model are presented. Our effort to develop security metrics and PKI applications ability to validate X.509 certificates will be based on these two models.

There are many difficulties related to understanding and describing validation, assessment, verification and evaluation of IT security functionality. Schneir [Schneir 2000] underlines that normal security testing fails because first, security flaws can appear anywhere either in the trust model, the system design, the algorithm and protocol, the implementation, the source code, the human computer interface, the procedure, the underlying computer system (hardware or system software). A single flaw can break the security of the entire product.

Different IT security criterias evolved from 1980's. In the early 1985 the Trusted Computer System Evaluation Criteria (TCSEC) [TCSEC 1985] were developed in the United States. TCSEC presents different focus and have been used in different ways to address confidentiality security service including the fact that it was used as an input of other IT security criteria which evolved latter.

In the succeeding decade, various countries began initiatives to develop evaluation criteria that built upon the concepts of the TCSEC but were more flexible and adaptable to the evolving nature of IT in general. In Europe, European Community published the Information Technology Security Evaluation Criteria (ITSEC), version 1.2, in 1991 [ITSEC 1991]. This was achieved by a joint development by the nations of France, Germany, the Netherlands, and the United Kingdom. Unlike TCSEC, ITSEC addresses confidentiality, integrity and availability

In Canada, the Canadian Trusted Computer Product Evaluation Criteria (CTCPEC)

version 3.0 was published in early 1993 [CTCPEC 1993]. CTCPEC addresses

confidentiality, integrity, availability and accountability.

In the United States, the draft Federal Criteria for Information Technology Security

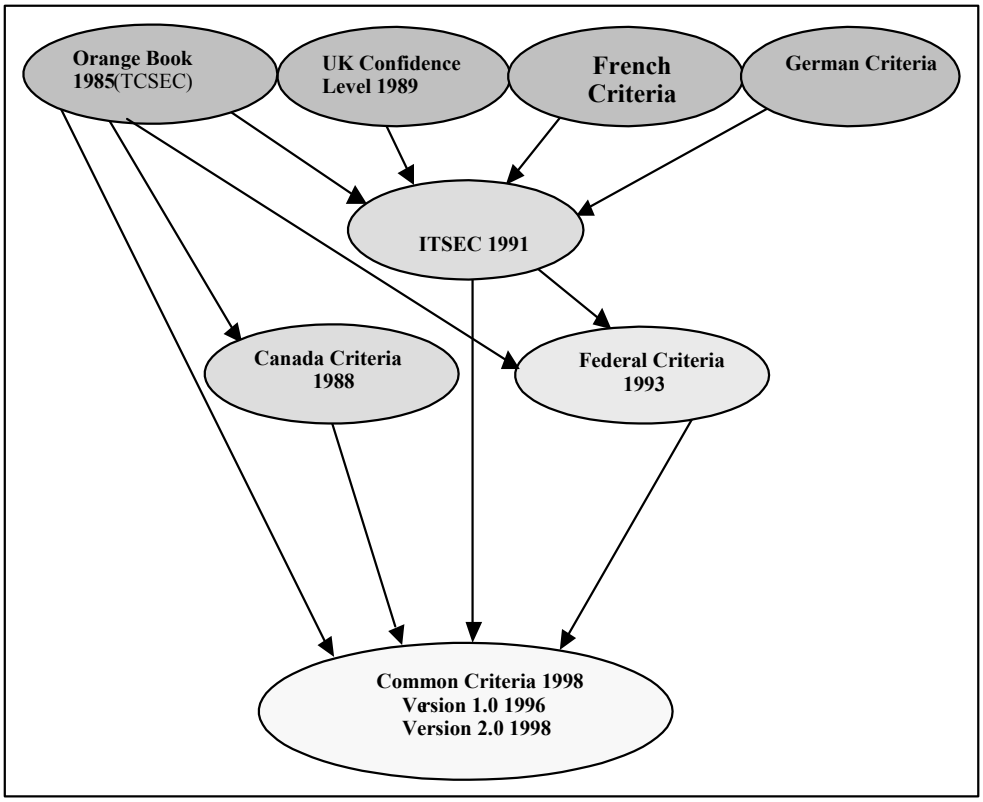
(FC) version 1.0 was also published in early 1993 [FC 1993], as an approach to combine

North American and European concepts for evaluation criteria.

The Common Criteria development work had begun in 1990 in the International Organisation for Standardisation (ISO) to develop a set of international standard evaluation criteria for general use in a global IT market [CC 1998]. This version of the Common Criteria for Information Technology Security Evaluation (CC 2.1) is a revision that aligns it with International Standard ISO/IEC 15408:1999 and IT products/or systems shown to be compliant with such specifications, are considered to be ISO/IEC

15408:1999 compliant. The Common Criteria (CC) represents the outcome of efforts to develop criteria for evaluation of IT security that are broadly useful within the international community.

The common Criteria (CC)



CC Evolution

CC is useful as a guide for IT developers and users of security products and systems. When CC is in use such products and systems are referred to as Target Of Evaluation (TOE) [CCMIB-99-031]. Such TOEs includes, for example, computer networks, applications, operating systems, distributed systems, databases and including security measures installed in firmware, hardware and software. CC defines two requirements namely the assurance requirement and functional requirements.

CC assurance requirement

Assurance provides confidence that an IT product or system meets its security objective. CC provides assurance through active investigation. Active investigation is an evaluation of an IT product or system in order to determine its security properties. Evaluation is the basis for CC approach [CCMB-99-033]. CC defines that greater assurance results from the application of greater evaluation effort. Effort is scaled into Scope, depth and rigor. This is because effort can be greater due to the following: Larger portion of the IT product or system is tested, it is deployed to a finer level of design and implementation details, and because it is applied in a more structured and formal manner.

CC Security Functions

Security functions, as outlined in the Common Criteria, comprises all hardware, software, and firmware of the product to be evaluated that must be relied upon for the correct enforcement of how the assets are managed, distributed and protected within the product under evaluation. CC specifies the security functions as: Identification and Authentication, Cryptographic support, Identification and Authentication, Security management, Privacy, Communication, Protection of The products security function, Resource utilization, Security functional requirements application notes, Trusted path, User data protection, and Security audit. These security functions are discussed further later in this chapter.

Common Criteria limitations

Some topics are peripheral to IT security but they have an impact on the way the IT security functions to counter the threats. Such topics includes

- Administrative measures such as organizational, personnel, physical and procedural controls.
- Physical aspects like electromagnetic emanation control
- Administrative and legal framework under which CC may be applied
- Procedures for use of evaluation results for instance accreditation process
- CC does not cover the assessment of inherent qualities of cryptographic algorithms. If crypto algorithms are part of TOE, the evaluation scheme under which CC is applied must provide for such assessment [CCIMB-99-031].

Evaluation Criteria, method, scheme and certification

The certification body is necessary for final evaluation results inspection and approval. The certification body requires some knowledge about the criteria, methodology, scheme of evaluation and TOE.

Certification process can be considered as one way of gaining consistency in the process of applying the evaluation criteria. Expert judgment and background knowledge is a preferred method for many criterias [CCMB-99-031]. The evaluation scheme defines the scope of evaluation. For example if the evaluation criteria in use is not comprehensive enough, like the Common Criteria, to cover all components of the target of evaluation, the scheme will have to define how these components will be evaluated.

CC Evaluation Assurance Levels

CC defines seven Evaluation Assurance Levels [CCIMB-99-031]

Functionally tested

Functionally tested is applicable where some confidence in correct operation is required, but the threats to security are not viewed as serious. It will be of value where independent assurance is required to support contention that due care has been exercised with respect to the protection of personal or similar information. This level provides an evaluation of

the Target of Evaluation TOE as made available to the consumer, including independent testing against a specification, and an examination of the guidance documentation.

Structurally tested

Structurally tested requires the cooperation of the developer in terms of the delivery of design information and test results, but should not demand more effort on the part of the developer than is consistent with good commercial practice. As such it should not require a substantially increased investment of cost or time. This is applicable in those circumstances where developers or users require a low to moderate level of independently assured security in the absence of ready availability of the complete development record. Such a situation may arise when securing legacy systems, or where access to the developer may be limited.

Methodically tested and checked

Methodically tested and checked permits a conscientious developer to gain maximum assurance from positive security engineering at the design stage without substantial alteration of existing sound development practices. It is applicable in those circumstances where developers or users require a moderate level of independently assured security, and require a thorough investigation of the TOE and its development without incurring substantial reengineering costs.

This evaluation provides for selective confirmation of the developer test results, and evidence of a developer search for obvious vulnerabilities. Development environmental controls and TOE configuration management are also required.

Methodically designed, tested and reviewed

Permits a developer to maximize assurance gained from positive security engineering based on good commercial development practices. Although rigorous, these practices do not require substantial specialist knowledge, skills, and other resources. It is the highest level at which it is likely to be economically feasible to retrofit to an existing product line. It is applicable in those circumstances where developers or users require a moderate to high level of independently assured security in conventional commodity TOEs, and are prepared to incur additional security-specific engineering costs.

This level of evaluation provides an analysis supported by the low-level design of the modules of the TOE, and a subset of the implementation. Testing is supported by an independent search for vulnerabilities. Development controls are supported by a life-cycle model, identification of tools, and automated configuration management.

Types of evaluation

Protection Profile (PP) evaluation

Protection Profile is set of security requirement for a category of products (TOE) that are implementation-independent that meet specific consumer needs. The goal of this evaluation is to demonstrate that the PP is complete, consistent, and technically sound and suitable for use as a statement of requirements for an evaluating the TOE.

Security Target (ST) evaluation

Security Target is a set of security requirements and specifications to be used as a basis for evaluation of the target product (TOE). The goal of such an evaluation is twofold: first to demonstrate that the ST is complete, consistent, and technically sound and hence suitable for use as the basis for the corresponding TOE; second, in the case where an ST claims conformance to a PP, to demonstrate that the ST properly meets the requirements of the PP.

Target Of Evaluation (TOT) evaluation

The Target of Evaluation is the product and associated documents that is subject of an evaluation. The goal of such an evaluation is to demonstrate that the TOE meets the security requirements contained in the ST.

Assurance maintenance (AM)

TOE assurance maintenance is carried out against the evaluation criteria using a previously evaluated TOE as the basis. The goal is to derive confidence that assurance already established in a TOE is maintained and that the TOE will continue to meet its security requirements as changes are made to the TOE or its environment.

Security Functions

Security functions, as outlined in the Common Criteria, comprises all hardware, software, and firmware of the product to be evaluated that must be relied upon for the correct enforcement of how the assets are managed, distributed and protected within the product under evaluation. The CC specifies the security functions as: Identification and Authentication, Cryptographic support, Identification and Authentication, Security management, Privacy, Communication, Protection of The products security function, Resource utilization, Security functional requirements application notes, Trusted path, User data protection, and Security audit.

Identification and Authentication

This is a common security requirement to unambiguously identify the entity performing functions in a security product. This involves establishing the claimed identity of each user, but also verifying that each entity is indeed who claims to be. The unambiguous identification of authorised entities and the correct association of security attributes with entities and subjects is critical to the enforcement of the security policies.

Cryptographic support

The implementation of cryptographic support could be in hardware, firmware and/or software. These are important for identification and authentication, non-repudiation,

trusted path, trusted channel and data separation. This class is used when the PKI implements cryptographic functions.

Privacy

These requirements provide a user protection against discovery and misuse of identity by other users. It can be classified into four categories: Anonymity, Pseudonymity, unlinkability and unobservability [CC]. Anonymity ensures that the user may use resources without disclosing the user identity. Pseudonymity ensures that the user may user services and resources without disclosing the identity but is still accountable for his actions. Unlinkability ensures that the user may use multiple resources or services without others being able to link these uses together. Unobservability ensures that the user uses services and resources without a third party being able to detect that the service is being used.

Communication

Communication provides two families that are related to assuring the identity of the originator of transmitted information and assuring the identity of the recipient of transmitted information. These families ensure that an originator cannot deny having sent the message, nor can the recipient deny having received it (Non-Repudiation of origin and Non-Repudiation of Receipt).

Protection of the products security function

Protection of the products security function ensures functional requirements integrity and management of the mechanisms that provide the product's security function and to the integrity of product's security function data. For example this class defines the rules for the prevention of loss of availability of PSF data moving between the PSF and a remote trusted IT product. This data could, for example, be PSF critical data such as passwords, keys, audit data, or PSF executable code.

Resource utilization

Resource utilization functional class ensures the following: The TOE will maintain correct operation even in the event of failures, The requirements of this family allow the PSF to control the use of resources such that high priority activities will always be accomplished without undue interference or delay caused by low priority activities and the requirements of this family allow the PSF to control the use of resources by users and subjects such that denial of service will not occur because of unauthorized monopolisation of resources.

Security functional requirements application documentation

The user documents contain information that is of interest to users. The evaluator documents contain any information that is of interest to developers and evaluators of TOEs. This can include clarifications of meaning and specification of the way to interpret requirements, as well as caveats and warnings of specific interest to evaluators.

Trusted path

Trusted channel provides non-repudiation characteristics with respect to the identity of the sides of the channel. A trusted path provides a means for users to perform functions through an assured direct interaction with the TSF. Trusted path is usually desired for user actions such as initial identification and/or authentication, but may also be desired at other times during a user's session.

User data protection

User data protection ensures secure user data within a security product, during import, export, and storage as well as security attributes directly related to user data.

Security audit

Security auditing involves recognizing, recording, storing, and analyzing information related to security relevant activities. The resulting audit records can be examined to determine which security relevant activities took place which user is responsible for them. Security audit operations are usually based on the security policy.

Systems Security Engineering Capability maturity model (SSE-CMM)

.

The SSE-CMM has a relationship to ISO/IEC TR 15504, Information technology software process assessment, particularly part 2, a reference model for processes and process capability, as both are concerned with process improvement and capability maturity assessment. However, TR 15504 is specifically focussed on software processes, whereas the SSE-CMM is focussed on security.

The SSE-CMM initiative began as an NSA-sponsored effort in April 1993 with research into existing work on Capability Maturity Models (Cams) and investigation of the need for a specialized CMM to address security engineering. The information security community was invited to participate in the effort at the First Public Security Engineering CMM Workshop in January 1995. Representatives from over 60 organizations reaffirmed the need for such a model. As a result of the community's interest, Project Working Groups were formed at the workshop, initiating the Develop Phase of the effort. The first meetings of the working groups were held in March 1995. Development of the model and appraisal method was accomplished through the work of the SSE-CMM Steering, Author, and Application Working Groups with the first version of the model published in October 1996 and of the appraisal method in April 1997.

Definition of Security Engineering

Security engineering is an evolving discipline. As such, a precise definition with community consensus does not exist today [CEM-99/045]. However, some generalizations are possible. Some goals of security engineering are listed below:

- Gain understanding of the security risks associated with an enterprise
- Establish a balanced set of security needs in accordance with identified risks
- Transform security needs into security guidance to be integrated into the activities of other disciplines employed on a project and into descriptions of a system configuration or operation
- Establish confidence or assurance in the correctness and effectiveness of security mechanisms

- Determine that operational impacts due to residual security vulnerabilities in a system or its operation are tolerable (acceptable risks)
- Integrate the efforts of all engineering disciplines and specialties into a combined understanding of the trustworthiness of a system

Capability maturity model

A capability maturity model (CMM) such as the SSE-CMM describes the stages through which processes progress as they are defined, implemented, and improved. The model provides a guide for selecting process improvement strategies by determining the current capabilities of specific processes and identifying the issues most critical to quality and process improvement within a particular domain. A CMM may take the form of a reference model to be used as a guide for developing and improving a mature and defined process.

A CMM may also be used to appraise the existence and institutionalization of a defined process that implements referenced practices. A capability maturity model covers the processes used to perform the tasks of the specified domain, (e.g., security engineering). A CMM can also cover processes used to ensure effective development and use of human resources, as well as the insertion of appropriate technology into products and tools used to produce them.

SSE-CMM Maturity Levels

These five levels are informally described below,
Level 1, Performed Informally, focuses on whether an organization or project performs a process that incorporates the base practices.
Level 2, Planned and Tracked, focuses on project-level definition, planning, and performance issues.
Level 3, Well Defined, focuses on disciplined tailoring from defined processes at the organization level.

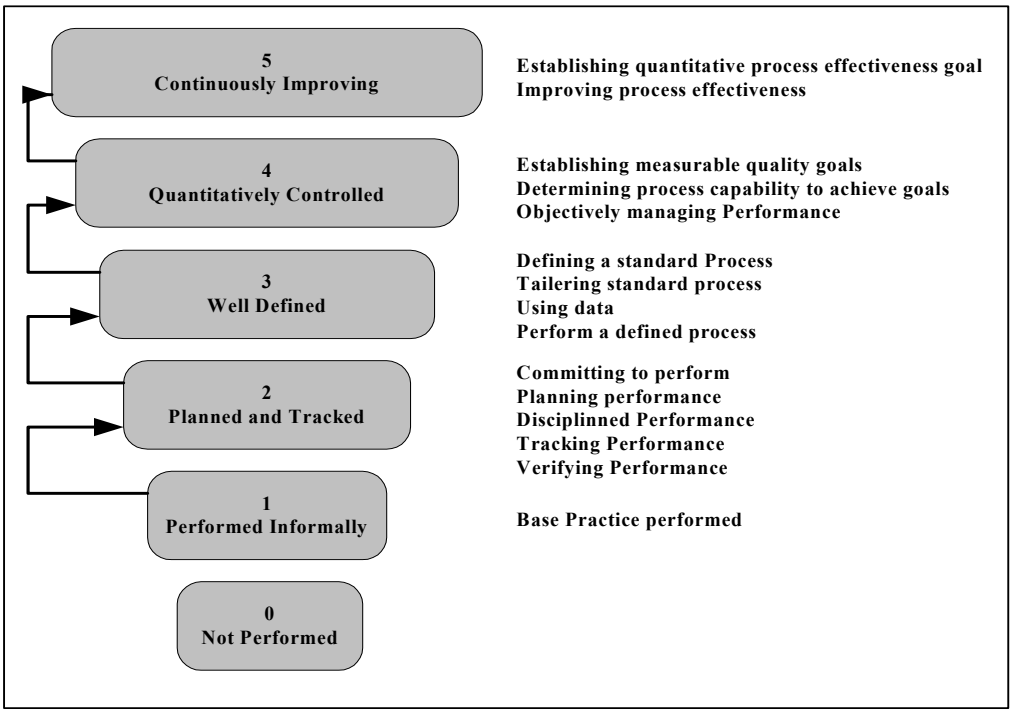


Figure 2.3 Levels of maturity of Security Engineering Organisation

Level 4, Quantitatively Controlled, focuses on measurements being tied to the business goals of the organization. Although it is essential to begin collecting and using basic project measures early, measurement and use of data is not expected organization wide until the higher levels have been achieved.
Level 5, Continuously Improving, gains leverage from all the management practice improvements seen in the earlier levels, then emphasizes the cultural shifts that will sustain the gains made.

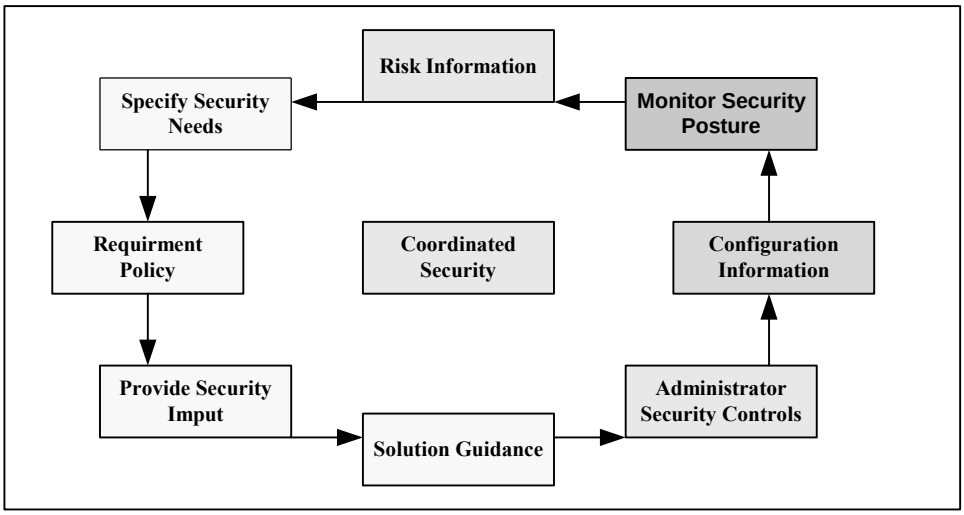
Security Engineering Process

The SSE-CMM divides security engineering into three basic areas: risk, engineering, and assurance. While these areas are by no means independent from one another, it is possible to consider them separately. At the simplest level, the risk process identifies and prioritizes dangers inherent to the developed product or system. The security engineering process works with the other engineering disciplines to determine and implement solutions to the problems presented by the dangers. Finally, the assurance process establishes confidence in the security solutions and conveys this confidence to the customers.

Security Engineering

Security engineering, like other engineering disciplines, is a process that proceeds through concept, design, implementation, test, deployment, operation, maintenance, and decommission. Throughout this process, security engineers should work closely with the other parts the system engineering team.

The SSE-CMM emphasizes that security engineers are part of a larger team and need to coordinate their activities with engineers from other disciplines. This helps to ensure that security is an integral part of the larger process, and not a separate and distinct activity. Using the information from the risk process described above, and other information about system requirements, relevant laws, and policies, security engineers work with the customer to identify security needs. Once needs are identified, security engineers identify and track specific requirements.



Security as an Integral part of Engineering [SSE-CMM]

The process of addressing security problems generally involves identifying possible alternatives and then evaluating the alternatives to determine which is the most practical. The difficulty in integrating this activity with the rest of the engineering process is that the solutions cannot be selected on security considerations alone. Rather, a wide variety of other considerations, including cost, performance, technical risk, and ease of use must be addressed. Typically, these decisions should be captured to minimize the need to revisit issues. The analyses produced also form a significant basis for assurance efforts. Later in the lifecycle, the security engineer is called on to ensure that products and systems are properly configured in relation to the perceived risks, ensuring that new risks do not make the system unsafe to operate.

Assurance

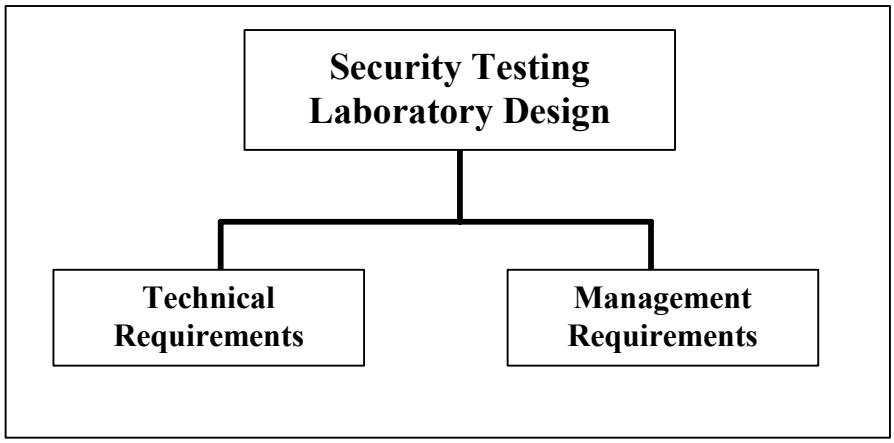
Assurance is defined as the degree of confidence that security needs are satisfied [NIST94a 1994]. There are many forms of assurance. The SSECMM contributes to one aspect, the confidence in the repeatability of the results from the security engineering process. The basis for this confidence is that a mature organization is more likely to repeat results than an immature organization. The detailed relationship between different forms of assurance is the subject of ongoing research. Assurance does not add any additional controls to counter risks related to security, but it does provide the confidence that the controls that have been implemented will reduce the anticipated risk.

The SSE-CMM activities themselves involve the production of assurance relevant evidence. For example, process documentation can indicate that the development has followed a well-defined and mature engineering process that is subject to continuous improvement. Security verification and validation play a large role in establishing the trustworthiness of a product or system. Many of the example work products included within the process areas will contribute to, or form part of that evidence. Modern statistical process control suggests that higher quality and higher assurance products can be produced more cost effectively and repeatedly by focusing on the process used to produce them. The maturity of the organizational practices will influence and contribute to the process.

Security Lab

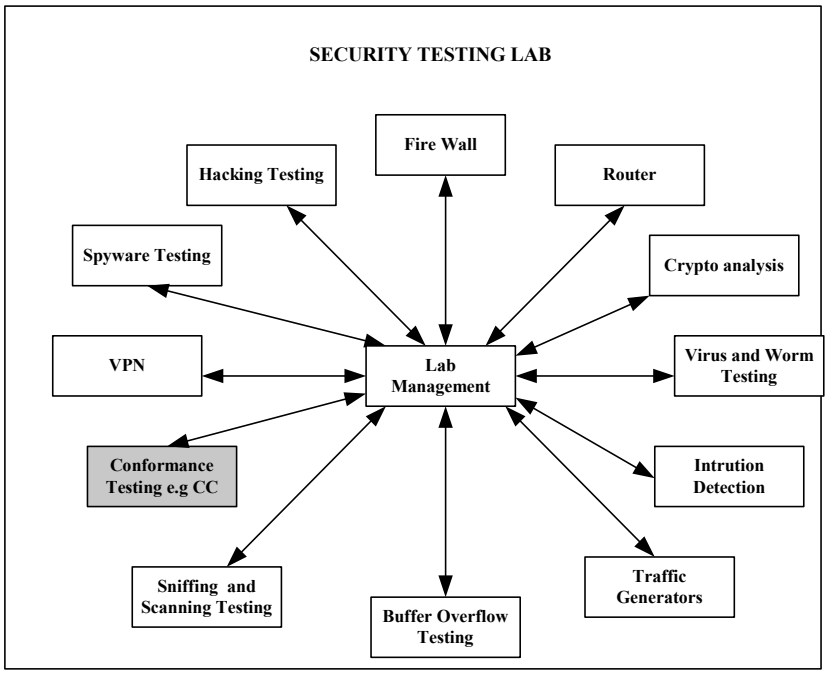
Security testing lab is a networking environment designed for conformance testing, evaluating exploits, viruses, and similar security related software, and in the case of training institution, it provides a facility for pen-test training, practice, and exercises. There are security measures that should be put in place to minimize the risk associated with therefore mentioned activities.

The administration and management computers can be designated for handling backups, logging, and creation of user. Hacking computers are used for hosting insecure and vulnerable software that trainees are allowed to hack into. Traffic generators are required because some experiments and exercises need background traffic. Scanning and network sniffing by using tools and techniques that gather information about hosts on a network, or that retrieves passwords from insecure protocols.



Security Lab Design

Firewall exercise enable trainees get acquainted with the firewall implementation. During the exercise trainees learn both how to configure a firewall and what different firewall rules mean in practice. By first constructing and implementing firewall rules and later testing them in practice, the trainees become familiar with the area of firewalls. Spyware investigation may include some of the most popular Peer-to-Peer tools available on the Internet. During this exercise the trainees investigate if these P2P tools are bundled with any components that may compromise user privacy. Security lab provides hands on exercise to trainees.



Security Lab components

However, precautions have to be taken especially when trainees exercise how to make virus and traffic generation. The best practice should be to isolate the lab physically from the cooperate network.

Appendix C: PKI large scale implementation problems

Application programming interface

One of thorniest problems in Large-scale implementation of PKI is interoperability between different PKI products. There is a need for a high-level Application Programming Interface (API) for public-key based cryptographic services to be developed. This can make the support across multiple PKI products easy and hence PKI enabled applications will be widely deployed.

The application requiring security services is any application that needs digital signature and/or encryption services. The security services/vendor products are the existing vendor products that provide the signing and encryption functionality. The product API is the vendor-specific interface provided by the product for calling the signing and encryption services. The high-level PKI services API is the common API that can be pacified to provide a consistent interface to signing and encryption services irrespective of the product being used. The high-level API is designed to hide the complexity of the underlying security mechanisms but facilitate service requests through simple service calls. The binding layer is the code necessary to translate the high-level PKI services API into the product API.

Certificate chain verification

Certificate chain verification involves tracking the certification path. A certificate path is an ordered list of certificates starting with a certificate issued by the relying party's trust root, and ending with the target certificate that needs to be validated. Certification path validation procedures are based on the algorithm supplied in ITU-T Recommendation X.509 and further defined in Internet Engineering Task Force (IETF) Request for Comments (RFC) 3280. Certification path processing verifies the binding between the subject distinguished name and/or subject alternative name and the subject public key defined in the target certificate. The binding is limited by constraints, which are specified in the certificates that comprise the path, and inputs that are specified by the relying party.

To ensure secure interoperation of PKI-enabled applications, the path validation must be done in accordance with the X.509 and RFC 3280 specifications. This process turns out to be more time consuming in case the certificate chain is long. It is time consuming in the sense that in large scale PKI the number of users is voluminous and every user will verify certificates. As depicted in Figure 3.4, to establish a certification path, every user enquires certificates from the directory or the other communicating party can provide them as part of the initial communication. The path verification is done in line with Certificate revocation verification. It is advisable that, systems should provide graceful certificate revocations, as this will enable smooth overall performance of the PKI infrastructure. The issue of certificate revocation is discussed in depth in section.

Certificate revocation mechanisms

The distribution of revocation has the potential to be the most costly aspect of running a large-scale public key infrastructure (PKI). In order to realize the cost effectiveness,

several alternative revocation distribution mechanisms have been proposed. Some of the proposed certificate revocation mechanisms includes the On line certificate status protocol, Delta certificate revocation list, complete or base CRL, sliding window delta CRL, authority revocation list, Freshest CRL, Redirect CRL, Dynamic CRL distribution point, Indirect CRL, etc.

Large scale PKI will involve voluminous users accessing the repository at the same time; in these scenario issues of trust, request rate, reliability and timeliness are of great importance. For this reason implications of any certificate revocation mechanism to end-user applications are drawing so much discussion. In order for a certificate to be valid, the certificate user must trust the issuing party utilizing any of the revocation status verification mechanisms. In order to maintain this environment of trust it is vital that the revocation process is well defined, implemented, and enforced without any ambiguity existing as to the status of a certificate.

A certificate revocation list (CRL) is a list containing the serial numbers of all certificate issued by a given certification authority that have been revoked and have not yet expired. Validity period is one of the attributes of the certificate; the certificate is expected to be in use in the entire validity period if there is no circumstance causing the certificate to become invalid. The certificate may become invalid due to various factors; some of these factors include change of name, change of association between the subject and the CA, compromise of the private key or suspected compromise of the private key, certificate no longer required by the subject, improper or faulty issue of the certificate and change of status of the subject. Under such circumstances, the CA needs to revoke the certificate. Once the relying party has obtained the CRL, the CRL may be cached for future validation but after certain point a newer CRL must be obtained in order to ensure that validation is based on up-to-date certificate status information.

Certificate Revocation List (CRL)

Among the standards that exist today and those being developed, the Certificate Revocation List is generally the preferred model. There are many variations to this model but they are all based around the same basic structure. The Certificate Revocation List is a periodically published data structure that contains a list of revoked certificate serial numbers. The CRL is time-stamped and digitally signed by typically the issuer of the certificates. However, other trusted third-party entities such as those providing revocation services may also publish and sign the CRL. Generally a CRL is published within an X.500 directory that also stores the certificates for the particular CA domain.

The publishing period should be determined by relying party business needs and therefore the associated Certificate Policy. Protocols used to extract revocation information need not employ signed transactions as the digital signature on the CRL maintains the integrity of the CRL itself. CRLs are currently defined in the X.509 standard with two CRL versions being defined. As with many standards the detail of implementation is open to interpretation.

One revocation method is defined in the X.509 v2 standard. This method involves each CA issuing a signed data structure called certificate revocation list (CRL). Any system using the certificate checks the validity of the certificate by verifying not only the signature but also checks the recent CRL to find if the certificates serial number is not on the CRL. A CA issues a new CRL on a regular periodic basis. An entry is added to the CRL as part of the next update following notification of revocation and is removed from

the list after the end of revocation validity period. The CRL may be distributed by exactly the same method as certificates themselves i.e. using unsecured communication and server.
The advantages of the X.509 V2 CRL is that the CRL may be used in a wide range of application and environment covering a broad spectrum of interoperability goals and even broader spectrum of operational and assurance requirements. Also common locations within CRL for frequently used attributes as well as common presentations are defined.

The limitation of CRL distribution method, using unsecured communication and server, is that the time granularity of revocation is limited to the CRL issue period. For example, if revocation is reported now, that revocation will not be reliably notified to certificate using systems until the next periodic CRL is issued. This depends on the frequency that the CA issued CRLs.

Complete or Base CRL

This is the implementation of a CRL as described above that is limited to containing all revocation information of a single CA domain. Successful use of this model would only be affected provided the number of end-entities was relatively small.

Disadvantages with the use of complete CRLs include: Scalability issues due to the volume of posted data, that can escalate significantly given that revocation information in the CRL must remain available until expiry of the certificate. If certificate validity periods are reduced this helps to alleviate this problem. As with all CRL variants timeliness can be an issue so it is important to align the posting periodicity with business requirements and certificate policies. As the volumes of posted data increase this may enforce a minimum CRL refresh period due to unacceptable performance degradation beyond a certain threshold. The single data structure of a base CRL limits the ability to distribute the network load especially as the data size expands.

Over-Issued CRL

Over-issuing is a method to spread out request for revocation information. With over issuing a CA issues a CRL more often than necessary. For example a CA may issue CRL for every 6 hours even if the CRL is valid for 24 hours. The result is that CRL in the relaying party caches will expire at different time so requests to the repository for new CRL will be spread out.
Over issuing certificates has the advantage of significantly reducing the request rate in the repository. In the traditional method of issuing CRLs the requests of for certificate revocation is not evenly distributed across time; when a new CRL is issued, the request rate is initially the same as the validation rate. The request rate then drops exponentially as an increased number of the relying parties perform validation-using CRLs in their caches that were obtained to perform previous validations.

Delta CRL

A delta CRL is a CRL that only provides the information about certificates whose status have been changed since the issuance of a specific previously issued CRL. A client in need of specific up to date information, that had already obtained a copy of previously

issued CRL, can download the latest delta CRL instead of downloading latest full CRL. Since delta-CRL tend to be significantly smaller than the full CRL, this will tend to reduce significantly the load on the repository and improve the response time for the client.

The advantage of system in which delta-CRL are available is that most of the request for full CRL will be replaced by delta-CRL. Which generally are serviced more quickly. By replacing most request of the full CRL with request of delta-CRLs, the average request rate for full CRLs can be substantially reduced.

The use of delta-CRL will not reduce the request rate of revocation information from the repository. In order to realize full benefit of delta-CRL the peak request rate for full CRL must be substantially reduced. This disadvantage cannot be eliminated if traditional certificate distribution method is used to distribute the delta-CRL. Presenting a new way of distributing delta-CRL, the sliding window delta-CRL, has solved this problem. A brief explanation of sliding window delta-CRL is discussed in the next paragraph.

Sliding window delta-CRL

Each delta-CRL provides information about any certificate whose status has changed between times the base CRL referenced by baseCRL number was issued and the time the delta-CRL was issued. In other words, the delta-CRL provides information about all status changes that occurred during a certain window of time. For example a window size for delta-CRL can vary between 20minutes and 6hours. The problem of issuing delta-CRL using tradition method is that the window sizes of the delta-CRL vary. The request rate for base-CRL drops as the window size increases and jumps up when the window size is reduced. So, the larger the window sizes of the delta-CRL, the lower the request rate will be for base CRL. In the sliding window model, the idea is to have for each delta certificate the same large size of the window instead of using variable window size as with the traditional method.

Partitioned CRLs

The CRL Distribution Points (also known as CRL Distribution Points) scheme allows a single CA domain to post revocation information on multiple CRLs. Certificates have knowledge of the CRL distribution point by utilizing the *CRL* distribution point extension as specified in X.509 Version 3. This therefore ensures that the relying party does not need to have prior knowledge of where the revocation information for a particular certificate might be located. Another and more significant advantage is that revocation information is spread across a number of more manageable partitions to enhance scalability and improve performance by distributing both the maximum and average loads. The drawback with this is that each end entity is likely to require access to multiple partitions therefore increasing the average request rate. Scalability may also be restricted in that the CRL partitions are fixed or static, which leads to the next model.

Redirect CRL

A Redirect or Referral CRL is based on existing standards and protocols but expands the concept of standard CRL distribution points by allowing for a more flexible partitioning approach. This structure is basically an empty CRL in that it contains no certificate revocation entries, but importantly it does contain specific Redirect Pointers within CRL extensions that identify the location of CRL partitions. If re-partitioning is required then

this provides a flexible means of re-defining pointers with the important feature of being an issuer signed structure.

Dynamic CRL Distribution Points

The concept of Dynamic CRL Distribution Points (previously referred to as Enhanced CRL Distribution Points) was developed to overcome the static partitioning of CRL Distribution Points. Once the associated certificate is issued, the CRL partition pointed to by the CRL Distribution Point extension is fixed for the life of that certificate. In addition to this, the issuing CA must have prior knowledge of the partitioning structure and that this structure cannot change over time. It is of course desirable, especially with evolving PKIs that implementations are able to evolve with the needs of the PKI community.

The answer to this is to implement a flexible and dynamic partitioning capability, which is defined through scooping statements with associated pointers. A scope statement specifies a range of certificates that are covered by a particular CRL partition while the pointer defines the associated CRL Distribution Point. These requirements have been met by proposing a CRL Scope field and a CRL Status Field to be incorporated in amendments to X.509 certificate extensions. The original proposed concept known as Open CRL Distribution Points employed this strategy but with unsigned pointers open to denial-or-service attacks. The concept of Dynamic CRL Distribution Points addresses this issue by defining pointers in Redirect CRLs as previously discussed.

Indirect CRL

A PKI domain that may utilize multiple CAs or a trusted third party service provider is able to publish revocation information in a single CRL structure using this mechanism. The relying party is therefore able to avoid the retrieval of revocation information from multiple CRLs being issued by multiple CAs. This Indirect CRL can therefore be considered as an aggregate of a number of Base CRLs.
The implementation of an Indirect CRL follows the same construct as that of a Base CRL, except for an identifying attribute in the Issuing Distribution Point extension. When this extension is set for an Indirect CRL there is also a requirement for a CRL entry to specify the issuing CA of the certificate using the Certificate Issuer extension. Both of these CRL extensions are specified in the X.509 Version 2 CRL standard.

Online Certificate Status Protocol (OCSP)

Currently the preferred online revocation mechanism amongst standards and implementations is the Online Certificate Status Protocol. This is specified in the proposed standard X.509 Internet Public Key Infrastructure – Online Certificate Status Protocol [RFC2560]. The basic process consists of a request/response protocol that obtains online revocation information from a trusted entity referred to as an OCSP Responder, with all responses being digitally signed. Although these responses are available in real time, the validity interval of the provided information is only as timely as the back-end mechanisms generating the data. It is important to note that OCSP purely provides revocation status information and does not verify whether a certificate is within its validity period.

Freshest CRL

There are often varying requirements by relying parties on the timeliness or freshness of available certificate revocation information. The Freshest CRL is a method of meeting these ranging needs in a cost and infrastructure efficient manner.
For those relying party applications that require more timely information, then a CRL (typically the latest Delta CRL) with short latency postings is made available via a Freshest CRL extension. Other relying parties with less stringent latency requirements may opt to utilize a Base CRL or other alternative. This service differentiation could be based on a user pays approach ensuring service level segmentation is applied on a business needs basis thereby distributing network load.

Authority Revocation List (ARL)

An ARL is a CRL that is used solely to publish revocation information for CAs cross certificate. It therefore does not contain any revocation information pertaining to relaying parties' (end users) certificates. As an ARL is used to revoke the certificates of CAs it is issued by either a superior CA; one which has the responsibility of revoking subordinate CAs or the issuing CA is revoking a cross-certificate issued by that CA. The ARL is identified using the issuing distribution point extension as implemented in X.509 for Version 2 CRL extensions. When validating a certificate path, a valid ARL must be available for each CA that has signed certificates in the path, the exception being for the self-signed root CA within a particular domain. The revocation of CA certificates is likely to be rare therefore It can be expected that an ARL will remain empty or small, This advantage will ensure that performance is significantly improved over Complete CRLs.

Segmented CRL

One of the existing options to improve the performance over the traditional method of distributing certificates is segment the CRL. Segmenting the CRL may not reduce the peak request rate for CRLs; it will reduce the size of each CRL. This may allow the repository to service the CRL requests faster. This suggests that the use of segmented CRL would allow the use of less powerful repository to handle requests for segmented CRLs.
The drawbacks of using the segmented CRL are that while the peak request rate is not affected by segmentation, the average request rate increases with the number of segments used since the request rate drops more slowly. Segmented CRL may not also be preferred by relying party because each time the relying party needs a CRL segment that is not already in its cache to perform validation, it must send a request to the repository and wait for the response before it complete the validation. As the number of CRL segments increases so does the number of times the relying party will have to wait for CRL information from the repository.

Trust

Trust is the most popular measure of how much confidence can be assigned to an IT security system. Today there exist several trust models that can be deployed. But most of them are appropriate for a particular business environment for example a single CA model is suitable for deployment in an organization that is small and central certificates

management is convenient. If this organization would, for some reasons, communicate with another organization, then, it will have to use cross certificates or bridge certificates.

When the constellation of organizations grows significantly we have what we call in this research as LSPKI where issues of interoperability and scalability becomes difficulty. For instance, when users authenticate themselves to an organisation billing application, the application's cryptographic software will verify the user certificate's signature using the public key of the CA that created the certificate. If the CA's key is not listed as a root key, then the certificate containing it will also have to be validated with the public key of the CA that signed that certificate; and so on until a certificate in the 'trust-chain' can be verified with a trusted root key. The validated chain then implies authenticity of all the certificates, including the end-user's one. In this section the pro and cons of various trust models is presented.

Web of Trust Model

This is one of the easiest trust models for a small group of users. This is the model which is employed by Pretty Good Privacy (PGP). In this type of system each user creates and signs certificates for the people he knows. Therefore, no central infrastructure is need.

This model works very well for small groups, who have pre-existing relationships, but it doesn't scale well for large groups or where consistency of assurance (e.g. level of authentication required before a certificate is issued) is important. Communication of certificate status to Relying Parties is also very difficult with this model.

Single CA Model

A simpler model for larger groups is to assign one person or organisation the role of Certificate Authority. In the Single CA model each person user is provided with the CA's public key in a secure out-of-band way and there is a repository to check whether a particular certificate has been revoked or not. The Single CA model is often extended by having Registration Agents CAs (RCAs) who is remote from the CA but local to specific user groups, for instance located at each departmental branch office.

The RA is responsible for verifying the Subscriber's identity and, if necessary, authorisations, before approving the application for a certificate. They are also often tasked with setting up a preliminary trust relationship between the Subscriber and the CA either through a shared secret or by exchanging public keys. In larger or more diverse groups there may be a requirement to have more than one CA. In this scenario, there need to be a capability for subscribers from each CA to be able to inter-operate with those from the other CAs. This problem is solved by the introduction of other models which are described below.

Hierarchical Model

The traditional more than one CA (multiple-CA) implementations are formed around a very hierarchical structure with a Root CA at the top, one or two layers of CAs below that (i.e. with their public keys in certificates signed by the Root CA) and then Subscribers and RAs under (signed by) them. Each user's most trusted key (the 'root' of any certificate chain they verify as a Relying Party) is the Root CA's public key. This model allows enforcement of policies and standards throughout the infrastructure, producing a

higher level of overall assurance than the other multiple- CA models. However, the hierarchical nature may not fit so well with the peer-to-peer business relationships between departments or different nations.

Browser Trust-list Model

Browser Trust-list model - also referred to as the CA list or User-centric model where each user application has a list of the public keys for all the CAs that user trusts is the most common PKI implementation. This model is implemented in Netscape and Microsoft web browsers. It gives users a great deal of flexibility to add and remove CAs from the trust list.

The primary concern with this model, however, is that there is no differentiation between a strong PKI and a weak one; for instance most users of VeriSign certificates will not normally look to see which policy a specific certificate is issued under before they rely on it.

Policy Trust list model

Policy Trust List Model would let users restrict access based on the policy under which the certificate was issued. With the development of a new standard like Certificate Trust Lists, users are given great precision over which certificates to trust, and these decisions can be made on a per application basis.

Cross-certificate Model

A model that is in effect a compromise between the preceding two models is the Cross-certificate model as used in Entrust's PKI architecture. In this model each CA creates certificates for the CAs that it has verified as of equivalent strength to its own. As in the hierarchical model, each user only has one root public key, but in this model that key is their local CA's key not the key of a central Root CA. The chain of certificates is navigated in the same way, but the view of the structure differs depending on the user's CA.

One problem with this model is the difficulty for the user application to determine a certificate chain between users whose CAs do not have a direct cross-certificate link. This model does confront the issue of who is the root CA? allowing CAs to be in peer structures rather than hierarchies, but like the Web Of Trust model, cross certification struggles to produce uniform or deterministic levels of assurance across the whole system.

Bridge CA Model

Because of the various issues raised with the preceding models, cross-organisational implementations sometimes employ a Bridge CA. For instance, the national Bridge CA will provide a trust bridge through cross-certificate pairs between the various Hierarchical and Cross-certificate PKIs being developed by government institutions and private companies. This implementation - crossing multiple vendors and models - may end up being highly complex LSPKI and might require modification of the end-user PKI

modules if they are to be capable of finding a trust chain between any two users within the structure. Interoperability between Government, commercial and international domains appears reasonable to exist for many reasons, but is likely to be very complex to work into the end-user cryptographic software.

Privacy

Privacy as a social and legal issue has for a long time been a concern of social scientists, philosophers, and lawyers. With the arrival of the computer and increasing capabilities of modern IT-systems and communication networks, individual privacy is increasingly endangered [Hubner 2001]. Especially on the way to a Global Information Society with different national programmes for the further development of data highways, there are severe privacy risks. Privacy as a fundamental human right recognised in the UN Declaration of Human Rights, the International Convenant on Civil and Political Rights and in many other international and regional treaties has to be protected in a democratic society.

Simone Hubner strongly urges about the privacy intrusive nature of IT in general [Hubner 2001]. However, they're others who point at PKI in particular to be a serious threat to individual's privacy [Greenleaf]. The authors (Graham Greenleaf and Roger Clarke) of the paper, Privacy Implications of Digital Signatures, give the following statement:
"There is an increasing likelihood that public key cryptography will fail to become generally applied for the purposes of authentication" [Greenleaf]. My view, regarding this, is that this may turn out to be true if there will be no efforts in the near future, to resolve how privacy issues are implicated in the following areas of PKI:

Private key generation: A first concern relates to the manner in which private keys are generated. From a security viewpoint, it is essential that key-generation is undertaken entirely under the control of the individual concerned, and that the private key never leaves the possession of that person without strong security precautions being taken. If any other approach is taken (such as generation by a service organisation, or by a government authority), serious security and privacy issues arise, because the scope exists for the individual to be convincingly impersonated. Recommendations do exists that if any user-generated pairs would have to comply with guidelines set up by the relevant CA, otherwise, the key pair might not be secure.

Private key revocation: When grounds exist for believing that a private key may have been compromised, the key pair must be withdrawn, or revoked. This involves identification of the party who is requesting the revocation. This identification is necessarily intrusive, because the risk exists of an impersonator requesting revocation, and certification of a replacement key. This would only need to be achieved during a few key minutes in order for a fraud to be perpetrated, e.g. in relation to the purchase or sale of shares, or the transfer of funds from a bank account. The consequences for individuals of wrongful key revocation are sufficiently important that there should be legal right to compensation if a key is wrongfully revoked.

Registers of public keys: If any central public registers of all public keys is maintained, then in order to sufficiently describe the person who holds each digital signature, personal information such as addresses, date-of-birth, position in the organization and country may be included, leading to problems of secondary uses for other purposes. This register, as it contains identification information about every holder of a digital signature is subject to be misused by governments.

Certificate Revocation Lists (CRLs): The great dangers of a central register arise from its potentials for political abuse and for surveillance. If it becomes routine for signature recipients to check the repository for non-revocation of digital signatures, then it is possible that logs can be kept and eventually they will become a centralised surveillance facility, capable of indicating which cyberspace entities a person is transacting with over a period of time. Police and other legal enforcement agencies are likely to show a keen interest, as they already do with telephone call data held by carriers.

Expectations of identification: There are strong pressures towards increasing expectations that members of the public should identify themselves when they conduct transactions. These pressures include: Digital signature technology adds a new dimension to the technological arsenal, because it provides apparently high-reliability identification of the individual who conducts a transaction. Applications like Secure Electronic Transaction, which PKI enables, do not provide anonymous transactions although it provides a mechanism to hide bank information to the merchant and it hides purchase details to the bank. In my view SET is still not good enough as far as individual privacy is concerned.

Performance and availability

One area that has already been identified as having a significant impact on overall PKI performance is that of certificate validation. Since the security of any PKI transaction is based on the current status of the participants' certificates, there are currently two approaches under consideration namely Certificate Revocation Lists (CRLs) and Online certificate Status Protocol (OCSP). It is generally recognized that a CRL approach will not scale to user populations in the millions since these lists are anticipated to grow into size that will quickly be unmanageable. This makes CRLs an unacceptable choice for downloading to each relying party/end user in the system.

Consequently, program planners are looking at the implications of deploying an OCSP-based validation system. OCSP solves the payload size problem present in the CRL approach since it only sends the status information of the certificate in question (not the entire list), is inter-operable with CRL issuing Certification Authorities (CAs), and provides centralized certificate status management. However this approach dramatically increases the network traffic for data that is critical to each transaction. The result is a degradation of overall system performance and validation responder availability.

Tracking down referral chains between certificates.

Certificate verification is a process that is performed by the user to prove the authenticity of a particular certificate and not the identity of the signer. This process is always performed before verifying the data itself. Authenticating the signer information is

critical to prevent anyone generating key-pairs with some one else's identity. The CA who carries out reasonable identity checks before issuing a certificate to a person or company.

The CA will have its own key pair and its own certificate, which is self signed, also known as a root certificate. The CA's private key is used to sign all issued certificates. Sometimes a CA will be a registration CA, Policy CA or Local CA thus a subsidiary. In such a case the signatures can be traced up along certificate chain to a root certificate authority. The root certificate is self-signed. Unless the entire certificate chain is validated, a given certificate and a given signature cannot be guaranteed to be valid.

To validate a certificate, one needs to validate the digital signature on it. This requires parsing the X.509 certificate and identifying the content and the digital signature.
Once the data content and the digital signature are located, the data must be hashed to obtain a digest. Next, the certificate authority's root CA certificate must be parsed to extract the public key. This certificate is typically placed in a trusted root certificates repository. The public key is then used to decrypt the digital signature on the certificate and reveal the hash. This hash must match with the computed hash on the data.

If the issuing CA's certificate is not marked as 'trusted' in the certificate repository, one must go further up the chain by identifying the next higher-level CA. This can be done by parsing and inspecting the issuing CA's root certificate. The digital signature on the issuing CA's root certificate now must be verified using the root CA certificate of the higher level CA. And if this CA happens to be not trusted, go another level up. When one finds a root CA certificate that is marked as trusted, the chain verification process stops successfully.

Chain verification is a resource intensive process in LSPKI. This process will definitely course network bottleneck and requires scalable processing power of the repository server. Slower network and network server is not acceptable in some of the components of LSPKI such as banking system.

Legal environment and Security Policy

New legislation is also impacting policies. Legislatures and regulators make decisions that have far reaching implications on PKI and PKI enabled applications. For example digital signature initiative impacts the type of services that may be offered, who may offer them and the procedures that must be followed. Digital signature initiative has been passed in many nations.

Just as traditional handwritten (holographic) signatures link people to the content of their agreements in a legally recognized manner, digital signatures can provide similar (but not identical) functions for electronic commerce and other purposes. Perhaps most importantly, digital signatures contribute to non-repudiation - a security service that is increasingly appreciated within the legal and business communities to provide important benefits.

The advances we have pointed out in this part of our report are revitalized in Europe and United States of America. Albeit many nations in the third world use Commonwealth laws, variations of regional and national legal environments and policies become a real negative factor for PKI interoperability.

The European Directive 1999/93/EC: The European directive 1999/93/EC, established a new legal framework guaranteeing European Union recognition of electronic signature. It specifically states that electronic signature cannot be legally discredited against solely on the ground that it is in electronic form. It further states that any resulting electronic signatures are as legally valid as the hand written signatures and they can be used as evidence in legal proceedings.

In 1989, electronic funds transfer laws, such as Article 4A of the Uniform Commercial Code and later the United Nations Commission on International Trade Laws (UNCITRAL's) Model Law on International Credit Transfers, adopted authentication procedures rather than traditional signatures as the basis for verifying transactions and apportioning liability.

In 1990, the U.S. Department of Justice issued its Guidelines on the Admissibility of Electronically Filed Federal Records as Evidence, which emphasized the reliability and trustworthiness of computer-based data for evidentiary purposes.

In 1991, the comptroller general of the United States issued a decision entitled "Use Of Electronic Data Interchange Technology to Create Valid Obligations" that authorized EDI for government contractual obligations "using properly secured EDI systems" and considered the permissible uses of digital signatures.
In 1994, the first comprehensive legal study of digital signature infrastructure was published, Federal Certification Authority Liability and Policy, under the auspices of the U.S. government.

In1995 commenced the adoption or consideration of digital signature legislation in various U.S. states. The first Digital Signature Act became law in Utah in May 1995, followed shortly thereafter by California, and other states are contemplating various forms of digital signature legislation.

The Government paperwork elimination act (GPEA): This requires government agencies to offer services electronically and to maintain records electronically by October 21, 2003. The Health Insurance Portability and Accountability (HIPAA) were passed in 1996. Its purpose was designed to improve efficiency through us of electronic data exchange mechanisms for health information. The E-Sign, the electronic Signature and National Commerce act, was signed into law in June 30, 2000. This removed a major legal barrier to the use of digital signature in electronic commerce.

Appendix D: PKI Interoperability based on various approaches

This section gives an overview of how different security products make use of various types of crypto systems and public key certificates to allow interoperability. Also we discuss the drawbacks these systems in regard to scalability, wider choice of cryptographic algorithms, public key management and certificate standard. In all the systems certificates are central for authentication, identification, confidentiality, nonrepudiation, access control and data integrity.

IPSec

Internet Protocol Security (IPSec) provides security between the initiator and the responder at the IP layer by enabling systems to negotiate the required security protocol, put in place any cryptographic keys required to provide the security service and to determine the algorithm to use for the service. Two IPSec systems cannot interoperate unless they are using the same algorithms. The security is provided for all the traffic at the IP level using the Authentication Header (AH) and the Encapsulating Security Payload (ESP). There are a number of algorithms that can be used in IPSec; these include the following: Triple DES, CAST, DES, RC5, IDEA, AES, Triple IDEA, Blowfish and Diffie Hellman [Stallings 2000].

The strength of IPSec is based on the fact that all traffic crossing the perimeter is secured. This makes IPSec ideal to be implemented in routers, firewalls and VPN. IPSec is transparent to end-users therefore when user's employment is terminated there is no need to revoke users certificates and no user training is required. IPSec implementation does not affect the application layer therefore no changes have to be performed on the application to support IPSec. Despite all these advantages the fact that IPSec encrypts all the traffic may cause some bottlenecks in the systems performance if the encryption algorithm is resource intensive. It is desirable sometimes that the user decides what to encrypt and what not depending on the type of information the user is communicating.
IPSec services are:

- Data origin authentication
- Access control
- Confidentiality
- Connectionless Integrity

AH provides access control, integrity and authentication while the ESP provides confidentiality in addition to what is provided by AH.
IPSec supports two types of key management namely manual and automatic. In the manual key management the system administrator configures the system with its own keys and the keys of the responder. This is not suitable if scalability is critical. The automatic key management allows keys to be automatically created on demand. This is suitable for distributed large-scale systems. Oakley and ISAKMP are protocols that enable an automated Key use and creation [RFC 2408] [RFC 2412].
Oakley protocol can be used for authentication in three different ways:

- Digital signature – Mutually obtainable hash is signed and each party signs the hash with its private key

- Public key encryption – ID and nonces are encrypted using private key to authenticate the exchange
- Symmetric key encryption – exchange authentication can done by encrypting exchange parameters using a key that can be generated using out of band mechanism

ISAKMP combines the security concepts of authentication, key management, and security associations to establish the required security.
ISAKMP has basic requirements for its authentication and key exchange components. These requirements guard against denial of service, replay, man-in-the-middle, and connection hijacking attacks. This is important because these are the types of attacks that are targeted against protocols. Complete Security (Association SA) support, which provides mechanism and algorithm independence, and protection from protocol threats are the strengths of ISAKMP.

PGP

Pretty good privacy (PGP) provides confidentiality and authentication security services for storage and electronic mail applications. PGP is base on SHA-1 for hash coding; RSA, DSS, and Diffie Hellman for Public key encryption; IDEA, CAST-128, and TDEA for conventional encryption. Confidentiality and authentication can be used for the same message. First a signature is generated for the message and appended to the message. Then the message and the appended signature are encrypted by using CAST-128, IDEA or TDEA, and the session key is encrypted using RSA. When both services are used, the initiator first signs the message with its own private key, then encrypts the message with a session key, and then encrypts the session key with the recipient's public key. PGP makes use of four types of keys namely: public key, private key, one-time session key and passphrase-based conventional keys.

Each PGP entity must have the following capability: maintain one file for its own public and private keys pair and public keys of responders, users may wish to have more than one pair of keys and a capability of generating unpredictable session keys. CAST-128 is used to generate 128-bit session key. Key identifiers are used to identify public key that have been used to encrypt the message. PGP does not have a rigid formal public-key management scheme and any specification for establishing CA. However, each public key entry in the store is associated with a field called trust flag byte that indicates the extend to which PGP will trust the certificate. The content of the trust flag are key legitimacy filed, signature trust field and owner trust field.

The key legitimacy field indicates the extent to which the PGP will trust that the public key for this user is valid. The signature trust field indicates the user trust to the signer to certify public keys and the owner trust field indicates the degree to which the public key is trusted to sign other public key. In case a user, for some reason, wants to revoke a certificate he will have to send to all users as fast as he can another certificate which includes an indication that the purpose of this certificate is to revoke the use of a particular public key. IETF established the Open-PGP work group in order develops interoperable applications Internet standard based on the Open-PGP format [44]. It does

not deal with storage and implementation questions. Open-PGP software uses a combination of strong public-key and symmetric cryptography to provide security services for electronic communications and data storage. These services include confidentiality, key management, authentication, and digital signatures.

PGP Limitations

The PGP's currently applied trust model, the web of trust, where the PGP users certify the keys of other PGP users, has some serious inherent problems. One is that some user may take signing of another's key too lightly, i.e. sign without having proved the identity of the certificate holder. In other trust models, say the hierarchy model, the CA is responsible to verify the identity of end entity. Other drawbacks with PGP are in terms of manageability (e.g. revocation management) and of verifying certificates, caused by the missing possibility to delete information in a once published public key in combination with the high probability that some keys in the web of trust loose their trustworthiness. A further drawback of the current PGP technology is that the current PGP public keys server is not distributed. If the increase of numbers of PGP users is significant, it becomes obsolete because it is not scalable.

Kerberos

Kerberos, a terminology from Greek mythology that means a many-headed dog, is a network authentication protocol developed at MIT. It is based on the Needham and Schroder authentication protocol [Kerberos]. It is designed to provide strong authentication for client/server applications by using secret-key cryptography. The Kerberos protocol uses strong cryptography so that a client can prove its identity to a server (and vice versa) across an insecure network connection. After a client and server have used Kerberos to prove their identity, they can also encrypt all of their communications to assure privacy and data integrity as they messages.

IETF defines Kerberos version 5 (RFC 1510) as Internet standard and has a work group that works out to address Kerberos interoperability issues. Kerberos version 4 is still in use and its deficiencies have been addressed in Kerberos version 5. Version 4 requires the use of DES PCBC mode, IP protocol, Ticket lifetime is about 21 hrs; Credential forwarding is not allowed; inter-realm relationship is in the order of N^2 where N is the number of realms. In Kerberos version 5 these shortcomings have been corrected where DES CBC mode is used, the number of realms where the ticket will be used is pre defined/limited, any network address can be used and in version 5 ticket credentials can be forwarded on behalf of the user. Kerberos provides single sign-on, key exchange, and a way to delegate user's credentials to back-end servers.

The Kerberos protocol shares a secret key with every entity in the network. The Kerberos client acts on behalf of the user, which may be a person or a process. The client operation depends on servers called Ticket-Granting Service (TGS) and Authentication server (AS). In brief description Kerberos works as following:

- First the user logs on to workstation and request service on AS. Every time the user logs on a ticket granting ticket (TGT) is requested from the AS

- Second after verifying user's access rights, creates a TGT and a session key. The result of the TGT and the SK is encrypted using a key derived from the user's password. Then eventually the AS sends the ticket and session key to the host
- Third the workstation prompts user for the password and uses the password to decrypt the message from the AS, then encrypts and sends the TGT and authenticator that contains the user name, network address time to the TGS.
- Fourth the TGS decrypts the message and upon verification of request it issues to the workstation a ticket for the requested server once per type of service
- Fifth the workstation sends the ticket to the server which provides the service for example printing services
- Sixth the server that provides the service verifies the ticket and authenticator then grants access to the service to the workstation.

The collection of Kerberos AS, TGS, Client and servers that provide services is called a realm or kerbery. Users from one realm can only be able to use services from another realm if and only if there is an inter-realm relationship. Inter-realm communication makes Kerberos one of the choices for distributed or large-scale security solution. Some important fields of ticket are: Name of the principal, the session key, Client's name, Time of the initial authentication, Time after which the ticket is valid, Time after which the ticket will not be honored, renew-till time, and Client's address.

Kerberos Limitations

Kerberos has a number of limitations. Although Kerberos supports time stamping, symmetric key encryption, mult kerbery, delegation of authentication credential to back end servers (single sign-on) and the potential of being a candidate for WWW security there are two problems in my view that needs to be addressed. First the dependence of users on a single server (AS) where they have to be registered in order to be able to use network resources is a serious problem. If the authentication server fails the whole kerbery fails. Secondly key management (ticket revocation and storage) in a large-scale implementation will be a problem if the number of users grows to a substantial big number.

Secure Socket Layer (SSL) and Transport Layer Security (TLS)

Netscape developed secure Socket Layer and latter it was standardized by IETF as TLS [RFC 2246], [Stallings 2000]. TLS is viewed as SSL Version 3.1. SSL makes use of TCP to provide secure end-to-end communication. It provides security to higher layer protocols like HTTP. Most importantly SSL provides peer authentication and data integrity and confidentiality. Data confidentiality service is provided by encrypting SSL payload using a shared secret key that is defined handshake protocol. Data Integrity security service is provided through a message authentication code (MAC) which is calculated using a secret key which is also defined by the handshake protocol.

The handshake protocol allows the peers to authenticate each other and negotiate X.509 Version3 cryptographic keys, MAC and cryptographic algorithms to be used during the session. Algorithms that are currently being used in SSL are: IDEA, RC2, DES, 3DES, SHA-1, Fortezza MD5, RSA, and Diffie Hellman.
Regard to certificates processing SSL support certificate chain verification and signing. Prior to data transfer the peers will exchange a secret key that will be encrypted with the

receiver's public key. The certificate request message contains a list of the distinguished names of acceptable certificate authorities. These parameters are paramount for the certificate verification process. If the peers are a client and a server the session could be established in the following order:

- Establish security capabilities: In this phase a logical connection is established, compression method is determined, cryptographic algorithms which are supported are listed and a session ID is defined
- Server authentication and key exchange: The server will send its X.509 certificate
- Client authentication and key exchange: The client will verify the certificate or chain of certificates from the server and if the verification is successful the client may send its certificate.
- Secure connection established: Peers will exchange messages that will indicate that the certificate verification is successful.

SSL provides a good connection into the server, and the server has been authenticated. Thus plaintext passwords and similar methods vulnerable to passive attacks can be used safely to authenticate the user for the server. However, SSL does not support single sign-on, and delegation or secure transfer of session credentials over connections. By single sign-on we mean user's ability to use several applications with a single authentication. There have been activities in the IETF TLS Working Group to provide help for delegation problems: cipher suites allowing use of Kerberos with TLS have been designed, which, when implemented, would solve the single sign on and delegation problem.

TLS provides authentication and encryption for communication stream. The communication stream from client to the server and visa versa are both protected. TLS is often used to protect web content, but it can be used with any stream oriented application protocol because both SSL and TLS are application protocol independent. TLS provides stream –oriented security with three properties:

Authentication. The handshaking protocol uses certificates and digital signature verification to confirm the identity of the remote application

Integrity. The application protocol data is protected from undetected modifications. The record protocol employs an integrity check value, computed using HMAC to conform that the data is unaltered.

Confidentiality. After the handshake protocol establishes the symmetric encryption key, the record protocol encrypts the remainder of the session.

Certificates are integral part of both key management and authentication services offered by TLS. The services depend on the binding of identity to the public key. Domain names are especially suited to identification of web servers. In client server authentication the server provides a certificate resulting to authentication of server to client. When the server sends to the client a handshake certificate request message the client must have a certificate and present it to the server.

Certificates include the key usage extension. This extension indicates the appropriate usage of the public key contained in the certificate. TLS includes several usage extension rules, these are first the digital signature key usage must be set to allow signature verification, the key encipherment key usage must be set to allow RAS encryption and the key agreement key usage must be set to allow Diffie-Hellma operation.

The server must be sure that the public key belongs to the client; also the client must be sure that the public key of the server really belongs to the server. The certificate provides the needed binding of the public key to the client's identity.

Secure Electronic Transaction (SET)

SET was designed to protect transactions on the Internet for credit card users. It is a product of combined efforts from IBM, Netscape, RSA, Terisa, Verisign, VISA, Master Card, and Microsoft. SET is incorporated in The Internet Open Trading Protocol (IOTP) which is defined in (RFC 2801) to provides an interoperable framework for Internet commerce
SET is designed to provide the following security services and functionality [Stallings 2000]:

- Provides information confidentiality of orders and payment
- Ensures integrity of all transmitted data
- Provides authentication of the card holder to the credit card account and a merchant relationship with a bank
- It is independent of platforms
- SET uses x.509 V3 certificates
- Supported algorithms are: RSA, DES, SHA-1,

The major components of SET are: CA, Merchant, Issuer, Acquirer, Cardholder and the Payment gateway. Upon customer's identity verification, the cardholder receives an X.509 V3 certificate that is signed by the bank. The merchant possesses two public keys one for signing messages and one for key exchange. The cardholder (customer) will use the merchant's public key certificate to verify the authenticity of the merchant. Upon order confirmation the merchant verifies the customer using the customer's certificate. The Customer's bank information such as the credit card number is encrypted such that the merchant cannot open it and the bank should not be able to know the customers order details. This method provides privacy to the customer at the same time linking the two in a way so that in case of a dispute the two-merged document should still be reliable. The method of signing separately the band information and order information then merging the two is termed as dual signature. The payment gateway will ensure the merchant that the credit is sufficient for the purchase and also the payment gate will process the payment to the merchant after the order is delivered. The strong part of SET the dual signature and introduction of the CA who is a third part trusted by the merchant, cardholder, and payment gateway. However the strength of DES algorithm has been questionable in recent years due to increased processing power of PCs. Replacing DES with 3DES or AES will suffice.

Secure Multipurpose Internet Mail Extension (SMIME)

SMIME is IETF standard based on RSA and is defined in a number of RFCs of which RFC 2632 defined SMIME version3 certificate handling, RFC 2631 defines Diffie Hellman key agreement method [RFC2632], [RFC2631]. SMIME uses Elgamal to provide encryption and decryption as an alternative to RSA. SHA-1 is used to create digital signature. For message encryption 3DES is used. RC2 is used due to DES export restriction in the US to allow product compliance.

SMIME is a security enhancement to MIME that is based on technology from RSA Data security and is an extension to RFC- 822 [RFC 822]. SMIME uses certificate extension that conforms to Version 3 of X.509. S/MIME user must configure each client with a list of trusted certificates and with revocation lists. This means, the responsibility is local to maintain the certificates needed to verify incoming and sign out going messages.
SMIME uses X.509 Version3 certificates. The Key management scheme used by SMIME is partly a strictly hierarchy scheme and partly a flexible web of trust like that which is

used in PGP. Certificates with CRLs are locally maintained or may be maintained by a local administrative entity on behalf of the user or a number of users. Certificates are used to verify incoming messages and sign outgoing messages. Currently there are a number of Certification authorities that provide certificates for e-mail users and enterprise solutions. A few to mention are Nortel, Verisign, GT and the US postal. Of these Verisign is a root
CA for Europe and probably is the globe most widely used.

SMIME provides several advanced security services. One is which provides proof of delivery to the originator of the message. This service is important in case of a dispute between the sender and the recipient it can be used to prove to the third party that the recipient actually received the message. Also the sensitivity of the content can be indicated using security label service. Security labels may be used for access control by indicating which users are permitted access to the object.

Certificates are central to all the services offered by S/MIME Version 2 and 3. Encryption, signature and receipts processing all rely on the binding of an e-mail address to the public key. The e-mail address should be present in the subject alternative name extension. S/MIME determines if the subjects name contained in a certificate matches to a particular e-mail address. For encryption, the originator must be sure that the public key used to distribute the content encryption key belongs to the intended recipient. If the incorrect public key is used then unintended recipient will have the ability to decrypt the message content. For signature, the recipient must compare the e-mail address from the sender with the e-mail address in the certificate. Similarly for receipts, the receipt validatory must compare the e-mail address from the receipt request to the e-mail address in the certificate.

A signed receipt may be requested in order to provide proof of delivery to the originating party. This will allow the originator to demonstrate to a third party that the recipient received the message. A security label is a set of security information regarding the sensitivity of the content that is being protected by S/MIME encapsulation. This may be used for access control, by indicating which users are permitted to access the object. Also priorities or roles may be set in the label, these are for example secret, confidential, restricted and so on. Roles include which kind of people may see the information for example medical billing agents, medical statisticians, etc.

DEPARTMENT OF COMPUTER AND SYSTEMS SCIENCES
Stockholm University/KTH
www.dsv.su.se/eng/publikationer/index.html
Ph.D. theses:

No 91-004 **Olsson, Jan**
An Architecture for Diagnostic Reasoning Based on Causal Models
No 93-008 **Orci, Terttu**
Temporal Reasoning and Data Bases
No 93-009 **Eriksson, Lars-Henrik**
Finitary Partial Definitions and General Logic
No 93-010 **Johannesson, Paul**
Schema Integration Schema Translation, and Interoperability in Federated Information Systems
No 93-018 **Wangler, Benkt**
Contributions to Functional Requirements Modelling
No 93-019 **Boman, Magnus**
A Logical Specification for Federated Information Systems
No 93-024 **Rayner, Manny**
Abductive Equivalential Translation and its Application to Natural-Language Database Interfacing
No 93-025 **Idestam-Almquist, Peter**
Generalization of Clauses
No 93-026 **Aronsson, Martin**
GCLA: The Design, Use, and Implementation of a Program Development
No 93-029 **Boström, Henrik**
Explanation-Based Transformation of Logic programs
No 94-001 **Samuelsson, Christer**
Fast Natural Language Parsing Using Explanation-Based Learning
No 94-003 **Ekenberg, Love**
Decision Support in Numerically Imprecise Domains
No 94-004 **Kowalski, Stewart**
IT Insecurity: A Multi-disciplinary Inquiry
No 94-007 **Asker, Lars**
Partial Explanations as a Basis for Learning
No 94-009 **Kjellin, Harald**
A Method for Acquiring and Refining Knowledge in Weak Theory Domains
No 94-011 **Britts, Stefan**
Object Database Design
No 94-014 **Kilander, Fredrik**
Incremental Conceptual Clustering in an On-Line Application
No 95-019 **Song, Wei**
Schema Integration: - Principles, Methods and Applications
No 95-050 **Johansson, Anna-Lena**
Logic Program Synthesis Using Schema Instantiation in an Interactive Environment

No 95-054 **Stensmo, Magnus**
Adaptive Automated Diagnosis
No 96-004 **Wærn, Annika**
Recognising Human Plans: Issues for Plan Recognition in Human - Computer Interaction
No 96-006 **Orsvärn, Klas**
Knowledge Modelling with Libraries of Task Decomposition Methods
No 96-008 **Dalianis, Hercules**
Concise Natural Language Generation from Formal Specifications
No 96-009 **Holm, Peter**
On the Design and Usage of Information Technology and the Structuring of Communication and Work
No 96-018 **Höök, Kristina**
A Glass Box Approach to Adaptive Hypermedia
No 96-021 **Yngström, Louise**
A Systemic-Holistic Approach to Academic Programmes in IT Security
No 97-005 **Wohed, Rolf**
A Language for Enterprise and Information System Modelling
No 97-008 **Gambäck, Björn**
Processing Swedish Sentences: A Unification-Based Grammar and Some Applications
No 97-010 **Kapidzic Cicovic, Nada**
Extended Certificate Management System: Design and Protocols
No 97-011 **Danielson, Mats**
Computational Decision Analysis
No 97-012 **Wijkman, Pierre**
Contributions to Evolutionary Computation
No 97-017 **Zhang, Ying**
Multi-Temporal Database Management with a Visual Query Interface
No 98-001 **Essler, Ulf**
Analyzing Groupware Adoption: A Framework and Three Case Studies in Lotus Notes Deployment
No 98-008 **Koistinen, Jari**
Contributions in Distributed Object Systems Engineering
No 99-009 **Hakkarainen, Sari**
Dynamic Aspects and Semantic Enrichment in Schema Comparison
No 99-015 **Magnusson, Christer**
Hedging Shareholder Value in an IT dependent Business society - the Framework BRITS
No 00-004 **Verhagen, Henricus**
Norm Autonomous Agents
No 00-006 **Wohed, Petia**
Schema Quality, Schema Enrichment, and Reuse in Information Systems Analysis
No 01-001 **Hökenhammar, Peter**
Integrerad Beställningsprocess vid Datasystemutveckling
No 01-008 **von Schéele, Fabian**
Controlling Time and Communication in Service Economy
No 01-015 **Kajko-Mattsson, Mira**
Corrective Maintenance Maturity Model: Problem Management

No 01-019 **Stirna, Janis**
The Influence of Intentional and Situational Factors on Enterprise Modelling Tool Acquisition in Organisations
No 01-020 **Persson, Anne**
Enterprise Modelling in Practice: Situational Factors and their Influence on Adopting a Participative Approach
No 02-003 **Sneiders, Eriks**
Automated Question Answering: Template-Based Approach
No 02-005 **Eineborg, Martin**
Inductive Logic Programming for Part-of-Speech Tagging
No 02-006 **Bider, Ilia**
State-Oriented Business Process Modelling: Principles, Theory and Practice
No 02-007 **Malmberg, Åke**
Notations Supporting Knowledge Acquisition from Multiple Sources
No 02-012 **Männikkö-Barbutiu, Sirkku**
SENIOR CYBORGS- About Appropriation of Personal Computers Among Some Swedish Elderly People
No 02-028 **Brash, Danny**
Reuse in Information Systems Development: A Qualitative Inquiry
No 03-001 **Svensson, Martin**
Designing, Defining and Evaluating Social Navigation
No 03-002 **Espinoza, Fredrik**
Individual Service Provisioning
No 03-004 **Eriksson-Granskog, Agneta**
General Metarules for Interactive Modular Construction of Natural Deduction Proofs
No 03-005 **De Zoysa, T. Nandika Kasun**
A Model of Security Architecture for Multi-Party Transactions
No 03-008 **Tholander, Jakob**
Constructing to Learn, Learning to Construct - Studies on Computational Tools for Learning
No 03-009 **Karlgren, Klas**
Mastering the Use of Gobbledygook - Studies on the Development of Expertise Through Exposure to Experienced Practitioners' Deliberation on Authentic Problems
No 03-014 **Kjellman, Arne**
Constructive Systems Science - The Only Remaining Alternative?
No 03-015 **Rydberg Fåhræus, Eva**
A Triple Helix of Learning Processes - How to cultivate learning, communication and collaboration among distance-education learners
No 03-016 **Zemke, Stefan**
Data Mining for Prediction - Financial Series Case
No 04-002 **Hulth, Anette**
Combining Machine Learning and Natural Language Processing for Automatic Keyword Extraction
No 04-011 **Jayaweera, Prasad M**.
A Unified Framework for e-Commerce Systems Development: *Business Process Patterns Perspective*
No 04-013 **Söderström, Eva**
B2B Standards Implementation: Issues and Solutions

No 04-014 **Backlund, Per**
Development Process Knowledge Transfer through Method Adaptation, Implementation, and Use
No 05-003 **Davies, Guy**
Mapping and Integration of Schema Representations of Component Specifications
No 05-004 **Jansson, Eva**
Working Together when Being Apart – An Analysis of Distributed Collaborative Work through ICT from an Organizational and Psychosocial Perspective
No 05-007 **Cöster, Rickard**
Algorithms and Representations for Personalised Information Access
No 05-009 **Ciobanu Morogan, Matei**
Security System for Ad-hoc Wireless Networks based on Generic Secure Objects
No 05-010 **Björck, Fredrik**
Discovering Information Security Management
No 05-012 **Brouwers, Lisa**
Microsimulation Models for Disaster Policy Making
No 05-014 **Näckros, Kjell**
Visualising Security through Computer Games
Investigating Game-Based Instruction in ICT Security: an Experimental approach
No 05-015 **Bylund, Markus**
A Design Rationale for Pervasive Computing
No 05-016 **Strand, Mattias**
External Data Incorporation into Data Warehouses
No 05-020 **Casmir, Respickius**
A Dynamic and Adaptive Information Security Awareness (DAISA) approach
No 05-021 **Svensson, Harald**
Developing Support for Agile and Plan-Driven Methods
No 05-022 **Rudström, Åsa**
Co-Construction of Hybrid Spaces
No 06-005 **Lindgren, Tony**
Methods of Solving Conflicts among Induced Rules
No 06-009 **Wrigstad, Tobias**
Owner-Based Alias Management
No 06-011 **Skoglund, Mats**
Curbing Dependencies in Software Evolution
No 06-012 **Zdravkovic, Jelena**
Process Integration for the Extended Enterprise
No 06-013 **Theresia Olsson Neve**
Capturing and Analysing Emotions to Support Organisational Learning: The Affect Based Learning Matrix