3-22-2012

Security Standards and Best Practice Considerations for Quantum Key Distribution (QKD)

Carole A. Harper

SECURITY STANDARDS AND BEST PRACTICE CONSIDERATIONS FOR QUANTUM KEY DISTRIBUTION (QKD)

THESIS

Carole A. Harper, Captain, USAF

AFIT/GSE/ENV/12-M05

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT/GSE/ENV/12-M05

SECURITY STANDARDS AND BEST PRACTICE CONSIDERATIONS FOR QUANTUM KEY DISTRIBUTION (QKD)

THESIS

Presented to the Faculty

Department of Systems and Engineering Management

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the

Degree of Master of Science in Systems Engineering

Carole A. Harper, BS

Captain, USAF

March 2012

AFIT/GSE/ENV/12-M05

SECURITY STANDARDS AND BEST PRACTICE CONSIDERATIONS FOR QUANTUM KEY DISTRIBUTION (QKD)

Carole A. Harper, BS

Captain, USAF

Approved:

Michael R. Grimaila, PhD, CISM, CISSP (Chairman)	Date
Lt Col Jeffrey Humphries, PhD (Member)	Date
Gerald Baumgartner, PhD (Member)	Date

Abstract

QKD systems combine cryptographic primitives with quantum information theory to produce a theoretic unconditionally secure cryptographic key. However, real-world implementations of QKD systems are far from ideal and significantly differ from the theoretic model. Because of this, real-world QKD systems require additional practical considerations when implemented to achieve secure operations. In this thesis, a content analysis of the published literature is conducted to determine if established security and cryptographic standards and best practices are addressed in real world, practical QKD implementations. The research reveals that most published, real world QKD implementations do not take advantage of established security and cryptographic standards and best practices. Based upon an analysis of existing security and cryptographic standards and best practices, systems architecture methodology is used to make recommendations for how these standards can and should be applied to establish a practical, secure, QKD system framework.

To my family, thank you always for your love and encouragement.

Acknowledgments

I would like to express my sincere appreciation to my research advisor, Dr. Michael Grimaila, for his enthusiasm and guidance throughout this effort. I would also like to thank my committee members, Lt Col Jeffrey Humphries and Mr. Gerry Baumgartner, whose expertise was invaluable. Finally, I would like to thank the QKD Project Team for their efforts, discussions and insights. I am incredibly grateful for the opportunity to work on such an exciting project.

Carole A. Harper

Table of Contents

Page

List of Figures

List of Tables

SECURITY STANDARDS AND BEST PRACTICE CONSIDERATIONS FOR QUANTUM KEY DISTRIBUTION (QKD)

I. Introduction

1.1 Background

Classical cryptographic methods rely on the computing time necessary to solve difficult mathematical problems such as discrete logarithms and factoring large prime numbers to provide security. They work by ensuring the time cost verses benefit gained to break the algorithm does not make solving the problem feasible for an attacker [1]; however, increases in computer processing speeds over time have prompted newer and more sophisticated methods of encrypting and protecting data for purposes of information security.

The past few decades, research has yielded a new technology called Quantum Key Distribution (QKD) which utilizes several quantum mechanics principles in conjunction with cryptographic primitives as a way to provide theoretically unconditional security. The appeal of QKD is driven by the unconditional security it provides despite any advances made in computing power or mathematics. As QKD becomes a more viable alternative to existing cryptographic technologies, researchers have sought to determine just how much security a QKD system provides through both mathematic and experimental rigor. While this research has produced a great deal of important discoveries for the future of QKD, very little has been published investigating whether systems meet existing security standards.

1.2 Problem Statement

To understand the security of a QKD system, an understanding of its fundamental principles is required. A QKD system claims to theoretically provide unconditional security by combining three key concepts. First, it utilizes a cryptographic primitive known as a one-time pad [2:10-12]. A one-time pad is a symmetric cryptographic algorithm that requires a random key the same length as the message to be encrypted and, provided the key is never reused, is the only information theoretically secure encryption primitive. Information theoretically secure means that it has been formally proven that knowing the cipher text message in no way gives information regarding the plain text [1]. Second, it employs a message authentication primitive which utilizes a fraction of the key in a Universal 2-Hash function [2:10-12]. Third, the principle that makes QKD a truly unique system is the key distribution primitive which relies on Quantum Information Theory that prevents bits sent on a quantum channel from being copied or intercepted without notice [2:10-12]. Ideally, when these three concepts are implemented in a QKD system, the message sent is secure.

Unfortunately, while theoretical proofs hold great value in determining strengths and weaknesses of a system, the real world implementations very rarely meet ideal conditions. These non idealities often introduce vulnerabilities. Standards and processes are developed to govern implementation, address non-idealities and to determine whether a system meets security requirements. In the case of a system such as QKD where security is a main system function, it becomes absolutely essential to use security standard considerations when developing a baseline architecture. This research seeks to answer the following question: Does existing QKD research consider security standards?

Further, this research seeks to synthesize a prototypical QKD system utilizing systems architecture that incorporates industry security standards and best practices.

1.3 Research Objectives and Hypothesis

This research surveys cryptographic and information technology security standards to examine their use to date in researching QKD and to suggest a starting framework for secure QKD design. It considers accepted practices from the standards community as well as systems engineering processes related to architectural development and definition to provide a baseline for consideration. Specifically, this thesis seeks to answer the following research questions:

1) To what extent do published QKD systems meet security standards?

2) Does systems architecture methodology provide a blueprint for future QKD development?

1.4 Methodology

This research will be conducted utilizing content analysis and will synthesize a prototypical QKD system using architectural definition. To support the stated objectives, a minimal systems engineering architecture will be built as an example of how an engineer may develop a QKD system baseline fulfilling system and security requirements. These requirements are based off the functional requirements of QKD, technical standards and user needs.

1.5 Assumptions and Limitations

By virtue of the nascent technological nature of QKD as well as the methodology described above and in Ch 3, there are several limitations to this research.

1) QKD systems are limited to hardware/software components that are currently available. For example, an ideal system requires a single photon generator to transmit bits using individual photons. In reality, single photon generators are not available for use and so another method, such as an attenuated weak laser pulse may be used instead.

2) Systems do not function in isolation and design must include the system context and so any system presented here will be strictly a general baseline for consideration, not a complete model for validation.

3) The methodology used in this research is limited by the availability of public literature that addresses QKD implementation. Most research papers are necessarily constricted by length and effectively focus on specific aspects of the system rather than developing a coherent whole.

4) The scope of this research is limited. As such, only 10 published papers and four industry standards were selected for review. An attempt was made to present a sampling of papers. Industry standards were selected based on applicability and generalness of use.

1.6 Implications

QKD public research to date has largely been focused on technology and theory development, not engineering rigor. As a result most available security investigations

have relied on either theoretical proofs or laboratory experiments. Although at the time of this research QKD standards requirements are in the process of being formalized [2,4,5,6,7], the newness of the technology has meant that production and use of QKD has outpaced these efforts. This thesis is an attempt to show that industry approaches applied with sufficient engineering rigor are a methodology that should be considered and to provide a foundational architecture for decision makers and future research and development.

1.7 Preview

This thesis is organized into 5 chapters. The introductory chapter discusses the system security considerations in terms technical standards and system architectural definition.

- Ch 2 examines and classifies general QKD information, security standards, selected QKD literature and systems engineering architecture development.
- Ch 3 describes the research methodology and introduces the Security Criteria Matrix and architectural development process used to conduct the research.
- Ch 4 provides results of the Security Criteria Matrix and presents a proposed prototypical QKD architecture
- Ch 5 draws conclusions regarding research objectives, answers the investigative questions and proposes future research.

II. Literature Review

2.1 Chapter Overview

This chapter will provide a review of key literature applicable to QKD research. It will give a general overview of QKD development and discusses ten published papers that will be analyzed in Chapter 4. Additionally it will review the four IT security standards that will be addressed in Chapter 4: Department of Defense Trusted Computer System Evaluation Criteria [18], Common Criteria for Information Technology Security Evaluation[19], Security Requirements for Cryptographic Modules[20], and the five ETSI documents [2,4,5,6,7]. Finally, this chapter will provide an overview of DoDAF v2.0 architecture guidelines [22].

2.2 Description/Background

The birth of quantum cryptography was a paper on conjugate coding by Stephen Wiesner in the late 1960's [33]. Wiesner postulated how quantum mechanics could be used to produce bank notes unable to be counterfeited. His research was mostly disregarded until Charles H. Bennett and Gilles Brassard discovered how to combine Wiesner's ideas with public key cryptography [34]. Shortly after, the real breakthrough was the realization that photons could be used to transmit information [9:2]. Eventually, this realization lead to a paper published in 1984 which put forth the now well known BB84 quantum key distribution protocol [8].

In the QKD protocol proposed by Bennett and Brassard in 1984, single photons are polarized in one of four potential orientations using two possible bases. The

polarization of the photon is assigned based on the desired bit and basis to be sent. The process depicted in Figure 1 is:

1. Alice randomly selects a bit and basis and polarizes photons accordingly.

2. Alice sends the polarized photons to Bob.

3. Bob receives polarized photons through his own randomly chosen basis

4. Alice and Bob then communicate via public channel to reveal which basis they selected for each photon. Photons where matching bases were chosen are kept; photons where bases did not match are ignored.

5. Alice and Bob then perform error correction/verification and in an ideal system what they have left is a secure key [8].

QUANTUM TRANSMISSION															
Alice's Random Bits	0	1	1	0	1	1	0	0	1	0	1	1	0	0	1
Random Sending Bases	D	R	D	R	R	R	R	R	D	D	R	D	D	D	R
Photons Alice Sends	↙	↕	↘	↔	↕	↕	↔	↔	↘	↗	↕	↘	↗	↗	↕
Random Receiving Bases	R	D	D	R	R	D	D	R	D	R	D	D	D	D	R
Bits as Received by Bob	1		1		1	0	0	0		1	1	1		0	1
PUBLIC DISCUSSION															
Bob Reports Bases of Received Bits	R		D		R	D	D	R		R	D	D		D	R
Alice Says Which Bases Were Correct			OK		OK			OK				OK		OK	OK
Presumably Shared Information (if no eavesdropping)			1		1			0				1		0	1
Bob Reveals Some Key Bits at Random					1									0	
Alice Confirms Them					OK									OK	
OUTCOME															
Remaining Shared Secret Bits			1					0				1			1

Figure 1. BB84 Protocol [8]

Since then, experimental quantum cryptography has continued to progress. Eventually, it was postulated that by utilizing a quantum distribution system, combined with a symmetric key known as a one-time-pad, and an appropriate hash function, information could be encrypted that was theoretical secure [2:10-12]. Problems that arise in experimental systems stem from non-idealities that occur such as equipment constraints, environmental context, and protocol or procedural weaknesses. As a result, much work has been done to investigate both the theoretical and experimental weaknesses in QKD and several well-known attacks have been published and mitigated [4].

2.3 Quantum Key Distribution Published Papers

Three distinct types of QKD papers will be reviewed in this thesis. This research will review early papers on QKD, various practical implementations, and finally vulnerability analyses.

Two critical early papers proposing quantum key distribution protocols are [8, 9]. “Quantum Cryptography: Public Key Distribution and Coin Tossing” published by Bennett and Brassard was one of the first papers to propose a protocol for using quantum particles to transmit information over a quantum channel in such a way that it would be impossible to eavesdrop without being detected [8]. This protocol, known as BB84, describes the secure distribution of random key information between two parties. The paper entitled “Experimental Quantum Cryptography” describes a more detailed design of the first experimental implementation of a QKD channel between two users that share no initial secret information [9]. The paper provides an illustration of the original QKD

protocol, the modifications necessary to implement the protocol experimentally, the apparatus used, possible sources of information leakage and the actual experimental data transmitted.

More recent proposals for implementing a QKD system include [10,11,12]. "Quantum Key Distribution over 122 km of Standard Telecom Fiber" reports the first demonstration of QKD using standard fiber over 100 km [10]. It presents a practical implementation and discusses the error rate, key formation rate and other factors limiting maximum fiber length and therefore maximum distance between QKD nodes. "How to Implement Decoy-State Quantum Key Distribution for a Satellite Uplink with 50 dB Channel Loss" addresses how to implement QKD via satellite, which act as a trusted node to link two or more ground stations [11]. The system presented handles up to 57dB photon loss, which is normally considered to be a very high transmission loss, and confirms the viability of a satellite uplink QKD system. "Optical Networking for Quantum Key Distribution and Quantum Communications" discusses leveraging existing fiber infrastructures for quantum communications [12]. This paper describes a potential architecture to support widespread quantum communications that provides a more efficient networking solution than the more common fixed end-to-end connections between Alice and Bob. It does so by experimentally demonstrating several fundamental capabilities of optical networking that apply to QKD and examining the practical impact to quantum signals.

The third class of papers reviewed is specific attacks and security vulnerabilities in practical QKD [13,14,15,16,17]. "Has Quantum Cryptography Been Proven Secure" presents an analysis of the current state of quantum cryptography with particular

emphasis on engineering issues [13]. The purpose is to demonstrate the need for more rigorous examination of presumed weaknesses. The paper disregards hardware and software shortcomings in favor of focusing on defining the problem to be solved in terms of mathematical and theoretical rigor. “After-gate Attack on a Quantum Cryptosystem” presents a specific intercept-resend attack against a specific QKD system [14]. The paper examines how the intercept-resend can be accomplished by targeting a specific component of QKD, the gated single-photon detectors, using bright pulses as faked states and outlines how to mitigate this vulnerability. “Information Leakage via Side Channels in Freespace BB84 Quantum Cryptography” analyzes a free space BB84 implementation using polarization encoded attenuated pulses [15]. The report focuses on potential side channels by measuring all degrees of freedom. “Time-Shift Attack in Practical Quantum Cryptosystems” discusses a vulnerability caused by an efficiency mismatch between two single photon detectors [16]. The paper examines what circumstances cause the vulnerability and discusses measures to mitigate. “Effects of Detector Efficiency Mismatch on Security of Quantum Cryptosystems” is another paper that addresses vulnerabilities presented from detector efficiency mismatches in QKD systems [17]. Experimental data is presented as well as protection measures to prevent exploitation.

2.4 Security Standards

Over the years, many security Information Technology (IT) standards have been published from the existing body of knowledge. Some are tailored for military use, while others are designed for commercial application. Some standards are meant to be general,

while others have detailed specific requirements. The standards reviewed in this research are outlined in the proceeding section.

Published in 1983, the "Orange Book," also titled "Department of Defense Trusted Computer System Evaluation Criteria" is part of the well known "Rainbow Series" of computer standards used by the US government in the 1980s and 1990s [35]. They were developed to provide a way of evaluating commercially available trusted automatic data processing systems. The document itself declares three stated purposes. They are [18]:

1) "To provide a standard to manufacturers as to what security features to build into their new and planned, commercial products in order to provide widely available systems that satisfy trust requirements (with particular emphasis on preventing the disclosure of data) for sensitive applications.

2) To provide DoD components with a metric with which to evaluate the degree of trust that can be placed in computer systems for the secure processing of classified and other sensitive information.

3) To provide a basis for specifying security requirements in acquisition specifications."

With these purposes in mind, the Orange Book breaks security requirements into four main categories: Security Policies, Accountability, Assurance and Documentation. Security policies require a set of rules used by the system to determine allowed access for users and accurately determine information sensitivity. Accountability mandates that access to information be managed based on who is attempting the access and requires the use of audit information to hold the system accountable. Assurance

specifies that mechanisms be evaluated, protected and that and each requirement is enforced. Documentation supports the first three requirements [18].

The Common Criteria (CC) for Information Technology Security Evaluation is a set of documents developed by the international community that form an agreement by which IT can be evaluated to determine the fulfillment of certain security properties. They are designed to protect assets from "unauthorized disclosure, modification, or loss of use" [19: v1, 10]. The CC provides requirements for security functionality and assurance for hardware, firmware and software. In general the CC is intended to be flexible and enable a range of security properties to be looked at for a range of products. Given this flexibility, any meaning derived from the CC must be evaluated within appropriate context. Additionally, there are certain topics that are stated as beyond the scope of the criteria.

The CC is broken into three parts. Part 1 provides an introduction, Part 2 outlines security functional components and Part 3 presents the security assurance components. Additionally the CEM, an accompanying document to the CC, provides guidance for an evaluator to conduct and evaluation based on the CC. This research focuses on CC Part 2 and providing the basis for security functional requirements expressed in a protection profile or security target. A protection profile is defined as an implementation independent security needs statement while a security target is an implementation dependent statement of security needs. The security requirements expressed in Part 2 are intended to counter threats in the operating environment and to be used to create trusted products meeting user needs. As user security requirements change, the functional requirements in the CC document may also change. [19]

The Federal Information Processing Standards (FIPS) Publication 140-2, Security Standards for Cryptographic Modules is used by the National Institute of Standards and Technology to specify the security requirements that must be satisfied by cryptographic modules used to protect sensitive, but unclassified, information. Requirements for more restrictive information are not addressed in this thesis. FIPS 140-2 covers requirements that relate to the design and implementation of a secure module. For additional requirements related to specific functions within a cryptographic module a list of cross referenced documents are provided within the standard. These additional requirements will be noted in this thesis, but not addressed in detail. [20]

Thus far standards mentioned have applied generically to secure IT systems or to classical cryptographic systems. The European Telecommunications Standards Institute (ETSI) has been in the process of developing and releasing a set of five standards documents that apply specifically to QKD systems. They are Components and Internal Interfaces, Module Security Specification, Security Proofs, Use Cases and Application Interface.

The Security Proofs standard examines the generic requirements for quantum security proofs and serves as a reference for developing evaluation criteria. The main themes of the Security Proofs document examine two key challenges to quantum security proofs. They are: the subtlety in security definitions of a quantum cryptographic protocol and the challenges to enforce assumptions in a practical QKD system. Specifically, the Security Proofs standard intends to:

1) “Make precise the nature of the security claim, including its statistical component”

2) “List meaningful restrictions of adversarial action”

3) “Clarify the difference between security claim of a protocol (based on models) and the security claim of its implementation”

4) “To carefully list all the usual components of a QKD protocol with their critical characterizations.” [5]

The specific nature of this document makes evaluating QKD security more straight forward than previous and more generic standards discussed, but a difference between ideal and implementation, between generic requirements and specific limitations of an operating environment, are still a major concern. This will be true of the other two ETSI standards as well. In face the Security Proofs standards specifically does not “give specific parameters for successful QKD as these numbers change with time” or “endorse particular proofs [5].”

Module Security Specification presents the requirements for QKD utilized as part of a telecommunications security system. It establishes a set of minimum specifications that QKD must fulfill based on eleven security aspects identified. Compliance with these specifications is stated as “necessary but not sufficient” to ensure security. The document does not consider varying security levels of degrees of sensitivity of information. The purpose of the Module Security Specification document is to establish requirements that will detect any system penetration with high probability. [4]

The Components and Internal Interfaces standard defines the properties of QKD system components and internal interfaces. Specifically it catalogues relevant requirements for interfaces between components commonly used in most QKD systems.

This standard emphasizes the need for sufficient definitions of parameters, components and operating conditions when implementing QKD. [6]

Finally, the Application Interface standard describes the interface between security applications and key management and the Use Cases standard describes potential implementations for QKD. [7]

2.5 Systems Engineering and Architecture

Systems engineering is an interdisciplinary, top down approach to realizing a successful system. It is the business of integrating all disciplines and specialties into a concerted and structured development process that addresses the entire lifecycle of a problem [21].

Systems architecting is a sub skill of systems engineering. The architect is responsible for organizing the system components, guiding principles and relationships between components and the external environment. The fundamental purpose of systems architecture is a successful mission or vision. The architecture itself is a means to an end and should be tailored to fit the purpose. Formal systems architecture is designed to promote interoperability and support decision-making processes and solutions [22].

Systems engineering and systems architecture processes often go hand in hand and are particularly useful for emerging technology such as QKD due to their interdisciplinary nature and use of best practices over the entire lifecycle of a project. Applying architectural rigor to defining a system visualizes the practical implementation aspects and allows the user and architect to work together to decide upon the level of abstraction most useful in a tailored analysis.

The US Federal Government has established several laws and policies encouraging the use of architectures in support of decisions [23,24,25,26,27,28]. The formalized framework developed to meet federal guidance is the Department of Defense Architecture Framework Version 2.0 (DoDAF V2.0). “DoDAF is the structure for organizing architecture concepts, principles, assumptions, and terminology about operations and solutions into meaningful patterns to satisfy specific DoD purposes [22].” The structure visualization is achieved via models, which consist of documents, spreadsheets or graphical representations, and serve to provide a template for organizing data in a way that is easier to understand. These individual models, when populated, are referred to as views. Several views comprise a viewpoint and viewpoints comprise the architectural description. DoDAF V2.0 also allows for user-defined views to be created as necessary that allow data to be presented via agency specific methods or preferences. The specific visualization used is less important than the data that is collected, organized, and stored and as such the models are not prescribed, but rather the data contained within. This allows for greater tailorability and the freedom to create and scope architectures that meet user requirements. DoDAF V2.0 enumerates eight viewpoints shown in Figure 2 to select from when organizing data. [22]

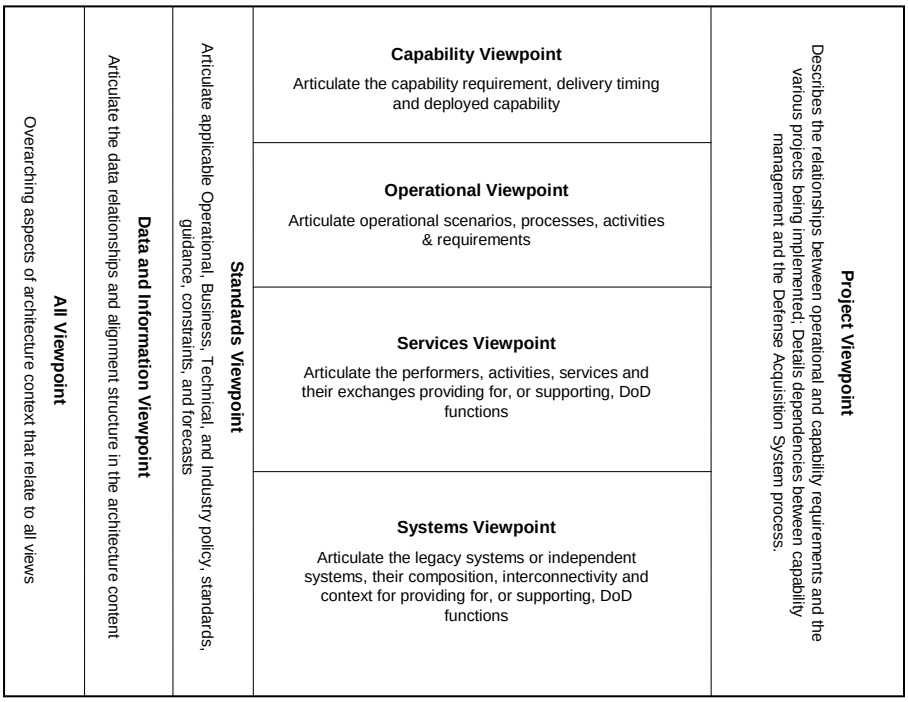

Figure 2. Architecture Viewpoints in DoDAF V2.0 [22]

The All Viewpoint contains information relevant to the entire architectural description. The scope, time frame, and setting provide a context for the description and can include conditions such as goals, vision, doctrine, procedures, etc. Within the All Viewpoint are the Overview and Summary Information view which describes the vision, goals, and conditions, and the Integrated Dictionary view, which provides a repository of definitions for all terms.

The Capability Viewpoint provides a strategic context for capabilities. It presents the goals associated with the ability to achieve desired effects through tasks. The models in this viewpoint are high level and meant to communicate the strategic vision and capabilities to decision makers. Within the Capability Viewpoint are seven individual

views or models that an architect may choose to develop. The Vision model provides the overall vision and context described at a high level. The Capability Taxonomy model gives a hierarchy of capabilities. The Capability Phasing model shows planned capability achievement at various periods of time and conditions. The Capabilities Dependencies model describes dependencies and logical grouping of capabilities to be used for identification purposes as well as impact analysis, disposal and other management functions. The Capability to Organizational Development Mapping model provides planned capability deployment and solutions. The Capability to Operational Activities Mapping model maps required capabilities to operational activities. The Capability to Services Mapping model links capabilities to the services they enable.

The Data and Information Viewpoint organizes business information requirements and structural process rules. It consists of three views: Conceptual Data Model, Logical Data Model, and Physical Data Model. The Conceptual Data Model view gives high level information concepts such as information items, entities, attributes and relationships. The Logical Data Model view documents data requirements and activity rules. While specific format is not specified, this may be done with a class or object diagram. The Physical Data Model view documents how data elements in the Logical Data Model may be implemented and is often described utilizing class or object diagrams.

The Operational Viewpoint describes organizations, tasks and activities that must be performed to accomplish the mission. There are nine views within the Operational Viewpoint. The High Level Operational Concept Graphic is a graphical and textual concept description. The Operational Resource Flow Description shows resource flow

between activities. The Operational Resource Flow Matrix describes both resources exchanged and their attributes. The Organizational Relationships Chart provides role relationships and context between organizations involved in the mission. The Operational Activity Decomposition Tree organizes operational activities into a hierarchy. The Operational Activity Model provides context to operational activities and their inputs and outputs. The Operational Rules Model identifies business rules that provide operational constraints. The State Transition Description illustrates operational activities and their responses to events. The Event-Trace Description traces a sequence of events in a scenario.

The Project Viewpoint contains three views that document the organizational relationships between programs. The Project Portfolio Relationships view describes dependency between organizations and projects. The Project Timelines view gives a timeline with key milestones and interdependencies. The Project to Capability Mapping view maps programs to capabilities to show how the elements achieve capability.

The Services Viewpoint uses 13 views to describe system, service and interconnection functionality. The Services Context Description provides the composition and interaction of services while incorporating the human element. The Services Resource Flow Description lists resource flow between services. The Systems-Services Matrix shows the relationship between systems and their services. The Services-Services Matrix shows the relationships between services. The Services Functionality Description provides functions performed by each service and activities between them. The Operational Activity to Services Traceability Matrix maps service activities to operational activities. The Services Resource Flow Matrix shows elements being

exchanged between services and the attributes. The Services Measures Matrix provides metrics of service elements. The Services Evolution Description lists planned steps for evolving services. The Services Technology and Skills Forecast describes emerging technologies and skills that may be available during the project timeframe. The Services Rules Model describes services functionality by enumerating design or implementation constraints. The Services State Transition Description illustrates service functionality by identifying service responses to events. The Services Event-Trace Description describes services functionality by providing service relevant specifics to critical event sequences.

The Standards Viewpoint contains two views that capture the minimal rules governing system parts or elements as individuals or part of the system. This is the viewpoint that will enumerate the applicable technical and engineering implementation guidelines. The Standards Profile lists current standards and the Standards Forecast describes emerging standards that may apply during project timelines.

The Systems Viewpoint describes the supporting automated systems, interconnectivity, and functionality using 13 views. The Systems Interface Description shows systems and interconnections. The Systems Resource Flow Description shows resource flow between systems. The Systems-Systems Matrix shows relationships of interest such as interfaces between systems. The System Functionality Description describes activities and data flows among systems. The Operational Activity to Systems Function Traceability Matrix maps system activities to operational activities. The Operational Activities to System Traceability Matrix maps the systems to operational activities. The Systems Resource Flow Matrix provides system to system resource flow exchange elements and attributes. The Systems Measures Matrix lists metrics for model

elements. The Systems Evolution Description describes planned steps for an evolving suite of systems. The Systems Technology and Skills Forecast provides technologies and skill emerging within the project timeline. The Systems Rules Model describes systems functionality by identifying constraints due to design or implementation. The Systems State Transition Description describes system functionality by identifying responses to events. This is often illustrated by a state machine diagram. The Systems Event-Trace Description describes system functionality by providing a system specific view for critical the operational sequences. [22]

In addition to the viewpoint guidelines above, architectures may incorporate several fit-for-purpose views. For example, a block diagram or various UML techniques may be appropriate ways of representing a system within an IT industry context. Whichever model chosen to comprise an architectural definition, there are six key steps to follow [29]:

1. Determine Intended Use of Architecture
2. Determine Scope
3. Determine Supporting Data Required
4. Collect, Organize, Correlate, and Store Data
5. Conduct Analysis in Support of Objectives
6. Document Results for Decision Maker Needs

2.6 Chapter Summary

This chapter provided a review of key literature applicable to QKD research. It gave a general overview of QKD development and discusses ten published

papers that will be analyzed in Chapter 4. Additionally it reviewed the four IT security standards that will be addressed: Department of Defense Trusted Computer System Evaluation Criteria [18], Common Criteria for Information Technology Security Evaluation[19], Security Requirements for Cryptographic Modules[20], and the five ETSI documents [2,4,5,6,7]. Finally, this chapter provided an overview of DoDAF v2.0 architecture guidelines [22].

III. Methodology

3.1 Chapter Overview

The purpose of this chapter is to present a methodology and approach for addressing the research questions. It will discuss Content Analysis and describe the method used for data collection. It will also describe the development of the standards matrix evaluation criteria and discuss the architectural process by which the prototypical QKD system will be developed.

3.2 Research Strategy

Content analysis is used to determine the presence of words or concepts within text. It allows a researcher to quantify the presence of such concepts formally or informally or as broadly or specifically as the researcher decides. A content analysis calls for a text to be broken down into manageable categories and then examined using one of the basic content analysis methods. The content analysis methods chosen for this research is conceptual analysis, which quantifies the presence, either implicit or explicit, of a specified concept within a text [36].

3.3 Data Collection

Conceptual analysis begins with identifying the research questions to be answer and determining what text will be analyzed. Once selected the text must be broken into content categories and then analyzed for specified concepts [36]. This analysis seeks to determine whether existing QKD research considers security standards. Based on the

research question and selected methodology, an approach for determining the use of IT security standards in QKD research is developed here.

The content to be analyzed is drawn from published QKD research papers. There have been a great many papers published on this topic. This research attempts to analyze a cross sampling of papers published from the 1980s to present time and does so by breaking the content into three main categories: earlier concepts for QKD implementation when less work had been done towards making it a commercially viable concept, different possible implementations and uses, known security vulnerabilities in practical QKD implementations. From these three categories, 10 papers were chosen for review.

The next step in the research is to specify the concepts to be identified in the coding scheme. The concepts were drawn from IT security standards documents. There are many standards documents present to select from; however, for this research four main documents were used to develop the criteria. These four standards are chosen based on general applicability to both secure IT and cryptographic systems and to QKD.

To determine concepts to be analyzed, each standard requirement is examined, not as detailed specifications but rather as general criteria against several published QKD papers to see if standard security requirements were considered by the authors. Coding is done based on a "met," "partially met," or "does not meet" basis. For example, the security requirement to authenticate will not be considered met or not met based on a detailed explanation of how authentication was accomplished in each set up examined. Instead, if the author makes mention of the need to have an authentication mechanism, the requirement will be considered met and annotated with an "x." If the requirement was

explicitly met in its entirety the matrix be annotated with an "x*." This research primarily evaluates whether standards criteria were considered, not whether the system presented is considered a complete security validation. It would be infeasible given the focus and brevity of most published QKD literature to expect a complete published security profile; however, it is important to consider certain concepts in discussing any secure cryptographic system. With a generalist attitude in mind, security criteria and their interpreted applicability to this research and for potential QKD systems are discussed here and an analysis of criteria conceptualized in literature will be presented in Chapter 4.

From the department of defense Trusted Computer System Evaluation Criteria, the Common Criteria, the Federal Information Processing Standards Publication 140-3, and the ETSI QKD Standards a top level list of minimum considerations was consolidated into a standards matrix. A brief explanation of each of the criteria chosen for review follows.

The Trusted Computer System Evaluation Criteria will be analyzed in four main categories: Security Policy, Accountability, Assurance and Documentation. The security policy consists of four main sub areas which are discretionary access control, object reuse, labeling, and mandatory access control. Discretionary access control requires that access be defined and controlled between users and objects. Object reuse requires information within a storage object be revoked prior to assignment to a subject. Labeling requires that sensitivity labels be assigned to each resource that is accessible outside the system.

Accountability defines ten sub categories for the scope of this research: identification and authentication, audit, system architecture, system integrity, covert

channel analysis, trusted recovery, security testing, design specification and verification, configuration management, and trusted distribution. Identification and authentication specifies requirements for users to identify themselves prior to performing any actions. Audit requirements demand that an audit trail be created, maintained and protected for all relevant events. The system architecture requires that the system domain be maintained for its own execution and be protected from tampering. System integrity determines if there are hardware/software features that may be used to validate correct operations. Covert channel analysis requires a bandwidth determination of each channel identified. Trusted recovery allows the system to recover without compromise after a system failure. Security testing provides proof that the system security mechanisms have been tested. Design specification and verification presents a formal, top level model of the system security policy. Configuration management requires a control of changes to descriptive top-level specification, design data, documentation and code for all security relevant hardware, firmware, and software. Trusted distribution requires a trusted system control and distribution facility to maintain integrity between the master data and the on-site copy.

Assurance criteria require that hardware and software provides assurance that security requirements are enforced. This may be accomplished by developing a system architecture that defines the system domain and hardware/software integrity.

Documentation criteria can be sub divided into two categories for this research: test documentation and design documentation. Test documentation specifies a test plan with procedures showing how security mechanisms are tested and stating their results.

Design documentation provides the manufacturer's philosophy of protection. This may include interfaces between modules, security policies and protection mechanisms. [18]

The Common Criteria Standards, being designed to cover a large variety of systems, enumerates many requirements to be met by secure systems. For the purposes of this research, these standards have been summarized by addressing the key concepts presented within the documents, rather than the many specific requirements. The major areas addressed by the Common Criteria that this research will review are: Security Audit; Non-Repudiation; Cryptographic Key Management to include generation, distribution, access and destruction; User Data Protection; Identification and Authentication; Security Management; Privacy; Resource Utilization; User Session Access; and Trusted Paths/Channels. A brief discussion of each follows.

Security audits require the generation of data capable of being audited. As such a secure system should be able to record, store and select event data relevant to the security of a system. What this data is and how it is generated, stored and selected should be specified. Additionally, the methods and policies for analysis and the responses to any potential security violation should be specified.

Non-repudiation requires some mechanism in place to ensure that a sender cannot deny sending a transmission and a receiver cannot deny having received it. This includes the identification of the user, the information transmitted, the destination, and the invocation of a non-repudiation service.

Cryptographic Key Management refers to how cryptographic keys are managed throughout their lifecycle. The lifecycle of a cryptographic key covers its generation, distribution, access, and eventual destruction. Each of these tasks is required to be done

in accordance with a specified method and applicable standard. For key generation an algorithm, key size and standard should be given. For key distribution, access, and destruction the method and standard should be stated. The Common Criteria includes requirements for cryptographic operations as well; however, the focus of most QKD systems to date is managing the key itself, not the functions performed utilizing that key. While it will be important to consider cryptographic operations when implementing in the larger context, it is not a particular concern in simply developing a QKD module to manage cryptographic keys and so the operations portion of this criteria is acknowledged, but not emphasized in this research with a place in the standards matrix.

User data protection includes the requirements related to protected user data during import export and storage of information. It also specifies any security attributes that relate to user data. This may include access control functions and policies that relate to this requirement.

Identification and authentication establishes requirements to verify user identities. Many other security attributes as well as the security of the entire QKD system rely on the ability to determine that Alice is in fact Alice and Bob is in fact Bob. To meet this critical requirement, mechanisms should be specified that not only authenticate a user's identity but also declare how to handle authentication failures, and quality metrics.

Security management encompasses the management of security roles, attributes, functions and data. Security management also includes revocation of security attributes for entities as well as attribute expiration, or enforcing time limits on security attributes. This specifies the attributes to be managed and by whom, the time limits restricting them

and the actions to be taken in the event that requirements are not met or time limits expire.

Privacy requirements are designed to provide the user with protection against identity discovery and misuse. Privacy ensures that a user may use the system without publicly releasing their identity, but can still be accountable for that use. It also ensures that a user may use a resource without it being publicly known that the resource is in use.

Protection of the Target of Evaluation Security Functionality involves protecting QKD system security functionality. Many mechanisms appear to duplicate user data protection requirements, but the consideration for the criteria emphasizes protecting security function data. This requirement considers the implementation and external data that governs security functions as well as any external entities that may be required to enforce security functional requirements. Examples of specific considerations for this requirement are assuring the availability, confidentiality and integrity of exported data, trusted recovery, fail safe functions, and time stamps.

Resource utilization specifies three main concepts: fault tolerance, priority of service and resource allocation. Fault tolerance ensures that the system continues to function despite failures. It requires that failures can be detected and that system capabilities are maintained or, depending on the failure, the system is shut down. Priority of service allows users to prioritize tasks. Resource allocation prevents a denial of service occurring due to monopolization of resources. In other words, a minimum amount of resources are always reserved for priority tasks.

User session access enumerates the functional requirements for establishing a user session. This may include functions relating to limiting selectable attributes for user

sessions, limiting the number of user sessions, session locking, session establishment, session termination, and access history. This may also include any warning banners designed to notify users of appropriate uses of the system.

The final conceptual requirement from the Common Criteria addressed in this research is trusted paths and channels. This requirement accounts for trusted path between the user and the system as well as between the system and other IT products. To be a trusted path, the channel generally isolates a subset of data and commands from the rest of the data and can provide assurance that the user is communicating with the right system and the system is communicating with the right user. [19]

FIPS Publication 140-2 contains 11 main security requirements governing cryptographic modules: cryptographic module specification; cryptographic module ports and interfaces; roles, services, and authentication; finite state model; physical security; operational environment; cryptographic key management; electromagnetic interference and compatibility; self-tests; design assurance; mitigation of other attacks. In Chapter 4, each of these will be examined to see if the intent of these criteria are met in part or in full by the literature presented. Following is a brief explanation of what is meant by each FIPS 140-2 requirement chosen for the purposes of the standards matrix presented in this paper.

A complete cryptographic module specification includes a detailed accounting of all hardware, software and firmware. It also includes specification of the cryptographic boundary, the security policies, algorithms and approved modes of operation.

Cryptographic module ports and interfaces require the specification of all interfaces for input and output data paths. Depending on the level of security required in

the module, it may also require data ports for critical security parameters to be separate from other ports.

Roles, services and authentication refer to the need for identity based operator authentication. In this case, the primary operators would be Alice and Bob and this requirement is primarily addressing the need for an authentication mechanism as part of the operating protocol. Additionally, services to be specified in a cryptographic module include showing status, performing self-tests, and performing approved security functions.

A cryptographic module should include finite state model. The finite state model should address all operational and error states of the module as well as any transitions between states. Input and output events that result in or from transitions should be specified. At a minimum the model requires on/off states, crypto officer states such as key management and initialization, key entry states, user states, self-test states, and error states.

Physical security encompasses all mechanisms that restrict unauthorized physical access. Mechanisms may include automatic zeroization, tamper detection and responses. Additionally this includes environmental failure protection which protects against environmental conditions and fluctuations whether accidental or induced. As part of this requirement, documentation should specify the normal operating ranges of the module.

The operational environment includes the hardware, firmware and software required to operate the cryptographic module. The operating system is a key feature of the operational environment. Like many of the criteria addressed here, each security level has its own requirements for the operational environment.

Cryptographic key management like in the Common Criteria refers to key management over the entire lifecycle of the cryptographic key. This includes random number and key generation, key establishment, key distribution, key entry/output, key storage, and key zeroization.

Electromagnetic interference and compatibility requires documentation meeting standards put forth in 47 Code of Federal Regulations.

Self-tests should be documented for secure cryptographic modules. They include two main types, power-up tests and conditional tests. Power-up tests occur whenever the module is powered up and conditional tests are required whenever a security function is invoked. Conditional tests may include pairwise consistency tests for public and private keys, software/firmware load tests, manual key entry tests, continuous random number generator tests and bypass tests. Test documentation should include successes and failures as well as any conditions and actions needed to re-enter normal operations.

Design assurance refers to use of best practice for configuration management, development, delivery and operation and guidance for the user and crypto officer. Configuration management mandates that the functional requirements and specifications are met when a system is implemented. It requires that the module, module components within the cryptographic boundary and all associated documentation have a configuration number. Design assurance during development means assurance that the implementation corresponds to the security policy and functional specification. Delivery and operation means ensuring that the module is securely delivered to authorized users and is correctly installed and implemented. Finally, the guidance provides warnings and instructions for

use as well as guidance on how to administer the module such as assumptions, security parameters, and administrative functions.

Mitigation of other attacks covers any attacks for which there were no security requirements known when the standards were published. This is of particular importance in QKD given the uniqueness of the technology and relatively limited commercial field time. When considering mitigation of other attacks the cryptographic module's security policy should specify security mechanisms used. These mechanisms will be validated as requirements and appropriate tests are developed. [20]

The ETSI QKD standards documents released in recent years consist of five parts: Application Interface, Components and Internal Interface, Module Security Specification, Security Proofs, and Use Cases. For the purpose of this research, we will focus primarily on the Module Security Specification which enumerates a number of specific security requirements for QKD. The standards matrix pulls all documentation requirements from this module and examines each requirement individually to see if it has been met. To a less specific extent, but no less important, requirements from the other four standards are addressed. The Components and Interface standard acknowledge that while different implementations of QKD possess different components, there are some that are most commonly used. These components are defined in terms of parameters, operating conditions, and component configuration where possible. From this, three general requirements for parameters, operating conditions and defining components will be added to the standards matrix. The Application Interface describes the key management layer that de-multiplexes secure bits into separate groups and passes to their associated applications. The Security Proofs standard is not represented in the standards matrix

explicitly. While incredibly useful in developing a QKD system and understanding the assumption and principles on which its security is based, the philosophy required by this standard is covered in requirements elsewhere [2,4,5,6,7]

3.4 QKD Architecture

The architectural framework that will be utilized for this research is the DoDAF V2.0 discussed in Chapter 2. DoDAF V2.0 is the framework guidance proposed for DoD managers and process owners to specify requirements and control development. The development is a six step process:

1. Determine Intended Use of Architecture
2. Determine Scope
3. Determine Supporting Data Required
4. Collect, Organize, Correlate, and Store Data
5. Conduct Analysis in Support of Objectives
6. Document Results for Decision Maker Needs [29]

This thesis will attempt to demonstrate a basic prototypical framework for a QKD system that utilizes industry standards as the primary requirements considerations for a new technology, not a complete working implementation. In depth analysis of specific system implementations is left for future research. As such, the architecture presented here will not present a complete meta-model, but rather utilize DoDAF v2.0 guidelines to produce several top level architectural viewpoints that best illustrate IT security standards practices.

3.5 Summary

This chapter presented a methodology and approach for addressing the research questions. It discussed Content Analysis and described the method used for data collection. It also described the development of the standards matrix evaluation criteria and discussed the architectural process by which the prototypical QKD system will be developed.

IV. Analysis and Results

4.1 Chapter Overview

This chapter will present the completed matrix and architecture.

4.2 Standards Matrix

The fully populated standards matrix is sparse, which is not an unexpected result considering the published research surveyed addressed either general concerns for implementing QKD or very specific vulnerabilities. Most of the positive results where published papers are determined to have addressed standards criteria in the matrix only partially or indirectly acknowledge them as requirements. Most of the literature did not attempt to formalize them. There were no specified or derived criteria in the matrix that were entirely addressed by the surveyed literature. The specific positive results of the standards matrix are explained below.

Table 1. Department of Defense Trusted Computer System Evaluation Matrix

		Quantum Cryptography: Public Key Distribution and Coin Tossing	Experimental Quantum Cryptography	Quantum Key Distribution Over 122 km of Standard Telecom Fiber	How to Implement Decoy-State Quantum Key Distribution for a Satellite Uplink with 50 dB Channel Loss	Optical Networking for Quantum Key Distribution and Quantum Communications	Has Quantum Cryptography Been Proven Secure	After Gate Attack on Quantum Cryptosystems	Information Leakage via Side Channels in Freespace BB84 Quantum Cryptography	Time-Shift Attack in Practical Cryptosystems	Effects of D on S
Orange Book ("Department of Defense Trusted Computer System Evaluation Criteria, 1983)	Security Policy: Discretionary Access Control										
	Security Policy: Object Reuse										
	Security Policy: Labels										
	Security Policy: Mandatory Access Control										
	Accountability: Identification and Authentication	x	x		x						
	Accountability: Audit										
	Assurance: System Architecture			x	x	x			x		
	Accountability: System Integrity					x					
	Accountability: Covert Channel Analysis				x	x					
	Accountability: Trusted Recovery										
	Accountability: Security Testing							x		x	x
	Accountability: Design Specification and Verification										
	Accountability: Configuration Management										
	Accountability: Trusted Distribution										
	Documentation: Test Documentation										x
	Documentation: Design Documentation										

Table 2. Common Criteria Matrix

		Quantum Cryptography: Public Key Distribution and Coin Tossing	Experimental Quantum Cryptography	Quantum Key Distribution Over 122 km of Standard Telecom Fiber	How to Implement Decoy-State Quantum Key Distribution with 50 dB Channel Loss	Optical Networking for Quantum Key Distribution for a Satellite Uplink	Has Quantum Cryptography Been Proven Secure	After Gate Attack on Quantum Cryptosystems	Information Leakage via Side Channels in Freespace BB84 Quantum Cryptography	Time Shift Attack in Practical Quantum Cryptosystems	Effects of Decoherence on Quantum Secret Key
Common Criteria	Security Audit				x			x			
	Communication: Non-Repudiation										
	Cryptographic Key Management: Generation	x	x							x	x
	Cryptographic Key Management: Distribution	x	x		x					x	x
	Cryptographic Key Management: Access										
	Cryptographic Key Management: Destruction										
	User Data Protection										
	Identification and Authentication	x	x								
	Security Management										
	Privacy										
	Resource Utilisation				x						x
	TOE Access: User Session Access										
	Trusted Path/Channels	x	x								x

Table 3. Security Standards for Cryptographic Modules Matrix

		Quantum Cryptography: Public Key Distribution and Coin Tossing	Experimental Quantum Cryptography	Quantum Key Distribution Over 122 km of Standard Telecom Fiber	How to Implement Decoy-State Quantum Key Distribution with 50 dB Channel Loss	Optical Networking for Quantum Key Distribution for a Satellite Uplink	Has Quantum Cryptography Been Proven Secure	After Gate Attack on Quantum Cryptosystems	Information Leakage via Side Channels in Freespace BB84 Quantum Cryptography	Time Shift Attack in Practical Quantum Cryptosystems	Effects of Decoherence on Quantum Secret Key
Federal Information Processing Standards Publication 140-2, Security Standards for Cryptographic Modules	Cryptographic Module Specification		x	x	x	x			x		
	Cryptographic Module Ports and Interfaces										
	Roles, Services, and Authentication	x	x		x						
	Finite State Model										
	Physical Security				x						
	Operational Environment										
	Cryptographic Key Management	x	x							x	x
	Electromagnetic Interference/Electromagnetic Compatibility				x	x					
	Self-Tests				x	x					
	Design Assurance										
	Mitigation of Other Attacks		x		x		x	x	x	x	x

Table 4. ETSI Matrix

		Quantum Cryptography: Public Key Distribution and Coin Tossing	Experimental Quantum Cryptography	Quantum Key Distribution Over 122 km of Standard Telecom Fiber	How to Implement Decoy-State Quantum Key Distribution for a Satellite Uplink with 50 dB Channel Loss	Optical Networking for Quantum Key Distribution and Quantum Communications	Has Quantum Cryptography Been Proven Secure	After Gate Attack on Quantum Cryptosystems	Information Leakage via Side Channels in Freespace BB84 Quantum Cryptography	Time-Shift Attack in Practical Quantum Cryptosystems	Effects of Detector Mismatch on Quan
ETSI QKD Standards	Parameters Defined	x	x	x	x	x		x	x	x	x
	Operating Conditions Defined			x	x	x					
	Components Defined		x	x	x	x				x	x
	Specification of the hardware and software configuration items		x								
	Specification of any hardware or software configuration items of a QKD module that are excluded from the security requirements		x								
	Specification of the physical ports and logical interfaces.										
	Specification of the manual or logical controls, status indicators, and applicable physical, logical, and electrical characteristics.										
	List of all security functions, both Approved and non-Approved, specification of all modes of operation										
	Block diagram depicting all of the major hardware components										
	Specification of the design of the hardware and software.		x	x	x	x			x		
	Specification of all security-related information whose disclosure or modification can compromise the security.										
	Specification of a Security Policy										
	Specification of the physical ports and logical interfaces and all defined input and output data paths.										
	Specification of all authorized roles supported.										
	Specification of the services, operations, or functions provided, both Approved and non-Approved										
	Specification of any services provided for which the operator is not required to assume an authorized role										
	Specification of the authentication mechanisms supported	x	x								
	Documentation shall specify which approved software integrity techniques are used.										
	Documentation shall specify the MSI commands employed.										
	Specification of the operational environment.					x					
	Specification of the physical security mechanisms that are employed.										
	Specification of the maintenance access interface and how plaintext secret and private keys and other CSPs are to be zeroized when the maintenance access interface is accessed.										

	If split knowledge procedures are used, proof that if knowledge of n key components is required to reconstruct the original key, then knowledge that any n-1 key components provides no information about the original key other than the key's length.										
	Specification of the SSP storage methods.										
	Specification of the SSP zeroization methods.										
	Specification of self-tests.					x					
	Specification of the error states.										
	Specification of all security functions critical to the secure operationand identification of the applicable pre-operational, conditional, and critical functions tests.										
	If a QKD module implements a bypass capability, specification of the mechanism or logic governing the switching procedure.										
	Specification of procedures for secure installation, generation, and start-up.										
	Specification of the procedures for maintaining security while distributing and delivering.										
	Specification of the correspondence between the design of the hardware/software of a QKD module and the Security Policy.										
	If a QKD module contains software, specification of the source code for the software.										
ETSI QKD Standards	If a cryptographic module contains hardware, specification of the schematics and/or the HDL listings for the hardware.		x	x	x	x			x	x	
	Functional specification that informally describes a QKD module.		x								
	Specification of a formal model that describes the rules and characteristics of the Security Policy.										
	Specification of a rationale that demonstrates the consistency and completeness of the formal model with respect to the Security Policy.										
	Specification of an informal proof of the correspondence between the formal model and the functional specification.										
	For each hardware/software component, source code annotation with comments.										
	Specification of an informal proof of the correspondence between the design of the QKD module and the functional specification.										
	Cryptographic Officer guidance										
	User guidance										
	If a QKD module is designed to mitigate one or more specific attacks, specification of security mechanisms employed to mitigate the attack(s).		x	x	x			x	x	x	x
	Definition of QKD Module Security Policy										
	Specification of a Cryptographic Module Security Policy	x	x							x	x

ETSI QKD Standards	Specification of the normal operating ranges. Specification of the environmental failure protection features.				x	x					x
	Specification of the mitigation techniques against applicable Timing Analysis attacks.										
	Specification of the mitigation techniques against applicable SPA attacks.										
	Specification of the mitigation techniques against applicable DPA attacks.										
	Specification of the mitigation techniques against applicable EME attacks.										
	Specification of all cryptographic keys, cryptographic key components.										
	Specification of each RBG (Approved RBGs and non-Approved RBG entropy sources).		x		x						
	Specification of each of the key generation methods (Approved and non-Approved).	x	x		x	x			x		x
	Specification of the key establishment methods.	x	x		x	x			x		x
	Specification of the key entry and output methods.										

The first paper analyzed using the standards matrix is [8]. As one of the earlier papers presenting the possibility of a practical QKD system, the authors effectively present several key concepts to consider. The primary concern is the protocol used to generate and distribute the cryptographic key. To present this then novel idea, the authors presented a scheme, known today as the BB84 protocol, to generate and transmit enough random bits through a quantum channel to serve as a one-time pad. Bennett and Brassard also provided a basic test for eavesdropping via bit agreement and suggested that an eavesdropper would cause a 25% bit error rate. Additionally, the authors recommended

utilizing a Wegman-Carter [31] authentication tag for messages over the public channel to prevent active eavesdropping. The authentication tag would create a trusted channel, which the Common Criteria defines as "a communication channel that may be initiated by either side of the channel, and provides non-repudiation characteristics with respect to the identity of the sides of the channel." Given this definition, Trusted Path/Channels were also partially addressed by mention of authentication requirements. In total 12 out of 95, or 12.6% of the criteria included in the standards matrix were partially addressed by this proposed system.

The 1991 paper "Experimental Quantum Cryptography" presented results for the first experimental QKD channel. Like the 1984 paper above it stated the need for a public channel authentication scheme and presented a basic key generation and distribution protocol. The 1991 also paper proposed a specific method for randomness and presented a physical description of the apparatus while defining some of the required parameters and operating conditions. The experimental set up addressed two specific QKD attacks: intercept/resend and beam splitting as well has how to determine the information leaked if exploited. Finally, it presented test results to see if results are as expected [9]. In total, 22 of 95, or 23.2% of the examined criteria were partially addressed in some form.

In "Quantum Key Distribution over 122 km of Standard Telecom Fiber" the authors presented a basic set up of the quantum module. There was no description of the classical channel included. The paper addressed causes of error including physical imperfections in the system. As a result, error rates are parameterized. Furthermore, this paper discussed a pulse splitting attack and method to mitigate [10]. These key points partially considered 8 of 95, or 8.4% of the standards requirements.

"Has Quantum Cryptography Been Proven Secure" asserted that additional engineering rigor is required before QKD can be declared secure. It presented a discussion of various QKD assertions and the impact of specific attacks. It does not however, propose a specific system or address any security criteria other than specifying attacks [13]. As a result, on 1 of 95 or 1.1% of the criteria were addressed in this paper.

"Optical Networking for Quantum Key Distribution and Quantum Communications" provided a look at how we would utilize existing fiber infrastructures to implement QKD. Key properties and parameters addressed in this paper were traffic distribution, bit rate, wavelength and power levels. Environmental effects were considered. Software was mentioned, but not elaborated upon and standard QKD protocols such as error correction, privacy amplification and authentication were addressed. Auto-synchronization was also presented. Finally, a basic network diagram was explained [12]. These key discussion points addressed 16 of 95, or 16.8% of the criteria analyzed.

A paper covering implementing a satellite outlined the basic cryptographic key process in a section under system configuration which provided a basic implementation diagram wherein Alice was the transmitter, Bob the receiver and satellites act as trusted nodes. Several parameters were identified, most notably wavelength, timing/precision, phase, intensity, key rate and Quantum Bit Error Rate (QBER). Randomness was handled through passive basis selection. Environmental concerns and the effects of background noise and attenuation were addressed. Further, the need to synchronize systems and implement mechanisms for systems leaving their stated parameters was noted. Processing overhead on the classical channel was a concern and channel performance was analyzed.

Finally, a decoy state protocol was proposed to mitigate specific vulnerabilities [11]. In total, these stated concerns, while not completely developed in a formal specification, lead to this paper addressing 22 out of 95 or 23.2% of the criteria.

"After-gate Attack on a Quantum Cryptosystem" addressed criteria by discussing various known attacks, how to mitigate them and identified the parameters and timing issues required. In particular, this paper addressed the component parameters of an Avalanche Photo Diode which is the most common type of Single Photon Detector used in QKD [14]. This considers 5 of 95 or 5.3% of the criteria.

"Information Leakage via Side Channels" was the second of the specific vulnerability papers and as such provided a discussion of the vulnerability associated with side channels. Additionally, it presented an informal sketch of the physical set up and an explanation of the cryptographic key protocol with accounting for a digital calibration of light pulses. Parameters specified included wavelength, APD efficiency, jitter, and some test parameters for lasers [15]. These specifics imply consideration for 9 of 95, or 9.5% of the criteria.

"Time-Shift in Practical Quantum Cryptosystems" discussed how to exploit an imperfection in the efficiency of single photon detectors, which are components typically used in QKD. In doing so, the paper discussed synchronization, provided a partial sketch of the physical system and declared use of BB84 protocol for key generation and distribution [16]. These key points addressed 10 of 95 or 10.5% of the criteria.

The final paper, "Effects of Detector Efficiency Mismatch on Security of Quantum Cryptosystems" focused on the photon number splitting attack while specifying BB84 protocol for key generation and distribution. Due to the attack focus most

parameters discussed are for single photon detectors: experimental detector sensitivity results, wavelength requirements, and quantum bit error rate. Additionally this paper discussed the set of all possible input signals and how to respond, including responses to failure/input outside expected parameters [17]. It partially discussed 15 of 95, or 15.7% of standards criteria.

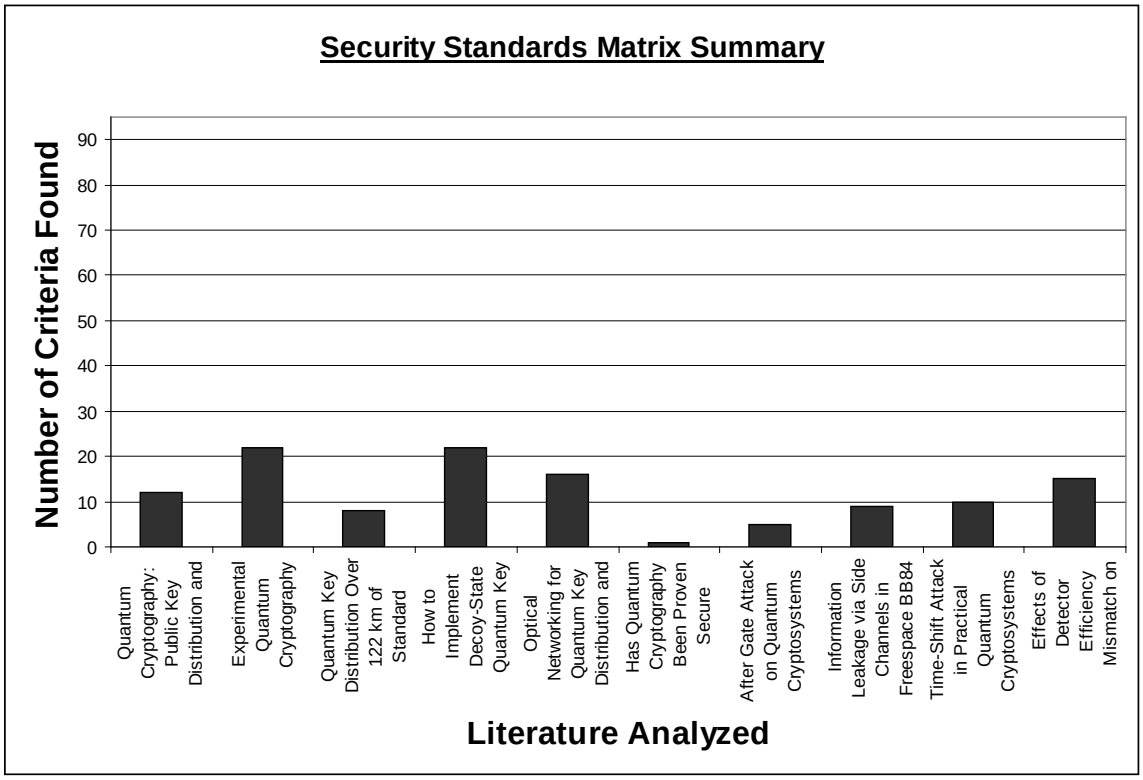

Figure 3, Security Standards Matrix Summary

Figure 3 above, summarizes the results of the content analysis. This answers the question: To what extent do published QKD systems meet security standards? It is seen that while evidence of some security standards concepts investigated can be found in the

published literature analyzed, no papers revealed a definitive discussion or emphasis on the criteria. The best results addressed approximately 23% of the standards body of knowledge.

4.3 QKD Architecture

This section will address the results of the architecture process outlined in Chapter 3. Step 1, Determine Intended Use of Architecture is addressed by the second research question of this thesis. The purpose of the architecture is to determine how to develop a prototypical systems architecture definition for future QKD development. Particularly, it will be tailored reflect the security standards requirements examined in this thesis, as well as identify critical components, primary functions, and non-idealities to the system.

With this overarching purpose in mind, we move on to Step 2, Determine Scope. The architecture contained within this section is scoped to also primarily address the research questions discussed in Chapter 1. Rather than developing a complete meta model, only views deemed most important are presented in this chapter. The information is kept sufficiently high level to avoid issues associated with the multiple methods for implementing QKD, but detailed enough to provide researchers a guideline for implementation considerations. First and foremost, this architecture is meant to highlight concepts and process that should be applied to future development. Both the Use and the Scope of the architecture are addressed in a DoDAF AV-1 model below in Table 5.

Table 5. AV-1

This AV-1 is an executive-level summary of the (QKD) architecture. This initial version of the AV-1 focuses the architecture development effort by documenting the scope and intended usage.	
Architecture Project Identification	
Name	Quantum Key Distribution (QKD) Architecture
Assumptions and Constraints	The QKD architecture: • Will make maximum use of DoD Architecture Framework (DoDAF v.0) and SysML/UML products with changes made as necessary • Will be tailored for maximum design flexibility and usefulness • Will use only optical and electrical hardware components that are currently available • Will use software (e.g. operating systems, classical communications protocol, etc.) that is currently available. • Will emphasize industry security standards.
Approval Authority	• AFIT, Department of Systems Engineering Management • AFIT, Department of Computer and Electrical Engineering
Date Completed	Feb 2012
LOE and Costs	Level of effort will be consistent with Master's Thesis requirements.
Scope: Architecture View(s) and Products Identification	
Products Developed	This QKD architecture consists of the set of integrated architecture products -- AV-1, AV-2, SV-1, SV-10, StdV-2, Block Diagram, Use Cases, Fit-for-purpose policy
Scope	The scope of the QKD architecture is to demonstrate how architectural definition identifies functions, and technology that are required in order to successfully develop a secure quantum key distribution system.
Time Frames Addressed	The QKD architecture would serve as the basis for further research and development of QKD implementations.
Organizations Involved	Development of initial QKD architecture would realistically involve organizations from the DoD as follows: • Air Force Institute of Technology • Laboratory for Telecommunications Sciences • Sandia National Laboratory
Purpose and Viewpoint	
Purpose	This architecture will provide a prototype for future research and recommendations for a QKD system.
Questions to be Answered	The following questions are considered critical to successful completion of the architecting effort. The QKD architecture should be capable of sufficiently answering how systems architecture definition provides a blueprint for future QKD development. In particular: a) What standards apply to QKD? b) What are the main system functions? c) What are the critical system components? d) How do actual hardware components differ from the ideal system assumed in the security proofs?

Architecture Viewpoint	The QKD architecture is developed primarily from DoDAF V2.0.
Context	
Mission	QKD is a response by the scientific community to the threat posed by a quantum computer to some traditional cryptographic algorithms. While the QKD "algorithm" can be proven mathematically secure, this proof is based upon certain assumptions about the hardware/software used to implement QKD. The actual security provided by QKD will depend upon the physical implementation of the QKD "algorithm" and how well real hardware/software matches the ideal assumed in security proofs. QKD is envisioned as a next generation cryptographic key distribution system capitalizing on quantum mechanics principles to generate shared keys. These shared key will be used as a One Time Pad or as shared symmetric secret keys.
Goals	The goal of a QKD system is to allow two or more parties to share a secret.

Rules, Criteria, and Conventions Followed	• IT Standards for Secure Systems and Cryptographic Modules
	  • European Telecommunications Standards Institute
	  • National Institute of Standards and Technology
	  • Department of Defense
	  • Common Criteria
	• Quantum Mechanics Principles
	• Cryptographic Principles
	• DoDAF v2.0 and AP233 (SysML) will be utilized in developing the system architecture.

Tools and File Formats Used	
Enterprise Architect v8.0, Microsoft; Word and Excel, Adobe Acrobat	

In addition to the AV-1, Step 2 also begins the development of the AV-2 or integrated dictionary. Throughout the architecture process it is important to keep a common vocabulary. Terms should be collected and defined to clarify any ambiguity that may arise from inconsistencies. A model dictionary is prepared from common terms found in QKD below in Table 6. This is far from a complete listing of every component and parameter possible, but is tailored to provide a reference point for terms presented within the scope of this thesis. Additional parameters and components that should be

included in an in depth dictionary and architecture are many and depend on the type of implementation being defined.

By analyzing the data collected in the AV-1, a developer begins to understand Step 3, Determine Supporting Data. The supporting data required is decided upon based on the mission and project requirements outlined in the AV-1 during Step 2. This context allows the architect to consider what additional views should be developed to organize and catalogue this information. The information contained in the project overview requires at a minimum this QKD architecture should include the AV-1 or project overview itself, the AV-2 or integrated project dictionary, the OV-1 or concept graphic, the SV-10 state transition diagram, StdV-1 or current applicable standards listing, a block diagram, use cases, and a fit-for-purpose view. By developing these views or models, the architecture will address the major requirements put forth in the standards studied both by enumerating them in the StdV-1 standards listing and by reflecting them throughout the architecture development as well as meet the additional objectives defined by the project.

Table 6. AV-2

Object	Type	Description
Avalanche Photodiode	Component	A device that transforms a weak optical signal into a (more) detectable signal with finite probability. The most common type of Single Photon Detector.
Beam Splitter	Component	Splits an optical pulse into two weaker (typically equal intensity) pulses. The reflected pulse undergoes a 90 degree phase shift.
Buffers	Component	The hardware and connections related to QKD buffers. This includes input/output buffers, plaintext/ciphertext buffers and control buffers
Classical Photo Detector	Component	A device that transforms an optical signal into an electrical signal.
Classical/Public Channel	Component	A non-quantum transmission medium that is used to perform sifting, error correction, privacy amplification and transmit initial authentication. It may consist of phone lines, radio, ethernet, or other classical media. In this architecture, "classical channel" is assumed to be ethernet or fiber connected to a generic computer communications network.
Clock	Component	A device that provides timing pulses to the rest of the machine.
Control Electronics	Component	The physical components of the system controller.
Fiber	Component	Fiber Optic Cable made of optical fibers, usually glass filaments, that can transmit data in the form of light pulses.
Intensity Modulator	Component	A device that can actively set the intensity of a photon pulse that is passing through it.
Interference Filter	Component	Reduces the intensity and spectral width of the laser pulse, typically to select a portion of the spectrum at which the photodetectors have a high quantum efficiency
Key Storage	Component	The hardware and connections used for key storage.
Laser Pulse Generator	Component	A quantum signal generator using laser pulses. Either a continuous wave laser that is pulsed by a switch, or a pulsed laser (e.g. q-switched).
Memory	Component	The hardware and connections used to provide working and program memory.
Phase/Signal Modulator	Component	A device that alters the phase of a given pulse of light. This can be as simple as a small delay line.
Power Supply	Component	The hardware and connections used to provide power to the system.
Quantum Channel	Component	A communications channel for transmitting quantum signals. The physical medium varies based on system implementation. For the purposes of this architecture, the quantum channel is assumed to be optical fiber.
Random Number Generator/Random Bit Generator	Component	A device that outputs unpredictable binary bit sequences. This can be accomplished using physical features such as quantum noise.
Single Photon Generator/Quantum Signal Generator	Component	An optical source that emits, at most, one photon at a time. In practical implementations, a weak attenuated pulse is substituted for a single photon generator.
Alice	Entity	Alice is the quantum information sender
Bob	Entity	Bob is the quantum information receiver
Classical Module	Entity	The portion of the QKD system that functions on a classical communication channel and whose function and security are based on classical proofs.
Control Module	Entity	Controls all cryptographic entities
Quantum Module	Entity	The portion of the QKD system that functions on a quantum communication channel and whose security and functions are based on quantum mechanics principles and proofs.
Afterpulse	Parameter	The probability that a single photon detector will register a photon detection event tiggered by a previous photon detection
Dark Count	Parameter	The probability that a single photon detector will register a photon when none is present.
Error Rate	Parameter	The expected percentage of bit errors in a raw quantum key based on probabilistic quantum properties and physical system limitations. An error rate over a specified threshold indicates a potential problem within the QKD system.
Jitter	Parameter	Uncertainty in detection time for a single photon detector.
Authentication	Term Definition	The process by which a user such as Alice or Bob confirms their identity.
Error Correction	Term Definition	The process by which Alice and Bob ensure their sifted keys are identical.
Heralded Photons	Term Definition	Low photon-count weak coherent pulses that are preceded in time by a brighter pulse of light.
Key Sifting/Distillation	Term Definition	The process by which Alice and Bob agree on a cryptographic key.
Perfect Security	Term Definition	Information theoretic perfect security. By definition, secure even against an adversary with unlimited computing power. QKD proofs use quantum physics to achieve this level of security, at least on a mathematical level.
Polarization Basis	Term Definition	A pair of orthogonal polarizations. The three polarization bases are rectilinear (horizontal and vertical), diagonal (+45 and -45 degrees), and rotational (left- and right-circular). In QKD, each polarization in each basis is arbitrarily defined as either a 0 or a 1. Of note is that any polarization measured in an incorrect basis has an equal probability of being measured as either a 0 or a 1.
Privacy Amplification	Term Definition	A method for reducing the amound of information gained by an unauthorized third party during key
Synchronization	Term Definition	The process by which Alice and Bob determine their clock timing.

Step 4, Collect, Organize, Correlate and Store Data is carried out by collecting and storing data. In the case of this research, data was collected from a background review of information pertaining to QKD and standards. The data was reviewed for security requirements, system functions and component configuration. This data was then organized into the views presented here.

Step 5 of the architecting process, Conduct Analysis in Support of Objectives, involves determining the adherence to requirements. It also identifies additional steps needed to complete the description. By applying the standards criteria synthesized in the matrix to the architecture developed in this Chapter, the results show that it addresses the majority of the requirements from a high level perspective. A summary comparison of all the papers and the architecture discussed here is presented in Figure 3. When this architecture is developed in lower levels of abstraction, these considerations will be specified in detail.

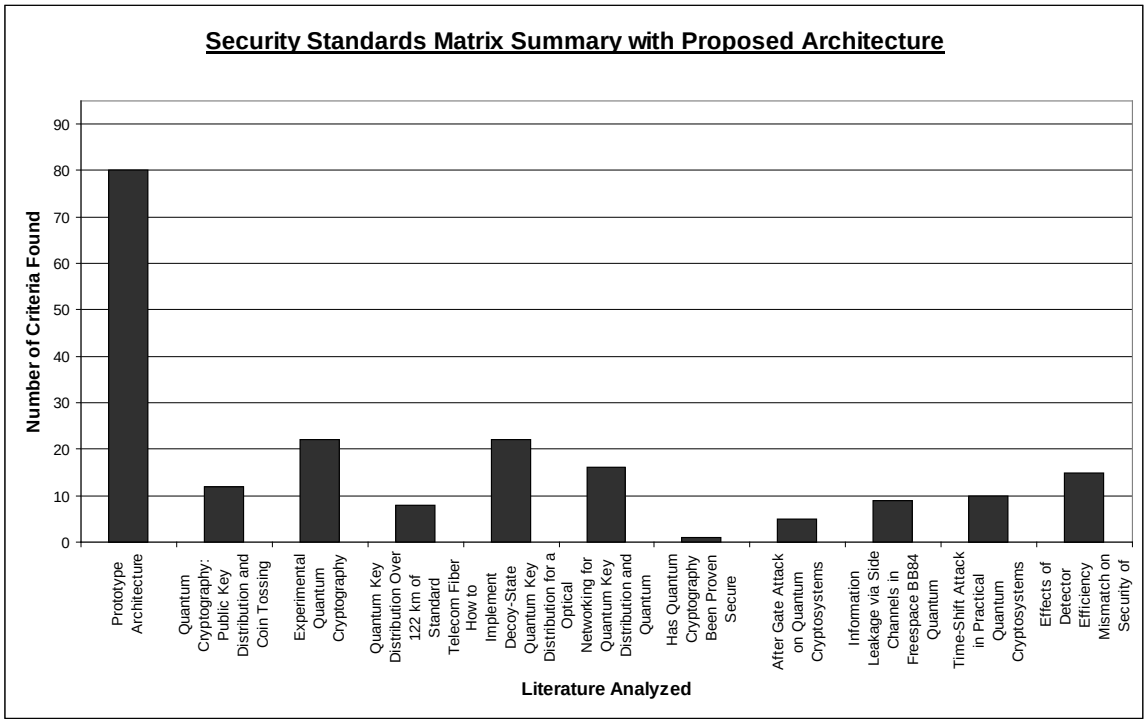

Figure 4. Standards Consideration Comparison Summary

In addition to the products already developed, results of the comparison indicate where the top level architecture is not sufficient to meet standards. This allows

recommendations to be made for additional development. Areas where this proposed architecture fails to consider standards are primarily in software documentation. For now, the architecture products discussed in Steps 3 provide a prototype of the hardware configuration, functions and processes that must be addressed. Step 6 will illustrate the remaining views in this research.

Step 6, Document Results in Accordance with Decision-Maker Needs presents the actual architecture for review. The StdV-1 in Table 7 provides the information for relevant standards documents. The standards enumerated in Table 7 are limited to those discussed in this thesis.

Table 7. StdV-1

Name	Author	Date	Version
Quantum Key Distribution (QKD) QKD Module Security Specification	European Telecommuncations Standards Institute	2010	V1.1.1
Quantum Key Distribution (QKD); Security Proofs	European Telecommuncations Standards Institute	2010	V1.1.1
Quantum Key Distribution (QKD); Components and Internal Interfaces	European Telecommuncations Standards Institute	2010	V1.1.1
Quantum Key Distribution (QKD); Application Interface	European Telecommuncations Standards Institute	2010	V1.1.1
Quantum Key Distribution; Use Cases	European Telecommuncations Standards Institute	2010	V1.1.1
Federal Information Processing Standards Publication, Security Requirements for Cryptographic Modules	National Institute of Standards and Technology	2001	140-2
Department of Defense Trusted Computer System Evaluation Criteria	United States Department of Defense	1985	
Common Criteria for Information Technology Security Evaluation Part	Australia The Defence Signals Directorate	2009	3.1
	New Zealand Government Communications Security Bureau		
	Canada Communications Security Establishment		
	France Direction Centrale de la Sécurité des Systèmes d'Information		
	Germany Bundesamt für Sicherheit in der Informationstechnik		
	Japan Information Technology Promotion Agency		
	Netherlands National Communications Security Agency		
	Spain Ministerio de Administraciones Públicas and Centro Criptológico Nacional		
	United Kingdom Communications-Electronics Security Group		
	United States National Security Agency		
	United States National Institute of Standards and Technology		

The OV-1 concept graphic is shown below in Figure 4. It demonstrates the simplest top level concept view of a QKD system. A QKD system is designed to share a secret between at least two people. It consists of two or more nodes. Each node contains a quantum module and a classical module that communicate with the necessary components.

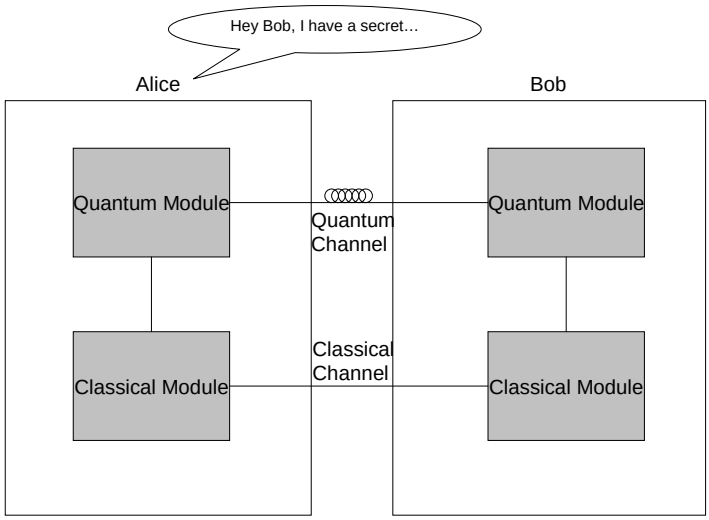

Figure 5. OV-1

The concept graphic must be developed further into other views. The next view developed here is a block definition diagram. The block definition diagram is explicitly required by the ETSI standards and provides an opportunity to examine the critical components and interfaces in a QKD system by delving another level into the quantum and classical modules. The block diagram allows an architect to begin to provide a module specification.

The classical module contains the components that perform classical communications functions as well as interface with the classical or public channel. The quantum module contains the apparatus unique to quantum key distribution and the quantum modules send data across a quantum channel. Additionally, the classical and quantum modules must interface with each other. Within a quantum module the photon pulses are generated, attenuated, encoded and detected. In this illustration, we assume that Alice is the secret sender and will generate and encode the photons and Bob is the secret receiver and will receive and decode the photons. It is important to note that Alice as the secret sender does not always generate the signal for all possible implementations, but for simplicity and the need to discuss component architecture, we assume it to be the case here. Additionally, the quantum channel can be considered a trusted channel. The full implications of this should be explored in a more detailed architecture.

The block diagram in Figure 5 depicts Alice and Bob containing a quantum and classical module. The critical classical components depicted are a clock, power supply, key storage, buffers and memory. The quantum apparatus or quantum module contains an element of randomness, a way of altering the quantum signal, and a component(s) that either generate a quantum signal, such as a laser, or detect a quantum signal, such as an Avalanche Photo-Diode (APD). At this level of abstraction the architecture begins to utilize non ideal components. The laser that generates a quantum signal does not generate single photons required by security protocols and the APD is not a perfect single photon detector. If an architect were to define another layer in each component, these non-ideal components could begin to be analyzed from an architectural perspective.

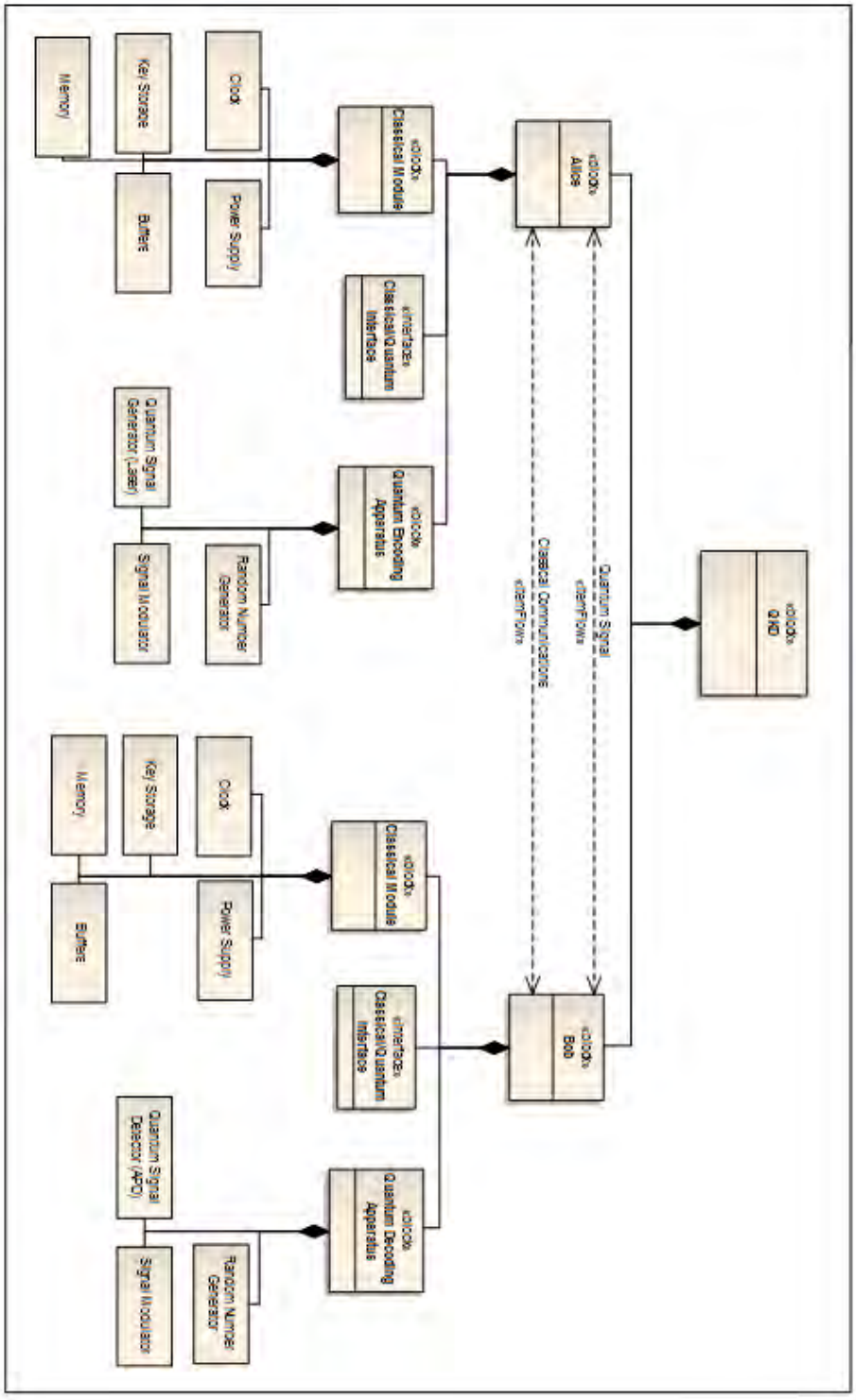

Figure 6. Block Diagram

The classical and quantum modules will need to interface. The interface will eventually need to define the physical and logical interconnections, but that detail is left for developers. Additionally, the interface will need to consider the possibility of different levels of security requirements as the quantum channel is generally considered secure and the classical channel is public. Finally, at this level, Alice and Bob will need a way for information to flow on a classical and quantum channel. As physical and logical definition gets more detailed, configuration management will also become important.

Also specified in the ETSI standards is a Finite State Model. In this research, the Finite State Model is represented using an SV-10 State Transition Diagram. The main states of a QKD system as well as their event triggers are shown in Figure 6. In a lower level architecture this would be developed sufficiently to prove that the module complies with all standards requirements. Details on the security implications of each state and transition will be provided. Security standards require the following states to be included: Power on/off, Initialization, Crypto Officer, CSP Entry, Approved, Self-test and Error. The model below also includes a Calibration, Synchronization and Authentication. Theses states were displayed separately due to their security function criticality. Only a single error state is shown, however as event triggers and other events are provided in greater detail, the error state will need to be better specified. Additionally, it should be noted that “Approved Operations” encompasses the actual QKD key-exchange process.

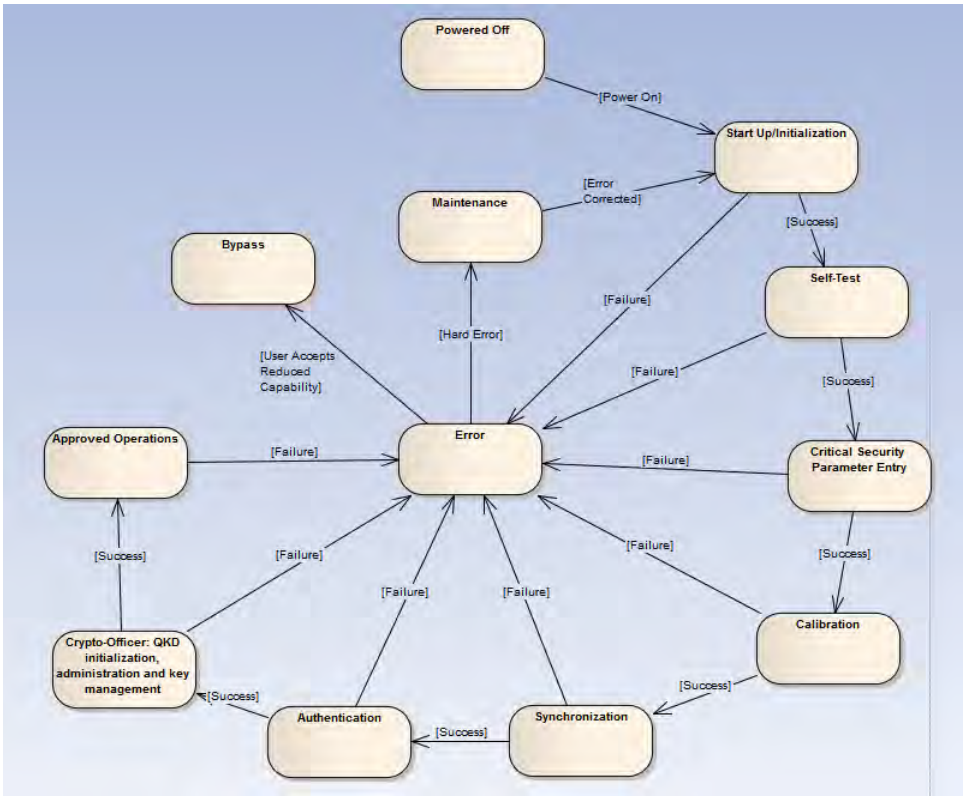

Figure 7. State Transition Diagram

UML Use Cases must also be developed to describe the critical functions a QKD system must perform. The use cases below describe these main functions. Use cases are developed in a casual format.

Use-Case: Start System

Brief Description

To successfully start a QKD system, the components must be powered on and pass a series of start-up self tests. The start-up tests should be designed to determine if all components are present and operating within acceptable parameters, are functioning without compromise and they must not radiate information beyond the cryptographic boundary (i.e. the data output must be disabled). The system should operate in initial start

up mode at minimum long enough to eliminate correlation between initial state and operational state.

Preconditions

The QKD system is securely delivered and installed.

Alternative Flows

Error State

If at any point in the main use case, events do not go as anticipated, system failure or error states must be specified along with any response or operating capability restrictions.

Use-Case: Perform Self Test

Brief Description

Self-tests ensure all components of the module are present and functioning correctly and in a secure state. A QKD module performs self-test for three specific purposes: pre-operational checks for software integrity, cryptographic algorithm implementation, and bypass capability logic; conditional checks for conditions specifying pair-wise consistency tests, software load tests, manual key entry tests, continuous RBG tests, RBG entropy source tests and conditional bypass tests; and other critical functions specified for secure operation. Pre-operational tests must be performed after the system is powered up, woken from hibernation state, or switched from one mode of operation to another, but prior to authentication or any other security function. Conditional and critical function self-tests are performed when conditions are met for specified tests or critical security functions, periodically, or when specified by a user. Tests to verify the mitigation

of one or more specific known QKD attacks will also be conducted to validate security mechanisms.

Alternative Flows

Error State

In the event a self-test fails, system will enter an error state. Error states must be specified along with any response or operating capability restrictions. Response to error state should include a need for trusted recovery when results show module to be insecure.

Degraded Capability

In the event of self-test failure, system may be operated within a specified approved mode of operation supporting degraded capability.

Maintenance Mode

If self-test results indicate a problem, system may enter maintenance mode until resolved.

Special Requirements

Data Output Interface

All data (except status data output via the status output interface) that is output from a cryptographic module (including plaintext data, ciphertext data, cryptographic keys, authentication data, and control information for another module) shall exit via the "data output" interface. All data output via the data output interface shall be inhibited when an error state exists and during self-tests

Specification Requirements

Conditions, critical security functions and testing method should be specified.

Pre-operational Tests

Pre-operational Tests should be automatic.

Use-Case: Calibrate

Brief Description

Before beginning operations, a QKD system must be calibrated within specification. Additionally, periodic re-calibration will be needed for continuous operations. To account for the photon number splitting attack when utilizing an attenuated optical pulse, the intensity, photon number statistics and source stability will be calibrated. The optical source will be calibrated at high power and then attenuated down to single-photon level. The detector will be calibrated utilizing source intensity determined by measuring the un-attenuated laser.

Preconditions

List all assumptions. Consider that a QKD system includes an optical, electronics, and classical layer. Any assumption made may impact the overall security of the system.

Alternative Flows

Error State

If at any point in the main use case, events do not go as anticipated, system failure or error states must be specified along with any response or operating capability restrictions.

Use-Case: Authenticate

Brief Description

QKD requires a secure authentication function and protocol that will be used to authenticate the sender, receiver, message text and time-stamp to ensure that the message originated and was sent where intended and was not modified during transmission. QKD

requires two-factor identity based authentication. The recommended secure function is a two-universal hash function. The protocol for the authentication function should ideally be submitted as a security proof. Authentication is done on the public channel and initial authentication may be handled using a pre-distributed secret. Subsequent authentication will use a fraction of the generated key to perform authentication.

Alternative Flows

Error State

If at any point in the main use case, events do not go as anticipated, system failure or error states must be specified along with any response or operating capability restrictions.

Special Requirements

Authentication Strength

The strength of the authentication mechanism will meet the following:

- For each attempt to use the authentication mechanism, the probability shall be equal to or less than one in 134 217 728 that a false acceptance will occur
- For multiple attempts to use the authentication mechanism during a one-minute period, the probability shall be equal to or less than one in 4 294 967 296 that a false acceptance will occur.
- Time between consecutive attempts will be no less than 2 seconds.
- Authentication shall be met by implementation and not rely on documented procedural controls or security rules (e.g. password size restrictions).

- If passwords are utilized, then restrictions shall be enforced by the module on password selection.
- Feedback of authentication data to an operator shall be obscured during authentication (e.g. no visible display of characters when entering a password).
- Feedback provided during authentication shall not weaken the strength of the authentication beyond that required.
- For first-time authentication, the default authentication data shall be unique to each module.

Object Reuse

Each session must be authenticated and no information not pertaining to that session may be available to users.

User Data Protection

Authentication mechanisms must be in place that protects user data during the authentication process, information flow and export.

Use-Case: Synchronize

Brief Description

Alice and Bob must synchronize their respective clocks prior to beginning operations to provide correct, secure operation of the QKD system. Re-synchronization must occur at pre-defined conditions to account for frequency drift and jitter. One method of synchronization is to use signal and decoy pulses that are attenuated to a single photon. After attenuated pulses are created they are wavelength division multiplexed with a much stronger clock pulse to synchronize Alice and Bob's electronics.

Alternative Flows

Error State

If at any point in the main use case, events do not go as anticipated, system failure or error states must be specified along with any response or operating capability restrictions.

Special Requirements

Frequency Drift

Frequency drift will be less than 10^{-8} parts per clock cycle.

Jitter

Jitter will be 10^{-2} % of a detector gate length.

Use-Case: Generate Quantum Signal

Brief Description

Generate quantum signal describes the process for generating and encoding a photon with a bit of information. In practical QKD system, this is done using an attenuated optical pulse. For example, Alice generates a weak coherent pulse and attenuates to a single photon level. Alice randomly chooses a base value for coding and uses a phase modulator to set the base. Alice then stores the base and value for later use.

Alternative Flows

Error State

If at any point in the main use case, events do not go as anticipated, system failure or error states must be specified along with any response or operating capability restrictions.

Use-Case: Transmit Quantum Signal

Brief Description

The attenuated laser pulse is transmitted over a quantum channel (i.e. fiber or free space) to Bob.

Alternative Flows

Error State

If at any point in the main use case, events do not go as anticipated, system failure or error states must be specified along with any response or operating capability restrictions.

Use-Case: Detect Quantum Signal

Brief Description

Bob's single photon detectors enter ready state. Bob will randomly select a base value and use a phase modulator to set the base. Bob will then set open the set detection gate to receive a photon. Once Bob has recorded detection, he will store the base and value and the detectors will ender a dead period where they detect no photons.

Alternative Flows

Error State

If at any point in the main use case, events do not go as anticipated, system failure or error states must be specified along with any response or operating capability restrictions.

No Detection

Because single photon detectors are probabilistic, there may be no detection event even if a photon is present. In this case the detectors will remain in ready state.

False Positive

Single photon detectors may yield false positives (ex: dark counts). In this case, detectors will ender a dead time and then move back to a ready state.

Use-Case: Final Key Distillation

Brief Description

Once an agreed upon number of photons have been sent and received, Alice and Bob communicate over the public channel the base used to send and measure each photon. If the bases match, the bits are kept, if they differ, the bits are discarded. If the error rate is below a specified threshold, the key is considered secure.

Alternative Flows

Error State

If at any point in the main use case, events do not go as anticipated, system failure or error states must be specified along with any response or operating capability restrictions.

Error Threshold

If the bit error rate is above a certain threshold, the key is discarded as unsecure and cause must be determined.

Use-Case: Error Correction

Brief Description

After an agreed number of photons sent that guarantee a minimum predefined length, Alice and Bob have blocks of bits specifying index, base and bit value are stored. An error reconciliation protocol such as CASCADE or LDPC is implemented over the public channel to ensure both keys are identical.

Preconditions

After key distillation, the error rate must be below the specified threshold to begin error correction.

Use-Case: Privacy Amplification

Brief Description

An approved privacy amplification procedure may be used on the results of the reconciled blocks. Privacy amplification reduces public knowledge about the final key by producing a new, shorter key via a specified function.

Preconditions

After key distillation, the error rate must be below the specified threshold to begin error correction.

Use-Case: Monitor Performance

Brief Description

Performance statistics will be collected and periodically review to ensure system is operating in an approved and expected state. Additionally, performance monitoring should indicate if resources are being appropriately utilized and to perform covert channel analysis. Parameters monitored should be specified.

Alternative Flows

Error State

If at any point in the statistics collected indicate unapproved operation, system failure or error states must be specified along with any response or operating capability restrictions.

Use-Case: React to Failure

Brief Description

If at any time the system fails or begins to operate outside of approved modes or parameters, steps must be taken to ensure no loss of secret information. As such, a method and conditions for stopping and re-starting operations at any point must be specified that meet requirements for trusted recovery.

Use-Case: Audit

Brief Description

The system must have a mechanism to record modification, access, deletion, and addition of data. This information must be recorded and stored via approved mechanisms to prevent unauthorized disclosure, modification or deletion of audit information. Audit information must be available upon request by authorized persons. In the event an audit detects a potential security event, steps to be taken should be specified.

Use-Case: Cryptographic Key Management

Brief Description

The QKD protocol accounts for random number generation, key generation and key establishment. The Cryptographic Key Management use case must describe the Key Entry and Output, Key Storage, and Key Zeroization methods.

Alternative Flows

Error State

If at any point in the main use case, events do not go as anticipated, system failure or error states must be specified along with any response or operating capability restrictions.

Use-Case: Manage Security

Brief Description

A process must be established for managing security attributes, data and functions. It should specify access control and capability. Additionally, management roles and interaction must be defined.

Alternative Flows

Error State

If at any point in the main use case, events do not go as anticipated, system failure or error states must be specified along with any response or operating capability restrictions.

Finally, within the documentation for a QKD system, there must be some fit for purpose view that details the policies the developer plans to ensure trusted distribution and physical security. This can be as simple as a policy document describing the methods or more complex if the user or developer requires. This view is acknowledged here, but the specific format is left up to the developer and user.

Use-Case: Entropy Loss Estimation

Brief Description

This is the process where all information regarding the QKD system (QBER, calibration data, information leaked during error correction, etc.) is utilized to determine how much privacy amplification must be done in order to ensure a secure key is generated.

The above architectural products provide key views and critical information to consider when developing a QKD system. They are formatted to provide easily analyze how well the prototypical system meets standards, but not to provide any meaningful analysis on system performance or quantify security.

4.5 Summary

This chapter demonstrates the extent to which industry technical standards, specifically the Common Criteria, FIPS 140-2, DoD Trusted Computer System Evaluation Criteria and the ETSI QKD standards are utilized in published QKD research. Additionally, it presents a prototypical architecture for consideration in future development and discusses the extent to which it meets the standards presented.

V. Conclusions and Recommendations

5.1 Chapter Overview

This chapter will answer the investigative questions and discuss shortfalls and areas for future research.

5.2 Investigative Questions Answered

Published QKD systems address partial security standards requirements peripherally, and do not directly consider them when discussing system implementation and security concerns. No papers surveyed displayed a definitive discussion of parameters found in the industry standards surveyed. The four standards that are selected in this research to apply to QKD are the Common Criteria, FIPS 140-2, DoD Trusted Computer System Evaluation Criteria and the ETSI QKD standards. These are the minimum that should be considered for high level discussions. When developing more detailed architecture, additional standards will need to be considered.

By conducting a review of security requirements found in the above mentioned documents, the six step systems architecture process and DoDAFv2.0 guidelines provide prototypical system documentation for further research and development. The views developed in this thesis should be considered the minimum architectural models for security considerations not a complete meta-model for analysis. The Use Cases and State Transition Diagram provide a reference for main system functions and states. The critical components are documented in a block diagram and by delving additional layers into this view the specific hardware components used in any particular implementation can be

identified. It can then be determined how they differ from hardware presented in security proofs. At the level developed for this thesis, we see that the single photon detector is in fact an avalanche photo-diode which functions probabilistically based on several parameters. We also see the laser which is incapable of generating the single photon required by security proofs. These are two major components defined differently from the security proofs. The common parameters to consider are identified in the Use Cases.

5.3 Research Limitations

By utilizing this process, a top level architecture prototype was developed that met 84% of the standards considered. The 16% not met can be attributed to the several shortfalls. A large portion of the standards not met were software concerns. This thesis focused primarily on hardware and processes; however, in an actual system implementation software integrity and functionality will be important and logical interfaces must be defined.

From a content analysis perspective, the scope of this thesis is incredibly restrictive. Only analyzing 10 research papers against four standards is a relatively small sampling of content to review. A larger sampling would hold more meaning, but would be difficult to accomplish within the constraints of this thesis.

Architecturally, no actual or simulated QKD system was analyzed to determine accuracy of architecture. The architecture was left vague for scoping purposes, but means that simulation or testing is impossible. Additionally, because QKD limitations and vulnerabilities stem from real world implementation, not having test documentation from

a physical system in its actual operational environment severely limits the effectiveness of an architectural analysis.

5.4 Recommendations for Future Research

I recommend the limitations within the standards presented by this research be addressed further. The four standards analyzed above do apply and should be used when building a QKD system they are not a perfect fit. There are strengths and weaknesses within each criteria chosen.

The intent of the Trusted Computer System Evaluation Criteria definitely applies to QKD. However, as this document has been superseded by more updated standards such as the Common Criteria, this should only be used for academic consideration of older systems.

The Common Criteria are designed to be flexible and allow a range of security concerns to be looked at. They provide a methodology for evaluating the security of a system. As such, they are useful when developing and examining QKD. If used, their flexibility should be utilized and they should be tailored where applicable. The tailoring of the Common Criteria will need to consider the specific implementation of QKD being evaluated as various implementations of QKD differ in greatly in system setup and needed security level. The Common Criteria, however, do not address the unique physical components or probabilistic security nature of QKD.

FIPS 140-2 as a federal cryptographic standard is very applicable to QKD, but will need to be tailored based on the implementation and use of the specific QKD system being evaluated. For example, guidance applying to use of public keys may not be

relevant to QKD and therefore not need to be considered. It provides a suite of tests with four security levels in regards to physical security, key management, roles and services, etc. FIPS 140-2, like the Common Criteria, does not address the unique nature of QKD security.

As the ETSI standards were created specifically for QKD modules, they are the most relevant. They address known parameters of the system and major components as well as discuss QKD specific protocols for key generation. However, they are not sufficient to evaluate the security of a system. For example, a stated goal within the QKD Security Proofs standard is to, "clarify which parameters need to be monitored continuously or periodically to assure the generation of a secret key for the different security levels [4]." The document provides needed discussion of parameters that affect security, but does not quantify the security level or address how it should be validated. Needed security can vary greatly based on system purpose, but thresholds are not given within the QKD Security Proofs document to quantify the various levels.

Additionally, processes for dealing with multiple security levels within a QKD module should be defined. For example, the interface between the quantum and classical modules in a QKD system must be carefully reviewed. Due to the nature of the system, a quantum module operates at a higher security level than a classical module. Exactly how the security levels differ and what restrictions should apply as they are forced to interact needs to be addressed in detail. The standards reviewed do not address this particular concern that becomes apparent as the system is developed in more detail.

A formal methodology is needed to quantify the security of real-world QKD systems and components and provide for independent testing and validation. Standards

could be developed that rigorously define the security within components, protocols and software used in QKD. For example, the laser that generates a single photon, the detector designed to detect a single photon, the configuration of the quantum channel are three main physical areas that are key to QKD security. The ETSI standards provide guidance as to how these should be implemented, but for independent validation, calibration, testing and other concerns, there may be additional standards that should be met. Developing a measurement framework and explicitly defining component and system parameters is a step that has begun to be taken, but must be developed further. There is an ongoing effort that began in September 2011 designed to provide a measurement framework. Metrology for Industrial Quantum Communications is attempting to define the operating parameters for photon emitters, quantum channels and photon receivers used in QKD [32]. This provides a start to developing independent verification and definition of security.

I also recommend that future research utilize all existing applicable security standards for both cryptographic modules and trusted systems. This research is limited to four; however, there is a much larger body of knowledge that should be addressed in any practical attempt to develop QKD. These standards will need to be reviewed and tailored for the operation and configuration selected by each implementation. Some standards to consider are: NIST 800 series, ISO 27000 series, FIPS Publication 199 Standards for Security Categorization of Federal Information and Information Systems, and other protocol specific standards published by the National Institutes for Standards and Technology.

The final recommendation derived from this research is to further develop a coherent integrated QKD architecture. A coherent and integrated architecture would utilize the industry security standards considered in this research as well as those omitted due to scoping limitations and help identify areas where standards fall short or need to be tailored for unique QKD concerns. Its formal process would generate a discussion of the future research concerns above and assist in testing and analysis of the many varied QKD configurations.

5.5 Summary

This chapter provides a summary of research conducted in this thesis. It concludes that IT industry standards are not considered in QKD research publications. It enumerates the limitations found in this research and it provides discussion for future research, both for industry standards and for architectural development.

Appendix A: Glossary of Terms

BB84 – Bennett and Brassard 1984 quantum key distribution protocol. A quantum key distribution protocol using single, polarized photons to encode information.

DoDAF v2.0 – Department of Defense Architecture Framework v2.0. The overarching framework and conceptual model enabling development of architectures for Department of Defense decision makers.

ETSI – European Telecommunications Standards Institute. International body that seeks to produce globally applicable standards for Information and Communications Technology.

IT – Information Technology

CC – Common Criteria. Part of an international technical basis by which Information Technology products can be evaluated by independent laboratories to determine fulfillment of security properties.

CEM – Common Methodology for Information Technology System Evaluation. Part of an international technical basis by which Information Technology products can be evaluated by independent laboratories to determine fulfillment of security properties.

FIPS – Federal Information Processing Standards. United States standards issued by the National Institute for Standards and Technology and approved by the Secretary of Commerce.

AV – DoDAF v2.0 All Viewpoint. This viewpoint contains two models that describe the overarching aspects of the architectural context.

OV - DoDAF v2.0 Operational Viewpoint. This viewpoint contains nine models that describes operational scenarios and activities requirements.

SV - DoDAF v2.0 Systems Viewpoint. This viewpoint contains 13 models that describe the design for solutions by articulating their systems, interconnectivity, and context.

StdV - DoDAF v2.0 Standards Viewpoint. This viewpoint contains two models that articulate the present and projected policies, standards and guidance.

QBER – Quantum Bit Error Rate. The ratio of an error rate to the key rate.

Appendix B: Example In Depth Standards Requirements

The standards presented in this thesis were largely abridged due to constraints caused by the need to scope this effort. Below is an example of what a more in depth look would require for the DoD Trusted Computer System Evaluation Criteria. Of note are the differing levels of security evaluation requirements based on the security concerns of the system. An in depth look would require classifying each major module of the QKD system and addressing them based on the security required within each subsystem. This greatly increases the complexity of the interactions, interfaces, and related analysis required.

Discretionary Access Control (C1): The Trusted Computing Base (TCB) shall define and control access between named users and named objects (e.g., files and programs) in the ADP system. The enforcement mechanism (e.g., self/group/public controls, access control lists) shall allow users to specify and control sharing of those objects by named individuals or defined groups or both.
Identification and Authentication (C1): The TCB shall require users to identify themselves to it before beginning to perform any other actions that the TCB is expected to mediate. Furthermore, the TCB shall use a protected mechanism (e.g., passwords) to authenticate the user's identity.The TCB shall protect authentication data so that it cannot be accessed by any unauthorized user.
System Architecture (C1): The TCB shall maintain a domain for its own execution protects it from external interference or tampering (e.g., by modification of its code or data structures). Resources controlled by the TCB may be a defined subset of the subjects and objects in the ADP system.
System Integrity (C1): Hardware and/or software features shall be provided that can be used to periodically validate the correct operation of the on-site hardware and firmware elements of the TCB.

Security Testing (C1): The security mechanisms of the ADP system shall be tested and found to work as claimed in the system documentation. Testing shall be done to assure that there are no obvious ways for an unauthorized user to bypass or otherwise defeat the security protection mechanisms of the TCB. (See the Security Testing Guidelines.)
Security Features User's Guide (C1): A single summary, chapter, or manual in user documentation shall describe the protection mechanisms provided by the TCB, guidelines on their use, and how they interact with one another.
Trusted Facility Manual (C1): A manual addressed to the ADP System Administrator shall present cautions about functions and privileges that should be controlled when running a secure facility.
Test Documentation (C1): The system developer shall provide to the evaluators a document that describes the test plan, test procedures that show how the the security mechanisms were tested, and results of the security mechanisms' functional testing.
Design Documentation (C1): Documentation shall be available that provides a description of the manufacturer's philosophy of protection and an explanation of how this philosophy is translated into the TCB. If the TCB is composed of distinct modules, the interfaces between these modules shall be described.
Discretionary Access Control (C2): The TCB shall define and control access between named users and named objects (e.g., files and programs) in the ADP system. The enforcement mechanism (e.g., self/group/public controls, access control lists) shall allow users to specify and control sharing of those objects by named individuals, or defined groups of individuals, or by both, and shall provide controls to limit propagation of access rights. The discretionary access control mechanism shall, either by explicit user action or by default, provide that objects are protected from unauthorized access. These access controls shall be capable of including or excluding access to the granularity of a single user. Access permission to an object by users not already possessing access permission shall only be assigned by authorized users.
Object Reuse (C2): All authorizations to the information contained within a storage object shall be revoked prior to initial assignment, allocation or reallocation to a subject from the TCB's pool of unused storage objects. No information, including encrypted representations of information, produced by a prior subject's actions is to be available to any subject that obtains access to an object that has been released back to the system.

Identification and Authentication (C2): The TCB shall require users to identify themselves to it before beginning to perform any other actions that the TCB is expected to mediate. Furthermore, the TCB shall use a protected mechanism (e.g., passwords) to authenticate the user's identity. The TCB shall protect authentication data so that it cannot be accessed by any unauthorized user. The TCB shall be able to enforce individual accountability by providing the capability to uniquely identify each individual ADP system user. The TCB shall also provide the capability of associating this identity with all auditable actions taken by that individual.
Audit (C2): The TCB shall be able to create, maintain, and protect from modification or unauthorized access or destruction an audit trail of accesses to the objects it protects. The audit data shall be protected by the TCB so that read access to it is limited to those who are authorized for audit data. The TCB shall be able to record the following types of events: use of identification and authentication mechanisms, introduction or objects into a user's address space (e.g., file open, program initiation), deletion of objects, and actions taken by computer operators and system administrators and/or system security officers, and other security relevant events. For each recorded event, the audit record shall identify: date and time of the event, user, type of event, and success or failure of the event. For identification/authentication events the origin of request (e.g., terminal ID) shall be included in the audit record. For events that introduce an object into a user's address space and for object deletion events the audit record shall include the name of the object. The ADP system administrator shall be able to selectively audit the actions of any one or more users based on individual identity.
System Architecture (C2): The TCB shall maintain a domain for its own execution that protects it from external interference or tampering (e.g., by modification of its code or data structures). Resources controlled by the TCB may be a defined subset of the subjects and objects in the ADP system. The TCB shall isolate the resources to be protected so that they are subject to the access control and auditing requirements.
System Integrity (C2): Hardware and/or software features shall be provided that can be used to periodically validate the correct operation of the on-site hardware and firmware elements of the TCB.
Security Testing (C2): The security mechanisms of the ADP system shall be tested and found to work as claimed in the system documentation. Testing shall be done to assure that there are no obvious ways for an unauthorized user to bypass or otherwise defeat the security protection mechanisms of the TCB. Testing shall also include a search for obvious flaws that would allow violation of resource isolation, or that would permit unauthorized access to the audit or authentication data. (See the Security Testing guidelines.)
Security Features User's Guide (C2): A single summary, chapter, or manual in user documentation shall describe the protection mechanisms provided by the TCB, guidelines on their use, and how they interact with one another.

Trusted Facility Manual (C2): A manual addressed to the ADP system administrator shall present cautions about functions and privileges that should be controlled when running a secure facility. The procedures for examining and maintaining the audit files as well as the detailed audit record structure for each type of audit event shall be given.
Test Documentation (C2): The system developer shall provide to the evaluators a document that describes the test plan, test procedures that show how the security mechanisms were tested, and results of the security mechanisms' functional testing.
Design Documentation (C2): Documentation shall be available that provides a description of the manufacturer's philosophy of protection and an explanation of how this philosophy is translated into the TCB. If the TCB is composed of distinct modules, the interfaces between these modules shall be described.
Discretionary Access Control (B1): The TCB shall define and control access between named users and named objects (e.g., files and programs) in the ADP system. The enforcement mechanism (e.g., self/group/public controls, access control lists) shall allow users to specify and control sharing of those objects by named individuals, or defined groups of individuals, or by both, and shall provide controls to limit propagation of access rights. The discretionary access control mechanism shall, either by explicit user action or by default, provide that objects are protected from unauthorized access. These access controls shall be capable of including or excluding access to the granularity of a single user. Access permission to an object by users not already possessing access permission shall only be assigned by authorized users.
Object Reuse (B1): All authorizations to the information contained within a storage object shall be revoked prior to initial assignment, allocation or reallocation to a subject from the TCB's pool of unused storage objects. No information, including encrypted representations of information, produced by a prior subject's actions is to be available to any subject that obtains access to an object that has been released back to the system.
Labels (B1): Sensitivity labels associated with each subject and storage object under its control (e.g., process, file, segment, and device) shall be maintained by the TCB. These labels shall be used as the basis for mandatory access control decisions. In order to import non-labeled data, the TCB shall request and receive from an authorized user the security level of the data, and all such actions shall be auditable by the TCB.

Label Integrity (B1): Sensitivity labels shall accurately represent security levels of the specific subjects or objects with which they are associated. When exported by the TCB, sensitivity labels shall accurately and unambiguously represent the internal labels and shall be associated with the information being exported.
Exportation of Labeled Information (B1): The TCB shall designate each communication channel and I/O device as either single-level or multi level. Any change in this designation shall be done manually and shall be auditable by the TCB. The TCB shall maintain and be able to audit any change in the security level or levels associated with a communication channel or I/O device.
Exportation to Multilevel Devices (B1): When the TCB exports an object to a multilevel I/O device, the sensitivity label associated with that object shall also be exported and shall reside on the same physical medium as the exported information and shall be in the same form (i.e., machine-readable or human-readable form). When the TCB exports or imports an object over a multilevel communication channel, the protocol used on that channel shall provide for the unambiguous pairing between the sensitivity labels and the associated information that is sent or received.
Exportation to Single-Level Devices (B1): Single-level I/O devices and single-level communication channels are not required to maintain the sensitivity labels of the information they process. However, the TCB shall include a mechanism by which the TCb and an authorized user reliably communicate to designate the single security level of information imported or exported via single-level communication channels or I/O devices.

Mandatory Access Control (B1): The TCB shall enforce a mandatory access control policy over all subjects and storage objects under its control (e.g., processes, files, segments, devices). These subjects and objects shall be assigned sensitivity labels that are a combination of hierarchical classification levels and non-hierarchical categories, and the labels shall be used as the basis for mandatory access control decisions. The TCB shall be able to support two or more such security levels. (See the Mandatory Access Control Guidelines.) The following requirements shall hold for all accesses between subjects and objects controlled by the TCB: a subject can read an object only if the hierarchical classification in the subject's security level is greater than or equal to the hierarchical classification in the object's security level and the non- hierarchical categories in the subject's security level include all the non-hierarchical categories in the object's security level. A subject can write an object only if the hierarchical classification in the subject's security level is less than or equal to the hierarchical classification in the object's security level and all the non-hierarchical categories in the subject's security level are included in the non-hierarchical categories in the object's security level. Identification and authentication data shall be used by the TCB to authenticate the user's identity and to ensure that the security level and authorization of subjects external to the TCB that may be created to act on behalf of the individual user are dominated by the clearance and authorization of that user.
Identification and Authentication (B1): The TCB shall require users to identify themselves to it before beginning to perform any other actions that the TCB is expected to mediate. Furthermore, the TCB shall maintain authentication data that includes information for verifying the identity of individual users (e.g., passwords) as well as information for determining the clearance and authorizations or individual users. This data shall be used by the TCB to authenticate the user's identity and to ensure that the security level and authorizations of subjects external to the TCB that may be created to act on behalf of the individual user are dominated by the clearance and authorization of that user. The TCB shall protect authentication data so that it cannot be accessed by any unauthorized user. The TCB shall be able to enforce individual accountability by providing the capability to uniquely identify each individual ADP system user. The TCB shall also provide the capability of associating this identity with all auditable actions taken by that individual.
Audit (B1): The TCB shall be able to create, maintain, and protect from modification or unauthorized access or destruction an audit trail of accesses to the objects it protects. The audit data shall be protected by the TCB so that read access to it is limited to those who are authorized for audit data. The TCB shall be able to record the following types of events: use of identification and authentication mechanisms, introduction of objects into a user's address space (e.g., file open, program initiation), deletion of objects, and actions taken by computer operators and system administrators and/or system security officers and other security relevant events. The TCB shall also be able to audit any override of human-readable output markings. For each recorded event, the audit record shall identify: date and time of the event, user, type of event, and success or failure of the event. For identification/authentication events the origin of request (e.g., terminal ID) shall be included in the audit record. For events that introduce an object into a user's address space and for object deletion events the audit record shall include the name of the object and the object's security level. The ADP system administrator shall be able to selectively audit the actions of any one or more users based on individual identity and/or object security level.

System Architecture (B1): The TCB shall maintain a domain for its own execution that protects it from external interference or tampering (e.g., by modification of its code or data structures). Resources controlled by the TCB may be a defined subset of the subjects and objects in the ADP system. The TCB shall maintain process isolation through the provision of distinct address spaces under its control. The TCB shall isolate the resources to be protected so that they are subject to the access control and auditing requirements.
System Integrity (B1): Hardware and/or software features shall be provided that can be used to periodically validate the correct operation of the on-site hardware and firmware elements of the TCB.
Security Testing (B1): The security mechanisms of the ADP system shall be tested and found to work as claimed in the system documentation. A team of individuals who thoroughly understand the specific implementation of the TCB shall subject its design documentation, source code, and object code to thorough analysis and testing. Their objectives shall be: to uncover all design and implementation flaws that would permit a subject external to the TCB to read, change, or delete data normally denied under the mandatory or discretionary security policy enforced by the TCB; as well as to assure that no subject (without authorization to do so) is able to cause the TCB to enter a state such that it is unable to respond to communications initiated by other users. All discovered flaws shall be removed or neutralized and the TCB retested to demonstrate that they have been eliminated and that new flaws have not been introduced. (See the Security Testing Guidelines.)
Design Specification and Verification (B1): An informal or formal model of the security policy supported by the TCB shall be maintained over the life cycle of the ADP system and demonstrated to be consistent with its axioms.
Security Features User's Guide (B1): A single summary, chapter, or manual in user documentation shall describe the protection mechanisms provided by the TCB, guidelines on their use, and how they interact with one another.

Trusted Facility Manual (B1): A manual addressed to the ADP system administrator shall present cautions about functions and privileges that should be controlled when running a secure facility. The procedures for examining and maintaining the audit files as well as the detailed audit record structure for each type of audit event shall be given. The manual shall describe the operator and administrator functions related to security, to include changing the security characteristics of a user. It shall provide guidelines on the consistent and effective use of the protection features of the system, how they interact, how to securely generate a new TCB, and facility procedures, warnings, and privileges that need to be controlled in order to operate the facility in a secure manner.
Test Documentation (B1): The system developer shall provide to the evaluators a document that describes the test plan, test procedures that show how the security mechanisms were tested, and results of the security mechanisms' functional testing.
Design Documentation (B1): Documentation shall be available that provides a description of the manufacturer's philosophy of protection and an explanation of how this philosophy is translated into the TCB. If the TCB is composed of distinct modules, the interfaces between these modules shall be described. An informal or formal description of the security policy model enforced by the TCB shall be available and an explanation provided to show that it is sufficient to enforce the security policy. The specific TCB protection mechanisms shall be identified and an explanation given to show that they satisfy the model.
Discretionary Access Control (B2): The TCB shall define and control access between named users and named objects (e.g., files and programs) in the ADP system. The enforcement mechanism (e.g., self/group/public controls, access control lists) shall allow users to specify and control sharing of those objects by named individuals, or defined groups of individuals, or by both, and shall provide controls to limit propagation of access rights. The discretionary access control mechanism shall, either by explicit user action or by default, provide that objects are protected from unauthorized access. These access controls shall be capable of including or excluding access to the granularity of a single user. Access permission to an object by users not already possessing access permission shall only be assigned by authorized users.

Object Reuse (B2): All authorizations to the information contained within a storage object shall be revoked prior to initial assignment, allocation or reallocation to a subject from the TCB's pool of unused storage objects. No information, including encrypted representations of information, produced by a prior subject's actions is to be available to any subject that obtains access to an object that has been released back to the system.
Labels (B2): Sensitivity labels associated with each ADP system resource (e.g., subject, storage object, ROM) that is directly or indirectly accessible by subjects external to the TCB shall be maintained by the TCB. These labels shall be used as the basis for mandatory access control decisions. In order to import non-labeled data, the TCB shall request and receive from an authorized user the security level of the data, and all such actions shall be auditable by the TCB.
Label Integrity (B2): Sensitivity labels shall accurately represent security levels of the specific subjects or objects with which they are associated. When exported by the TCB, sensitivity labels shall accurately and unambiguously represent the internal labels and shall be associated with the information being exported.
Exportation of Labeled Information (B2): The TCB shall designate each communication channel and I/O device as either single-level or multilevel. Any change in this designation shall be done manually and shall be auditable by the TCB. The TCB shall maintain and be able to audit any change in the security level or levels associated with a communication channel or I/O device.
Exportation to Multilevel Devices (B1): When the TCB exports an object to a multilevel I/O device, the sensitivity label associated with that object shall also be exported and shall reside on the same physical medium as the exported information and shall be in the same form (i.e., machine-readable or human-readable form). When the TCB exports or imports an object over a multilevel communication channel, the protocol used on that channel shall provide for the unambiguous pairing between the sensitivity labels and the associated information that is sent or received.

Exportation to Single-Level Devices (B1): Single-level I/O devices and single-level communication channels are not required to maintain the sensitivity labels of the information they process. However, the TCB shall include a mechanism by which the TCb and an authorized user reliably communicate to designate the single security level of information imported or exported via single-level communication channels or I/O devices.
Subject Sensitivity Labels (B2): The TCB shall immediately notify a terminal user of each change in the security level associated with that user during an interactive session. A terminal user shall be able to query the TCB as desired for a display of the subject's complete sensitivity label.
Device Labels (B2): The TCB shall support the assignment of minimum and maximum security levels to all attached physical devices. These security levels shall be used by the TCB to enforce constraints imposed by the physical environments in which the devices are located.
Mandatory Access Control (B2): The TCB shall enforce a mandatory access control policy over all resources (i.e., subjects, storage objects, and I/O devices that are directly or indirectly accessible by subjects external to the TCB. These subjects and objects shall be assigned sensitivity labels that are a combination of hierarchical classification levels and non-hierarchical categories, and the labels shall be used as the basis for mandatory access control decisions. The TCB shall be able to support two or more such security levels. (See the Mandatory Access Control guidelines.) The following requirements shall hold for all accesses between All subjects external to the TCB and all objects directly or indirectly accessible by these subjects: A subject can read an object only if the hierarchical classification in the subject's security level is greater than or equal to the hierarchical classification in the object's security level and the non- hierarchical categories in the subject's security level include all the non-hierarchical categories in the object's security level. A subject can write an object only if the hierarchical classification in the subject's security level is less than or equal to the hierarchical classification in the object's security level and all the non-hierarchical categories in the subject's security level are included in the non-hierarchical categories in the object's security level. Identification and authentication data shall be used by the TCB to authenticate the user's identity and to ensure that the security level and authorization of subjects external to the TCB that may be created to act on behalf of the individual user are dominated by the clearance and authorization of that user.

Identification and Authentication (B2): The TCB shall require users to identify themselves to it before beginning to perform any other actions that the TCB is expected to mediate. Furthermore, the TCB shall maintain authentication data that includes information for verifying the identity of individual users (e.g., passwords) as well as information for determining the clearance and authorizations of individual users. This data shall be used by the TCB to authenticate the user's identity and to ensure that the security level and authorizations of subjects external to the TCB that may be created to act on behalf of the individual user are dominated by the clearance and authorization of that user. The TCB shall protect authentication data so that it cannot be accessed by any unauthorized user. The TCB shall be able to enforce individual accountability by providing the capability to uniquely identify each individual ADP system user. The TCB shall also provide the capability of associating this identity with all auditable actions taken by that individual.
Trusted Path (B2): The TCB shall support a trusted communication path between itself and user for initial login and authentication. Communications via this path shall be initiated exclusively by a user.
Audit (B2): The TCB shall be able to create, maintain, and protect from modification or unauthorized access or destruction an audit trail of accesses to the objects it protects. The audit data shall be protected by the TCB so that read access to it is limited to those who are authorized for audit data. The TCB shall be able to record the following types of events: use of identification and authentication mechanisms, introduction of objects into a user's address space (e.g., file open, program initiation), deletion of objects, and actions taken by computer operators and system administrators and/or system security officers, and other security relevant events. The TCB shall also be able to audit any override of human-readable output markings. For each recorded event, the audit record shall identify: date and time of the event, user, type of event, and success or failure of the event. For identification/ authentication events the origin of request (e.g., terminal ID) shall be included in the audit record. For events that introduce an object into a user's address space and for object deletion events the audit record shall include the name of the object and the object's security level. The ADP system administrator shall be able to selectively audit the actions of any one or more users based on individual identity and/or object security level. The TCB shall be able to audit the identified events that may be used in the exploitation of covert storage channels.
System Architecture (B2): The TCB shall maintain a domain for its own execution that protects it from external interference or tampering (e.g., by modification of its code or data structures). The TCB shall maintain process isolation through the provision of distinct address spaces under its control. The TCB shall be internally structured into well-defined largely independent modules. It shall make effective use of available hardware to separate those elements that are protection-critical from those that are not. The TCB modules shall be designed such that the principle of least privilege is enforced. Features in hardware, such as segmentation, shall be used to support logically distinct storage objects with separate attributes (namely: readable, writeable). The user interface to the TCB shall be completely defined and all elements of the TCB identified.

System Integrity (B2): Hardware and/or software features shall be provided that can be used to periodically validate the correct operation of the on-site hardware and firmware elements of the TCB.
Covert Channel Analysis (B2): The system developer shall conduct a thorough search for covert storage channels and make a determination (either by actual measurement or by engineering estimation) of the maximum bandwidth of each identified channel. (See the covert channels guideline section.)
Trusted Facility Management (B2): The TCB shall support separate operator and administrator functions.
Security Testing (B2): The security mechanisms of the ADP system shall be tested and found to work as claimed in the system documentation. A team of individuals who thoroughly understand the specific implementation of the TCB shall subject its design documentation, source code, and object code to thorough analysis and testing. Their objectives shall be: to uncover all design and implementation flaws that would permit a subject external to the TCB to read, change, or delete data normally denied under the mandatory or discretionary security policy enforced by the TCB; as well as to assure that no subject (without authorization to do so) is able to cause the TCB to enter a state such that it is unable to respond to communications initiated by other users. The TCB shall be found relatively resistant to penetration. All discovered flaws shall be corrected and the TCB retested to demonstrate that they have been eliminated and that new flaws have not been introduced. Testing shall demonstrate that the TCB implementation is consistent with the descriptive top-level specification. (See the Security Testing Guidelines.)
Design Specification and Verification (B2): A formal model of the security policy supported by the TCB shall be maintained over the life cycle of the ADP system that is proven consistent with its axioms. A descriptive top-level specification (DTLS) of the TCB shall be maintained that completely and accurately describes the TCB in terms of exceptions, error messages, and effects. It shall be shown to be an accurate description of the TCB interface.

Configuration Management (B2): During development and maintenance of the TCB, a configuration management system shall be in place that maintains control of changes to the descriptive top-level specification, other design data, implementation documentation, and source code, the running version of the object code, and test fixtures and documentation. The configuration management system shall assure a consistent mapping among all documentation and code associated with the current version of the TCB. Tools shall be provided for generation of a new version of the TCB from source code. Also available shall be tools for comparing a newly generated version with the previous TCB version in order to ascertain that only the intended changes have been made in the code that will actually be used as the new version of the TCB.
Security Features User's Guide (B2): A single summary, chapter, or manual in user documentation shall describe the protection mechanisms provided by the TCB, guidelines on their use, and how they interact with one another.
Trusted Facility Manual (B2): A manual addressed to the ADP system administrator shall present cautions about functions and privileges that should be controlled when running a secure facility. The procedures for examining and maintaining the audit files as well as the detailed audit record structure for each type of audit event shall be given. The manual shall describe the operator and administrator functions related to security, to include changing the security characteristics of a user. It shall provide guidelines on the consistent and effective use of the protection features of the system, how they interact, how to securely generate a new TCB, and facility procedures, warnings, and privileges that need to be controlled in order to operate the facility in a secure manner. The TCB modules that contain the reference validation mechanism shall be identified. The procedures for secure generation of a new TCB from source after modification of any modules in the TCB shall be described.
Test Documentation (B2): The system developer shall provide to the evaluators a document that describes the test plan, test procedures that show how the security mechanisms were tested, and results of the security mechanisms' functional testing. It shall include results of testing the effectiveness of the methods used to reduce covert channel bandwidths.

Design Documentation (B2): Documentation shall be available that provides a description of the manufacturer's philosophy of protection and an explanation of how this philosophy is translated into the TCB. The interfaces between the TCB modules shall be described. A formal description of the security policy model enforced by the TCB shall be available and proven that it is sufficient to enforce the security policy. The specific TCB protection mechanisms shall be identified and an explanation given to show that they satisfy the model. The descriptive top-level specification (DTLS) shall be shown to be an accurate description of the TCB interface. Documentation shall describe how the TCB implements the reference monitor concept and give an explanation why it is tamper resistant, cannot be bypassed, and is correctly implemented. Documentation shall describe how the TCB is structured to facilitate testing and to enforce least privilege. This documentation shall also present the results of the covert channel analysis and the tradeoffs involved in restricting the channels. All auditable events that may be used in the exploitation of known covert storage channels shall be identified. The bandwidths of known covert storage channels the use of which is not detectable by the auditing mechanisms, shall be provided. (See the Covert Channel Guideline section.)
Discretionary Access Control (B3): The TCB shall define and control access between named users and named objects (e.g., files and programs) in the ADP system. The enforcement mechanism (e.g., access control lists) shall allow users to specify and control sharing of those objects, and shall provide controls to limit propagation of access rights. The discretionary access control mechanism shall, either by explicit user action or by default, provide that objects are protected from unauthorized access. These access controls shall be capable of specifying, for each named object, a list of named individuals and a list of groups of named individuals with their respective modes of access to that object. Furthermore, for each such named object, it shall be possible to specify a list of named individuals and a list of groups of named individuals for which no access to the object is to be given. Access permission to an object by users not already possessing access permission shall only be assigned by authorized users.
Object Reuse (B3): All authorizations to the information contained within a storage object shall be revoked prior to initial assignment, allocation or reallocation to a subject from the TCB's pool of unused storage objects. No information, including encrypted representations of information, produced by a prior subjects actions is to be available to any subject that obtains access to an object that has been released back to the system.

Labels (B3): Sensitivity labels associated with each ADP system resource (e.g., subject, storage object, ROM) that is directly or indirectly accessible by subjects external to the TCB shall be maintained by the TCB. These labels shall be used as the basis for mandatory access control decisions. In order to import non-labeled data, the TCB shall request and receive from an authorized user the security level of the data, and all such actions shall be auditable by the TCB.
Label Integrity (B3): Sensitivity labels shall accurately represent security levels of the specific subjects or objects with which they are associated. When exported by the TCB, sensitivity labels shall accurately and unambiguously represent the internal labels and shall be associated with the information being exported.
Exportation of Labeled Information (B3): The TCB shall designate each communication channel and I/O device as either single-level or multilevel. An change in this designation shall be done manually and shall be auditable by the TCB. The TCB shall maintain and be able to audit any change in the security level or levels associated with a communication channel or I/O device.
Exportation to Multilevel Devices (B3): When the TCB exports an object to a multilevel I/O device, the sensitivity label associated with that object shall also be exported and shall reside on the same physical medium as the exported information and shall be in the same form (i.e., machine-readable or human-readable form). When the TCB exports or imports an object over a multilevel communication channel, the protocol used on that channel shall provide for the unambiguous pairing between the sensitivity labels and the associated information that is sent or received.
Exportation to Single-Level Devices (B3): Single-level I/O devices and single-level communication channels are not required to maintain the sensitivity labels of the information they process. However, the TCB shall include a mechanism by which the TCB and an authorized user reliably communicate to designate the single security level of information imported or exported via single-level communication channels or I/O devices.

Subject Sensitivity Labels (B3): The TCB shall immediately notify a terminal user of each change in the security level associated with that user during an interactive session. A terminal user shall be able to query the TCB as desired for a display of the subject's complete sensitivity label.
Device Labels (B3): The TCB shall support the assignment of minimum and maximum security levels to all attached physical devices. These security levels shall be used by the TCB to enforce constraints imposed by the physical environments in which the devices are located.
Mandatory Access Control (B3): The TCB shall enforce a mandatory access control policy over all resources (i.e., subjects, storage objects, and I/O devices) that are directly or indirectly accessible by subjects external to the TCB. These subjects and objects shall be assigned sensitivity labels that are a combination of hierarchical classification levels and non-hierarchical categories, and the labels shall be used as the basis for mandatory access control decisions. The TCB shall be able to support two or more such security levels. (See the Mandatory Access Control guidelines.) The following requirements shall hold for all accesses between all subjects external to the TCB and all objects directly or indirectly accessible by these subjects: A subject can read an object only if the hierarchical classification in the subject's security level is greater than or equal to the hierarchical classification in the object's security level and the non-hierarchical categories in the subject's security level include all the non-hierarchical categories in the object's security level. A subject can write an object only if the hierarchical classification in the subject's security level is less than or equal to the hierarchical classification in the object's security level and all the non-hierarchical categories in the subject's security level are included in the non- hierarchical categories in the object's security level. Identification and authentication data shall be used by the TCB to authenticate the user's identity and to ensure that the security level and authorization of subjects external to the TCB that may be created to act on behalf of the individual user are dominated by the clearance and authorization of that user.
Identification and Authentication (B3): The TCB shall require users to identify themselves to it before beginning to perform any other actions that the TCB is expected to mediate. Furthermore, the TCB shall maintain authentication data that includes information for verifying the identity of individual users (e.g., passwords) as well as information for determining the clearance and authorizations of individual users. This data shall be used by the TCB to authenticate the user's identity and to ensure that the security level and authorizations of subjects external to the TCB that may be created to act on behalf of the individual user are dominated by the clearance and authorization of that user. The TCB shall protect authentication data so that it cannot be accessed by any unauthorized user. The TCB shall be able to enforce individual accountability by providing the capability to uniquely identify each individual ADP system user. The TCB shall also provide the capability of associating this identity with all auditable actions taken by that individual.

Trusted Path (B3): The TCB shall support a trusted communication path between itself and users for use when a positive TCB-to- user connection is required (e.g., login, change subject security level). Communications via this trusted path shall be activated exclusively by a user of the TCB and shall be logically isolated and unmistakably distinguishable from other paths.
Audit (B3): The TCB shall be able to create, maintain, and protect from modification or unauthorized access or destruction an audit trail of accesses to the objects it protects. The audit data shall be protected by the TCB so that read access to it is limited to those who are authorized for audit data. The TCB shall be able to record the following types of events: use of identification and authentication mechanisms, introduction of objects into a user's address space (e.g., file open, program initiation), deletion of objects, and actions taken by computer operators and system administrators and/or system security officers and other security relevant events. The TCB shall also be able to audit any override of human-readable output markings. For each recorded event, the audit record shall identify: date and time of the event, user, type of event, and success or failure of the event. For identification/authentication events the origin of request (e.g., terminal ID) shall be included in the audit record. For events that introduce an object into a user's address space and for object deletion events the audit record shall include the name of the object and the object's security level. The ADP system administrator shall be able to selectively audit the actions of any one or more users based on individual identity and/or object security level. The TCB shall be able to audit the identified events that may be used in the exploitation of covert storage channels. The TCB shall contain a mechanism that is able to monitor the occurrence or accumulation of security auditable events that may indicate an imminent violation of security policy. This mechanism shall be able to immediately notify the security administrator when thresholds are exceeded, and if the occurrence or accumulation of these security relevant events continues, the system shall take the least disruptive action to terminate the event.
System Architecture (B3): The TCB shall maintain a domain for its own execution that protects it from external interference or tampering (e.g., by modification of its code or data structures). The TCB shall maintain process isolation through the provision of distinct address spaces under its control. The TCB shall be internally structured into well-defined largely independent modules. It shall make effective use of available hardware to separate those elements that are protection-critical from those that are not. The TCB modules shall be designed such that the principle of least privilege is enforced. Features in hardware, such as segmentation, shall be used to support logically distinct storage objects with separate attributes (namely: readable, writeable). The user interface to the TCB shall be completely defined and all elements of the TCB identified. The TCB shall be designed and structured to use a complete, conceptually simple protection mechanism with precisely defined semantics. This mechanism shall play a central role in enforcing the internal structuring of the TCB and the system. The TCB shall incorporate significant use of layering, abstraction and data hiding. Significant system engineering shall be directed toward minimizing the complexity of the TCB and excluding from the TCB modules that are not protection-critical.
System Integrity (B3): Hardware and/or software features shall be provided that can be used to periodically validate the correct operation of the on-site hardware and firmware elements of the TCB.

Covert Channel Analysis (B3): The system developer shall conduct a thorough search for covert channels and make a determination (either by actual measurement or by engineering estimation) of the maximum bandwidth of each identified channel. (See the Covert Channels Guideline section.)
Trusted Facility Management (B3): The TCB shall support separate operator and administrator functions. The functions performed in the role of a security administrator shall be identified. The ADP system administrative personnel shall only be able to perform security administrator functions after taking a distinct auditable action to assume the security administrator role on the ADP system. Non-security functions that can be performed in the security administration role shall be limited strictly to those essential to performing the security role effectively.
Trusted Recovery (B3): Procedures and/or mechanisms shall be provided to assure that, after an ADP system failure or other discontinuity, recovery without a protection compromise is obtained.
Security Testing (B3): The security mechanisms of the ADP system shall be tested and found to work as claimed in the system documentation. A team of individuals who thoroughly understand the specific implementation of the TCB shall subject its design documentation, source code, and object code to thorough analysis and testing. Their objectives shall be: to uncover all design and implementation flaws would permit a subject external to the TCB to read, change, or delete data normally denied under the mandatory or discretionary security policy enforced by the TCB; as well as to assure that no subject (without authorization to do so) is able to cause the TCB to enter a state such that it is unable to respond to communications initiated by other users. The TCB shall be found resistant to penetration. All discovered flaws shall be corrected and the TCB retested to demonstrate that they have been eliminated and that new flaws have not been introduced. Testing shall demonstrate that the TCB implementation is consistent with the descriptive top-level specification. (See the Security Testing Guidelines.) No design flaws and no more than a few correctable implementation flaws may be found during testing and there shall be reasonable confidence that few remain.
Design Specification and Verification (B3): A formal model of the security policy supported by the TCB shall be maintained over the life cycle of the ADP system that is proven consistent with its axioms. A descriptive top-level specification (DTLS) of the TCB shall be maintained that completely and accurately describes the TCB in terms of exceptions, error messages, and effects. It shall be shown to be an accurate description of the TCB interface. A convincing argument shall be given that the DTLS is consistent with the model.

Configuration Management (B3): During development and maintenance of the TCB, a configuration management system shall be in place that maintains control of changes to the descriptive top-level specification, other design data, implementation documentation, source code, the running version of the object code, and test fixtures and documentation. The configuration management system shall assure a consistent mapping among all documentation and code associated with the current version of the TCB. Tools shall be provided for generation of a new version of the TCB from source code. Also available shall be tools for comparing a newly generated version with the previous TCB version in order to ascertain that only the intended changes have been made in the code that will actually be used as the new version of the TCB.
Security Features User's Guide (B3): A single summary, chapter, or manual in user documentation shall describe the protection mechanisms provided by the TCB, guidelines on their use, and how they interact with one another.
Trusted Facility Manual (B3): A manual addressed to the ADP system administrator shall present cautions about functions and privileges that should be controlled when running a secure facility. The procedures for examining and maintaining the audit files as well as the detailed audit record structure for each type of audit event shall be given. The manual shall describe the operator and administrator functions related to security, to include changing the security characteristics of a user. It shall provide guidelines on the consistent and effective use of the protection features of the system, how they interact, how to securely generate a new TCB, and facility procedures, warnings, and privileges that need to be controlled in order to operate the facility in a secure manner. The TCB modules that contain the reference validation mechanism shall be identified. The procedures for secure generation of a new TCB from source after modification of any modules in the TCB shall be described. It shall include the procedures to ensure that the system is initially started in a secure manner. Procedures shall also be included to resume secure system operation after any lapse in system operation.
Test Documentation (B3): The system developer shall provide to the evaluators a document that describes the test plan, test procedures that show how the security mechanisms were tested, and results of the security mechanisms' functional testing. It shall include results of testing the effectiveness of the methods used to reduce covert channel bandwidths.

Design Documentation (B3): Documentation shall be available that provides a description of the manufacturer's philosophy of protection and an explanation of how this philosophy is translated into the TCB. The interfaces between the TCB modules shall be described. A formal description of the security policy model enforced by the TCB shall be available and proven that it is sufficient to enforce the security policy. The specific TCB protection mechanisms shall be identified and an explanation given to show that they satisfy the model. The descriptive top-level specification (DTLS) shall be shown to be an accurate description of the TCB interface. Documentation shall describe how the TCB implements the reference monitor concept and give an explanation why it is tamper resistant, cannot be bypassed, and is correctly implemented. The TCB implementation (i.e., in hardware, firmware, and software) shall be informally shown to be consistent with the DTLS. The elements of the DTLS shall be shown, using informal techniques, to correspond to the elements of the TCB. Documentation shall describe how the TCB is structured to facilitate testing and to enforce least privilege. This documentation shall also present the results of the covert channel analysis and the tradeoffs involved in restricting the channels. All auditable events that may be used in the exploitation of known covert storage channels shall be identified. The bandwidths of known covert storage channels, the use of which is not detectable by the auditing mechanisms, shall be provided. (See the Covert Channel Guideline section.
(A1): A formal model of the security policy must be clearly identified and documented, including a mathematical proof that the model is consistent with its axioms and is sufficient to support the security policy.
(A1): An (Formal Top Level Specification) FTLS must be produced that includes abstract definitions of the functions the TCB performs and of the hardware and/or firmware mechanisms that are used to support separate execution domains.
(A1): The FTLS of the TCB must be shown to be consistent with the model by formal techniques where possible (i.e., where verification tools exist) and informal ones otherwise.
(A1): The TCB implementation (i.e., in hardware, firmware, and software) must be informally shown to be consistent with the FTLS. The elements of the FTLS must be shown, using informal techniques, to correspond to the elements of the TCB. The FTLS must express the unified protection mechanism required to satisfy the security policy, and it is the elements of this protection mechanism that are mapped to the elements of the TCB.
(A1): Formal analysis techniques must be used to identify and analyze covert channels. Informal techniques may be used to identify covert timing channels. The continued existence of identified covert channels in the system must be justified.

Discretionary Access Control (A1): The TCB shall define and control access between named users and named objects (e.g., files and programs) in the ADP system. The enforcement mechanism (e.g., access control lists) shall allow users to specify and control sharing of those objects, and shall provide controls to limit propagation of access rights. The discretionary access control mechanism shall, either by explicit user action or by default, provide that objects are protected from unauthorized access. These access controls shall be capable of specifying, for each named object, a list of named individuals and a list of groups of named individuals with their respective modes of access to that object. Furthermore, for each such named object, it shall be possible to specify a list of named individuals and a list of groups of named individuals for which no access to the object is to be given. Access permission to an object by users not already possessing access permission shall only be assigned by authorized users.
Object Reuse (A1): All authorizations to the information contained within a storage object shall be revoked prior to initial assignment, allocation or reallocation to a subject from the TCB's pool of unused storage objects. No information, including encrypted representations of information, produced by a prior subject's actions is to be available to any subject that obtains access to an object that has been released back to the system.
Labels (A1): Sensitivity labels associated with each ADP system resource (e.g., subject, storage object, ROM) that is directly or indirectly accessible by subjects external to the TCB shall be maintained by the TCB. These labels shall be used as the basis for mandatory access control decisions. In order to import non-labeled data, the TCB shall request and receive from an authorized user the security level of the data, and all such actions shall be auditable by the TCB.
Label Integrity (A1): Sensitivity labels shall accurately represent security levels of the specific subjects or objects with which they are associated. When exported by the TCB, sensitivity labels shall accurately and unambiguously represent the internal labels and shall be associated with the information being exported.
Exportation of Labeled Information (A1): The TCB shall designate each communication channel and I/O device as either single-level or multilevel. Any change in this designation shall be done manually and shall be auditable by the TCB. The TCB shall maintain and be able to audit any change in the security level or levels associated with a communication channel or I/O device.

Exportation to Multilevel Devices (A1): When the TCB exports an object to a multilevel I/O device, the sensitivity label associated with that object shall also be exported and shall reside on the same physical medium as the exported information and shall be in the same form (i.e., machine-readable or human-readable form). When the TCB exports or imports an object over a multilevel communication channel, the protocol used on that channel shall provide for the unambiguous pairing between the sensitivity labels and the associated information that is sent or received.
Exportation to Single Level Devices (A1): Single-level I/O devices and single-level communication channels are not required to maintain the sensitivity labels of the information they process. However, the TCB shall include a mechanism by which the TCB and an authorized user reliably communicate to designate the single security level of information imported or exported via single-level communication channels or I/O devices.
Subject Sensitivity Labels (A1): The TCB shall immediately notify a terminal user of each change in the security level associated with that user during an interactive session. A terminal user shall be able to query the TCB as desired for a display of the subject's complete sensitivity label.
Device Labels (A1): The TCB shall support the assignment of minimum and maximum security levels to all attached physical devices. These security levels shall be used by the TCB to enforce constraints imposed by the physical environments in which the devices are located.

Mandatory Access Control (A1): The TCB shall enforce a mandatory access control policy over all resources (i.e., subjects, storage objects, and I/O devices) that are directly or indirectly accessible by subjects external to the TCB. These subjects and objects shall be assigned sensitivity labels that are a combination of hierarchical classification levels and non-hierarchical categories, and the labels shall be used as the basis for mandatory access control decisions. The TCB shall be able to support two or more such security levels. (See the Mandatory Access Control guidelines.) The following requirements shall hold for all accesses between all subjects external to the TCB and all objects directly or indirectly accessible by these subjects: A subject can read an object only if the hierarchical classification in the subject's security level is greater than or equal to the hierarchical classification in the object's security level and the non-hierarchical categories in the subject's security level include all the non-hierarchical categories in the object's security level. A subject can write an object only if the hierarchical classification in the subject's security level is less than or equal to the hierarchical classification in the object's security level and all the non-hierarchical categories in the subject's security level are included in the non- hierarchical categories in the object's security level. Identification and authentication data shall be used by the TCB to authenticate the user's identity and to ensure that the security level and authorization of subjects external to the TCB that may be created to act on behalf of the individual user are dominated by the clearance and authorization of that user.
Identification and Authentication (A1): The TCB shall require users to identify themselves to it before beginning to perform any other actions that the TCB is expected to mediate. Furthermore, the TCB shall maintain authentication data that includes information for verifying the identity of individual users (e.g., passwords) as well as information for determining the clearance and authorizations of individual users. This data shall be used by the TCB to authenticate the user's identity and to ensure that the security level and authorizations of subjects external to the TCB that may be created to act on behalf of the individual user are dominated by the clearance and authorization of that user. The TCB shall protect authentication data so that it cannot be accessed by any unauthorized user. The TCB shall be able to enforce individual accountability by providing the capability to uniquely identify each individual ADP system user. The TCB shall also provide the capability of associating this identity with all auditable actions taken by that individual.

Trusted Path (A1): The TCB shall support a trusted communication path between itself and users for use when a positive TCB-to- user connection is required (e.g., login, change subject security level). Communications via this trusted path shall be activated exclusively by a user or the TCB and shall be logically isolated and unmistakably distinguishable from other paths.
Audit (A1): The TCB shall be able to create, maintain, and protect from modification or unauthorized access or destruction an audit trail of accesses to the objects it protects. The audit data shall be protected by the TCB so that read access to it is limited to those who are authorized for audit data. The TCB shall be able to record the following types of events: use of identification and authentication mechanisms, introduction of objects into a user's address space (e.g., file open, program initiation), deletion of objects, and actions taken by computer operators and system administrators and/or system security officers, and other security relevant events. The TCB shall also be able to audit any override of human-readable output markings. For each recorded event, the audit record shall identify: date and time of the event, user, type of event, and success or failure of the event. For identification/ authentication events the origin of request (e.g., terminal ID) shall be included in the audit record. For events that introduce an object into a user's address space and for object deletion events the audit record shall include the name of the object and the object's security level. The ADP system administrator shall be able to selectively audit the actions of any one or more users based on individual identity and/or object security level. The TCB shall be able to audit the identified events that may be used in the exploitation of covert storage channels. The TCB shall contain a mechanism that is able to monitor the occurrence or accumulation of security auditable events that may indicate an imminent violation of security policy. This mechanism shall be able to immediately notify the security administrator when thresholds are exceeded, and, if the occurrence or accumulation of these security relevant events continues, the system shall take the least disruptive action to terminate the event.
System Architecture (A1): The TCB shall maintain a domain for its own execution that protects it from external interference or tampering (e.g., by modification of its code or data structures). The TCB shall maintain process isolation through the provision of distinct address spaces under its control. The TCB shall be internally structured into well-defined largely independent modules. It shall make effective use of available hardware to separate those elements that are protection-critical from those that are not. The TCB modules shall be designed such that the principle of least privilege is enforced. Features in hardware, such as segmentation, shall be used to support logically distinct storage objects with separate attributes (namely: readable, writeable). The user interface to the TCB shall be completely defined and all elements of the TCB identified. The TCB shall be designed and structured to use a complete, conceptually simple protection mechanism with precisely defined semantics. This mechanism shall play a central role in enforcing the internal structuring of the TCB and the system. The TCB shall incorporate significant use of layering, abstraction and data hiding. Significant system engineering shall be directed toward minimizing the complexity of the TCB and excluding from the TCB modules that are not protection-critical.
System Integrity (A1): Hardware and/or software features shall be provided that can be used to periodically validate the correct operation of the on-site hardware and firmware elements of the TCB.

Covert Channel Analysis (A1): The system developer shall conduct a thorough search for covert channels and make a determination (either by actual measurement or by engineering estimation) of the maximum bandwidth of each identified channel. (See the Covert Channels Guideline section.) Formal methods shall be used in the analysis.
Trusted Facility Management (A1): The TCB shall support separate operator and administrator functions. The functions performed in the role of a security administrator shall be identified. The ADP system administrative personnel shall only be able to perform security administrator functions after taking a distinct auditable action to assume the security administrator role on the ADP system. Non-security functions that can be performed in the security administration role shall be limited strictly to those essential to performing the security role effectively.
Trusted Recovery (A1): Procedures and/or mechanisms shall be provided to assure that, after an ADP system failure or other discontinuity, recovery without a protection compromise is obtained.
Security Testing (A1): The security mechanisms of the ADP system shall be tested and found to work as claimed in the system documentation. A team of individuals who thoroughly understand the specific implementation of the TCB shall subject its design documentation, source code, and object code to thorough analysis and testing. Their objectives shall be: to uncover all design and implementation flaws that would permit a subject external to the TCB to read, change, or delete data normally denied under the mandatory or discretionary security policy enforced by the TCB; as well as to assure that no subject (without authorization to do so) is able to cause the TCB to enter a state such that it is unable to respond to communications initiated by other users. The TCB shall be found resistant to penetration. All discovered flaws shall be corrected and the TCB retested to demonstrate that they have been eliminated and that new flaws have not been introduced. Testing shall demonstrate that theTCB implementation is consistent with the formal top- level specification. (See the Security Testing Guidelines.) No design flaws and no more than a few correctable implementation flaws may be found during testing and there shall be reasonable confidence that few remain. Manual or other mapping of the FTLS to the source code may form a basis for penetration testing.

Design Specification and Verification (A1): A formal model of the security policy supported by the TCB shall be maintained over the life-cycle of the ADP system that is proven consistent with its axioms. A descriptive top-level specification (DTLS) of the TCB shall be maintained that completely and accurately describes the TCB in terms of exceptions, error messages, and effects. A formal top-level specification (FTLS) of the TCB shall be maintained that accurately describes the TCB in terms of exceptions, error messages, and effects. The DTLS and FTLS shall include those components of the TCB that are implemented as hardware and/or firmware if their properties are visible at the TCB interface. The FTLS shall be shown to be an accurate description of the TCB interface. A convincing argument shall be given that the DTLS is consistent with the model and a combination of formal and informal techniques shall be used to show that the FTLS is consistent with the model. This verification evidence shall be consistent with that provided within the state-of-the-art of the particular computer security center-endorsed formal specification and verification system used. Manual or other mapping of the FTLS to the TCB source code shall be performed to provide evidence of correct implementation.
Configuration Management (A1): During the entire life-cycle, i.e., during the design, development, and maintenance of the TCB, a configuration management system shall be in place for all security- relevant hardware, firmware, and software that maintains control of changes to the formal model, the descriptive and formal top-level specifications, other design data, implementation documentation, source code, the running version of the object code, and test fixtures and documentation. The configuration management system shall assure a consistent mapping among all documentation and code associated with the current version of the TCB. Tools shall be provided for generation of a new version of the TCB from source code. Also available shall be tools, maintained under strict configuration control, for comparing a newly generated version with the previous TCB version in order to ascertain that only the intended changes have been made in the code that will actually be used as the new version of the TCB. A combination of technical, physical, and procedural safeguards shall be used to protect from unauthorized modification or destruction the master copy or copies of all material used to generate the TCB.
Trusted Distribution (A1): A trusted ADP system control and distribution facility shall be provided for maintaining the integrity of the mapping between the master data describing the current version of the TCB and the on-site master copy of thecode for the current version. Procedures (e.g., site security acceptance testing) shall exist for assuring that the TCb software, firmware, and hardware updates distributed to a customer are exactly as specified by the master copies.
Security Features User's Guide (A1): A single summary, chapter, or manual in user documentation shall describe the protection mechanisms provided by the TCB, guidelines on their use, and how they interact with one another.

Trusted Facility Manual (A1): A manual addressed to the ADP system administrator shall present cautions about functions and privileges that should be controlled when running a secure facility. The procedures for examining and maintaining the audit files as well as the detailed audit record structure for each type of audit event shall be given. The manual shall describe the operator and administrator functions related to security, to include changing the security characteristics of a user. It shallprovide guidelines on the consistent and effective use of the protection features of the system, how they interact, how to securely generate a new TCB, and facility procedures, warnings, and privileges that need to be controlled in order to operate the facility in a secure manner. The TCB modules that contain the reference validation mechanism shall be identified. The procedures for secure generation of a new TCB from source after modification of any modules in the TCB shall be described. It shall include the procedures to ensure that the system is initially started in a secure manner. Procedures shall also be included to resume secure system operation after any lapse in system operation.
Test Documentation (A1): The system developer shall provide to the evaluators a document that describes the test plan, test procedures that show how the security mechanisms were tested, and results of the security mechanisms' functional testing. It shall include results of testing the effectiveness of the methods used to reduce covertchannel bandwidths. The results of the mapping between the formal top-level specification and the TCB source code shall be given.
Design Documentation (A1): Documentation shall be available that provides a description of the manufacturer's philosophy of protection and an explanation of how this philosophy is translated into the TCB. The interfaces between the TCB modules shall be described. A formal description of the security policy model enforced by the TCB shall be available and proven that it is sufficient to enforce the security policy. The specific TCB protection mechanisms shall be identified and an explanation given to show that they satisfy the model. The descriptive top-level speci- fication (DTLS) shall be shown to be an accurate description of the TCB interface. Documentation shall describe how the TCB implements the reference monitor concept and give an explana- tion why it is tamper resistant, cannot be bypassed, and is correctly implemented. The TCB implementation (i.e., in hardware, firmware, and software) shall be informally shown to be consistent with the formal top-level specification (FTLS). The elements of the FTLS shall be shown, using informal techniques, to correspond to the elements of the TCB. Documentation shall describe how the TCB is structured tofacilitate testing and to enforce least privilege. This documentation shall also present the results of the covert channel analysis and the tradeoffs involved in restricting the channels. All auditable events that may be used in the exploitation of known covert storage channels shall be identified. The bandwidths of known covert storage channels, the use of which is not detectable by the auditing mechanisms, shall be provided. (See the Covert Channel Guideline section.) Hardware, firmware, and software mechanisms not dealt with in the FTLS but strictly internal to the TCB (e.g., mapping registers, direct memory access I/O) shall be clearly described.

Bibliography

1. Trappe, Wade and Washington, Lawrence, *Introduction to Cryptography with Coding Theory Second Edition*, NJ: Pearson Prentice Hall, 2006.

2. European Telecommunications Standards Institute. *Quantum Key Distribution (QKD); Use Cases*. ETSI GS QKD 008 v1.1.1, Sophia Antipolis Cedex – FRANCE, 2010

3. European Telecommunications Standards Institute. *Quantum Key Distribution (QKD); QKD Module Security Specification*. ETSI GS QKD 008 v1.1.1, Sophia Antipolis Cedex – FRANCE, 2010

4. European Telecommunications Standards Institute. *Quantum Key Distribution (QKD); Security Proofs*. ETSI GS QKD 008 v1.1.1, Sophia Antipolis Cedex – FRANCE, 2010

5. European Telecommunications Standards Institute. *Quantum Key Distribution (QKD); Components and Internal Interfaces*. ETSI GS QKD 008 v1.1.1, Sophia Antipolis Cedex – FRANCE, 2010

6. European Telecommunications Standards Institute. *Quantum Key Distribution (QKD); Application Interface*. ETSI GS QKD 008 v1.1.1, Sophia Antipolis Cedex – FRANCE, 2010

7. Bennet, Charles H., and Gilles Brassard. "Quantum Cryptography: Public Key Distribution and Coin Tossing," *Proceeding of the International Conference on Computers, Systems & Signal Processing*. 1984.

8. Bennet, Charles, Francois Bessette, Filles Brassard, Louis Salvail, and John Smolin, "Experimental Quantum Cryptography," *Journal of Cryptology*. (1992).

9. Gobby, C. Z.L. Yuan, and A. J. Shields. "Quantum Key Distribution Over 122 km of Standard Telecom Fiber," *Applied Physics Letters*, 84: 3762-3764 (2009).

10. Meyer-Scott, Evan, Zhizhong Yan, Allison MacDonald, Jean-Philippe Bourgoin, Hannes Hubel, and Thomas Jennwein. "How to Implement Decoy-state Quantum Key Distribution for a Satellite Uplink with 50-dB Channel Loss," *A Physical Review*, (2011).

11. Chapuran, T E, P Toliver, N A Peters, J Jackel, M S Goodman, R J Runser, S R McNown, N Dallmann, R J Hughes, K P McCabe, J E Nordholt, C G Peterson, K T Tyagi, L. Mercer, and H. Dardy. "Optical Networking for Quantum Key Distribution and Quantum Communications," *New Journal of Physics*, 11 (2009).

12. Nakassis, Tassos, J.C. Bienfang, P. Johnson, A. Mink, D. Rogers, X. Tang, and C.J. Williams. “Has Quantum Cryptography Been Proven Secure?” Quantum Information and Computation, 6244 (2006).

13. Wiechers, C, L Lydersen, C Wittmann, D Elser, K Skaar, Ch Marquardt, V, Makarov, and G Leuchs. “After-gate Attack on a Quantum Cryptosystem,” *New Journal of Physics,* 13 (January 2011).

14. Nauerth, Sebastian, Martin Furst, Tobias Schmitt-Manderbach, Henning Weier, and Harold Weinfurter. “Information Leakage via Side Channels in Freespace BB84 Quantum Cryptography,” *New Journal of Physics,* 11 (3 June 2009).

15. Bing, Qi, Chi-Hang Fred Fung, Hoi-Kwong Lo, and Xiongfeng Ma. “Time-shift Attack in Practical Quantum Cryptosystems,” *Quantum Information & Computation*, 7: 73-82 (2007).

16. Makarov, Vadim, Andrey Anisimov, and Johannes Skaar. “Effects of Detector Efficiency Mismatch on Security of Quantum Cryptosystems,” *A Physical Review*, 78 (31 June 2008).

17. Department of Defense. *Department of Defense Trusted Computer System Evaluation Criteria*. DoD 5200.28-STD. Washington, 15 August 1983.

18. The Defence Signals Directorate and others, *Common Criteria for Information Technology Security Evaluation*. CCMB-2009-07-001, July 2009

19. Information Technology Laboratory National Institute of Standards and Technology. *Security Requirements for Cryptographic* Modules. FIPS PUB 140-2, Gaithersburg, MD, 11 January 1994.

20. *Architecture and Principles of Systems Engineering*, C.E. Dickerson, D.N. Mavris, 2010 Taylor and Francis Group, LLC.

21. Department of Defense. *DoD Architecture Framework Version 2.0*. Washington: GPO, 28 May 2009.

22. United States Congress. *Clinger-Cohen Act of 1996*. Public Law No. 104-106, 104th Congress, 2nd Session. Washington: GPO, 1996.

23. United States Congress. *E-Government Act of 2002*. Title 44, Ch 36, 107th Congress, 2nd Session. Washington: GPO, 2002.

24. Office of Management and Budget. “Management of Federal Information Resources,” *Office of Management and Budget Circular,* A-130, (February 1996).

25. Federal Enterprise Architecture Program Management Office. *FEA Consolidated Reference Model Document Version 2.3*. Washington: Office of Management and Budget, October 2007.

26. Office of Management and Budget. *Improving Agency Performance and Using Information and Information Technology (Enterprise Architecture Assessment Framework v3.1).* Washington: Office of Management and Budget, June 2009.

27. United States General Accounting Office. *Information Technology A Framework for Assessing and Improving Enterprise Architecture Management (Version 1.1).* GAO-03-584, April 2003.

28. International Council on Systems Engineering (INCOSE). *Systems Engineering Handbook A Guide For System Life Cycle Processes and Activities*. INCOSE−TP−2003−002−03.2. San Diego, CA, January 2011.

29. Ghernaouti-Helie, Tashi, Langer, Monyk, "*Quantum Cryptography: An Innovation in the Domain of Secure Information Transmission*," (Sep 2008).

30. M. Wegman and L. Carter, "New Hash Functions and Their Use in Authentication and Set Equality," *J. Comp.Sys.Sci*, 22, 265-279, (1981).

31. Metrology for Industrial Quantum Communications. http://projects.npl.co.uk/MIQC/index.html. 15 February 2012.

33. Wiesner, S., "Conjugate Coding," *Sigact News*, Vol 15, no.1, 78-88, (1983).

34. Bennet, C. H., G. Brassard, S. Breidbart and S. Wiesner, "Quantum Cryptography or Unforgeable Subway Tokens," in *Advances in Cryptology: Proceedings of Crypto '82*, Plenum Press, 1982.

35. National Institute of Standards and Technology. "Rainbow Series." Security Publications. http://csrc.nist.gov/publications/secpubs/rainbow/. Retrieved October 2011

36. Busch, Carol, Paul S. De Maret, Teresa Flynn, Rachel Kellum, Sheri Le, Brad Meyers, Matt Saunders, Robert White, and Mike Palmquist. (2005). *Content Analysis.* Writing@CSU. Colorado State University Department of English. from http://writing.colostate.edu/guides/research/content/. Retrieved Mar 2012

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 074-0188

PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From – To)*
22 Mar 2012	Master's Thesis	20 Aug 2010 - 22 Mar 2012

4. TITLE AND SUBTITLE

Security Standards and Best Practice Considerations for Quantum Key Distribution (QKD)

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Harper, Carole, A., Captain, USAF

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S)

Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 Hobson Way, Building 640
WPAFB OH 45433-8865

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT/GSE/ENV/12-M05

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Dr. Gerry Baumgartner
Laboratory for Telecommunications Sciences
8080 Greenmead Drive
College Park MD 20740
(240) 373-2743

10. SPONSOR/MONITOR'S ACRONYM(S)

LTS

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

DISTRIBUTION STATEMENT A. APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

13. SUPPLEMENTARY NOTES

This material is declared work of the U.S. Government and is not subject to copyright protection in the United States.

14. ABSTRACT

QKD systems combine cryptographic primitives with quantum information theory to produce a theoretic unconditionally secure cryptographic key. However, real-world implementations of QKD systems are far from ideal and significantly differ from the theoretic model. Because of this, real-world QKD systems require additional practical considerations when implemented to achieve secure operations. In this thesis, a content analysis of the published literature is conducted to determine if established security and cryptographic standards and best practices are addressed in real world, practical QKD implementations. The research reveals that most published, real world QKD implementations do not take advantage of established security and cryptographic standards and best practices. Based upon an analysis of existing security and cryptographic standards and best practices, systems architecture methodology is used to make recommendations for how these standards can and should be applied to establish a practical, secure, QKD system framework.

15. SUBJECT TERMS

Quantum Key Distribution, Systems Engineering, Systems Architecture, Cryptography, Industry Technical Standards

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Dr. Michael Grimaila
U	U	U	UU	119	19b. TELEPHONE NUMBER *(Include area code)* (937) 255-3636, x 4800 (Michael.Grimaila@afit.edu)

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std. Z39-18