US Government Family of Protection Profiles

Public Key- Enabled Applications

(Individual PP Names are defined by the algorithm:

U.S. Government Basic Robustness PKE PP with <packages included in the PP, listed in the order in which they appear in the PP> **at** <Basic Robustness Assurance, EAL 3 with augmentation, or EAL 4 with augmentation, depending on the assurance level selected>

and are discussed further in the Foreword)

For

Basic Robustness Environments

Information
Assurance
Directorate

February 1, 2007

Version 2.77

Foreword

This family of PPs is written to support the development of a range of Public Key-Enabled applications and services that may be integrated into a computing platform. This family of PPs is written to address applications and products written for and used by the United States Government.

Given the range of applications to which it may be applied, the approach used in writing this family of PPs was to use the concept of "packages." A package, as defined by the CC, is an intermediate combination of functional or assurance components that define requirements that meet an identifiable set of security objectives. Packages may be thought of as sets of defined functionality requirements. All PKE applications are required to perform certain processes. Other processes may or may not be performed, depending upon the needs and functions of the application.

A set of IT Environment functional requirements was defined that must be met by the IT Environment of all PKE applications compliant with the PP. In addition, packages were defined that contain functionality that may or may not be included in a PKE application. The functionality contained in the packages is not "optional." Rather, the packages define additional PK functionality that may or may not be needed by an application (TOE). If a particular application (TOE) contains the functionality defined in a given package, then that package must be included in the ST for the TOE and the TOE must comply with the package requirements in full. In order to claim compliance with this PP, an ST must include at least one of the functional packages (other than the Audit Package) defined in this PP. An ST can claim conformance to this PP using Demonstrable Conformance technique. Demonstrable Conformance technique is specified in this PP.

In addition, this family of PPs contains three different Assurance Levels. The appropriate assurance level will be selected by the ST author depending upon the requirements of the application.

Thousands of possible PPs are included in this PP family, given the number of possible combinations of packages and the choice of assurance level. Rather than listing thousands of names, an algorithm was defined to generate the name of any given PP. The PP name is of the form:

> **U.S. Government Basic Robustness PKE PP with** <packages included in the PP, listed in the order in which they appear in the PP> **at** <Basic Robustness Assurance, EAL 3 with augmentation, or EAL 4 with augmentation, depending on the assurance level selected>

The words in bold print are included in every title and appropriate package names are listed for all of the packages included in the PP. Note that the list of packages in the title must be in the order in which they appear in this document in order to ensure consistency of naming.

Revision History

Version	Date	Description
0.1	March 15, 2002	Initial version of the Protection Profile
0.2	March 22, 2002	Draft version of the Protection Profile. Added threats, objectives, and requirements.
0.3	April 6, 2002	Added a new PP for PKI Based Entity Authentication Revised the PKI Credential Management package for additional security requirements Updated the Glossary to make the definitions more accurate Responded to comments received
1.0	April 30, 2002	Made editorial changes including cleaning up, adding references, expanding acronyms, adding acronyms list, etc. Revised the CC assurance requirements based on comments received. Provided for dependencies for cryptographic operations on explicitly stated requirements, as appropriate (e.g., FCS_COP.1 when a cryptographic operation is involved) Added optional trust anchor processing Added optional Audit functional package Revised Section 2 to provide a better approach to the reader regarding how to read the document. Added a new subsection "Approach" Revised threats and objectives based on comments received
1.1	May 28, 2002	Revised to explain assumption regarding the functional Vs. procedural aspects of path validation. Added an assumption regarding how the key recovery is out of scope Added the approach how multiple keys (e.g., due to key recovery, key history, re-key, etc.) are supported Made changes based on NIST comments, including: ▪ Added ALC_FLR.1 requirements ▪ Improved explanation of FIPS 140 series requirements in terms requiring it for all cryptographic modules ▪ Made the path validation packages incremental as opposed to self-contained ▪ Explained the path validation packages better ▪ Added rules for rejecting certificates, CRL, OCSP responses if critical extensions are not processed by the TOE ▪ Made minor revisions to the threats and objectives ▪ Made minor changes to SFR

Version	Date	Description
1.2	June 16, 2002	Responded to evaluator EORs and authors reviewed changes.
1.3	June 28, 2002	Responded to additional EORs.
1.3.1	July 1, 2002	Fixed type “envelop” to “envelope” in table 2.1
2.0	July 25, 2002	Responded to EORs.
2.1	August 1, 2002	Separated document into 28 PPs at the direction of NIAP.
2.2	September 22, 2002	Responded to decisions by NIAP regarding the use of packages.
2.3	September 26, 2002	Made minor updates to fix errors in previous version.
2.4	October 25, 2002	Responded to EORs: EOR_APE_DES.1-04, EOR_APE_DES.1-05, EOR_APE_DES.1.06, EOR_APE_ENV.1-02, EOR_APE_ENV.1-03, EOR_APE_OBJ.1-01, EOR_APE_OBJ.1-02, EOR_APE_OBJ.1-03, EOR_APE_REQ.1-06, EOR_APE_REQ.1-14, EOR_APE_REQ.1-15, EOR_APE_REQ.1-16, EOR_APE_REQ.1-17, EOR_Editorial_PP_v2-3
2.5	October 31, 2002	Responded to evaluator questions and revised the PP for editorial changes. No major material change was made.
2.6	October 31, 2003	Revised the PP based on ST development and evaluation. Clarified some of the requirements since the TOE does not perform base crypto operations. Also revised the PP to permit the ST author to define the combination of digital signature or non repudiation bits required for signature verification.
2.61	July 31, 2004	Made editorial changes to make this a U.S. Government PP and to make changes based on ST evaluation. Refined some of the PKE requirements.
2.62	August 31,2004	Added threats, objectives, requirements, and rationale for basic robustness consistency.
2.70	January 2, 2005	Moved base PP requirements to the IT Environment. Made significant changes to the basic threats, policies, assumptions, and objectives for the IT Environment. Allocated basic robustness threats, objectives, requirements, rationale, etc. to IT Environment. Revised the Audit Package to align with the basic robustness requirements. Added a statement requiring “Demonstrable” conformance. Added a statement requiring at least one of the packages other than Audit since there are no base TOE requirements. Added an element to the policy mapping package for compliance with X.509 Revised Entity Authentication Package since it no longer has iterated requirement. Revised continuous authentication package to remove the distinction between local and remote users. In summary, it is best to take a fresh look at the IT Environment, and

Version	Date	Description
		Audit.
2.71	January 23, 2005	Added an annex for confirming the IT Environment Requirements.
2.72	January 27, 2005	Replaced the term “extended requirement” with “explicitly stated requirement”; corrected some references; and added EXP to all explicitly stated requirements.
2.73	January 29, 2005	Put parentheses around EXP. Added rationale for selecting higher than basic robustness assurance. Added basic robustness assurance. Removed text for assurance components. This caused significant editorial changes throughout the document.
2.74	January 31, 2005	Made editorial changes per CCEVS request.
2.75	July 24, 2005	Revised based on the first evaluation by the CCTL.
2.76	November 20, 2006	Revised to provide guidance for testing requirements not claimed by an evaluated IT Environment (e.g., operating system)
2.77	February 1, 2007	Fixed the short form name of some of the requirements in Appendix E. Added mapping for OE.CORRECT_TSF_OPERATION to ATE assurance class. Removed a sentence requiring all of the IT Environment components to be CC validated; FIPS 140 validation is sufficient for cryptographic modules. Removed FDP_ITC_PKI_(EXP).1 from the IT Environment and associated assumption and objective. Removed audit requirements for FMT_MOF.1, FMT_MSA.1, FMT_MSA.3-NIAP-0429, FMT_MTD.1:1 through 5, FPT_TST_SOF_(EXP).1, FTA_SSL.1, FTA_SSL.2.

Table of Contents

Page

List of Tables

1 Introduction

This section contains document management and overview information. The Protection Profile (PP) Identification provides the labeling and descriptive information necessary to identify, catalogue, register, and cross-reference a PP or PP family. The PP Overview summarizes the profile in narrative form and provides sufficient information for a potential user to determine whether the PP family is of interest. The overview can also be used as a standalone abstract for PP catalogues and registers.

1.1 Identification

Title: Thousands of possible PPs are included in this PP family, given the number of possible combinations of packages and the choice of assurance level. Rather than listing the names, an algorithm was defined to generate the name of any given PP. The PP name is of the form:

> **U.S. Government Basic Robustness PKE PP with** <packages included in the PP, listed in the order in which they appear in the PP> **at** <Basic Robustness Assurance, EAL 3 with augmentation, or EAL 4 with augmentation, depending on the assurance selected>

The words in bold print are included in every title and appropriate package names are listed for all of the packages included in the PP. Note that the list of packages in the title must be in the order in which they appear in this document in order to ensure consistency of naming.

Assurance Level: This family of PPs includes Basic Robustness Assurance, Evaluation Assurance Level (EAL) 3 with Augmentation and EAL 4 with Augmentation. The functional requirements, objectives, threats, and assumptions are identical for each assurance level. The ST author will choose the appropriate assurance level depending upon the needs of the application. The Strength of Function (SOF) in all PPs is SOF Basic.

Version Number: Version 2.75

Date: July 24, 2005

PP Authors: Jean Petty and Swapna Katikaneni , CygnaCom Solutions, Inc; and Santosh Chokhani, Orion Security Solutions, Inc.

Sponsoring Organization: United States Marine Corps (USMC)

Registration: <To be filled in upon registration>

Keywords: Public Key Enabled (PKE), PKE, Public Key Infrastructure (PKI), PKI

1.2 Protection Profile Overview

This family of PPs describes the Information Technology (IT) security requirements for PKE Applications, based on the X.509 standard (see references below), integrated into computing platforms or systems. Public key technology provides digital signature generation and verification, public/private key encryption and decryption, public key distribution services, and various support functions. A PKE application may provide confidentiality, integrity, authentication, and non-repudiation, based on the use of public key technology security services. A variety of applications may be PK-enabled. This family of PPs should be used to specify the various PK services. Thousands of PPs can be defined depending upon the combination of functional packages and the assurance level chosen to meet the requirements of the application. Many functional requirements in the PPs represent extensions to the Common Criteria (CC), because the CC does not provide requirements for the X.509 processing rules that are critical to this family of PPs.

1.3 Related Documents

- Federal Information Processing Standard (FIPS) 196, Entity Authentication Using Public Key Cryptography, 18 February 1997
- International Organization for Standards/Internet Electrotechnical Committee (ISO/IEC) 9594-8:"Information Technology- Open Systems Interconnection-The Directory: Public Key and Attribute Certificate Frameworks" (X.509 Standard)
- X.509 Internet Public Key Infrastructure Certificate and CRL Profile, RFC 3280, April 2002
- X.509 Internet Public Key Infrastructure Online Certificate Status Protocol (OCSP), RFC 2560 June 1999.
- International Standard ISO/IEC 15408 Information technology — Security techniques — Evaluation criteria for IT security
- Common Methodology for Information Security Evaluation (CEM) Version 2.2, January 2004
- FIPS 140-2, Security Requirements for Cryptographic Modules, 25 May 2001 http://csrc.nist.gov/publications/fips/fips140-2/fips1402.pdf

1.4 PP Organization

This family of PPs includes thousands of possible PPs that can be used to specify a variety of PK services. Sections 2, 3, and 4 define TOE descriptions; assumptions, organizational security policies, and threats; and security objectives, respectively. The descriptions, threats, and security objectives are identified separately for each package defined. Section 5.1 contains the security functional requirements for the IT Environment. Section 5.2 contains the security functional requirements for all of the TOE packages. Sections 5.4 contains the three security assurance requirement packages, of which one must be selected by a PP/ST author. Rationale for selecting an assurance level is provided in Section 6.2.2.

Appendices provide supplemental information. References are provided in Appendix A. A glossary of PKI-related terms used in the protection profile (PP) is provided in Appendix B followed by a list of acronyms in Appendix C. Appendix D characterizes Basic Robustness. Appendix E elaborates on testing of requirement for the IT environment. Appendix F describes how "Demonstrable Conformance" is evaluated.

1.5 Common Criteria Conformance

This family of PPs has been built with Common Criteria (CC) Version 2.2 (ISO/IEC 15408 Evaluation Criteria for Information Technology Security; Part 1: Introduction and general model, Part 2: Security functional requirements, and Part 3: Security assurance requirements).

The PPs at Basic Robustness assurance level in this family of PPs are Common Criteria Version 2.2, Part 2 extended, and Part 3 conformant, at Evaluation Assurance Level 2 with augmentation. The PPs at assurance level EAL 3 augmented in this family of PPs are Common Criteria Version 2.2, Part 2 extended, and Part 3 conformant, at Evaluation Assurance Level 3 with Augmentation. The PPs at assurance level EAL 4 augmented in this family of PPs are Common Criteria Version 2.2, Part 2 extended, and Part 3 conformant, at Evaluation Assurance Level 4 with Augmentation.

1.6 PP Conformance

All of the following rules shall be followed when claiming conformance to a PP in this family of PPs:

- An ST may claim conformance to only one PP in this family of PPs;
- An ST may claim conformance only if meets the requirements of at least one functional package, other than the Audit Package, in this family of PPs;
- An ST may claim "demonstrable" conformance (as defined in Appendix F of this family of PPs) to any PP in this family of PPs, if the demonstrable conformance in evaluated in accordance with the requirements of Appendix F of this family of PPs; and

1.7 Basic Robustness Consistency

This family of PPs was designed to provide a general purpose tool to assess the security of PK enabled applications. Primary objective of this family of PPs is to eliminate the need for the specifiers, evaluators and validators to be crypto or PKI experts. This family of PPs was drafted with the "software only" TOE in mind. This family may be applied to toolkits, PK enabled applications and even TOEs that include hardware and system software such as operating system.

Given the nature of PK enabled applications and toolkits, many of the threats, objectives and requirements for Basic Robustness can only be applied to the IT Environment (generally meaning the underlying operating system) or the application integrating the TOE. Thus, the consistency guidance requirements have been applied to the IT Environment or the TOE, as appropriate. This approach ensures that the intent of Basic Robustness Consistency is maintained while maintaining this family of PPs as a useful tool rather than just another document.

2 TOE Description

2.1 Overview

An application is PK enabled if it:

- Securely manages private keys and trust anchors.
- Manages public key certificates.
- Uses one or more of the security services supported by a PKI by accepting and processing an X.509 public key certificate (also known as simply "certificate").
- Is able to obtain relevant certificates and revocation data.
- Checks each certificate for validity, using procedures described in the X.509 standard [ISO 9594-8], prior to reliance, including checking for revocation.
- Has access to accurate and reliable time in order to verify the dates on certificates, revocation data, and application data.
- Correctly interoperates with an appropriate cryptographic token.
- Collects, stores and maintains the data required to support digital signature verification in the future.
- Is able to automatically select from multiple private decryption keys if it performs public key based decryption.

2.2 Approach

This section defines the approach that was taken in developing this family of PPs. This document does not provide background information on the CC, PKI, PKE, or cryptography. The reader is assumed to possess appropriate knowledge of the CC, PKI, cryptography and related technology to understand the content of this document. There are, however, many ways to develop a PP and to address the subject matter of this family of PPs. This section provides specifics on the development approach used.

2.2.1 Package concept

This PP family provides a tool to specify and evaluate a broad range of PKE applications. Given the range of applications to which it may be applied, the approach used in writing this PP family was to use the concept of “packages.” A package, as defined by the CC, is an intermediate combination of functional or assurance components that define requirements that meet an identifiable set of security objectives. All PKE applications are required to perform certain processes. Other processes may or may not be performed, depending upon the needs and functions of the application.

A base set of functional requirements was defined that must be met by the IT Environment of all PKE applications compliant with any PP in this family of PPs. Packages were defined that contain functionality that may or may not be included in a PKE application. The functionality contained in the packages is not “optional.” Rather, the packages define additional PK functionality that may or may not be needed by an application (TOE). If a particular application (TOE) contains the functionality defined in a given package, then that package must be included in the ST for the TOE and the TOE must comply with the package requirements in full. Thousands of possible PPs are included in this PP family, given the number of possible combinations of packages and the choice of assurance level. Rather than listing thousands of names, an algorithm was defined to generate the name of any given PP. The PP name is of the form:

> **U.S. Government Basic Robustness PKE PP with** <packages included in the PP, listed in the order in which they appear in the PP> **at** <Basic Robustness Assurance, EAL 3 with augmentation, or EAL 4 with augmentation, depending on the assurance level selected>

The words in bold print are included in every title and appropriate package names are listed for all of the packages included in the PP. Note that the list of packages in the title must be in the order in which they appear in this document in order to ensure consistency of naming. Also, when specifying a PP, only one PP from this family should be specified, i.e., the PP with the largest number of packages. The ST author should not attempt to specify all of the possible PPs represented (which would include every possible combination of packages in the document). Instead, the ST author should name only the most comprehensive PP represented by the document.

The TOE functional requirements for the packages are defined in Section 5.2 and IT Environment requirements are defined in Section 5.1; appropriate assumptions, threats, and objectives are defined for the TOE requirements and for the IT Environment requirements in Sections 3 and 4. The applicable TOE requirements, IT Environment requirements and associated assumptions, threats, and objectives must be included in every ST compliant with a PP in this family of PPs.

Each package represents a discrete set of threats, objectives, and requirements. The packages are named and their corresponding threats, objectives and functional requirements are identified in separate subsections within Sections 3.5, 4.2, and 5.2. When a package is included in an ST, all of the components of the package must be included, i.e., all of the threats, objective, requirements, and rationale. The ST author is expected to maintain the modularity of the packages in the ST, since this will enhance the ability to evaluate PP conformance and the ability to evaluate a TOE in a modular fashion.

The packages define a subset of X.509 certificate and revocation processing capabilities as defined in the ISO and Internet Engineering Task Force (IETF) standards. Some of the examples of these various capabilities include:

- Ability to process certificatePolicies extension
- Ability to process all certificate policies related extensions
- Ability to process name constraints extension
- Ability to handle the various public key algorithms (e.g., Rivest, Shamir, Adelman (RSA); Digital Signature Algorithm (DSA); Diffie Hellman (DH); Elliptic Curve Diffie Hellman (ECDH); etc.)
- Ability to handle a variety of public key based mechanisms (e.g., signature generation, signature verification, encryption, decryption, entity authentication, etc.)

The packages provide the granularity for the above listed capabilities. The ST author is further provided a high degree of flexibility by the use of selections and assignments for the various security functional requirements.

2.2.2 Part 2 and Explicitly Stated Security Functional Requirements

Using Part 2 of the CC as the tool for specifying security relevant requirements, this family of PPs addresses only the security aspects of PK enablement. For example, the PP does not deal with mechanisms of how the certificates and Certificate Revocation Lists (CRLs) are obtained since the security of certificates and CRLs does not depend on "from where" or "how" they were obtained; their security is ensured through verification of digital signatures.

In the area of certification path validation, requirements are defined that are compliant with both the ISO X.509 and IETF PKIX Request for Comment (RFC) 3280. However, the certification path validation in these standards is procedural. In order to make the PP implementation neutral, certification path validation requirements are specified using non-procedural techniques.

CC access control related components are not appropriate to express the certificate and revocation information (e.g., Certificate Revocation List (CRL), OCSP response, etc.) processing requirements and hence explicitly stated requirements were used to address the processing of certificates and revocation information.

2.2.3 Technical Approach for PKI requirements

This subsection describes the technical approach used in selecting and developing the PKI requirements.

The certification path validation requirements were developed with meaningful names for the components to define X.509 input, processing, and output segments. Certificate policy calculation is included in the output components.

An analysis of X.509 certificate processing revealed that a set of processing rules are applied to all the certificates and some additional rules are applied to intermediate (i.e., CA) certificates. Thus, basic certificate processing and intermediate certificate processing components have been established.

Neither X.509 nor PKIX require any trust anchor processing rules. However, to provide a tool that can be used to specify rules for trust anchor processing when trust anchor is in the form of a self-signed certificate, trust anchor processing rules (including “none”) may be defined by the ST author as a part of the path validation initialisation.

The cryptographic operations that require the use of a public key must use the public key, public key parameters (if applicable) and subscriber identifying information from certification path validation in order to preserve the security. Functional packages for the various cryptographic operations have been developed to specify this linkage.

This PP family provides functional requirements for processing all of the certificate extensions and for complete certification path validation. While this PP family provides the ability to evaluate PKE applications (TOEs) that perform full X.509 path validation, it also provides the flexibility to evaluate applications (TOEs) that perform minimal to no policy and other extensions processing.

This family of PPs provides the capability to select public key cryptography algorithms since a PKI may use a variety of cryptographic algorithms. Packages for the public key cryptography algorithms are provided so that this family of PPs need not be revised to accommodate the various cryptographic algorithms.

The scope of this family of PPs excludes key recovery infrastructure-related functions since key recovery is an infrastructure function as opposed to a PKI application function. The ability to deal with multiple keys using the key identifier is addressed in appropriate locations in certification path validation output and in cryptographic operations. The PKE application could have multiple keys due a variety of reasons such as key recovery, key history and re-key.

This PP family specifies the following IT Environment requirements based on Basic Robustness consistency and general security principles:

- TSF Self-protection and isolation requirements;
- Identification and authentication requirements;
- Access control requirements;
- Audit requirements[1];
- The residual information protection for private and secret keys, which will be satisfied by a FIPS validated cryptographic module since the FIPS validated cryptographic module must provide for plaintext private keys and plaintext secret keys to be zeroized.

The following features are deferred for future revisions of this family of PPs:

- Processing partitioned CRLs
- Processing delta CRLs
- Processing indirect CRLs
- Processing server based validation responses, such as Simple Certificate Validation Protocol (SCVP), OCSP Version 2, etc.

2.2.4 Specifying and Evaluating a PP or Compliant ST from this PP Family

When several PPs can be constructed using some or any combination of component packages, it is desirable to minimize the number of evaluations, e.g., in the case of this PP family, thousands of evaluations would be required to evaluate separately every possible PP that can be specified. To illustrate, if there are n packages, there are $2^n - 1$ PPs. Clearly, even for a small number of packages, it becomes a very large number of possible PPs. In naming a PP or specifying compliance with a PP, the author must use the naming convention defined in the Foreword and repeated in Section 2.2.1. In particular, packages listed in the title must be specified in the order in which they occur in the PP and only one PP from this family, the most comprehensive PP, should be specified, i.e., the PP with the largest number of packages of interest and applicable to the ST author. The ST author should not attempt to specify compliance with all of the possible PPs in the PP family to which compliance might be claimed, instead, compliance should be claimed only for the most comprehensive PP.

When claiming compliance with a particular PP, it is sufficient for an ST to identify any PP in this PP family by simply naming the PP. This is sufficient because the name of the

[1] Some of the audit requirements must be satisfied by the TOE as described in the Audit Package.

PP clearly identifies all of the packages contained in the PP and the assurance level. The ST evaluator can then evaluate compliance with the PP by examining the ST and its compliance with the PP packages and assurance level identified in the title.

The approach used for this family of PPs, during the PP family evaluation, is to evaluate each package once, to evaluate inter-relationships among all packages once, and then to be confident about the validity of any PPs derived from this PP family. A PP derived from this family is considered to have passed the evaluation without any further work because in this PP family:

  - – The packages are constructed with constraints as described below under Section 2.2.4.1, Constraints,
  - – Each package is evaluated per the CEM; and
  - – The packages go through the additional evaluation work units during PP family evaluation described below under Section 2.2.4.3, Additional Evaluation Work Units.

- ▪ A unique name is generated for the PP using the algorithm described in Section 2.2.1.
- ▪ An ETR is produced during the PP family evaluation that is valid for all PPs derived from this family because the ETR covers all of the packages. Note that in the case of this PP family, multiple ETRs may be required: one for each assurance level.

2.2.4.1 Constraints

The following constraints were met in the development of this PP family:

1. Each package is complete, i.e., each package contains a name, TOE Description, threats, organization security policy (if applicable), secure usage assumptions (if applicable), security objectives for the TOE (if applicable), security objectives for the environment (if applicable), security functional requirements for the TOE (if applicable), IT security functional requirements for the environment (if applicable), non-IT security functional requirements for the environment (if applicable), security assurance requirements, security objectives rationale, security requirements rationale, dependencies rationale, and strength of function rationale. In other words, the package has all of the components of a PP.
2. A dependency rationale points to other packages to satisfy some of the requirements. Note that dependencies are specifically identified for packages both in Section 2.3 and in Section 5.2 of this document. Also, the requirement that packages dependencies must be included and achieve transitive closure is stated both in Section 2.3 and in Section 5.1 of this document.
3. Some material is included in a package by reference. For example, if assurance requirements and strength of function requirements are common to some or all packages, it is sufficient to include them only once as long as it is clear which packages are applicable.
4. From the TOE description, it is obvious that the security functionality provided by each package is different from functionality provided by other packages under evaluation.

5. The threats for each package are different from the threats for other packages. This means:
    a. A threat name appears in only one package, and
    b. Each threat description is distinct.
6. The objectives for each package are different from the objectives for other packages. This means:
    a. An objective name appears in only one package, and
    b. Each objective description is distinct.
7. The security functional requirements and security assurance requirements for all of the packages have the same label if and only if they are identical.
8. The authors describe the algorithm for naming the various composite PPs and show that they result in unique name for each possible composite PP.

2.2.4.2 Evaluating this PP family

In order to evaluate this family of PPs, the evaluator must do the following:

- The evaluator must evaluate the packages to verify that the assertions made in the previous section hold true.
- The evaluator must ensure that combining the packages will continue to be safe.
- The evaluator must verify that all the constraints listed above are satisfied by the packages.

A high-level methodology to perform this evaluation is described below.

For constraint items 1, 2, and 3 listed in Section 2.2.4.1, validation of these items falls naturally out of the evaluation of each package, as if that package or component were being evaluated in a normal PP evaluation. Thus, if each package passes the evaluation, items 1, 2, and 3 are satisfied.

For constraint item 4, the evaluator should compare the functions performed by the various packages. The functions must be distinct. The functions may be distinct in terms of one or more of the following:

- Security capability; or
- Security services; or
- Data to which the security capability and/or service applies.

Constraint items 5,6, and 7 can be executed using current CEM work units by treating the packages as if they are combined into a single composite PP. By analyzing all of the threats, objectives, and requirements at once, as if they were all contained in a single PP, any interactions or overlap between them can be identified.

For constraint item 8, the evaluator shall examine the composite PP and verify that the composite PP naming scheme will provide unique and unambiguous names. To perform this work unit, the evaluator will analyze the algorithm to make sure that the name clearly implies the packages that are either included, excluded or both. The evaluator shall also take some sample cases and see that each case results in a unique, meaningful and unambiguous name.

2.2.4.3 Additional Evaluation Work Units for this PP Family

The following additional work units must be carried out to ensure that when the packages are combined, the evaluation will continue to be valid.

1. The evaluator shall verify that the security objectives for the TOE and security objectives for the Environment do not conflict. The evaluator shall look at all the objectives for the packages and/or components collectively and apply the methodology used for APE_OBJ.1-9 to ensure that the objectives do not conflict.
2. The evaluator shall verify that the IT security requirements do not conflict. The evaluator shall look at all the IT security requirements for the packages and/or components collectively and apply the methodology used for APE_REQ.1-22 to ensure that the IT security requirements do not conflict.
3. If the same requirement appears in more than one package, it applies to mutually exclusive scope, e.g., to different data.
4. The evaluator shall examine the packages to ensure that either the iterations of the same component are properly applied or there is sufficiently detailed guidance provided to the ST author in order to uniquely and unambiguously label each iterated component.

2.2.4.4 Evaluating IT Environment for TOEs

Appendix E contains guidance to the TOE evaluators regarding the steps that must be followed in order to confirm that the IT Environment satisfies the IT Environment security requirements specified in the STs claiming compliance with a PP in this family of PPs.

2.3 Definition of TOE

The TOE and TSF boundaries will be defined by the ST author and will address what functionality is included in the TOE and what is included in the IT Environment. Some or all of the requirements allocated to the IT Environment may be satisfied by the TOE.

All of the PPs in this family assume that the IT Environment includes one or more cryptographic module(s) that are all validated at FIPS 140 series Level 1 or greater. This FIPS 140 series validated module or modules will perform one or more of the following: key pair generation, digital signature generation and verification, encryption, decryption, secure hash, random number generation, Hash based Message Authentication Code (HMAC) and/or other required cryptographic functions. Note that the TOE environment may contain more that one cryptographic module so that some functions, such as key pair generation, may be performed in a hardware cryptographic module, while others, such as secure hash, may be performed in a software module. Generally, private key operations will be performed in the hardware cryptographic module and public key and symmetric key operations will be performed either in the hardware or the software cryptographic module.

This PP family also assumes that certificates and status message, i.e., CRLs or OCSP responses are available as part of the PKI service.

For all of the PPs in this PP family, TOE user data is defined as any data that is encrypted, decrypted, signed, verified, imported or exported by the user. TOE user data may also include the user's cryptographic keys. The ST author will provide a specific definition of user data, depending upon the TOE.

For all of the PPs in this PP family, TSF data is defined as identification and authentication data, private keys owned by the system, security attributes and other data as defined by the ST author. Note that if the IT Environment performs the identification and authentication function or other security functions, then the associated data is not considered to be TSF data, since it is not within the TOE boundary.

This PP family defines a set of security requirements to be levied on TOEs. A TOE may be a stand-alone system or consist of components in a network or a distributed environment. The TOE may be a toolkit or may consist of an application running on one or more processors and associated peripherals and storage devices to be used by multiple users to perform a variety of PKI functions requiring controlled, shared access to the information stored on the system. The ST author will provide a specific definition of the TOE.

All of the PPs in this PP family contain a set of requirements for the IT Environment. These requirements are used to specify the ability to manage multiple private keys, associated certificates, and identifying data and associations among them. The term "manage" means the ability to do one or more of the following: generate, destroy, delete, use, import, export, modify, etc. The identifying data and association between private key and public key certificates are useful in selecting the appropriate cryptographic keys for cryptographic operations and for PKCS-7 type information generation. The IT Environment also maintains secure storage of trust anchors.

Table 2.1 provides a summary of the functionality contained in the packages included in this PP family. The following subsections describe the functionality of the packages. Note that each of the packages described in the following subsections have an assurance level of Basic Robustness, EAL 3 augmented, or EAL 4 augmented.

Note that some packages have dependencies on other packages, i.e., when a package with dependencies is included in a PP, the dependent package(s) must also be included in their entirety. A valid PP must contain all dependencies defined for packages in the PP. A summary of package dependencies is as follows:

- Certification Path Validation – Basic Package is a dependency of the following other packages, i.e., when the following packages are included in a PP, the Certification Path Validation – Basic Package must also be included in the PP:
  - Certification Path Validation – Basic Policy Package
  - Certification Path Validation – Policy Mapping Package
  - Certification Path Validation – Name Constraints Package
  - PKI Encryption using Key Transfer Algorithms
  - PKI Encryption using Key Agreement Algorithms
  - PKI Decryption using Key Agreement Algorithms
  - PKI Signature Verification
  - PKI Based Entity Authentication
  - Continuous Authentication
- Certification Path Validation – Basic Policy is a dependency of Certification Path Validation – Policy Mapping

- PKI Based Entity Authentication package is a dependency of Continuous Authentication Package

Table 2.1 lists any dependent packages for each of the packages included in this PP family. Note that if a package with dependencies is included in a PP or ST, then the dependency package(s) must also be included in the PP.

Table 2.1 – Summary of Packages

Package Name	Functionality	Dependency
Certification Path Validation (CPV) – Basic	Perform all X.509 validation checks except policy processing and name constraints processing	None
CPV – Basic Policy	Process certificatePolicies extension	CPV – Basic
CPV – Policy Mapping	Process policy mapping related extensions: policyMapping, policyConstraints, and inhibitAnyPolicy	CPV – Basic, CPV – Basic Policy
CPV – Name Constraints	Process nameConstraints extension	CPV – Basic
PKI Signature Generation	Use private key for signature generation Generate the signature information (e.g., Public Key Cryptography Standard 7 (PKCS 7) blob)	None
PKI Signature Verification	Process the signature information (e.g., PKCS 7 blob) Use public key to verify signature	CPV – Basic
PKI Encryption using Key Transfer Algorithms	Generate the encryption envelope information (e.g., PKCS 7 blob) Use public key for encryption	CPV – Basic
PKI Encryption using Key Agreement Algorithms	Generate the key agreement envelope information (e.g., PKCS 7 blob) Use decryptor public key for key agreement Use encryptor private key for key agreement	CPV – Basic
PKI Decryption using Key Transfer Algorithms	Process encryption envelope information (e.g., PKCS 7 blob) Use private key for decryption	None
PKI Decryption using Key Agreement Algorithms	Process the key agreement envelope information (e.g., PKCS 7 blob) Use encryptor public key for key agreement Use decryptor private key for key agreement	CPV – Basic
PKI Based Entity Authentication	Carry out the “assigned” authentication protocol(s) Use public key for authentication	CPV – Basic

Package Name	Functionality	Dependency
Online Certificate Status Protocol Client	Generate OCSP request in accordance with RFC 2560 Process OCSP response	None
Certificate Revocation List (CRL) Validation	Obtain CRL Process CRL	None
Audit	Generate Audit Log	None
Continuous Authentication	Perform Continuous Authentication	PKI Based Entity Authentication, CPV - Basic

2.3.1 Certification Path Validation – Basic Package

The Certification Path Validation – Basic Package (CPV – Basic) provides for all X.509 validation checks except policy processing and name constraints processing. The functionality in this package is the same regardless of the assurance level. This package addresses certification path development and certification path validation. The most likely implementation consists of developing a path (using a variety of techniques) and then validating the certification path. Certification path validation generally consists of validating certificates starting with the one certified by a trust anchor and ending with the one issued to the subscriber of interest. However, in order to be implementation neutral, this package does not mandate any ordering of certification path development and certification validation processes. A compliant implementation will only need to meet the security requirements specified in this package.

All processing defined is X.509 and PKIX compliant.

Public key certificates in a certification path can be categorized in three types for the purpose of certification path validation:

- Self-signed certificates: The trust anchors can be in the form of self-signed certificates. The trust anchor is used to obtain the Distinguished Name (DN), public key, algorithm identifier, and the public key parameters (if applicable). This package permits validation of trust anchor if it is in the form of a self-signed certificate, including validating signature and verifying that the self-signed certificate validity period has not expired.
- Intermediate certificates: These are the certificates issued to the CAs. All certificates in a certification path are intermediate certificates, except the last one.
- End certificate: This is the last certificate in the certification path and is issued to the subscriber of interest. This is typically an end-entity (i.e., not a CA) certificate. However, this package permits this certificate to be a CA certificate also.

This package includes processes for the following security related certificate extensions checks: no-check, keyUsage, extendedKeyUsage, and basicConstraints.

This PP family provides a capability to validate path as of the time of interest (TOI), which can be current time or earlier. It is assumed that the IT environment can provide certificates and revocation information (i.e., OCSP responses and/or CRL) for the TOI.

2.3.2 Certification Path Validation – Basic Policy Package

The Certification Path Validation – Basic Policy package is dependent on the CPV – Basic package. The functionality in this package is the processing of certificatePolicies extension. The functionality in this package is the same regardless of the assurance level.

2.3.3 Certification Path Validation – Policy Mapping Package

The Certification Path Validation – Policy Mapping package is dependent on the CPV – Basic Policy and the CPV – Basic packages. The functionality in this package is the processing of the following certificate policies related extension: policyMapping, inhibitAnyPolicy, and policyConstraints. The functionality in this package is the same regardless of the assurance level.

2.3.4 Certification Path Validation – Name Constraints Package

The Certification Path Validation – Name Constraints is dependent on the CPV – Basic package. The functionality in this package is the processing of nameConstraints extension. The functionality in this package is the same regardless of the assurance level.

2.3.5 PKI Signature Generation Package

The PKI Signature Generation package contains the following functionality:

- Select the appropriate private key;
- Invoke a signature generation function using the selected private key; and
- Generate and include signature information that identifies the signer and is useful in efficient signature verification.

The functionality in this package is the same regardless of the assurance level.

2.3.6 PKI Signature Verification Package

The PKI Signature Verification package is dependent on the CPV – Basic package. This package contains the following functionality:

- Process the signature information, e.g. the PKCS 7 blob;
- Invoke a signature verification function with the public key obtained from certification path validation; and
- Verify the signature information.

The functionality in this package is the same regardless of the assurance level.

2.3.7 PKI Encryption using Key Transfer Algorithms Package

The PKI Encryption using Key Transfer Algorithms package is dependent on the CPV – Basic package. The functionality of this package is to:

- Invoke a public key encryption function using a key transfer algorithm such as RSA; using a public key obtained from certification path validation.
- Generate and include additional information useful in efficient decryption.

The functionality in this package is the same regardless of the assurance level.

2.3.8 PKI Encryption using Key Agreement Algorithms Package

The PKI Encryption using Key Agreement Algorithms package is dependent on the CPV – Basic package. This package contains the following functionality:

- Invoke a public key encryption function using a key agreement algorithm such as DH or ECDH using:
    - Public key obtained from certification path validation; and
    - Appropriate private key
- Generate and include additional information useful in efficient decryption.

The functionality in this package is the same regardless of the assurance level.

2.3.9 PKI Decryption using Key Transfer Algorithms Package

The PKI Decryption using Key Transfer Algorithms package contains the following functionality:

- Process the encrypted information;
- Select the appropriate private key for decryption; and
- Invoke a public key decryption function using a key transfer algorithm such as RSA.

The functionality in this package is the same regardless of the assurance level. Since only the decrypting party's private key is used, this package does not require certificate path processing functionality.

2.3.10 PKI Decryption using Key Agreement Algorithms Package

The PKI Decryption using Key Agreement Algorithms package is dependent on the CPV – Basic package. This package contains the following functionality:

- Process the encrypted information;
- Verify the encryptor;
- Invoke a public key decryption function using a key agreement algorithm such as DH or ECDH using:
    - Public key obtained from certification path validation; and
    - Appropriate private key

The functionality in this package is the same regardless of the assurance level.

2.3.11 PKI Based Entity Authentication Package

The PKI Based Entity Authentication is dependent on the CPV – Basic package and allows PKI to be used for an entity authentication service. This package allows the ST author to select a PKI based entity authentication standard for identification and authentication of an entity. This package shall be used for initial authentication of the entity. The functionality in this package is the same regardless of the assurance level.

2.3.12 Online Certificate Status Protocol Client Package

The Online Certificate Status Protocol Client package allows the TOE to make Online Certificate Status Protocol (OSCP) requests and to validate OCSP responses. This package permits the use of the OCSP Responder as a trust anchor, as the CA, or an end entity authorized to sign OCSP responses. The ST author can assign additional rules to process OCSP extensions. If the OCSP implementation establishes trust in the OCSP responder by performing Certification Path Validation, then the CPV – Basic package may be used in combination with this package. The functionality in this package is the same regardless of the assurance level.

2.3.13 Certificate Revocation List (CRL) Validation Package

The Certificate Revocation List (CRL) Validation package allows the TOE to validate a CRL. This version of this package does not require processing of a CRL issuing distribution point (IDP) CRL or a delta CRL. Future versions may include that capability by codifying Annex B of X.509 standard.

It should be noted that this package may be used to process a CRL that is pointed to by a CRL Distribution Point (CRLDP) extension in a certificate as long as the CRL is a full CRL, indicated by the absence of IDP and deltaCRLIndicator extensions.

This package permits the use of the same public key for CRL signature verification as the one used for verifying the signature on the certificate, but does not mandate it. In other words, a compliant implementation can use that or develop a certification path. If the compliant implementation develops a certification path, then the CPV – Basic package may be used in combination with this package. The functionality in this package is the same regardless of the assurance level.

2.3.14 Audit Package

The Audit package generates PKE related audit events relevant to the TOE. Examples of audit events are:

- Signature verification success, date and time, and policies under which signatures were valid
- Signature verification failure, date and time, cause of failure (signature on the object failed, certification path failure, policy failure, etc.)
- User override events (CRL availability, accept policy failure, accept null policy, accept other policy, etc.)

The functionality in this package is the same regardless of the assurance level.

2.3.15 Continuous Authentication Package

This package is dependent on PKI Based Entity Authentication and the CPV – Basic packages. This package is used for continuous authentication of the protocol, command, packets etc. The functionality in this package is the same regardless of the assurance level.

2.4 Assurance Requirements

There are three assurance levels included in this family of PPs: Basic Robustness Assurance Level which is EAL 2 with augmentation; EAL 3 with augmentation; and EAL 4 with augmentation. Although higher EALs increase assurance, none meet the requirements of medium robustness; therefore all PKE PP assurance requirements are considered basic robustness. The ST author will determine the appropriate assurance requirements, based on application requirements.

3 TOE Security Environment

3.1 Relationship between Basic Robustness Level and the formation of applicable assumptions, threats and the policies of the TSE

Basic robustness TOEs falls in the upper left area of the robustness figures discussed in Appendix D. A Basic Robustness TOE is considered sufficient for low threat environments or where compromise of protected information will not have a significant impact on mission objectives. This implies that the motivation of the threat agents will be low in environments that are suitable for TOEs of this robustness. In general, basic robustness results in “good commercial practices” that counter threats based in casual and accidental disclosure or compromise of data protected by the TOE.

Threat agent motivation can be considered in a variety of ways. One possibility is that the value of the data process or protected by the TOE will generally be seen as of little value to the adversary (i.e., compromise will have little or no impact on mission objectives). Another possibility, (where higher value data is processed or protected by the TOE) is that procuring organizations will provide other controls or safeguards (i.e., controls that the TOE itself does not enforce) in the fielded system in order to increase the threat agent motivation level for compromise beyond a level of what is considered reasonable or expected to be applied.

3.2 Secure Usage Assumptions for all PPs in this PP family

Table 3.1 lists the Secure Usage Assumptions for the IT environment. These assumptions for the IT environment are included in every PP in this PP family.

Table 3.1 – Assumptions for the IT Environment

Assumption Name	Description
A.Configuration	The TOE will be properly installed and configured.
A.Low	The attack potential on the TOE is assumed to be low.
A.NO_EVIL	Administrators are non-hostile, appropriately trained and follow all administrator guidance.
A.PHYSICAL	It is assumed that the environment provides the TOE with appropriate physical security, commensurate with the value of the IT assets protected by the TOE.

3.3 Threat Agent Characterization

In addition to helping define the robustness appropriate for a given environment, the threat agent is a key component of the formal threat statements in the PP. Threat agents are typically characterized by a number of factors such as *expertise*, *available resources*, and *motivation*. Because each robustness level is associated with a variety of environments, there are corresponding varieties of specific threat agents (that is, the threat agents will have different combinations of motivation, expertise, and available resources) that are valid for a given level of robustness. The following discussion explores the impact of each of the threat agent factors on the ability of the TOE to protect itself (that is, the robustness required of the TOE).

The *motivation* of the threat agent seems to be the primary factor of the three characteristics of threat agents outlined above. Given the same expertise and set of resources, an attacker with low motivation may not be as likely to attempt to compromise the TOE. For example, an entity with no authorization to low value data none-the-less has low motivation to compromise the data; thus a basic robustness TOE should offer sufficient protection. Likewise, the fully authorized user with access to highly valued data similarly has low motivation to attempt to compromise the data, thus again a basic robustness TOE should be sufficient.

Unlike the motivation factor, however, the same can't be said for *expertise*. A threat agent with low motivation and low expertise is just as unlikely to attempt to compromise a TOE as an attacker with low motivation and high expertise; this is because the attacker with high expertise does not have the motivation to compromise the TOE even though they may have the expertise to do so. The same argument can be made for *resources* as well

Therefore, when assessing the robustness needed for a TOE, the motivation of threat agents should be considered a “high water mark”. That is, the robustness of the TOE should increase as the motivation of the threat agents increases.

Having said that, the relationship between expertise and resources is somewhat more complicated. In general, if resources include factors other than just raw processing power (money, for example), then expertise should be considered to be at the same “level” (low, medium, high, for example) as the resources because money can be used to purchase expertise. Expertise in some ways is different, because expertise in and of itself does not automatically procure resources. However, it may be plausible that someone with high expertise can procure the requisite amount of resources by virtue of that expertise (for example, hacking into a bank to obtain money in order to obtain other resources).

It may not make sense to distinguish between these two factors; in general, it appears that the only effect these may have is to lower the robustness requirements. For instance, suppose an organization determines that, because of the value of the resources processed by the TOE and the trustworthiness of the entities that can access the TOE, the motivation of those entities would be “medium”. This normally indicates that a medium robustness TOE would be required because the likelihood that those entities would attempt to compromise the TOE to get at those resources is in the “medium” range. However, now suppose the organization determines that the entities (threat agents) that are the least trustworthy have no resources and are unsophisticated. In this case, even though those threat agents have medium motivation, the likelihood that they would be able to mount a successful attack on the TOE would be low, and so a basic robustness TOE may be sufficient to counter that threat.

It should be clear from this discussion that there is no “cookbook” or mathematical answer to the question of how to specify exactly the level of motivation, the amount of resources, and the degree of expertise for a threat agent so that the robustness level of TOEs facing those threat agents can be rigorously determined. However, an organization can look at combinations of these factors and obtain a good understanding of the likelihood of a successful attack being attempted against the TOE. Each organization wishing to procure a TOE must look at the threat factors applicable to their environment; discuss the issues raised in the previous paragraph; consult with appropriate accreditation authorities for input; and document their decision regarding likely threat agents in their environment.

The important general points we can make are:

- The motivation for the threat agent defines the upper bound with respect to the level of robustness required for the TOE
- A threat agent’s expertise and/or resources that is “lower” than the threat agent’s motivation (e.g., a threat agent with high motivation but little expertise and few resources) may lessen the robustness requirements for the TOE (see next point, however).
- The availability of attacks associated with high expertise and/or high availability of resources (for example, via the Internet or “hacker chat rooms”) introduces a problem when trying to define the expertise of, or resources available to, a threat agent.

3.4 Threats to Security for all PPs in this PP Family

This subsection defines the base threats to the TOE, included in Table 3.2, below. The asset under attack is the information transiting the TOE. In general, the threat agent includes, but is not limited to: 1) people with TOE access who are expected to possess “average” expertise, few resources, and moderate motivation, or 2) failure of the TOE.

The following threats are included in every PP in this PP family. These threats must be included in every ST that claims compliance any one of the PPs in this family.

Table 3.2 – Base Threats to Security for all PPs in this PP Family

Threat Name	Threat Description
T.AUDIT_COMPROMISE	A user or process may view audit records, cause audit records to be lost or modified, or prevent future audit records from being recorded, thus masking a user’s action
T.CHANGE_TIME	An unauthorized user may change the TSF notion of time resulting in accepting old revocation information or expired certificates.
T.CRYPTO_COMPROMISE	A user or process may cause key, data or executable code associated with the cryptographic functionality to be inappropriately accessed (viewed, modified, or deleted), thus compromising the cryptographic mechanisms and the data protected by those mechanisms.

Threat Name	Threat Description
T.MASQUERADE	A user or process may masquerade as another entity in order to gain unauthorized access to data or TOE resources.
T.POOR_TEST	Lack of or insufficient tests to demonstrate that all TOE security functions operate correctly (including in a fielded TOE) may result in incorrect TOE behavior being undiscovered thereby causing potential security vulnerabilities.
T.RESIDUAL_DATA	A user or process may gain unauthorized access to data through reallocation of TOE resources from one user or process to another.
T.TSF_COMPROMISE	A user or process may cause, through an unsophisticated attack, TSF data, security attributes, or executable code to be inappropriately accessed (viewed, modified, or deleted).
T.UNATTENDED_SESSION	A user may gain unauthorized access to an unattended session.
T.UNAUTHORIZED_ACCESS	A user may gain access to user data for which they are not authorized according to the TOE security policy.
T.UNIDENTIFIED_ACTIONS	The administrator may not have the ability to notice potential security violations, thus limiting the administrator’s ability to identify and take action against a possible security breach.

This PP does not include the threats recommended by the Basic Robustness Consistency Instruction Manual that map solely to assurance requirements. This was done to keep the size and complexity of the family of PPs to manageable.

Some of the threats have been renamed by removing the term "accidental" since the mechanisms protect against accidental and well as intentional threats.

3.5 Threats to Security for Packages

The following subsections define security threats for each of the packages defined. The asset under attack is the information transiting the TOE. In general, the threat agent includes, but is not limited to: 1) people with TOE access who are expected to possess “average” expertise, few resources, and moderate motivation, or 2) failure of the TOE.

Note that in addition to the threats defined below for each package, every PP derived from this PP family also includes the base threats defined in Table 3.2.

3.5.1 Certification Path Validation – Basic Package

In addition to the base threats, the following threats are defined for the Certification Path Validation – Basic package. These threats apply to this package at all assurance levels.

Table 3.3 – Threats for the CPV – Basic Package

Threat Name	Threat Description
T.Certificate_Modi	An untrusted user may modify a certificate resulting in using a wrong public key.
T.DOS_CPV_Basic	The revocation information or access to revocation information could be made unavailable, resulting in loss of system availability.
T.Expired_Certificate	An expired (and possibly revoked) certificate as of TOI could be used for signature verification.
T.Untrusted_CA	An untrusted entity (Certification Authority (CA)) may issue certificates to bogus entities, permitting those entities to assume identity of other legitimate users.
T.No_Crypto	The user public key and related information may not be available to carry out the cryptographic function.
T.Path_Not_Found	A valid certification path is not found due to lack of system functionality.
T.Revoked_Certificate	A revoked certificate could be used as valid, resulting in security compromise.
T.User_CA	A user could act as a CA, issuing unauthorized certificates.

3.5.2 Certification Path Validation – Basic Policy Package

The following threats are defined for the Certification Path Validation – Basic Policy package. This threat applies to this package at all assurance levels.

Table 3.4 – Threats for the CPV – Basic Policy Package

Threat Name	Threat Description

T.Unknown_Policies	The user may not know the policies under which a certificate was issued.

3.5.3 Certification Path Validation – Policy Mapping Package

The following threats are defined for the Certification Path Validation – Policy Mapping package. These threats apply to this package at all levels.

Table 3.5 – Threats for the CPV – Policy Mapping Package

Threat Name	Threat Description
T.Mapping	The user may accept unacceptable certificates or reject acceptable certificates due to improper certificate policy mapping.
T.Wrong_Policy_Dec	The user may accept certificates that were not generated with the diligence and security acceptable to the user. The user may reject certificates that were generated with the diligence and security acceptable to the user.

3.5.4 Certification Path Validation – Name Constraints Package

The following threats are defined for the Certification Path Validation – Name Constraints Package. This threat applies to this package at all assurance levels.

Table 3.6 – Threats for the CPV – Name Constraints Package

Threat Name	Threat Description
T.Name_Collision	The user may accept certificates from CA where the CA's understanding and the user's understanding of the names differ, i.e., user and CA associate different identity with the same name.

3.5.5 PKI Signature Generation Package

The following threats are defined for the PKI Signature Generation package. This threat applies to this package at all assurance levels.

Table 3.7 – Threats for the PKI Signature Generation Package

Threat Name	Threat Description
T.Clueless_PKI_Sig	The user may try only inappropriate certificates for signature verification because the signature does not include a hint.

3.5.6 PKI Signature Verification Package

The following threats are defined for the PKI Signature Verification Package. These threats apply to this package at all assurance levels.

Table 3.8 – Threats for the PKI Signature Verification Package

Threat Name	Threat Description
T.Assumed_Identity_PKI_Ver	A user may assume the identity of another user in order to verify a PKI signature.
T.Clueless_PKI_Ver	The user may try only inappropriate certificates for signature verification because hints in the signature are ignored.

3.5.7 PKI Encryption using Key Transfer Algorithms Package

The following threats are defined for the PKI Encryption using Key Transfer Algorithms Package. These threats apply to this package at all assurance levels.

Table 3.9 – Threats for the PKI Encryption using Key Transfer Algorithms Package

Threat Name	Threat Description
T.Assumed_Identity_WO_En	A user may assume the identity of another user in order to perform encryption using Key Transfer algorithms.
T.Clueless_WO_En	The user may try only inappropriate certificates for encryption using Key Transfer algorithms in absence of hint.

3.5.8 PKI Encryption using Key Agreement Algorithms Package

The following threats are defined for the PKI Encryption using Key Agreement Algorithms package. These threats apply to this package at all assurance levels.

Table 3.10 – Threats for the PKI Encryption using Key Agreement Algorithms Package

Threat Name	Threat Description
T.Assumed_Identity_With_En	A user may assume the identity of another user in order to perform encryption using Key Agreement algorithms.
T.Clueless_With_En	The user may try only inappropriate certificates for encryption using Key Agreement algorithms in absence of hint.

3.5.9 PKI Decryption using Key Transfer Algorithms Package

The following threats are defined for the PKI Decryption using Key Transfer Algorithms package. These threats apply to this package at all assurance levels.

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

Threat Name	Threat Description
T.Garble_WO_De	The user may not apply the correct key transfer algorithm or private key, resulting in garbled data.

3.5.10 PKI Decryption using Key Agreement Algorithms Package

The following threats are defined for the PKI Decryption using Key Agreement Algorithms package. These threats apply to this package at all assurance levels.

Table 3.12 – Threats for the PKI Decryption using Key Agreement Algorithms Package

Threat Name	Threat Description
T.Assumed_Identity_With_De	A user may assume the identity of another user for decrypting using Key Agreement algorithms.
T.Clueless_With_De	The user may try only inappropriate certificates for decryption using Key Agreement algorithms in absence of hint.
T.Garble_With_De	The user may not apply the correct key agreement algorithm or private key, resulting in garbled data.

3.5.11 PKI Based Entity Authentication Package

The following threats are defined for the PKI Based Entity Authentication package. These threats apply to this package at all assurance levels.

Table 3.13 – Threats for the PKI Based Entity Authentication Package

Threat Name	Threat Description
T.Assumed_Identity_Auth	A user may assume the identity of another user to perform entity based authentication.
T.Replay_Entity	An unauthorized user may replay valid entity authentication data.

3.5.12 Online Certificate Status Protocol Client Package

The following threats are defined for Online Certificate Status Protocol Client package. These threats apply to this package at all assurance levels.

Table 3.14 – Threats for the OCSP Client Package

Threat Name	Threat Description
T.DOS_OCSP	The OCSP response or access to the OCSP response could be made unavailable, resulting in loss of system availability.
T.Replay_OCSP_Info	The user may accept an OCSP response from well before TOI resulting in accepting a revoked certificate.
T.Wrong_OCSP_Info	The user may accept a revoked certificate or reject a valid certificate due to a wrong OCSP response.

3.5.13 Certificate Revocation List (CRL) Validation Package

The following threats are defined for the Certificate Revocation List (CRL) Validation package. These threats apply to this package at all assurance levels.

Table 3.15 – Threats for the Certificate Revocation List (CRL) Validation Package

Threat Name	Threat Description
T.DOS_CRL	The CRL or access to CRL could be made unavailable, resulting in loss of system availability.
T.Replay_Revoc_Info_CRL	The user may accept a CRL issued well before TOI resulting in accepting a revoked certificate.
T.Wrong_Revoc_Info_CRL	The user may accept a revoked certificate or reject a valid certificate due to a wrong CRL.

3.5.14 Audit Package

The following threats are defined for the Audit package. These threats apply to this package at all assurance levels.

Table 3.16 – Threats for the Audit Package

Threat Name	Threat Description
T.PKE_Accountability	The PKE related audit events cannot be linked to individual actions.

3.5.15 Continuous Authentication Package

The following threat is defined for Continuous Authentication package. This threat applies to this package at all assurance levels.

Table 3.17 – Threats for the Continuous Authentication Package

Threat Name	Threat Description
T.Hijack	An unauthorized user may hijack an authenticated session.

3.6 Organizational Security Policies for all PPs in this PP Family

The policies described in the table below are included in every PP in this PP family. These policies must be included in every ST that claims compliance to any one of the PPs in this family.

Table 3.18 – Organizational Security Policies

Policy Name	Policy Description
P.ACCESS_BANNER	The IT Environment shall display an initial banner describing restrictions of use, legal agreements, or any other appropriate information to which users consent by accessing the system.
P.ACCOUNTABILITY	The authorized users of the TOE shall be held accountable for their actions within the TOE.
P.CRYPTOGRAPHY	Only NIST FIPS validated cryptography (methods and implementations) are acceptable for key management (i.e.; generation, access, distribution, destruction, handling, and storage of keys) and cryptographic services (i.e.; encryption, decryption, signature, hashing, key exchange, and random number generation services).

4 Security Objectives

4.1 Security Objectives for the Environment

The security objectives for the Environment are defined in Table 4.1, below. These security objectives are included in every PP in this PP family and must be included in every ST that claims compliance with any PP in this family of PPs.

There are four security objectives for the non-IT environment of the TOE: OE.Configuration, OE.NO_EVIL, OE.PHYSICAL, and OE.Low. The remaining objectives are for the IT environment.

Table 4.1 – Security Objectives for the Environment for all PPs in this PP Family

Objective Name	Objective Description
OE.AUDIT_GENERATION	The IT Environment will provide the capability to detect and create records of security-relevant events associated with users.
OE.AUDIT_PROTECTION	The IT Environment will provide the capability to protect audit information.
OE.AUDIT_REVIEW	The IT Environment will provide the capability to selectively view audit information,
OE.Configuration	The TOE will be installed and configured properly for starting up the TOE in a secure state.
OE.CORRECT_TSF_OPERATION	The IT Environment will provide the capability to test the TSF to ensure the correct operation of the TSF at a customer's site.
OE.CRYPTOGRAPHY	The TOE shall use NIST FIPS 140-2 validated cryptographic services provided by the IT Environment.
OE.DISPLAY_BANNER	The IT Environment will display an advisory warning regarding use of the TOE.
OE.Low	The Identification and Authentication functions in the IT Environment will be designed and implemented for a minimum attack potential of low as validated by the vulnerability assessment and Strength of Function analyses.
OE.MANAGE	The IT Environment will provide all the functions and facilities necessary to support the administrators in their management of the security of the TOE, and restrict these functions and facilities from unauthorized use.
OE.MEDIATE	The IT Environment will protect user data in accordance with its security policy.

Objective Name	Objective Description
OE.NO_EVIL	Sites using the TOE will ensure that administrators are non-hostile, appropriately trained and follow all administrator guidance.
OE.PHYSICAL	The non-IT environment will provide an acceptable level of physical security so that the TOE cannot be tampered with or be subject to side channel attacks such as the various forms of power analysis and timing analysis.
OE.RESIDUAL_INFORMATION	The IT Environment will ensure that any information contained in a protected resource within its Scope of Control is not released when the resource is reallocated.
OE.SELF_PROTECTION	The IT Environment will maintain a domain for its own execution that protects it and its resources from external interference, tampering, or unauthorized disclosure.
OE.TIME_STAMPS	The IT Environment will provide reliable time stamps and the capability for the administrator to set the time used for these time stamps.
OE.TIME_TOE	The IT Environment will provide reliable time for the TOE use.
OE.TOE_ACCESS	The IT Environment will provide mechanisms that control a user’s logical access to the TOE.
OE.TOE_PROTECTION	The IT Environment will protect the TOE and TOE resources from external interference, tampering, or unauthorized disclosure and modification.

4.2 Security Objectives for Packages

Security objectives for the packages in this PP family are defined in the following subsections. Note that in addition to the security objectives defined for each individual package, each PP derived from this PP family must include the security objectives for the IT Environment defined in Section 4.1.

4.2.1 Certification Path Validation – Basic Package

The following security objectives are defined for the Certification Path Validation – Basic PPs. These security objectives apply to this package at all assurance levels.

Table 4.2 – Security Objectives for CPV – Basic Package

Objective Name	Objective Description
O.Availability	The TSF shall continue to provide security services even if revocation information is not available.
O.Correct_Temporal	The TSF shall provide accurate temporal validation results.
O.Current_Certificate	The TSF shall only accept certificates that are not expired as of TOI.
O.Get_KeyInfo	The TSF shall provide the user public key and related information in order to carry out cryptographic functions.
O.Path_Find	The TSF shall be able to find a certification path from a trust anchor to the subscriber.
O.Trusted_Keys	The TSF shall use trusted public keys in certification path validation.
O.User	The TSF shall only accept certificates issued by a CA.
O.Verified_Certificate	The TSF shall only accept certificates with verifiable signatures.
O.Valid_Certificate	The TSF shall use certificates that are valid, i.e., not revoked.

Objectives O.Availability and O.Valid_Certificate mitigate threats T.DOS_CPV_Basic and T.Revoked_Certificate, respectively. But these objectives cannot completely counter the threats simultaneously. The ST author needs to find an approach for their application to by tailoring FDP_DAU_CPV_(EXP).1.3 security functional requirement element to decide if the threat T.DOS_CPV_Basic or the threat T.Revoked_Certificate will be fully mitigated.

4.2.2 Certification Path Validation – Basic Policy Package

The following security objective is defined for the Certification Path Validation – Basic Policy package. This security objective applies to this package at all assurance levels.

Table 4.3 – Security Objectives for CPV – Basic Policy Package

Objective Name	Objective Description
O.Provide_Policy_Info	The TSF shall provide certificate policies for which the certification path is valid.

4.2.3 Certification Path Validation – Policy Mapping Package

The following security objectives are defined for the Certification Path Validation – Policy Mapping package. These security objectives apply to this package at all assurance levels.

Table 4.4 – Security Objectives for CPV – Policy Mapping Package

Objective Name	Objective Description
O.Map_Policies	The TSF shall map certificate policies in accordance with user and CA constraints.
O.Policy_Enforce	The TSF shall validate a certification path in accordance with certificate policies acceptable to the user.

4.2.4 Certification Path Validation – Name Constraints Package

The following security objective is defined for the Certification Path Validation – Name Constraints package. This security objective applies to this package at all assurance levels.

Table 4.5 – Security Objectives for CPV – Name Constraints Package

Objective Name	Objective Description
O.Authorised_Names	The TSF shall validate a certificate only if the CA is authorized to issue a certificate to the subject.

4.2.5 PKI Signature Generation Package

The following security objective is defined for the PKI Signature Generation package. This security objective applies to this package at all assurance levels.

Table 4.6 – Security Objectives for PKI Signature Generation Package

Objective Name	Objective Description
O.Give_Sig_Hints	The TSF shall provide hints for selecting correct certificates for signature verification.

4.2.6 PKI Signature Verification Package

The following security objectives are defined for the PKI Signature Verification package. These security objectives apply to this package at all assurance levels.

Table 4.7 – Security Objectives for PKI Signature Verification Package

Objective Name	Objective Description
O.Use_Sig_Hints	The TSF shall use hints for selecting correct certificates for signature verification.
O.Linkage_Sig_Ver	The TSF shall use the correct user public key for signature verification.

4.2.7 PKI Encryption using Key Transfer Algorithms Package

The following security objectives are defined for the PKI Encryption using Key Transfer Algorithms package. These security objectives apply to this package at all assurance levels.

Table 4.8 – Security Objectives for PKI Encryption using Key Transfer Algorithms Package

Objective Name	Objective Description
O.Hints_Enc_WO	The TSF shall provide hints for selecting correct certificates or keys for PKI Encryption using Key Transfer Algorithms.
O.Linkage_Enc_WO	The TSF shall use the correct user public key for key transfer.

4.2.8 PKI Encryption using Key Agreement Algorithms Package

The following security objectives are defined for the PKI Encryption using Key Agreement Algorithms package. These security objectives apply to this package at all assurance levels.

Table 4.9 – Security Objectives for PKI Encryption using Key Agreement Algorithms Package

Objective Name	Objective Description
O.Hints_Enc_W	The TSF shall provide hints for selecting correct certificates or keys for PKI encryption using Key Agreement algorithms.
O.Linkage_Enc_W	The TSF shall use the correct user public key for key agreement during encryption.

4.2.9 PKI Decryption using Key Transfer Algorithms Package

The following security objectives are defined for the PKI Decryption using Key Transfer Algorithms package. These security objectives apply to this package at all assurance levels.

Table 4.10 – Security Objectives for PKI Decryption using Key Transfer Algorithms Package

Objective Name	Objective Description
O.Correct_KT	The TSF shall use appropriate private key and key transfer algorithm.

4.2.10 PKI Decryption using Key Agreement Algorithms Package

The following security objectives are defined for the PKI Decryption using Key Agreement Algorithms package. These security objectives apply to this package at all assurance levels.

Table 4.11 – Security Objectives for PKI Decryption using Key Agreement Algorithms Package

Objective Name	Objective Description
O.Hints_Dec_W	The TSF shall provide hints for selecting correct certificates or keys for PKI decryption using Key Agreement algorithms.
O.Linkage_Dec_W	The TSF shall use the correct user public key for key agreement during decryption.
O.Correct_KA	The TSF shall use appropriate private key and key agreement algorithm.

4.2.11 PKI Based Entity Authentication Package

The following security objectives are defined for the PKI Based Entity Authentication package. These security objectives apply to this package at all assurance levels.

Table 4.12 – Security Objectives for PKI Based Entity Authentication Package

Objective Name	Objective Description
O.I&A	The TSF shall uniquely identify all entities, and shall authenticate the claimed identify before granting an entity access to the TOE facilities.
O.Limit_Actions_Auth	The TSF shall restrict the actions an entity may perform before the TSF verifies the identity of the entity.
O.Linkage	The TSF shall use the correct user public key for authentication.
O.Single_Use_I&A	The TSF shall use the I&A mechanism that requires unique authentication information for each I&A.

4.2.12 Online Certificate Status Protocol Client Package

The following security objectives are defined for the Online Certificate Status Protocol Client package. These security objectives apply to this package at all assurance levels.

Table 4.13 – Security Objectives for Online Certificate Status Protocol Client Package

Objective Name	Objective Description
O.Accurate_OCSP_Info	The TSF shall accept only accurate OCSP responses.
O.Auth_OCSP_Info	The TSF shall accept the revocation information from an authorized source for OCSP transactions.
O.Current_OCSP_Info	The TSF accept only OCSP responses current as of TOI.
O.User_Override_Time_OCSP	The TSF shall permit the user to override the time checks on the OCSP response.

Objectives O.Current_OCSP_Info and O.User_Override_Time_OCSP mitigate threats T.Replay_OCSP_Info and T.DOS_OCSP, respectively. But these objectives cannot completely counter the threats simultaneously.

To fully mitigate the threat T.Replay_OCSP, the ST author can use request nonce as listed in the security functional requirements element FDP_DAU_OCS_(EXP).1.12.

To mitigate the threat T.DOS_OCSP, the ST author can perform operations on FDP_DAU_OCS_(EXP).1.9 security functional requirements element to ignore time checks.

4.2.13 Certificate Revocation List (CRL) Validation Package

The following security objectives are defined for the Certificate Revocation List Validation Package. These security objectives apply to this package at all assurance levels.

Table 4.14 – Security Objectives for Certificate Revocation List (CRL) Validation Package

Objective Name	Objective Description
O.Accurate_Rev_Info	The TSF shall accept only accurate revocation information.
O.Auth_Rev_Info	The TSF shall accept the revocation information from an authorized source for CRL.
O.Current_Rev_Info	The TSF shall accept only CRL that are current as of TOI.
O.User_Override_Time_CRL	The TSF shall permit the user to override the time checks on the CRL.

Objectives O.Current_Rev_Info and O.User_Override_Time_CRL mitigate threats TT.Replay_Revoc_Info_CRL and T.DOS_CRL, respectively. But these objectives cannot completely counter the threats simultaneously. To mitigate the threat T.DOS_CRL, the ST author can perform operations on FDP_DAU_CRL_(EXP).1.6 security functional requirements element to ignore time checks.

4.2.14 Audit Package

The following security objectives are defined for the Audit Package. These security objectives apply to this package at all assurance levels.

Table 4.15 – Security Objectives for Audit Package

Objective Name	Objective Description
O.PKE_Audit	The TSF shall audit security relevant PKE events.

4.2.15 Continuous Authentication Package

The following security objective is defined for the Continuous Authentication package. This security objective applies to this package at all assurance levels.

Table 4.16 – Security Objectives for Continuous Authentication Package

Objective Name	Objective Description
O.Continuous_I&A	The TSF shall continuously authenticate the entity.

5 IT Security Requirements

This section defines the TOE security functional requirements and assurance requirements, included for all of the PPs in this PP family. Requirements are drawn from the CC Parts 2 and 3 where possible. Explicitly stated requirements have been added, when necessary. Selections and assignments to be made by the ST author in Part 2 and explicitly stated requirements are enclosed in [square brackets] and text is in *italics*. A list of selections, identified as “Selection by the ST author,” allow the ST author to select one or more of the items listed as indicated. Assignments, identified as “Assignment by the ST author,” provide the ST author with the opportunity to insert specific information. Where the PP authors have made refinements in Part 2 requirements, the text is indicated by ***bold italics***. Assignments and selections in Part 2 requirements are indicated by *italics*. Iterations of requirements are indicated by a semicolon and number following the requirement number, e.g., FIA_UAU.1.1;1. In addition, the iterated requirement titles are indicated using a colon, e.g., FIA_UAU.1:1.

Each PP in this family of PPs is Part 2 extended. All functional requirements included in the family of PPs are listed in Table 5.1, below. Explicitly stated requirements are identified as “Part 2 extended.” And their name ends with "EXP" or a NIAP interpretation tag. Each PP in this family of PPs uses security functional requirements from the table below. In other words, each PP in this family of PPs uses a subset of security functional requirements from the table below.

Table 5.1 – Part 2 or Part 2 Extended

Requirement	**Part 2 or extended**
FAU_GEN.1-NIAP-0407:1 & 2	Part 2 Extended
FAU_GEN.2-NIAP-0410: 1 & 2	Part 2 Extended
FAU_SAR.1	Part 2
FAU_SAR.2	Part 2
FAU_SAR.3	Part 2
FAU_SEL.1-NIAP-0407	Part 2 Extended
FAU_STG.1-NIAP-0429	Part 2 Extended
FAU_STG.NIAP-0429-1	Part 2 Extended
FCS_CRM_FPS_(EXP).1	Part 2 Extended
FDP_ACC.1	Part 2
FDP_ACF.1-NIAP-0407	Part 2 Extended
FDP_RIP.2	Part 2
FIA_AFL.1	Part 2
FIA_ATD.1	Part 2
FIA_UAU.1	Part 2

Requirement	Part 2 or extended
FIA_UAU.2	Part 2
FIA_UAU.4	Part 2
FIA_UAU.6	Part 2
FIA_UAU.7	Part 2
FIA_UID.1	Part 2
FIA_UID.2	Part 2
FIA_USB.1	Part 2
FMT_MOF.1	Part 2
FMT_MSA.1	Part 2
FMT_MSA.3-NIAP-0429	Part 2 Extended
FMT_MTD.1:1 through 5	Part 2
FMT_SMF.1	Part 2
FMT_SMR.1	Part 2
FPT_RVM.1	Part 2
FPT_SEP.1	Part 2
FPT_SEP_ENV_(EXP).1	Part 2 Extended
FPT_STM.1	Part 2
FPT_TST_SOF_(EXP).1	Part 2 Extended
FTA_SSL.1	Part 2
FTA_SSL.2	Part 2
FTA_TAB.1	Part 2
FDP_CPD_(EXP).1	Part 2 Extended
FDP_DAU_CPV_(EXP).1	Part 2 Extended
FDP_DAU_CPV_(EXP).2	Part 2 Extended
FDP_DAU_CPV_(EXP).3	Part 2 Extended
FDP_DAU_CPV_(EXP).4	Part 2 Extended
FDP_DAU_CPV_(EXP).5	Part 2 Extended
FDP_DAU_CPI_(EXP).1	Part 2 Extended
FDP_DAU_CPI_(EXP).2	Part 2 Extended
FDP_DAU_CPI_(EXP).3	Part 2 Extended
FDP_DAU_CPI_(EXP).4	Part 2 Extended
FDP_DAU_CPO_(EXP).1	Part 2 Extended
FDP_DAU_CPO_(EXP).2	Part 2 Extended
FDP_DAU_CPO_(EXP).3	Part 2 Extended

Requirement	Part 2 or extended
FDP_DAU_CRL_(EXP).1	Part 2 Extended
FDP_DAU_ENC_(EXP).1	Part 2 Extended
FDP_DAU_ENC_(EXP).2	Part 2 Extended
FDP_DAU_ENC_(EXP).3	Part 2 Extended
FDP_DAU_OCS_(EXP).1	Part 2 Extended
FDP_DAU_SIG_(EXP).1	Part 2 Extended
FDP_ETC_ENC_(EXP).1	Part 2 Extended
FDP_ETC_ENC_(EXP).2	Part 2 Extended
FDP_ETC_SIG_(EXP).1	Part 2 Extended
FDP_ITC_ENC_(EXP).1	Part 2 Extended
FDP_ITC_ENC_(EXP).2	Part 2 Extended
FDP_ITC_SIG_(EXP).1	Part 2 Extended
FIA_UAU_SIG_(EXP).1	Part 2 Extended

All of the PPs in this family contain a set of IT Environment functional requirements. These requirements, which are common to all of the PPs, are included in Section 5.1 below. There are 15 packages and 3 different assurance levels defined in this family of PPs. A PP in this family is composed of the following:

- IT Environmental requirements defined in Section 5.1;
- One or more of the fifteen PP functional requirements packages defined in Section 5.2; and
- One of the assurance packages listed in Section 5.4.

5.1 IT Environment Security Functional Requirements

A list of the IT Environment security functional requirements is provided in Table 5.2. The full text of the security functional requirements is contained below.

The IT Environment requirements specify the ability to manage multiple private keys, associated certificates, and identifying data and associations among them. The term “manage” means the ability to do one or more of the following: generate, destroy, delete, use, import, export, modify, etc. The identifying data and association between private key and public key certificates are useful in selecting the appropriate cryptographic keys for cryptographic operations and for PKCS-7 type information generation. The IT Environment requirements also maintain secure storage of trust anchors.

Table 5.2 – IT Environment Security Functional Requirements included in all PPs in this PP Family

Functional Requirement	Title
FAU_GEN.1-NIAP-0407:1	Audit data generation
FAU_GEN.2-NIAP-0410:1	User identity association
FAU_SAR.1	Audit review
FAU_SAR.2	Restricted audit review
FAU_SAR.3	Selectable audit review
FAU_SEL.1-NIAP-0407	Selective audit
FAU_STG.1-NIAP-0429	Protected audit trail storage
FAU_STG.NIAP-0429-1	Site-configurable Prevention of audit data loss
FCS_CRM_FPS_(EXP).1	FIPS compliant cryptographic module
FDP_ACC.1	Subset access control – PKI Credential Management
FDP_ACF.1-NIAP-0407	Security attribute based access control – PKI Credential Management
FDP_RIP.2	Full residual information protection
FIA_AFL.1	Authentication failure handling
FIA_ATD.1	User attribute definition
FIA_UAU.2	User authentication before any action
FIA_UAU.7	Protected authentication feedback
FIA_UID.2	User identification before any action
FIA_USB.1	User-subject binding
FMT_MOF.1	Management of security function behavior
FMT_MSA.1	Management of security attributes
FMT_MSA.3-NIAP-0429	Static attribute initialization
FMT_MTD.1:1	Management of TSF data – I&A Data
FMT_MTD.1:2	Management of TSF data – Authentication Data
FMT_MTD.1:3	Management of TSF data – I&A Attempts
FMT_MTD.1:4	Management of TSF data – Trust Anchors
FMT_MTD.1:5	Management of TSF data – Time
FMT_SMF.1	Specification of management functions
FMT_SMR.1	Security roles
FPT_RVM.1	Non-bypassability of the TSP
FPT_SEP.1	TSF Domain separation
FPT_SEP_ENV_(EXP).1	Support for TSF domain separation

Functional Requirement	Title
FPT_STM.1	Reliable time stamps
FPT_TST_SOF_(EXP).1	TSF testing for Software only TOEs
FTA_SSL.1	TSF-initiated session locking
FTA_SSL.2	User-initiated locking
FTA_TAB.1	Default TOE access banners

5.1.1 Class FAU – Security Audit

FAU_GEN.1-NIAP-0407:1 Audit data generation

Hierarchical to: No other component

FAU_GEN.1.1-NIAP-0407;1 The ***IT Environment*** shall be able to generate an audit record of the following auditable events:

a) Start-up and shutdown of the audit functions;

b) All auditable events listed in Table 5-3; and

c) [selection: [assignment: *events at a basic level of audit introduced by the inclusion of additional SFRs determined by the ST author*], [assignment: *events commensurate with a basic level of audit introduced by the inclusion of explicit requirements determined by the ST author*], "*no additional events*"].

FAU_GEN.1.2-NIAP-0410;1 The ***IT Environment*** shall record within each audit record at least the following information:

a) Date and time of the event, type of event, subject identity (if applicable), and the outcome (success or failure) of the event; and

b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, information specified in column three of Table 5-3 below.

Dependencies: FPT_STM.1 Reliable time stamps

Table 5.3 – IT Environment Auditable Events

Requirement	Auditable Events	Additional Audit Record Contents
FAU_GEN.1-NIAP-0407:1	None	
FAU_GEN.2-NIAP-0410:1	None	
FAU_SAR.1	Opening the audit trail	The identity of the Audit Administrator performing the function
FAU_SAR.2	Unsuccessful attempts to read information from the audit records	The identity of the administrator performing the function

Requirement	Auditable Events	Additional Audit Record Contents
FAU_SAR.3	None	
FAU_SEL.1-NIAP-0407	All modifications to the audit configuration that occur while the audit collection functions are operating	The identity of the Security Administrator performing the function
FAU_STG.1-NIAP-0429	None	
FAU_STG.NIAP-0429	None	
FCS_CRM_FPS_(EXP).1.	None	
FDP_ACC.1	None	
FDP_ACF.1-NIAP-0407	All requests to perform an operation on an object covered by the SFP	Object identity
FDP_RIP.2	None	
FIA_AFL.1	Reaching of the threshold for the unsuccessful authentication attempts	
FIA_ATD.1	None	
FIA_UAU.2	All use of authentication mechanism	
FIA_UAU.7	None	
FIA_UID.2	All use of identification mechanism	User identity
FIA_USB.1	Success and failure of binding of user security attributes to a subject (e.g. success and failure to create a subject).	
FMT_SMF.1	Use of management function	Management function
FMT_SMR.1	Modifications to the group of users that are part of a role	
FPT_RVM.1	None	
FPT_SEP.1	None	
FPT_SEP_ENV_(EXP).1	None	
FPT_STM.1	Change to the time	
FTA_TAB.1	None	

FAU_GEN.2-NIAP-0410:1 User identity association

Hierarchical to: No other components.

FAU_GEN.2.1-NIAP-0410;1 For audit events resulting from actions of identified users, the ***IT Environment*** shall be able to associate each auditable event with the identity of the user that caused the event.

Dependencies: FAU_GEN.1 Audit data generation

FIA_UID.1 Timing of identification

FAU_SAR.1 **Audit review**

Hierarchical to: No other components.

FAU_SAR.1.1 The ***IT Environment*** shall provide *the administrator* with the capability to read *all audit information* from the audit records.

FAU_SAR.1.2 The ***IT Environment*** shall provide the audit records in a manner suitable for the user to interpret the information.

Dependencies: FAU_GEN.1 Audit data generation

FAU_SAR.2 **Restricted audit review**

Hierarchical to: No other components.

FAU_SAR.2.1 The ***IT Environment*** shall prohibit all users read access to the audit records, except those users that have been granted explicit read-access.

Dependencies: FAU_SAR.1 Audit review

FAU_SAR.3 **Selectable audit review**

Hierarchical to: No other components.

FAU_SAR.3.1 The ***IT Environment*** shall provide the ability to perform *searches and sorting* [selection of one or more by the ST author: ordering, *no other operation*] of audit data based on *date, time, user identity and [assignment* by the ST author: *criteria with logical relations*].

Dependencies: FAU_SAR.1 Audit review

FAU_SEL.1-NIAP-0407 **Selective audit**

Hierarchical to: No other components.

FAU_SEL.1.1-NIAP-0407 The ***IT Environment*** shall ***allow only the administrator*** to include or exclude auditable events from the set of audited events based on the following attributes:

a) user identity;

b) event type;

c) [selection of one or more by ST author: *object identity, subject identity, host identity, "none"*];

d) success of auditable security events;

e) failure of auditable security events; and

f) [selection by ST author: [assignment by ST author: *list of additional criteria that audit selectivity is based upon*], *no additional criteria*]].

Dependencies: FAU_GEN.1 Audit data generation

FMT_MTD.1 Management of TSF data

Application Note: *“event type” is to be defined by the ST author; the intent is to be able to include or exclude classes of audit events*.

FAU_STG.1-NIAP-0429 Protected audit trail storage

Hierarchical to: No other components.

FAU_STG.1.1-NIAP-0429 The ***IT Environment*** shall ***restrict the deletion of*** stored audit records in the audit trail ***to the administrator***.

FAU_STG.1.2-NIAP-0429 The ***IT Environment*** shall be able to *prevent* modifications to the audit records in the audit trail.

Dependencies: FAU_GEN.1 Audit data generation

FAU_STG.NIAP-0429-1 Site-configurable Prevention of audit data loss

Hierarchical to: FAU_STG.4

FAU_STG.NIAP-0429-1.1 The ***IT Environment*** shall provide an authorized administrator with the capability to select one or more of the following actions [*prevent auditable events, except those taken by the authorized user with special rights, overwrite the oldest stored audit records*] and [selection of one by the ST author: [assignment by the ST author: *other actions to be taken in case of audit storage failure*], "*no additional options"*] to be taken if the audit trail is full.

FAU_STG.NIAP-0429-1.2 The ***IT Environment*** shall [selection of one by the ST author: "*ignore auditable events", "prevent auditable events, except those taken by the authorized user with special rights", "overwrite the oldest stored audit records"*] and [assignment by the ST author: *other actions to be taken in case of audit storage failure, "no other action"*] if the audit trail is full and no other action has been selected.

Dependencies: FAU_STG.1 Protected Audit Trail Storage

FMT_MTD.1 Management of TSF Data

Application Note: The IT Environment provides the administrator the option of preventing audit data loss by preventing auditable events from occurring. The administrator’s actions under these circumstances are not required to be audited. The IT Environment also provides the administrator the option of overwriting “old” audit records rather than preventing auditable events, which may protect against a denial-of-service attack.

The ST writer should fill in other technology-specific actions that can be taken for audit storage failure (in addition to the two already specified), or select “no additional options” if there are no such technology-specific actions.

5.1.2 Class FCS – Cryptographic Support

FCS_CRM_FPS_(EXP).1 FIPS compliant cryptographic module

Hierarchical to: No other components.

FCS_CRM_FPS_(EXP).1.1 The IT environment shall provide all cryptographic modules necessary for the TSF.

FCS_CRM_FPS_(EXP).1.2 Each cryptographic module shall be FIPS 140 series Level 1 validated.

Dependencies: None.

5.1.3 Class FDP – User Data Protection

FDP_ACC.1 Subset access control – PKI Credential Management

Hierarchical to: No other components.

FDP_ACC.1.1 The ***IT Environment*** shall enforce the *PKI credential management SFP* on

Subjects: [assignment by the ST author: *list of subjects covered by the SFP*],

Objects: cryptographic key, public key certificate [assignment by ST author: *additional objects covered by the SFP*]*,*

Operations: [selection of one or more by the ST author:

a) Generate, import, export, destroy, and use private key

b) Import, export, and delete public key certificate

c) Use public key certificate

d) [Assignment by the ST author: *additional operations among subjects and objects covered by the SFP*]].

Application Note: *The terms object and subject refer to generic elements in the IT Environment. For a policy to be implemented, these entities must be clearly identified. For most systems there is only one type of subject, usually called a process or task, which needs to be specified in the ST. The ST author should specify the list of subjects, objects, and operations among subjects and objects covered by the SFP.*

Dependencies: FDP_ACF.1 Security attribute based access control

FDP_ACF.1-NIAP-0407 Security attribute based access control – PKI Credential Management

Hierarchical to: No other components.

FDP_ACF.1.1-NIAP-0407 The ***IT Environment*** shall enforce the *PKI credential management SFP* to objects based on the following: list of subjects: *all subjects;*

list of objects: *cryptographic keys and public key certificate;* list of subjects and object attributes: *identity of the subject and the set of roles that the subject is authorized to assume* [assignment by the ST author: *object attributes (e.g., owner)*].

FDP_ACF.1.2-NIAP-0407 The ***IT Environment*** shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed [selection of one or more by the ST author:

a) *Private keys may be generated, imported, exported, destroyed, used by* [selection of one or more by the ST author*: owner, administrator,* [assignment by the ST author: *other roles defined by the ST author*]].

b) *Public key certificates may be imported, exported, deleted by [*selection of one or more by the ST author*: owner, administrator,* [assignment by the ST author: *other roles defined by the ST author*]].

c) *Public key certificates may be used by anyone.*

d) [assignment by the ST author: *other rule(s)*].]

FDP_ACF.1.3--NIAP-0407 The ***IT Environment*** shall explicitly authorize access of subjects to objects based on the following additional rules: [Selection: [assignment by the ST author: *rules, based on security attributes that explicitly authorize access of subjects to objec*ts], *"no additional rules"*].

FDP_ACF.1.4--NIAP-0407 The ***IT Environment*** shall explicitly deny access of subjects to objects based on the [Selection: [assignment by the ST author: *rules, based on security attributes that explicitly deny access of subjects to objec*ts], *"no additional rules"*].

Dependencies: FDP_ACC.1 Subset access control, FMT_MSA.3 Static attribute initialization

FDP_RIP.2 Full residual information protection

Hierarchical to: FDP_RIP.1

FDP_RIP.2.1 The ***IT Environment*** shall ensure that any previous information content of a resource is made unavailable upon the [selection of one or more by the ST author: *allocation of the resource to, deallocation of the resource from*] all objects.

Dependencies: No dependencies

5.1.4 Class FIA – Identification and Authentication

FIA_AFL.1 Authentication failure handling

Hierarchical to: No other components

FIA_AFL.1.1 The ***IT Environment*** shall detect when *an administrator configurable positive integer within* [assignment by the ST Author: *range of acceptable values*] unsuccessful authentication attempts occur related to [assignment by the ST author: *list of authentication events*].

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been met or surpassed, the ***IT Environment*** shall prevent *all entities requesting authentication other than the administrator* from performing activities that require authentication until an action is taken by the administrator.

Dependencies: FIA_UAU.1 Timing of authentication

FIA_ATD.1 User attribute definition

Hierarchical to: No other components

FIA_ATD.1.1 The ***IT Environment*** shall maintain the following list of security attributes belonging to individual users: *user ID, role.*

Dependencies: No dependencies

FIA_UAU.2 User authentication before any action

Hierarchical to: FIA_UAU.1

FIA_UAU.2.1 The ***IT Environment*** shall require each user to be successfully authenticated before allowing any other ***IT Environment*** mediated actions on behalf of that user.

Dependencies: FIA_UID.1 Timing of identification

FIA_UAU.7 Protected authentication feedback

Hierarchical to: No other components

FIA_UAU.7.1 The ***IT Environment*** shall provide only [assignment by the ST author: *list of feedback*] to the user while the authentication is in progress.

Dependencies: FIA_UAU.1 Timing of authentication

FIA_UID.2 User identification before any action

Hierarchical to: FIA_UID.1

FIA_UID.2.1 The ***IT Environment*** shall require each user identify itself before allowing any other ***IT Environment*** mediated actions on behalf of that user.

Dependencies: No dependencies

FIA_USB.1 User-subject binding

Hierarchical to: No other components

FIA_USB.1.1 The ***IT Environment*** shall associate the following user security attributes with subjects acting on the behalf of that user: *all user security attributes*.

FIA_USB.1.2 The ***IT Environment*** shall enforce the following rules on the initial association of user security attributes with subjects acting on the behalf of users: *none*.

FIA_USB.1.3 The ***IT Environment*** shall enforce the following rules governing changes to the user security attributes associated with subjects acting on the behalf of users: *none*.

Dependencies: FIA_ATD.1 User attribute definition

5.1.5 Class FMT – Security Management

FMT_MOF.1 Management of security function behavior

Hierarchical to: No other components

FMT_MOF.1.1 The ***IT Environment*** shall restrict the ability to [selection of one or more by the ST author: *determine the behavior of, disable, enable, modify the behavior* of] the functions *audit*, [assignment by the ST author: *list of functions*] to *the administrator.*

Dependencies: FMT_SMF.1 Specification of Management Functions,

FMT_SMR.1 Security roles

FMT_MSA.1 Management of security attributes

Hierarchical to: No other components

FMT_MSA.1.1 The ***IT Environment*** shall enforce the *PKI credential management SFP* to restrict the ability to [s*election of one or more by the ST author: change_default, query, modify, delete,* [assignment by the ST author*: other specified operations*]] the security attributes [*selection of one or more by the ST author: user role, key identifier, association between private key and public key certificate,* [assignment by the ST author*: other security attributes*]] to [selection of one or more by the ST author*: owner, user, administrator,* [assignment by the ST author*: other role(s) defined*]].

Dependencies: FMT_SMF.1 Specification of Management Functions, FMT_SMR.1 Security roles, FDP_ACC.1 Subset access control

[FDP_ACC.1 Subset access control or FDP_IFC Subset information flow control]

FMT_MSA.3-NIAP-0429 Static attribute initialization

Hierarchical to: No other components

FMT_MSA.3.1-NIAP-0429 The ***IT Environment*** shall enforce the *PKI credential management SFP* to provide *specific* default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2-NIAP-0429 The ***IT Environment*** shall allow the [selection of one or more by the ST author: *owner, user, administrator,* [assignment by the ST author: *other role(s) defined*]] to specify alternative initial values to override the default values when an object or information is created.

Dependencies: FMT_SMR.1 Security roles, FMT_MSA.1 Management of security attributes

FMT_MTD.1:1 Management of TSF data – I&A Data

Hierarchical to: No other components

FMT_MTD.1.1;1 The ***IT Environment*** shall restrict the ability to *initialize and modify identification data and authentication data* to *administrator*.

Dependencies: FMT_SMF.1 Specification of Management Functions, FMT_SMR.1 Security roles

FMT_MTD.1:2 Management of TSF data – Authentication Data

Hierarchical to: No other components

FMT_MTD.1.1;2 The ***IT Environment*** shall restrict the ability to *modify authentication data* to *administrator and the user owning the account*.

Dependencies: FMT_SMF.1 Specification of Management Functions, FMT_SMR.1 Security roles

FMT_MTD.1:3 Management of TSF data – I&A Attempts

Hierarchical to: No other components

FMT_MTD.1.1;3 The ***IT Environment*** shall restrict the ability to *initialize and modify number of unsuccessful authentication* to *administrator*.

Dependencies: FMT_SMF.1 Specification of Management Functions, FMT_SMR.1 Security roles

FMT_MTD.1:4 Management of TSF data – Trust Anchors

Hierarchical to: No other components

FMT_MTD.1.1;4 The ***IT Environment*** shall restrict the ability to *add and delete trust anchors***,** to [selection of one or more by the ST author: *user, administrator,* [assignment by the ST author: *other role(s) defined*]].

Dependencies: FMT_SMF.1 Specification of Management Functions,
FMT_SMR.1 Security roles

FMT_MTD.1:5 Management of TSF data – Time

Hierarchical to: No other components

FMT_MTD.1.1;5 The ***IT Environment*** shall restrict the ability to *initialize and modify system time* to *administrator*.

Dependencies: FMT_SMF.1 Specification of Management Functions,
FMT_SMR.1 Security roles

FMT_SMF.1 Specification of Management Functions

Hierarchical to: No other components

FMT_SMF.1.1 The ***IT Environment*** shall be capable of performing the following security management functions: *audit management*, *user identity management, trust anchor management, system time management,* [assignment by ST author*: list of security management functions to be provided by the TSF*]*.*

Dependencies: No dependencies

FMT_SMR.1 Security roles

Hierarchical to: No other component

FMT_SMR.1.1 The ***IT Environment*** shall maintain the roles *user, administrator* [assignment by the ST author: none, *other role(s) defined*]].

FMT_SMR.1.2 The ***IT Environment*** shall be able to associate users with roles.

Dependencies: FIA_UID.1 Timing of identification

.

5.1.6 Class FPT – Protection of the TOE Security Functions

FPT_RVM.1 Non-bypassability of the IT Environment security policy

Hierarchical to: No other components

FPT_RVM.1.1 The ***IT Environment*** shall ensure that ***IT Environment security policy*** enforcement functions are invoked and succeed before each function within the ***IT Environment*** **Scope of Control** is allowed to proceed.

Dependencies: No dependencies

FPT_SEP.1 TSF Domain separation

Hierarchical to: No other components

FPT_SEP.1.1 The ***IT Environment*** shall maintain a security domain for its own execution that protects it from interference and tampering by untrusted subjects.

FPT_SEP.1.2 The ***IT Environment*** shall enforce separation between the security domains of subjects in the ***IT Environment Scope of Control***.

Dependencies: No dependencies

FPT_SEP_ENV_(EXP).1 Support for TSF domain separation

Hierarchical to: No other components

FPT_SEP_ENV_(EXP).1.1 The IT Environment shall maintain a security domain that protects the TSF from interference and tampering by untrusted subjects.

Dependencies: None.

Application Note: *This requirement is provide protection to the TSF. It is generally met by FPT_SEP and DAC on TSF executables, TSF Data, TSF User data including file systems objects and IPC mechanisms such as pipes, shared memory etc..*

FPT_STM.1 Reliable time stamps

Hierarchical to: No other components.

FPT_STM.1.1 The ***IT environment*** shall be able to provide reliable time stamps for its own ***and TSF*** use.

Dependencies: None.

FPT_TST_SOF_(EXP).1 TSF testing for Software only TOEs

Hierarchical to: No other components.

FPT_TST_SOF_(EXP).1.1 The ***IT Environment*** shall provide administrator with the capability to verify the integrity of the following TSF data: [assignment by the ST author: none, *list of TSF data*].

FPT_TST_SOF_(EXP).1.2 The ***IT Environment*** shall provide administrator with the capability to verify the integrity of stored TSF executable code.

Dependencies: No dependencies

5.1.7 Class FTA – TOE Access

FTA_SSL.1 TSF-initiated session locking

Hierarchical to: No other components.

FTA_SSL.1.1 The ***IT Environment*** shall lock an interactive session after [assignment by the ST author: time interval of user inactivity] by:

	a) clearing or overwriting display devices, making the current contents unreadable;
	b) disabling any activity of the user's data access/display devices other than unlocking the session.
FTA_SSL.1.2	The ***IT Environment*** shall require the following events to occur prior to unlocking the session: *authentication by the user.*
Dependencies:	FIA_UAU.1 Timing of authentication

FTA_SSL.2 User-initiated locking

Hierarchical to: No other components.

FTA_SSL.2.1	The ***IT Environment*** shall allow user-initiated locking of the user's own interactive session, by:
	a) clearing or overwriting display devices, making the current contents unreadable;
	b) disabling any activity of the user's data access/display devices other than unlocking the session.
FTA_SSL.2.2	The ***IT Environment*** shall require the following events to occur prior to unlocking the session: *authentication by the user*.
Dependencies:	FIA_UAU.1 Timing of authentication

FTA_TAB.1 Default TOE access banners

Hierarchical to: No other components.

FTA_TAB.1.1	Before establishing a user session, the ***IT Environment*** shall display an advisory warning message regarding unauthorized use of the ***System***.
Dependencies:	No dependencies

5.2 Security Functional Requirements for TOE

The following subsections define functional requirements for each package. Note that all PPs in this PP family must include the IT Environment functional requirements defined in Section 5.1, in addition to the unique requirements defined below for the particular packages selected for inclusion. There are 15 subsections below. Each subsection provides functional requirements for a package. Note that some packages have dependencies on other packages. A summary of package dependencies is as follows:

- Certification Path Validation – Basic Package is a dependency of the following other packages, i.e., when the following packages are included, the Certification Path Validation – Basic Package must also be included:
    - Certification Path Validation – Basic Policy Package
    - Certification Path Validation – Policy Mapping Package

  - Certification Path Validation – Name Constraints Package
  - PKI Encryption using Key Transfer Algorithms
  - PKI Encryption using Key Agreement Algorithms
  - PKI Decryption using Key Agreement Algorithms
  - PKI Signature Verification
  - PKI Based Entity Authentication
  - Continuous Authentication
- Certification Path Validation – Basic Policy is a dependency of Certification Path Validation – Policy Mapping Package
- PKI Based Entity Authentication is a dependency of Continuous Authentication Package

Note that functional requirements for packages remain the same, regardless of which assurance level is selected.

A summary of the functional requirements included in each package and package dependencies is provided in Table 5.4, below. Note that if a package has one or more dependency packages listed, then all the dependency package(s) must be included in the PP or ST when the dependent package is included in the PP. It is not valid under any circumstances to include a package with dependencies and not include the dependency packages in the PP or ST, i.e. dependencies must be included as specified in Table 5.4.

Note that the Audit requirements at basic level for explicitly stated requirements are implicitly defined by Table 5.5 which lists the Audit requirements for the various components in the TOE.

Table 5.4 – Summary of Security Functional Requirements in Packages

Package Name	Functional Requirement	Dependency Package
Certification Path Validation – Basic	FDP_CPD_(EXP).1	none
	FDP_DAU_CPI_(EXP).1	
	FDP_DAU_CPV_(EXP).1	
	FDP_DAU_CPV_(EXP).2	
	FDP_DAU_CPO_(EXP).1	
Certification Path Validation – Basic Policy	FDP_DAU_CPI_(EXP).2	Certification Path Validation – Basic
	FDP_DAU_CPO_(EXP).2	
Certification Path Validation – Policy Mapping	FDP_DAU_CPI_(EXP).3	Certification Path Validation – Basic, Certification Path Validation – Basic Policy
	FDP_DAU_CPV_(EXP).3	
	FDP_DAU_CPO_(EXP).3	
Certification Path Validation – Name Constraints	FDP_DAU_CPI_(EXP).4	Certification Path Validation – Basic
	FDP_DAU_CPV_(EXP).4	

Package Name	Functional Requirement	Dependency Package
	FDP_DAU_CPV_(EXP).5	
PKI Signature Generation	FDP_ETC_SIG_(EXP).1	none
PKI Signature Verification	FDP_ITC_SIG_(EXP).1	Certification Path Validation – Basic
	FDP_DAU_SIG_(EXP).1	
PKI Encryption using Key Transfer Algorithms	FDP_ETC_ENC_(EXP).1	Certification Path Validation – Basic
	FDP_DAU_ENC_(EXP).1	
PKI Encryption using Key Agreement Algorithms	FDP_ETC_ENC_(EXP).2	Certification Path Validation – Basic
	FDP_DAU_ENC_(EXP).2	
PKI Decryption using Key Transfer Algorithms	FDP_ITC_ENC_(EXP).1	None
PKI Decryption using Key Agreement Algorithms	FDP_ITC_ENC_(EXP).2	Certification Path Validation – Basic
	FDP_DAU_ENC_(EXP).3	
PKI Based Entity Authentication	FIA_UAU.1	Certification Path Validation – Basic
	FIA_UAU.4	
	FIA_UAU_SIG_(EXP).1	
	FIA_UID.1	
Online Certificate Status Protocol Client	FDP_DAU_OCS_(EXP).1	None
Certificate Revocation List Validation	FDP_DAU_CRL_(EXP).1	None
Audit	FAU_GEN.1-NIAP-0407:2	None
	FAU_GEN.2-NIAP-0410:2	
Continuous Authentication	FIA_UAU.6	PKI Based Entity Authentication, Certification Path Validation – Basic

In addition to the above dependencies, the following conditional dependencies may be invoked depending on the selections by the ST author:

- CPV – Basic package may depend on OCSP Client Package
- CPV – Basic package may depend on Certificate Revocation List (CRL) Validation Package
- OCSP Client Package may depend on CPV – Basic package
- Certificate Revocation List (CRL) Validation Package may depend on CPV – Basic package

5.2.1 Certification Path Validation – Basic Package

The functions in this package address the validation of the certification path. Certification path development is also a part of this package. It is realized that the most likely implementations consist of developing a path (using a variety of techniques) and then validating the certification path. It is further recognized that certification path validation generally consists of validating certificates starting with the one certified by a trust anchor and ending with the one issued to the subscriber of interest. However, in order to be implementation neutral, this package does not mandate any ordering of certification path development and certification validation processes. A compliant implementation will only need to meet the security requirements specified in this package.

All processing defined is X.509 and PKIX compliant. The certification path validation in these standards is procedural, but in keeping with the spirit of functional specification, certification path validation requirements are specified using non-procedural techniques.

From certification path processing perspective, certificates can be of up to three types:

- Self-signed trust anchor certificate: The trust anchor can be in the form of a self-signed certificate. The trust anchor is used to obtain the Distinguished Name (DN), public key, algorithm identifier, and the public key parameters (if applicable). This package permits validation of trust anchor if it is in the form of self-signed certificate, including validating signature and verifying that the self-signed certificate validity period has not expired.
- Intermediate certificates: These are the certificates issued to the CAs. All certificates in a certification path are intermediate certificates, except the last one.
- End certificate: This is the last certificate in the certification path and is issued to the subscriber of interest. This is typically an end-entity (i.e., not a CA) certificate. However, this package permits that certificate to be a CA certificate also.

This package processes the following security related certificate extensions checks: no-check, keyUsage, extendedKeyUsage, and basicConstraints.

This PKE PP family provides the capability to validate path as of a user-defined time called TOI which can be current time or earlier.

If revocation checking is selected, this package may depend on one or both of OCSP Client and CRL validation packages

5.2.1.1 Class FDP – User Data Protection

FDP_CPD_(EXP).1 Certification path development

Hierarchical to: No other components.

FDP_CPD_(EXP).1.1 The TSF shall develop a certification path from a trust anchor provided by [selection of one or more by the ST author: *user; administrator,* [assignment by the ST author*: other role defined*]] to the subscriber using matching rules for the following subscriber certificate fields or extensions: [selection of one or more by the ST author: *distinguished name, subject alternative names, subject key identifier, subject public key algorithm, certificate policies,* [assignment by the ST author*: other certificate fields or extensions*]].

FDP_CPD_(EXP).1.2 The TSF shall develop the certification path using the following additional matching rule: [selection of one by the ST author:

a) none,

b) keyUsage extension has nonRepudiation bit set,

c) keyUsage extension has digitalSignature bit set,

d) keyUsage extension has keyEncipherment bit set,

e) *key Usage extension has keyAgreement bit set*].

FDP_CPD_(EXP).1.3 The TSF shall develop the certification path using the following additional matching rule [selection of one by the ST author:

a) none,

b) extendedKeyUsage extension contains EFS or anyExtendedKeyUsage OID,

c) extendedKeyUsage extension contains SCL or anyExtendedKeyUsage OID,

d) extendedKeyUsage extension contains code signing or anyExtendedKeyUsage OID,

e) extendedKeyUsage extension contains OCSP signing or anyExtendedKeyUsage OID,

f) [assignment by the ST author*: other extended key usage OID related matching rules*]].

FDP_CPD_(EXP).1.4 The TSF shall bypass any matching rules except [selection of one or more by the ST author: *distinguished name, subject alternative names, subject key identifier, subject public key algorithm, certificate policies,* [assignment by the ST author*: other certificate fields or extensions, none*], *none*] if additional certification paths are required.

Dependencies: None

Application Note: In FDP_CPD_(EXP).1.2, the assignment nonRepudiation should be used if the path is being developed for signature verification; the assignment digitalSignature should be used if the path is being developed for entity authentication; the assignment keyEncipherment, should be used if the path is being developed for encryption certificate using a key transfer algorithm (e.g., RSA); the assignment keyAgreement should be used if the path is being developed for encryption certificate using a key calculation algorithm (e.g., DH, ECDH).

In FDP_CPD_(EXP).1.3, the selection of the matching rule should be made depending on the PKE application requirement. anyExtendedKeyUsage is a match for any application.

FDP_DAU_CPI_(EXP).1 Certification path initialisation -- basic

Hierarchical to: No other components.

FDP_DAU_CPI_(EXP).1.1 The TSF shall use the trust anchor provided by [selection of one or more by the ST author: *user, administrator,* [assignment by the ST author*: other role(s) defined*]].

FDP_DAU_CPI_(EXP).1.2 The TSF shall obtain the time of interest called "TOI' from a reliable source [selection of one by the ST author: *local environment,* [assignment by ST author: *other sources defined by ST author*]].

FDP_DAU_CPI_(EXP).1.3 The TSF shall perform the following checks on the trust anchor [selection of one or more by the ST author:

a) None;

b) Subject DN and Issuer DN match;

c) Signature verifies using the subject public key and parameter (if applicable) from the trust anchor;

d) notBefore field in the trust anchor <= TOI;

e) *notAfter field in the trust anchor => TOI*]

FDP_DAU_CPI_(EXP).1.4 The TSF shall derive from the trust anchor [selection of one or more by the ST author: *subject DN, subject public key, subject public key algorithm object identifier, subject public key parameters*]

Dependencies: FCS_COP.1, FPT_STM.1

Application Note: *While the PP requires the environment to provide accurate time to required precision, the ST author can choose other sources of accurate time*

.

FDP_DAU_CPV_(EXP).1 Certificate processing -- basic

Hierarchical to: No other components.

FDP_DAU_CPV_(EXP).1.1The TSF shall reject a certificate if any of the following checks fails:

a) Use parent-public-key, parent-public-key-algorithm-identifier, and parent-public-key-parameters to verify the signature on the certificate;

b) notBefore field in the certificate < = TOI;

c) notAfter field in the certificate > = TOI;

d) issuer field in the certificate = parent-DN; or

e) TSF is able to process all extensions marked critical

FDP_DAU_CPV_(EXP).1.2 The TSF shall bypass the revocation status check if the certificate contains no-check extension.

FDP_DAU_CPV_(EXP).1.3 The TSF shall bypass the revocation check if the revocation information is not available and [selection of one or more by the ST author: *none, user, administrator,* [assignment by the ST author*: other role(s) defined*]] overrides revocation checking.

FDP_DAU_CPV_(EXP).1.4 The TSF shall reject a certificate if the revocation status using [selection of one or more by the ST author: *CRL, OCSP*] demonstrates that the certificate is revoked.

FDP_DAU_CPV_(EXP).1.5 The TSF shall update the public key parameters state machine using the following rules:

a) Obtain the parameters from the subjectPublickeyInfo field of certificate if the parameters are present in the field; else

b) Retain the old parameters state if the subject public key algorithm of current certificate and parent public key algorithm of current certificate belong to the same family of algorithms, else

c) Set parameters = “null”.

Dependencies: FCS_COP.1, FPT_STM.1, [FDP_DAU_OCS_(EXP).1 or FDP_DAU_CRL_(EXP).1]

Application Note: *While each certificate is expected to be checked using only one of the revocation mechanisms, each certificate in a certification path can be checked using different revocation mechanism. That is why the selection is one or more.*

FDP_DAU_CPV_(EXP).2 Intermediate certificate processing -- basic

Hierarchical to: No other components.

FDP_DAU_CPV_(EXP).2.1 The TSF shall reject an intermediate certificate if any of the following additional checks fails:

a) basicConstraints field is present with cA = TRUE;

b) pathLenConstraint is not violated; or

c) if a critical keyUsage extension is present, keyCertSign bit is set

Dependencies: FDP_DAU_CPV_(EXP).1

FDP_DAU_CPO_(EXP).1 Certification path output -- basic

Hierarchical to: No other components.

FDP_DAU_CPO_(EXP).1.1 The TSF shall output certification path validation failure if any certificate in the certification path is rejected.

FDP_DAU_CPO_(EXP).1.2 The TSF shall output the following variables from the end certificate: subject DN, subject public key algorithm identifier, subject public key, critical keyUsage extension.

FDP_DAU_CPO_(EXP).1.3 The TSF shall output the following additional variables from the end certificate [selection of one or more by the ST author: *certificate, subject alternative names, extendedKeyUsage,* [assignment by the ST author*: other information*]].

FDP_DAU_CPO_(EXP).1.4 The TSF shall output the subject public key parameters from the certification path parameter state machine.

Dependencies: FDP_DAU_CPV_(EXP).1

5.2.2 Certification Path Validation – Basic Policy Package

The security functional requirements in this package address certificate path processing with the processing of certificatePolicies extension. This package is dependent upon the Certification Path Validation – Basic package.

5.2.2.1 Class FDP – User Data Protection

FDP_DAU_CPI_(EXP).2 Certification path initialisation – basic policy

Hierarchical to: No other components.

FDP_DAU_CPI_(EXP).2.1 The TSF shall use the initial-certificate-policies provided by [selection of one or more by the ST author: *user, administrator,* [assignment by the ST author*: other role(s) defined*]].

Dependencies: FDP_DAU_CPI_(EXP).1

FDP_DAU_CPO_(EXP).2 Certification path output – basic policy

Hierarchical to: No other components.

FDP_DAU_CPO_(EXP).2.1 The TSF shall output the certificate policies using the following rule: intersection of certificatePolicies extensions in all the certificates in certification path and initial-certificate-policies.

Dependencies: FDP_DAU_CPO_(EXP).1

5.2.3 Certification Path Validation – Policy Mapping Package

The security functional requirements in this package address certificate path processing, including the processing of the following certificate policies related extensions: policyMapping, inhibitAnyPolicy, and policyConstraints. This package is dependent upon the Certification Path Validation – Basic package and the Certification Path Validation – Basic Policy package.

5.2.3.1 Class FDP – User Data Protection

FDP_DAU_CPI_(EXP).3 Certification path initialisation – policy mapping

Hierarchical to: No other components.

FDP_DAU_CPI_(EXP).3.1 The TSF shall use the explicit-policy-indicator, policy-mapping-inhibit-indicator, inhibit-any-policy-indicator provided by [selection of one or more by the ST author: *user, administrator,* [assignment by the ST author*: other role defined*]].

Dependencies: FDP_DAU_CPI_(EXP).2

FDP_DAU_CPV_(EXP).3 Intermediate certificate processing – policy mapping

Hierarchical to: No other components.

FDP_DAU_CPV_(EXP).3.1 The TSF shall use the intermediate certificate to update the following state variables in accordance with X.509 Standard:

a) explicit-policy-indicator

b) policy-mapping-inhibit-indicator

c) inhibit-any-policy-indicator

Dependencies: FDP_DAU_CPV_(EXP).2

FDP_DAU_CPO_(EXP).3 Certification path output – policy mapping

Hierarchical to: No other components.

FDP_DAU_CPO_(EXP).3.1 The TSF shall perform policy processing in accordance with X.509 standard.

FDP_DAU_CPO_(EXP).3.2 The TSF shall map policies in the calculation of the policies intersection if and only if policy-mapping-inhibit-indicator is not set.

FDP_DAU_CPO_(EXP).3.3 During the calculation of the policy intersection, the TSF shall match any-policy to all policies if and only if inhibit-any-policy-indicator is not set.

FDP_DAU_CPO_(EXP).3.4 The TSF shall output certification path failure if the intersection of certificatePolicies (as modified by policy mapping and inhibit-any-policy) is null and explicit-policy-indicator is set.

FDP_DAU_CPO_(EXP).3.5 The TSF shall output certification path failure if the intersection of certificatePolicies (as modified by policy mapping and inhibit-any-policy) and initial-certificate-policies is null and explicit-policy-indicator is set.

FDP_DAU_CPO_(EXP).3.6 The TSF shall output policy mapping history.

FDP_DAU_CPO_(EXP).3.7 The TSF shall output policy qualifiers applicable to output policies.

Dependencies: FDP_DAU_CPO_(EXP).2

5.2.4 Certification Path Validation – Name Constraints Package

The security functional requirements in this package address certificate path processing, including the processing of the nameConstraints extension. This package is dependent upon the Certification Path Validation – Basic package.

5.2.4.1 Class FDP – User Data Protection

FDP_DAU_CPI_(EXP).4 Certification path initialisation – names

Hierarchical to: No other components.

FDP_DAU_CPI_(EXP).4.1 The TSF shall initialize the following: permitted-subtrees = ∞, excluded-subtrees = ∅

Dependencies: FDP_DAU_CPI_(EXP).1

FDP_DAU_CPV_(EXP).4 Certificate processing – name constraints

Hierarchical to: No other components.

FDP_DAU_CPV_(EXP).4.1The TSF shall reject a certificate if any one of the following is not satisfied:

a) subject DN is in at least one of the permitted-subtrees for DN;

b) subject DN is in none of the excluded-subtrees for DN;

c) each hierarchical name form of type [selection of one or more by the ST author: *DN, RFC-822, URL,* [assignment by the ST author*: other hierarchical name forms*]] in the subjectAlternateName field is in at least one of the permitted-subtrees for that name form; or

d) each hierarchical name form of type [selection of one or more by the ST author: *DN, RFC-822, URL,* [assignment by the ST author*: other hierarchical name forms*]] in the subjectAlternateName field is in none of the excluded-subtrees for that name form

Dependencies: FDP_DAU_CPV_(EXP).1

FDP_DAU_CPV_(EXP).5 Intermediate Certificate processing – name constraints

Hierarchical to: No other components.

FDP_DAU_CPV_(EXP).5.1 The TSF shall use the intermediate certificate to update the following states:

a) permitted-subtrees

b) excluded-subtrees

Dependencies: FDP_DAU_CPV_(EXP).2

5.2.5 PKI Signature Generation Package

The PKI Signature Generation package invokes a cryptographic module for digital signature generation. The package functionality includes generation of signature information that identifies the signer and is useful in efficient signature verification.

5.2.5.1 Class FDP – User Data Protection

FDP_ETC_SIG_(EXP).1 Export of PKI Signature

Hierarchical to: No other component

FDP_ETC_SIG_(EXP).1.1 The TSF shall invoke the cryptographic module with the user selected private key to generate digital signature.

FDP_ETC_SIG_(EXP).1.2 The TSF shall include the following information with the digital signature [selection of one or more by the ST author: *hashing algorithm, signature algorithm, signer public key certificate, signer DN, signer subject alternative name, signer subject key identifier,* [assignment by the ST author*: other information*]].

Dependencies: FCS_CRM_FPS_(EXP).1

5.2.6 PKI Signature Verification Package

The PKI Signature Verification package processes and verifies the signature information, and invokes a cryptographic module to verify digital signatures. This package is dependent upon the Certification Path Validation – Basic package. The signature verification package uses the Certification Path Validation package data as input.

5.2.6.1 Class FDP – User Data Protection

FDP_ITC_SIG_(EXP).1 Import of PKI Signature

Hierarchical to no other component

FDP_ITC_SIG_(EXP).1.1 The TSF shall use the following information from the signed data [selection of one or more by the ST author: *hashing algorithm, signature algorithm, signer public key certificate, signer DN, signer subject alternative name, signer subject key identifier,* [assignment by the ST author*: other information*]] during signature verification.

Dependencies: None

FDP_DAU_SIG_(EXP).1 Signature Blob Verification

Hierarchical to: No other components.

FDP_DAU_SIG_(EXP).1.1 The TSF shall invoke the cryptographic module with the following information from Certification Path Validation to verify digital signature on signed data: subject public key algorithm, subject public key, subject public key parameters.

FDP_DAU_SIG_(EXP).1.2 The TSF shall verify that the keyUsage extension output from the Certification Path Validation has the [selection by the ST author: *nonRepudiation, digitalSignature, nonRepudiation or digitalSignature, nonRepudiation and digitalSignature*] bit set.

FDP_DAU_SIG_(EXP).1.3 The TSF shall apply the following additional checks [selection of one or more by the ST author:

a) *Match the subject DN from the Certification Path Validation with that in the signed data.*

b) *Match the subject alternative name from the Certification Path Validation with that in the signed data.*

c) *Verify that the extendedKeyUsage from Certification Path Validation contains an OID for the PKE application or anyExtendedKeyUsage OID*.

d) [assignment by the ST author: *other checks defined*]].

Dependencies: FCS_CRM_FPS_(EXP).1, FDP_DAU_CPO_(EXP).1

5.2.7 PKI Encryption using Key Transfer Algorithms Package

This package supports the performance of public key encryption using key transfer algorithms such as RSA. Certification path validation is used to ensure that the correct public key of the decrypting party is used. This package is dependent upon the Certification Path Validation – Basic package.

5.2.7.1 Class FDP – User Data Protection

FDP_ETC_ENC_(EXP).1 Export of PKI Encryption – Key Transfer Algorithms

Hierarchical to: No other component

FDP_ETC_ENC_(EXP).1.1 The TSF shall include the following information with the encrypted data [selection of one or more by the ST author: *key encryption algorithm, data encryption algorithm, decryptor key identifier,* [assignment by the ST author*: other information*]].

FDP_ETC_ENC_(EXP).1.2 The TSF shall invoke the cryptographic module with the following information from Certification Path Validation to create encrypted data: subject public key algorithm, subject public key, subject public key parameters.

Dependencies: FCS_CRM_FPS_(EXP).1, FDP_DAU_CPO_(EXP).1

FDP_DAU_ENC_(EXP).1 PKI Encryption Verification – Key Transfer

Hierarchical to: No other components.

FDP_DAU_ENC_(EXP).1.1 The TSF shall verify that the keyUsage output from Certification Path Validation contains keyEncipherment bit set.

FDP_DAU_ENC_(EXP).1.2 The TSF shall apply the following additional checks [selection of one or more by the ST author:

a) *Match the subject DN from the Certification Path Validation with that of the subject of interest.*

b) *Match the subject alternative name from the Certification Path Validation with that of the subject of interest.*

c) *Verify that the extendedKeyUsage from Certification Path Validation contains an OID for the PKE application or anyExtendedKeyUsage OID*.

d) [assignment by the ST author: *other checks defined*]].

Dependencies: FDP_DAU_CPO_(EXP).1

Application Note: *This component is used to verify that the correct public key is used during encryption.*

5.2.8 PKI Encryption using Key Agreement Algorithms Package

This package supports the performance of public key encryption using key calculation algorithms such as DH or ECDH. Certification path validation is included to ensure that the correct public key of the decrypting party is used. This package is dependent upon the Certification Path Validation – Basic package.

5.2.8.1 Class FDP – User Data Protection

FDP_ETC_ENC_(EXP).2 Export of PKI Encryption – Key Agreement Algorithms

Hierarchical to: FDP_ETC_ENC_(EXP).1

FDP_ETC_ENC_(EXP).2.1 The TSF shall include the following information with the encrypted data [selection of one or more by the ST author: *key encryption algorithm, data encryption algorithm, decryptor key identifier,* [assignment by the ST author*: other information*]].

FDP_ETC_ENC_(EXP).2.2 The TSF shall invoke the cryptographic module with the following information from Certification Path Validation to create encrypted data: subject public key algorithm, subject public key, subject public key parameters.

FDP_ETC_ENC_(EXP).2.3 The TSF shall include the following additional information with the encrypted data [selection of one or more by the ST author: *encryptor public key certificate, encryptor DN, encryptor subject alternative name, encryptor subject key identifier,* [assignment by the ST author*: other information*]].

Dependencies: FCS_CRM_FPS_(EXP).1, FDP_DAU_CPO_(EXP).1

FDP_DAU_ENC_(EXP).2 PKI Encryption Verification – Key Agreement, Subject, Decryptor

Hierarchical to: No other components.

FDP_DAU_ENC_(EXP).2.1 The TSF shall verify that the keyUsage output from Certification Path Validation contains keyAgreement bit set.

FDP_DAU_ENC_(EXP).2.2 The TSF shall apply the following additional checks [selection of one or more by the ST author:

a) Match the subject DN from the Certification Path Validation with that of the decryptor.

b) Match the subject alternative name from the Certification Path Validation with that of the decryptor.

c) *Verify that the extendedKeyUsage from Certification Path Validation is contains the OID for the PKE application or anyExtendedKeyUsage OID*.

d) [assignment by ST author: *other checks defined*]].

Dependencies: FDP_DAU_CPO_(EXP).1

Application Note: *This component is used to verify that the correct public key is used during encryption.*

5.2.9 PKI Decryption using Key Transfer Algorithms Package

This package supports the performance of public key decryption using key transfer algorithms such as RSA. Since only the decrypting party's private key is used, this package does not depend upon certificate path processing.

5.2.9.1 Class FDP – User Data Protection

FDP_ITC_ENC_(EXP).1 Import of PKI Encryption – Key Transfer Algorithms

Hierarchical to: No other components

FDP_ITC_ENC_(EXP).1.1 The TSF shall invoke the cryptographic module with the following information from the encrypted data [selection of one or more by the ST author: *key encryption algorithm, data encryption algorithm, decryptor key identifier,* [assignment by the ST author*: other information*]] to perform decryption.

Dependencies: FCS_CRM_FPS_(EXP).1

5.2.10 PKI Decryption using Key Agreement Algorithms Package

This package supports the performance of public key decryption using key calculation algorithms such as DH or ECDH. This package is dependent upon the Certification Path Validation – Basic package.

5.2.10.1 Class FDP – User Data Protection

FDP_ITC_ENC_(EXP).2 Import of PKI Encryption – Key Agreement Algorithms

Hierarchical to: FDP_ITC_ENC_(EXP).1

FDP_ITC_ENC_(EXP).2.1 The TSF shall invoke the cryptographic module with the following information from the encrypted data [selection of one or more by the ST author: *key encryption algorithm, data encryption algorithm, decryptor key identifier,* [assignment by the ST author*: other information*]] to perform decryption.

FDP_ITC_ENC_(EXP).2.2 The TSF shall invoke the cryptographic module with the following additional information from Certification Path Validation during decryption: subject public key algorithm, subject public key, subject public key parameters.

FDP_ITC_ENC_(EXP).2.3 The TSF shall use the following additional information from the encrypted data [selection of one or more by the ST author: *encryptor public key certificate, encryptor DN, encryptor subject alternative name, encryptor subject key identifier,* [assignment by the ST author*: other information*]].

Dependencies: FCS_CRM_FPS_(EXP).1, FDP_DAU_CPO_(EXP).1

FDP_DAU_ENC_(EXP).3 PKI Encryption Verification – Key Agreement, Subject, Encryptor

Hierarchical to: No other components.

FDP_DAU_ENC_(EXP).3.1 The TSF shall verify that the keyUsage output from Certification Path Validation contains keyAgreement bit set.

FDP_DAU_ENC_(EXP).3.2 The TSF shall apply the following additional checks [selection of one or more by the ST author:

a) *Match the subject DN from the Certification Path Validation with that of the encryptor.*

b) *Match the subject alternative name from the Certification Path Validation with that of the encryptor.*

c) *Verify that the extendedKeyUsage from Certification Path Validation contains the OID for the PKE application or anyExtendedKeyUsage OID*.

d) [assignment by the ST author: *other checks defined*]].

Dependencies: FDP_DAU_CPO_(EXP).1

Application Note: *This component is used to verify that the correct public key is used during decryption.*

5.2.11 PKI Based Entity Authentication Package

This package provides for the use of PKI as an entity authentication service. The identification and authentication (I&A) requirements in this package have a different purpose than I&A requirements for the IT Environment in Section 5.1. The IT Environment requirements in Section 5.1 are always required and are used to manage and use the cryptographic keys, whereas this PKI Based Entity Authentication package is used when the PKE application (TOE) performs entity authentication (e.g., Secure Socket Layer (SSL), Transport Layer Security (TLS), etc.). The following characteristics are valid for the PP or ST regardless of the assurance level:

- This package is used to permit the use of a PKI based entity authentication standard for identification and authentication of a user. The standard may or may not determine the authentication failure, selection of secrets, and authentication feedback requirements. Thus, FIA_AFL and FIA_SOS families, and FIA_UAU.7 components were not selected for inclusion in this package.
- This package shall be used for initial authentication of the entity. A dependent package (Continuous Authentication) shall be used for continuous authentication of the protocol, command, packets etc.
- This package only requires a user to authenticate to the TOE. For two-way authentication (e.g., client and server) when each TOE includes the package for authentication of the other, two-way authentication is achieved. In addition, the specification of the standard (e.g., SSL v3) may imply two-way authentication.

This package is dependent upon the Certification Path Validation – Basic package.

5.2.11.1 Class FIA – Identification and Authentication

FIA_UAU.1 Timing of authentication

Hierarchical to: No other components

FIA_UAU.1.1 The TSF shall allow [assignment by the ST author*: list of TSF mediated actions*] on behalf of the user to be performed before the user is authenticated.

FIA_UAU.1.2 The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

Dependencies: FIA_UID.1 Timing of identification

FIA_UAU.4 Single-use authentication mechanisms

Hierarchical to: No other components

FIA_UAU.4.1 The TSF shall prevent reuse of authentication data related to [selection of one or more by the ST author: *FIPS 196, SSL v2, SSL v3, TLS,* [assignment by the ST author: *other PKI based authentication mechanism(s)*]].

Dependencies: No dependencies

FIA_UAU_SIG_(EXP).1 Entity Authentication

Hierarchical to: No other components.

FIA_UAU_SIG_(EXP).1.1 The TSF shall invoke the cryptographic module with the following information from Certification Path Validation to verify signature on response from the entity to the challenge from the TSF: subject public key algorithm, subject public key, subject public key parameters.

FIA_UAU_SIG_(EXP).1.2 The TSF shall verify that the keyUsage output from Certification Path Validation contains digitalSignature bit set.

FIA_UAU_SIG_(EXP).1.3 The TSF shall apply the following additional checks [selection of one or more by the ST author:

a) Match the subject DN from the Certification Path Validation with the entity being authenticated.

b) Match the subject alternative name from the Certification Path Validation with the entity being authenticated.

c) [assignment by the ST author: *other checks defined*]].

Dependencies: FCS_CRM_FPS_(EXP).1, FDP_DAU_CPO_(EXP).1

FIA_UID.1 Timing of identification

Hierarchical to: No other components

FIA_UID.1.1 The TSF shall allow [assignment by the ST author*: list of TSF-mediated actions*] on behalf of the user to be performed before the user is identified.

FIA_UID.1.2 The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.

Dependencies: No dependencies

5.2.12 Online Certificate Status Protocol Client Package

This package allows for making Online Certificate Status Protocol (OSCP) requests and validating OCSP responses. This package permits the use of the OCSP Responder as a trust anchor, as the CA, or an end entity authorized to sign OCSP responses. The ST author can assign additional rules to process OCSP extensions. If the OCSP implementation establishes trust in the OCSP responder by performing Certificate Path Validation, then CPV – Basic and other CPV packages may also be applicable, depending upon the implementation.

5.2.12.1 Class FDP – User Data Protection

FDP_DAU_OCS_(EXP).1 Basic OCSP Client

Hierarchical to: No other component

FDP_DAU_OCS_(EXP).1.1 The TSF shall formulate the OCSP requests in accordance with PKIX RFC 2560.

FDP_DAU_OCS_(EXP).1.2 The OCSP request shall contain the following extensions: [selection of one or more by the ST author: *none, nonce,* [assignment by the ST author: *other extensions*]].

FDP_DAU_OCS_(EXP).1.3 The TSF shall obtain the public key, algorithm, and public key parameters of the OCSP Responder from [selection of one by the ST author: *trust anchor, certificate signing CA, OCSP responder certificate,* [assignment by ST author: *other sources*]].

FDP_DAU_OCS_(EXP).1.4 The TSF shall perform the following additional function [selection of one by the ST author:

a) none; or

b) *establish trust in OCSP responder certificate using* [selection of one or more by the ST author: *certification path validation – basic, certification path validation – basic policy, certification path validation –policy mapping, certification path validation – name constraint*]].

FDP_DAU_OCS_(EXP).1.5 The TSF shall invoke the cryptographic module to verify signature on the OCSP response using trusted public key, algorithm, and public key parameters of the OCSP responder.

FDP_DAU_OCS_(EXP).1.6 The TSF shall verify that if the OCSP responder certificate contains extendedKeyUsage extension, the extension contains the PKIX OID for ocsp-signing or the anyExtendedKeyUsage OID.

FDP_DAU_OCS_(EXP).1.7 The TSF shall match the responderID in the OCSP response with the corresponding information in the responder certificate

FDP_DAU_OCS_(EXP).1.8 The TSF shall match the certID in a request with certID in singleResponse.

FDP_DAU_OCS_(EXP).1.9 The TSF shall reject the OCSP response for an entry if all of the following are true:

a) time checks are not overridden;

b) [selection of one by the ST author: always, TOI > producedAt + x where x is provided by [selection by the ST author: user, administrator, [assignment by the ST author: other role(s) defined]]];

c) [selection of one by the ST author: *always, TOI > thisUpdate for entry + x where x is provided by* [selection by the ST author: *user, administrator,* [assignment by the ST author: *other role(s) defined*]]]; and

d) [selection of one by the ST author: *always, TOI > nextUpdate for entry + x if nextUpdate is present and where x is provided by* [selection by the ST author: *user, administrator,* [assignment by the ST author: *other role(s) defined*]]].

FDP_DAU_OCS_(EXP).1.10 The TSF shall permit [selection of one or more by the ST author: *user, administrator,* [assignment by the ST author: *other role(s) defined*], *none*] to override time checks.

FDP_DAU_OCS_(EXP).1.11 The TSF shall reject OCSP response if the response contains “critical” extension(s) that TSF does not process.

FDP_DAU_OCS_(EXP).1.12 The TSF shall perform the following additional checks [selection of one or more by the ST author:

a) none,

b) request nonce = response nonce,

c) [assignment by ST author: *other rule(s)*]].

Dependencies: FCS_CRM_FPS_(EXP).1, FPT_STM.1

5.2.13 Certificate Revocation List (CRL) Validation Package

This package is used for validating a CRL. This version of the document does not require processing of CRL issuing distribution point (IDP) CRL or delta CRL. Future versions may include that capability by codifying Annex B of X.509 standard.

It should be noted that this package may be used to process a CRL that is pointed to by a CRL Distribution Point (CRLDP) extension in a certificate as long as the CRL is a full CRL, indicated by the absence of IDP and deltaCRLIndicator extensions.

This package permits the use of the same public key for CRL signature verification as the one used for verifying the signature on the certificate, but does not mandate it. In other words, a compliant implementation can use that or develop a certification path. If the compliant implementation develops a certification path, then CPV – Basic and other CPV packages may also be applicable, depending upon the implementation..

The ST author can assign additional rules to process Issuing Distribution Point CRL and Delta CRL.

5.2.13.1 Class FDP – User Data Protection

FDP_DAU_CRL_(EXP).1 Basic CRL Checking

Hierarchical to no other component

FDP_DAU_CRL_(EXP).1.1 The TSF shall obtain the CRL from [selection of one or more by the ST author: *local cache, repository, location pointed to by the CRL DP in public key certificate of interest, user*, [assignment: *other locations defined by the ST author*]].

FDP_DAU_CRL_(EXP).1.2 The TSF shall obtain the trusted public key, algorithm, and public key parameters of the CRL issuer.

FDP_DAU_CRL_(EXP).1.3 The TSF shall invoke the cryptographic module to verify signature on the CRL using trusted public key, algorithm, and public key parameters of the CRL issuer.

FDP_DAU_CRL_(EXP).1.4 The TSF shall verify that if a critical keyUsage extension is present in CRL issuer certificate, cRLSign bit in the extension is set in the certificate.

FDP_DAU_CRL_(EXP).1.5 The TSF shall match the issuer field in the CRL with what it assumes to be the CRL issuer.

FDP_DAU_CRL_(EXP).1.6 The TSF shall reject the CRL if all of the following are true:

a) Time check are not overridden;

b) [selection of one by the ST author: *always, TOI > thisUpdate + x where x is provided by* [selection by the ST author: *user, administrator,* [assignment by the ST author: *other role(s) defined*]]]; and

c) [selection of one by the ST author: *always, TOI > nextUpdate + x if nextUpdate is present and where x is provided by* [selection by the ST author: *user, administrator,* [assignment by the ST author: *other role(s) defined*]]].

FDP_DAU_CRL_(EXP).1.7 The TSF shall permit [selection by the ST author: *user, administrator,* [assignment by the ST author: *other role(s) defined*]*, none*] to override time checks.

FDP_DAU_CRL_(EXP).1.8 The TSF shall reject CRL if the CRL contains “critical” extension(s) that TSF does not process.

FDP_DAU_CRL_(EXP).1.9 The TSF shall perform the following additional checks [selection of one or more by the ST author:

a) none,

b) [assignment by ST author: *other rule(s)*]].

Dependencies: FCS_CRM_FPS_(EXP).1, FPT_STM.1

Application Note: The trusted public key, algorithm, and public key parameters of the CRL issuer should normally be the same as those used for verifying signature on the certificate being checked for revocation. If not, at least certificate path development – basic can be used to obtain the public key.

5.2.14 Audit Package

This package is used in order to generate and protect audit events relevant to the PKE applications (TOEs). Examples of PKE application audit events are:

- Signature verification success, date and time, and policies under which signatures were valid
- Signature verification failure, date and time, cause of failure (signature on the object failed, certification path failure, policy failure, etc.)
- User override events (CRL availability, accept policy failure, accept null policy, accept other policy, etc.)

The security functional requirements below provide an accurate and complete list of auditable events.

The dependencies for this package are satisfied by the IT Environment functional requirements. Examples of these dependencies include:

- Reliable time stamp
- User identification

The Rationale section of this PP family provides accurate and complete dependency analysis. Note that many of the audit requirements are not listed in this package since the TOE must use the IT Environment Audit Log facility to protect and manage the audit data.

5.2.14.1 Class FAU – Security Audit

FAU_GEN.1-NIAP-0407:2 Audit data generation – TOE

Hierarchical to: No other component

FAU_GEN.1.1-NIAP-0407;2 The TSF shall be able to generate an audit record of the following auditable events:

a) Start-up and shutdown of the audit functions;

b) All auditable events listed in Table 5-5; and

c) [selection: [assignment: *events at a basic level of audit introduced by the inclusion of additional SFRs determined by the ST author*], [assignment: *events commensurate with a basic level of audit introduced by the inclusion of explicit requirements determined by the ST author*], “*no additional events*”].

FAU_GEN.1.2-NIAP-0410;2 The TSF shall record within each audit record at least the following information:

c) Date and time of the event, type of event, subject identity (if applicable), and the outcome (success or failure) of the event; and

d) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, information specified in column three of Table 5-5 below.

Dependencies: FPT_STM.1 Reliable time stamps

Table 5.5 – TOE Auditable Events

Requirement	Auditable Events	Additional Audit Record Contents
FDP_CPD_(EXP).1	Success or failure to build path	For success, matching rules bypassed
FDP_DAU_CPI_(EXP).1	None	
FDP_DAU_CPV_(EXP).1	Success or failure of certificate processing Bypass of revocation status checking	For failure, reason(s) for failure
FDP_DAU_CPV_(EXP).2	Success or failure of certificate processing	For failure, reason(s) for failure
FDP_DAU_CPO_(EXP).1	None	
FDP_DAU_CPI_(EXP).2	None	
FDP_DAU_CPO_(EXP).2	None	
FDP_DAU_CPI_(EXP).3	None	
FDP_DAU_CPV_(EXP).3	None	
FDP_DAU_CPO_(EXP).3	Success or failure	
FDP_DAU_CPI_(EXP).4	None	
FDP_DAU_CPV_(EXP).4	Success or failure	
FDP_DAU_CPV_(EXP).5	None	
FDP_ETC_SIG_(EXP).1	Invocation of the function	
FDP_ITC_SIG_(EXP).1	None	
FDP_DAU_SIG_(EXP).1	Success or failure	In case of failure, reason for failure

Requirement	Auditable Events	Additional Audit Record Contents
FDP_ETC_ENC_(EXP).1	None	
FDP_DAU_ENC_(EXP).1	Success or failure	In case of failure, reason for failure
FDP_ETC_ENC_(EXP).2	Invocation of the function	
FDP_DAU_ENC_(EXP).2	Success or failure	In case of failure, reason for failure
FDP_ITC_ENC_(EXP).1	Invocation of the function	
FDP_ITC_ENC_(EXP).2	Invocation of the function	
FDP_DAU_ENC_(EXP).3	Success or failure	In case of failure, reason for failure
FIA_UAU.1	All use of authentication mechanism	
FIA_UAU.4	Attempt to reuse authentication data	
FIA_UAU_SIG_(EXP).1	Success or failure	In case of failure, reason for failure
FIA_UID.1	All use of identification mechanism	User identity
FDP_DAU_OCS_(EXP).1	Rejection of OCSP response Override time checks	Reason for rejection
FDP_DAU_CRL_(EXP).1	Rejection of CRL Override time checks	Reason for rejection
FAU_GEN.1-NIAP-0407:2	None	
FAU_GEN.2-NIAP-0410:2	None	
FIA_UAU.6	All re-authentication attempts	

In the table above, if a component is included in the ST, then and only then the audit record event for that component must be generated.

FAU_GEN.2-NIAP-0410:2 User identity association – TOE

Hierarchical to: No other components.

FAU_GEN.2.1-NIAP-0410;2 For audit events resulting from actions of identified users, the TSF shall be able to associate each auditable event with the identity of the user that caused the event.

Dependencies: FAU_GEN.1 Audit data generation

FIA_UID.1 Timing of identification

5.2.15 Continuous Authentication Package

This package provides for the use of the continuous authentication service of an entity. This package is dependent on the PKI Based Entity Authentication Package and the CPV – Basic package. This package is used for continuous authentication of an entity. The following characteristics are valid for a PP or ST regardless of the assurance level:

- This package only requires an user to authenticate to the TOE. For two-way authentication (e.g., client and server) when each TOE includes the package for authentication of the other, two-way authentication is achieved. In addition, the specification of the standard (e.g., SSL v3) may imply two-way authentication.

5.2.15.1 Class FIA – Identification and Authentication

FIA_UAU.6 Re-authenticating

Hierarchical to: No other components

FIA_UAU.6.1	The TSF shall re-authenticate the user under the conditions [selection of one or more by the ST author: *each packet, each command, each transaction,* [assignment by ST author: *list of conditions under which re-authentication is required*]].
Dependencies:	No dependencies
Application Note:	*It is acceptable to use the symmetric session cryptographic key established during the initial authentication in conjunction with integrity and authentication functions such as HMAC for re-authentication of commands, packets, transactions, etc.*

5.3 Strength of Function Requirement

The strength of function for the IT Environment authentication function is assumed to be SOF-basic. This SOF level matches an attack potential of low. The strength of cryptographic algorithms is outside the scope of the CC. Strength of function only applies to non-cryptographic, probabilistic or permutational mechanisms. The SOF requirement applies to the identification and authentication functionality within the IT Environment.

5.4 Security Assurance Requirements

The PP/ST author must select exactly one of the following assurance packages:

- Basic Robustness described in Section 5.4.1 below
- EAL 3 with Augmentation described in Section 5.4.2 below
- EAL 4 with Augmentation described in Section 5.4.3 below

Since this family of PPs only requires demonstrable conformance claims, the PP/ST author may augment and/or extend any of the assurance packages selected.

5.4.1 PPs with Basic Robustness Assurance

The PP/ST author may select basic robustness assurance. Basic Robustness TOE is considered sufficient for low threat environments or where compromise of protected information will not have a significant impact on mission objectives. This implies that the motivation of the threat agents will be low in environments that are suitable for TOEs of this robustness. In general, basic robustness results in “good commercial practices” that counter threats based in casual and accidental disclosure or compromise of data protected by the TOE. The basic robustness assurance requirements are based on this principle and consist of EAL 2 augmented with the following additions:

- ALC_FLR.2 Flaw Reporting Procedures
- AVA_MSU.1 Examination of Guidance

The following is a list of the assurance requirements needed for Basic Robustness. These Security Assurance Requirements are drawn from the Common Criteria for Information Technology Security Evaluation, Part 3, dated January 2004, Version 2.2

Table 5.6 – Basic Robustness Assurance Requirements

Class	Assurance Component	Assurance Component Description
Configuration Management	ACM_CAP.2	Configurations items
Delivery and Operation	ADO_DEL.1	Delivery procedures
	ADO_IGS.1	Installation, generation, and start-up procedures
Development	ADV_FSP.1	Informal Functional specification
	ADV_HLD.1	Descriptive high-level design
	ADV_RCR.1	Informal correspondence demonstration
Guidance Documents	AGD_ADM.1	Administrator guidance
	AGD_USR.1	User guidance

Class	Assurance Component	Assurance Component Description
Life Cycle Support	**ALC_FLR.2**	**Flaw Reporting Procedures**
Tests	ATE_COV.1	Evidence of coverage
	ATE_FUN.1	Functional testing
	ATE_IND.2	Independent testing - sample
Vulnerability Assessment	**AVA_MSU.1**	**Examination of guidance**
	AVA_SOF.1	Strength of TOE security function evaluation
	AVA_VLA.1	Developer Vulnerability analysis

5.4.2 PPs with EAL 3 With Augmentation Assurance

The PP/ST author may select assurance components of EAL3 augmented by ALC_FLR.2. EAL 3 with augmentation will be selected when the TOE requires a moderate level of independently assured security and requires a thorough investigation of the TOE and its development without substantial re-engineering. ALC_FLR.2 augmentation is done to ensure compliance with the Basic Robustness assurance requirements. The assurance components are listed in Table 5.7. These Security Assurance Requirements are drawn from the Common Criteria for Information Technology Security Evaluation, Part 3, dated January 2004, Version 2.2

Table 5.7 – EAL 3 with Augmentation Assurance Requirements

Class	Assurance Component	Assurance Component Description
Configuration Management	ACM_CAP.3	Authorisation controls
	ACM_SCP.1	TOE CM coverage
Delivery and Operation	ADO_DEL.1	Delivery procedures
	ADO_IGS.1	Installation, generation, and start-up procedures
Development	ADV_FSP.1	Informal Functional specification
	ADV_HLD.2	Security enforcing high-level design
	ADV_RCR.1	Informal correspondence demonstration
Guidance Documents	AGD_ADM.1	Administrator guidance
	AGD_USR.1	User guidance

Class	Assurance Component	Assurance Component Description
Life Cycle Support	ALC_DVS.1	Identification of security measures
	ALC_FLR.2	**Flaw Reporting Procedures**
Tests	ATE_COV.2	Analysis of coverage
	ATE_DPT.1	Testing: high-level design
	ATE_FUN.1	Functional testing
	ATE_IND.2	Independent testing - sample
Vulnerability Assessment	AVA_MSU.1	Examination of guidance
	AVA_SOF.1	Strength of TOE security function evaluation
	AVA_VLA.1	Developer Vulnerability analysis

5.4.3 PPs with EAL 4 With Augmentation Assurance

The PP/ST author may select assurance components of EAL 4 augmented by ALC_FLR.2. EAL 4 with augmentation will be selected in those circumstances where developers or users require a moderate to high level of independently assured security in conventional commodity TOEs and are prepared to incur additional security-specific engineering costs. EAL4 permits a PKE application developer to gain added assurance from positive security engineering based on good commercial development practices, which, though rigorous, do not require substantial specialist knowledge, skills, and other resources. EAL4 is the highest assurance level at which it is likely to be economically feasible to retrofit to an existing product line. ALC_FLR.2 augmentation is done to ensure compliance with the Basic Robustness assurance requirements. The assurance components are listed in Table 5.8. These Security Assurance Requirements are drawn from the Common Criteria for Information Technology Security Evaluation, Part 3, dated January 2004, Version 2.2

Table 5.8 – EAL 4 with Augmentation Assurance Requirements

Class	Assurance Component	Assurance Component Description
Configuration Management	ACM_AUT.1	Partial CM automation
	ACM_CAP.4	Generation support and acceptance procedures
	ACM_SCP.2	Problem tracking CM coverage
Delivery and Operation	ADO_DEL.2	Detection of modification
	ADO_IGS.1	Installation, generation, and start-up procedures

Class	Assurance Component	Assurance Component Description
Development	ADV_FSP.2	Fully defined external interfaces
	ADV_HLD.2	Security enforcing high-level design
	ADV_IMP.1	Subset of the implementation of the TSF
	ADV_LLD.1	Descriptive low-level design
	ADV_RCR.1	Informal correspondence demonstration
Guidance Documents	AGD_ADM.1	Administrator guidance
	AGD_USR.1	User guidance
Life Cycle Support	ALC_DVS.1	Identification of security measures
	ALC_FLR.2	**Flaw Reporting Procedures**
	ALC_LCD.1	Developer defined life-cycle model
	ALC_TAT.1	Well-defined development tools
Tests	ATE_COV.2	Analysis of coverage
	ATE_DPT.1	Testing: high-level design
	ATE_FUN.1	Functional testing
	ATE_IND.2	Independent testing - sample
Vulnerability Assessment	AVA_MSU.2	Validation of analysis
	AVA_SOF.1	Strength of TOE security function evaluation
	AVA_VLA.2	Independent vulnerability analysis

6 Rationale

This section provides further evidence and explanation to support the certification of this family of PPs.

6.1 Security Objectives Rationale

6.1.1 Base and Environmental Security Objectives Rationale

Table 6.1 maps base assumptions and threats to objectives, demonstrating that all assumptions and threats are mapped to at least one objective. Table 6.2 maps base objectives to threats and assumptions, demonstrating that all objectives are mapped to at least one threat or assumption.

Table 6.1 – Mapping the TOE Base Assumptions and Threats to Objectives

Assumption/Threat	Objectives
A.Configuration	OE.Configuration
A.Low	OE.Low
A.NO_EVIL	OE.NO_EVIL
A.PHYSICAL	OE.PHYSICAL
P.ACCESS_BANNER	OE.DISPLAY_BANNER
P.ACCOUNTABILITY	OE.AUDIT_GENERATION; OE.TIME_STAMPS; OE.TOE_ACCESS; OE.TIME_TOE
P.CRYPTOGRAPHY	OE.CRYPTOGRAPHY
T.AUDIT_COMPROMISE	OE.AUDIT_PROTECTION; OE.RESIDUAL_INFORMATION; OE.SELF_PROTECTION; OE.TOE_PROTECTION
T.CHANGE_TIME	OE.TIME_TOE
T.CRYPTO_COMPROMISE	OE.CRYPTOGRAPHY; OE.PHYSICAL
T.MASQUERADE	OE.TOE_ACCESS
T.POOR_TEST	OE.CORRECT_TSF_OPERATION
T.RESIDUAL_DATA	OE.RESIDUAL_INFORMATION
T.TSF_COMPROMISE	OE.RESIDUAL_INFORMATION; OE.SELF_PROTECTION; OE.TOE_PROTECTION; OE.MANAGE
T.UNATTENDED_SESSION	OE.TOE_ACCESS
T.UNAUTHORIZED_ACCESS	OE.MEDIATE
T.UNIDENTIFIED_ACTIONS	OE.AUDIT_REVIEW; OE.AUDIT_GENERATION; OE.TIME_STAMPS; OE.TIME_TOE

A.NO_EVIL states that administrators are non-hostile, appropriately trained and follow all administrator guidance. This assumption is mapped to:

- **OE.NO_EVIL**, which states that sites using the TOE will ensure that administrators are non-hostile, appropriately trained and follow all administrator guidance.

A.PHYSICAL states that environment provides the TOE with appropriate physical security, commensurate with the value of the IT assets protected by the TOE.. This assumption is mapped to:

- **OE.PHYSICAL**, which states that the non-IT environment will provide an acceptable level of physical security so that the TOE cannot be tampered with or be subject to side channel attacks such as the various forms of power analysis and timing analysis.

A.Configuration states that the TOE will be properly installed and configured. This assumption is mapped to:

- **OE.Configuration**, which states that the TOE shall be installed and configured properly for starting up the TOE in a secure state.

A.Low states that the attack potential on the TOE is assumed to be low. A.Low is mapped to:

- **OE.Low,** which states that the Identification and Authentication functions in the TOE will be designed for a minimum attack potential of low as validated by the vulnerability assessment and Strength of Function analyses.

In Table 6.2, the Base Objectives are mapped back to threats and assumptions, thereby demonstrating that every objective is mapped to a threat or assumption. Explanation of the mapping is defined above and is not repeated following Table 6.2. Note, once again, these threats and objectives are included in every PP in this PP family.

P.ACCESS_BANNER states that the IT Environment shall display an initial banner describing restrictions of use, legal agreements, or any other appropriate information to which users consent by accessing the system. This policy is mapped to:

- **OE.DISPLAY_BANNER** which states that the IT Environment will display an advisory warning regarding use of the TOE. **OE.DISPLAY_BANNER** satisfies this policy by ensuring that the TOE displays an administrator configurable banner that provides all interactive users with a warning about the unauthorized use of the TOE

P.ACCOUNTABILITY states that the authorized users of the TOE shall be held accountable for their actions within the TOE. This policy is mapped to:

- **OE.AUDIT_GENERATION** which states that the IT Environment will provide the capability to detect and create records of security-relevant events associated with users. **OE.AUDIT_GENERATION** addresses this policy by providing the administrator with the capability of configuring the audit mechanism to record the actions of a specific user, or review the audit trail based on the identity of the user. Additionally, the administrator's ID is recorded when any security relevant change is made (e.g. access rule modification, start-stop of the audit mechanism, establishment of a trusted channel, etc.).
- **OE.TIME_STAMPS** which states that the IT Environment will provide reliable time stamps and the capability for the administrator to set the time used for these time stamps. **OE.TIME_STAMPS** plays a role in supporting this policy by requiring the IT Environment to provide a reliable time stamp (configured locally

by the Security Administrator or via an external NTP server). The audit mechanism is required to include the current date and time in each audit record. All audit records that include the user ID, will also include the date and time that the event occurred.

- **OE.TIME_TOE** which states that the IT Environment will provide reliable time for the TOE use. **OE.TIME_STAMPS** plays a role in supporting this policy by permitting the TOE to provide reliable time on audit records generated by the TOE.
- **OE.TOE_ACCESS** which states that the IT Environment will provide mechanisms that control a user's logical access to the TOE. **OE.TOE_ACCESS** supports this policy by requiring the IT Environment to identify and authenticate all authorized users prior to allowing any TOE access or any TOE mediated access on behalf of those users.

P.CRYPTOGRAPHY states that only NIST FIPS validated cryptography (methods and implementations) are acceptable for key management (i.e.; generation, access, distribution, destruction, handling, and storage of keys) and cryptographic services (i.e.; encryption, decryption, signature, hashing, key exchange, and random number generation services). This policy is mapped to:

- **OE.CRYPTOGRAPHY** which states The TOE shall use NIST FIPS 140-2 validated cryptographic services provided by the IT Environment. **OE.CRYPTOGRAPHY** satisfies this policy by requiring the IT Environment to implement NIST FIPS validated cryptographic services. These services will provide confidentiality and integrity services as required by the IT Environment and the TOE.

T.AUDIT_COMPROMISE states that a user or process may view audit records, cause audit records to be lost or modified, or prevent future audit records from being recorded, thus masking a user's action. This threat is mapped to:

- **OE.AUDIT_PROTECTION** which states that the IT Environment will provide the capability to protect audit information. **OE.AUDIT_PROTECT** contributes to mitigating this threat by controlling access to the audit trail. Only an administrator is allowed to read the audit trail, no one is allowed to modify audit records, the administrator is the only one allowed to delete the audit trail, and the IT Environment has the capability to prevent auditable actions from occurring if the audit trail is full.
- **OE.RESIDUAL_INFORMATION** which states that the IT Environment will ensure that any information contained in a protected resource within its Scope of Control is not released when the resource is reallocated. **OE.RESIDUAL_INFORMATION** prevents a user not authorized to read the audit trail from access to audit information that might otherwise be persistent in a resource (e.g., memory). By ensuring the IT Environment prevents residual information in a resource, audit information will not become available to any user or process except those explicitly authorized for that data.
- **OE.SELF_PROTECTION** which states that the IT Environment will maintain a domain for its own execution that protects it and its resources from external interference, tampering, or unauthorized disclosure. **OE.SELF_PROTECTION** contributes to countering this threat by ensuring that the IT Environment can protect itself from users. If the IT Environment could not maintain and control its

domain of execution, it could not be trusted to control access to the resources under its control, which includes the audit trail which are always invoked is also critical to the migration of this threat.

- **OE.TOE_PROTECTION** which states The IT Environment will protect the TOE and TOE resources from external interference, tampering, or unauthorized disclosure and modification. **OE.TOE_PROTECTION** contributes to countering this threat by ensuring that the IT Environment can protect TOE. If the TOE could not be protected, it could not be trusted to provide accurate audit information.

T.CHANGE_TIME states that an unauthorized user may change the TSF notion of time resulting in accepting old revocation information or expired certificates. This threat is mapped to:

- **OE.TIME_TOE** which states that the IT Environment will provide reliable time for the TOE use. **OE.TIME_TOE** protects against this threat by ensuring that the IT Environment does not permit users to change the time.

T.CRYPTO_COMPROMISE states that a user or process may cause key, data or executable code associated with the cryptographic functionality to be inappropriately accessed (viewed, modified, or deleted), thus compromising the cryptographic mechanisms and the data protected by those mechanisms. This threat is mapped to:

- **OE.CRYPTOGRAPHY** which states that the TOE shall use NIST FIPS 140-2 validated cryptographic services provided by the IT Environment. **OE.CRYPTOGRAPHY** protects against this threat by ensuring that the cryptography used is sound and has been validated.
- **OE.PHYSICAL** which states that the non-IT environment will provide an acceptable level of physical security so that the TOE cannot be tampered with or be subject to side channel attacks such as the various forms of power analysis and timing analysis. **OE.PHYSICAL** contributes to protection against this threat by providing physical protection from side channel attacks protects against the attempts to compromise the cryptographic mechanisms.

T.MASQUERADE states that a user or process may masquerade as another entity in order to gain unauthorized access to data or TOE resources. This threat is mapped to:

- **OE.TOE_ACCESS** which states that the IT Environment will provide mechanisms that control a user's logical access to the TOE. **OE.TOE_ACCESS** mitigates this threat by controlling the logical access to the TOE and its resources. By constraining how and when authorized users can access the TOE, and by mandating the type and strength of the authentication mechanism this objective helps mitigate the possibility of a user attempting to login and masquerade as an authorized user. In addition, this objective provides the administrator the means to control the number of failed login attempts a user can generate before an account is locked out, further reducing the possibility of a user gaining unauthorized access to the TOE.

T.POOR_TEST states that lack of or insufficient tests to demonstrate that all TOE security functions operate correctly (including in a fielded TOE) may result in incorrect TOE behavior being undiscovered thereby causing potential security vulnerabilities. This threat is mapped to:

- **OE.CORRECT_TSF_OPERATION** which states that the IT Environment will provide the capability to test the TSF to ensure the correct operation of the TSF at a customer's site. **OE.CORRECT_TSF_OPERATION** ensures that once the TOE is installed at a customer's location, the capability exists that the integrity of the TSF (hardware and software) can be demonstrated, and thus providing end users the confidence that the TOE's security policies continue to be enforced.

T.RESIDUAL_DATA states that a user or process may gain unauthorized access to data through reallocation of TOE resources from one user or process to another. This threat is mapped to:

- **OE.RESIDUAL_INFORMATION** which states that the IT Environment will ensure that any information contained in a protected resource within its Scope of Control is not released when the resource is reallocated.
  OE.RESIDUAL_INFORMATION counters this threat by ensuring that TSF data and user data is not persistent when resources are released by one user/process and allocated to another user/process.

T.TSF_COMPROMISE states that a user or process may cause, through an unsophisticated attack, TSF data, security attributes, or executable code to be inappropriately accessed (viewed, modified, or deleted). This threat is mapped to:

- **OE.RESIDUAL_INFORMATION** which states that the IT Environment will ensure that any information contained in a protected resource within its Scope of Control is not released when the resource is reallocated.
  OE.RESIDUAL_INFORMATION is necessary to mitigate this threat, because even if the security mechanisms do not allow a user to explicitly view TSF data, if TSF data were to inappropriately reside in a resource that was made available to a user, that user would be able to inappropriately view the TSF data
- **OE.SELF_PROTECTION** which states that the IT Environment will maintain a domain for its own execution that protects it and its resources from external interference, tampering, or unauthorized disclosure. **OE.SELF_PROTECTION** is necessary to mitigate this threat to provide the TOE a domain for its own execution that protects itself and its resources from external interference, tampering, or unauthorized disclosure through its own interfaces. This feature in turn ensures that other processes can not interfere with the IT Environment and defeat the IT Environment mechanisms.
- **OE.TOE_PROTECTION** which states that the IT Environment will protect the TOE and TOE resources from external interference, tampering, or unauthorized disclosure and modification. **OE.TOE_PROTECTION** is necessary to mitigate this threat by ensuring that the IT Environment will protect the TOE. This feature ensures that other processes can not defeat the TOE protection mechanisms.
- **OE.MANAGE** which states that the IT Environment will provide all the functions and facilities necessary to support the administrators in their management of the security of the TOE, and restrict these functions and facilities from unauthorized use. **OE.MANAGE** is necessary because an access control policy is not specified to control access to TSF data. This objective is used to dictate who is able to view and modify TSF data, as well as the behavior of TSF functions

T.UNATTENDED_SESSION states that a user may gain unauthorized access to an unattended session. This threat is mapped to:

- **OE.TOE_ACCESS** which states that the IT Environment will provide mechanisms that control a user's logical access to the TOE. **OE.TOE_ACCESS** helps to mitigate this threat by including mechanisms that place controls on user's sessions. User and administrator's sessions are locked. Locking the session reduces the opportunity of someone gaining unauthorized access the session when the console is unattended.

T.UNAUTHORIZED_ACCESS states that a user may gain access to user data for which they are not authorized according to the TOE security policy. This threat is mapped to:

- **OE.MEDIATE** which states that the IT Environment will protect user data in accordance with its security policy. **OE.MEDIATE** ensures that all accesses to user data are subject to mediation, unless said data has been specifically identified as public data**.** The TOE requires successful authentication prior to gaining access to any controlled-access content. By implementing strong authentication to gain access to these services, an attacker's opportunity to successfully conduct a man-in-the-middle and/or password guessing attack is greatly reduced. Lastly, the IT Environment will ensure that all configured enforcement functions (authentication, access control rules, etc.) must be invoked prior to allowing a user to gain access to TOE or TOE mediated services. The IT Environment restricts the ability to modify the security attributes associated with access control rules, access to authenticated and unauthenticated services, etc to the Administrator. This feature ensures that no other user can modify the information flow policy to bypass the intended TOE security policy.

T.UNIDENTIFIED_ACTIONS states that The administrator may not have the ability to notice potential security violations, thus limiting the administrator's ability to identify and take action against a possible security breach. This threat is mapped to:

- **OE.AUDIT_REVIEW** which states that the IT Environment will provide the capability to selectively view audit information. **OE.AUDIT_REVIEW** helps to mitigate this threat by providing the Administrator with a required minimum set of configurable audit events that could indicate a potential security violation. By configuring these auditable events, the IT Environment and TOE monitors the occurrences of these events (e.g. set number of authentication failures, set number of information policy flow failures, self-test failures, etc.).
- **OE.AUDIT_GENERATION** which states that the IT Environment will provide the capability to detect and create records of security-relevant events associated with users. **OE.AUDIT_GENERATION** helps to mitigate this threat by recording actions for later review
- **OE.TIME_STAMPS** which states that the IT Environment will provide reliable time stamps and the capability for the administrator to set the time used for these time stamps. **OE.TIME_STAMPS** helps to mitigate this threat by ensuring that audit records have correct timestamps.
- **OE.TIME_TOE** which states that the IT Environment will provide reliable time for the TOE use. **OE.TIME_STAMPS** plays a role in supporting this policy by permitting the TOE to provide reliable time on audit records generated by the TOE.

Table 6.2 – Mapping the Base Objectives to Threats, Assumptions or OSP

Objective	Threats, Assumption or OSP

Objective	Threats, Assumption or OSP
OE.AUDIT_GENERATION	P.ACCOUNTABILITY; T.UNIDENTIFIED_ACTIONS
OE.AUDIT_PROTECTION	T.AUDIT_COMPROMISE
OE.AUDIT_REVIEW	T.UNIDENTIFIED_ACTIONS
OE.Configuration	A.Configuration
OE.CORRECT_TSF_OPERATION	T.POOR_TEST
OE.CRYPTOGRAPHY	P.CRYPTOGRAPHY; T.CRYPTO_COMPROMISE
OE.DISPLAY_BANNER	P.ACCESS_BANNER
OE.Low	A.Low
OE.MANAGE	T.TSF_COMPROMISE
OE.MEDIATE	T.UNAUTHORIZED_ACCESS
OE.NO_EVIL	A.NO_EVIL
OE.PHYSICAL	A.PHYSICAL. T.CRYPTO_COMPROMISE
OE.RESIDUAL_INFORMATION	T.AUDIT_COMPROMISE; T.RESIDUAL_DATA; T.TSF_COMPROMISE
OE.SELF_PROTECTION	T.AUDIT_COMPROMISE; T.TSF_COMPROMISE
OE.TIME_STAMPS	P.ACCOUNTABILITY; T.UNIDENTIFIED_ACTIONS
OE.TIME_TOE	P.ACCOUNTABILITY; T.CHANGE_TIME; T.UNIDENTIFIED_ACTIONS
OE.TOE_ACCESS	P.ACCOUNTABILITY; T.MASQUERADE; T.UNATTENDED_SESSION
OE.TOE_PROTECTION	T.AUDIT_COMPROMISE; T.TSF_COMPROMISE

6.1.2 Security Objectives Rationale for Packages

The following subsections provide the mapping and rationale for the security objectives and threats associated with each individual package.

6.1.2.1 CPV – Basic Package Security Objectives Rationale

The following tables demonstrate the mapping of threats to objectives and objectives to threats for the CPV – Basic package. Explanatory text is provided below the tables to support the mapping.

Table 6.3 – Mapping of Threats to Objectives for CPV – Basic Package

Threat	Objectives

T.Certificate_Modi	O.Verified_Certificate
T.DOS_CPV_Basic	O.Availability
T.Expired_Certificate	O.Correct_Temporal O.Current_Certificate
T.Untrusted_CA	O.Trusted_Keys
T.No_Crypto	O.Get_KeyInfo
T.Path_Not_Found	O.Path_Find
T.Revoked_Certificate	O.Valid_Certificate
T.User_CA	O.User

T.Certificate_Modi states that an untrusted user may modify a certificate resulting in using a wrong public key. This threat is mapped to:

- **O.Verified_Certificate**, which states that the TSF shall only accept certificates with verifiable signatures.

T.DOS_CPV_Basic states that the revocation information or access to revocation information could be made unavailable, resulting in loss of system availability. This threat is mapped to:

- **O.Availability**, which states that the TSF shall continue to provide security services even if revocation information is not available.

T.Expired_Certificate states that an expired (and possibly revoked) certificate as of TOI could be used for signature verification. This threat is mapped to:

- **O.Correct_Temporal**, which states that the TSF shall provide accurate temporal validation results.
- **O.Current_Certificate**, which states that the TSF shall only accept certificates that are not expired as of TOI.

T.Untrusted_CA states that an untrusted entity (Certification Authority (CA)) may issue certificates to bogus entities, permitting those entities to assume identity of other legitimate users. This threat is mapped to:

- **O.Trusted_Keys**, which states that the TSF shall use trusted public keys in certification path validation.

T.No_Crypto states that the user public key and related information may not be available to carry out the cryptographic function. This threat is mapped to:

- **O.Get_KeyInfo**, which states that the TSF shall provide the user public key and related information in order to carry out cryptographic functions.

T.Path_Not_Found states that a valid certification path is not found due to lack of system functionality. This threat is mapped to:

- **O.Path_Find**, which states that the TSF shall be able to find a certification path from a trust anchor to the subscriber.

T.Revoked_Certificate states that a revoked certificate could be used as valid, resulting in security compromise. This threat is mapped to:

- **O.Valid_Certificate**, which states that the TSF shall use certificates that are valid, i.e., not revoked.

T.User_CA states that a user could act as a CA, issuing unauthorized certificates. This threat is mapped to:

- **O.User**, which states that the TSF shall only accept certificates issued by a CA.

Table 6.4 maps objectives for the CPV – Basic Package to threats, demonstrating that every objective is mapped to a threat. The mapping is described in the text above and is not repeated following Table 6.4.

Table 6.4 – Mapping of Objectives to Threats for CPV – Basic Package

Objective	Threats
O.Availability	T.DOS_CPV_Basic
O.Correct_Temporal	T.Expired_Certificate
O.Current_Certificate	T.Expired_Certificate
O.Get_KeyInfo	T.No_Crypto
O.Path_Find	T.Path_Not_Found
O.Trusted_Keys	T.Untrusted_CA
O.User	T.User_CA
O.Verified_Certificate	T.Certificate_Modi
O.Valid_Certificate	T.Revoked_Certificate

6.1.2.2 CPV – Basic Policy Package Security Objectives Rationale

The mapping of threats to objectives for the CPV – Basic Policy package is shown in Table 6.5. Text that further supports for the mapping is provided following Table 6.5.

Table 6.5 – Mapping of Threats to Objectives for CPV – Basic Policy Package

Threat	Objectives
T.Unknown_Policies	O.Provide_Policy_Info

T.Unknown_Policies states that the user may not know the policies under which a certificate was issued. This threat is mapped to:

- **O.Provide_Policy_Info**, which states that the TSF shall provide certificate policies for which the certification path is valid.

Table 6.6 maps objectives for the CPV – Basic Policy package to threats, demonstrating that every objective is mapped to a threat. The mapping is described in the text above and is not repeated following Table 6.6.

Table 6.6 – Mapping of Objectives to Threats for CPV – Basic Policy Package

Objective	Threats
O.Provide_Policy_Info	T.Unknown_Policies

6.1.2.3 CPV –Policy Mapping Package Security Objectives Rationale

The mapping of threats to objectives for the CPV – Policy Mapping package is shown in Table 6.7. Text that further supports for the mapping is provided following Table 6.7.

Table 6.7 – Mapping of Threats to Objectives for CPV – Policy Mapping Package

Threat	Objectives
T.Mapping	O.Map_Policies
T.Wrong_Policy_Dec	O.Policy_Enforce

T.Mapping states that the user may accept unacceptable certificates or reject acceptable certificates due to improper certificate policy mapping. This threat is addressed by:

- **O.Map_Policies**, which states that the TSF shall map certificate policies in accordance with user and CA constraints.

T.Wrong_Policy_Dec states that the user may accept certificates that were not generated with the diligence and security acceptable to the user. The user may reject certificates that were generated with the diligence and security acceptable to the user. This threat is addressed by:

- **O.Policy_Enforce**, which states that the TSF shall validate a certification path in accordance with certificate policies acceptable to the user.

Table 6.8 maps objectives for the CPV – Policy Mapping package to threats, demonstrating that every objective is mapped to a threat. The mapping is described in the text above and is not repeated following Table 6.8.

Table 6.8 – Mapping of Objectives to Threats for CPV – Policy Mapping Package

Objective	Threats
O.Map_Policies	T.Mapping
O.Policy_Enforce	T.Wrong_Policy_Dec

6.1.2.4 CPV – Name Constraints Package Security Objectives Rationale

The mapping of threats to objectives for the CPV – Name Constraints Package is shown in Table 6.9. Text that further supports for the mapping is provided following Table 6.9.

Table 6.9 – Mapping of Threats to Objectives for CVP – Name Constraints Package

Threat	Objectives
T.Name_Collision	O.Authorised_Names

T.Name_Collision states that the user may accept certificates from CA where the CA's understanding and the user's understanding of the names differ, i.e., user and CA associate different identity with the same name. This threat is addressed by:

- **O.Authorised_Names**, which states that the TSF shall validate a certificate only if the CA is authorized to issue a certificate to the subject.

Table 6.10 maps objectives for the CPV – Name Constraints Package to threats, demonstrating that every objective is mapped to a threat. The mapping is described in the text above and is not repeated following Table 6.10.

Table 6.10 – Mapping of Objectives to Threats for CPV – Name Constraints Package

Objective	Threats
O.Authorised_Names	T.Name_Collision

6.1.2.5 PKI Signature Generation Package Security Objectives Rationale

The mapping of threats to objectives for the PKI Signature Generation package is shown in Table 6.11. Text that further supports for the mapping is provided following Table 6.11.

Table 6.11 – Mapping of Threats to Objectives for the PKI Signature Generation Package

Threat	Objectives
T.Clueless_PKI_Sig	O.Give_Sig_Hints

T.Clueless_PKI_Sig states that the user may try only inappropriate certificates for PKI signature verification because the signature does not include a hint. This threat is addressed by:

- **O.Give_Sig_Hints**, which states that the TSF shall give hints for selecting correct certificates or keys for PKI signature.

Table 6.12 maps objectives for the PKI Signature Generation package to threats, demonstrating that every objective is mapped to a threat. The mapping is described in the text above and is not repeated following Table 6.12.

Table 6.12 – Mapping of Objectives to Threats for the PKI Signature Generation Package

Objective	Threats
O.Give_Sig_Hints	T.Clueless_PKI_Sig

6.1.2.6 PKI Signature Verification Package Security Objectives Rationale

The mapping of threats to objectives for the PKI Signature Verification package is shown in Table 6.13. Text that further supports for the mapping is provided following Table 6.13.

Table 6.13 – Mapping of Threats to Objectives for the PKI Signature Verification Package

Threat	Objectives
T.Assumed_Identity_PKI_Ver	O.Linkage_Sig_Ver
T.Clueless_PKI_Ver	O.Use_Sig_Hints

T.Assumed_Identity_PKI_Ver states that a user may assume the identity of another user for PKI signature verification. This threat is addressed by:

- **O.Linkage_Sig_Ver**, which states that the TSF shall use the correct user public key for signature verification.

T.Clueless_PKI_Ver states that the user may try only inappropriate certificates for PKI signature verification by ignoring hints in the signature. This threat is addressed by:

- **O.Use_Sig_Hints**, which states that the TSF shall provide hints for selecting correct certificates or keys for signature verification.

Table 6.14 maps objectives The PKI Signature Verification package to threats, demonstrating that every objective is mapped to a threat. The mapping is described in the text above and is not repeated following Table 6.14.

Table 6.14 – Mapping of Objectives to Threats for the PKI Signature Verification Package

Objective	Threats
O.Use_Sig_Hints	T.Clueless_PKI_Ver
O.Linkage_Sig_Ver	T.Assumed_Identity_PKI_Ver

6.1.2.7 PKI Encryption using Key Transfer Algorithms Package Security Objectives Rationale

The mapping of threats to objectives for all of PKI Encryption using Key Transfer Algorithms package is shown in Table 6.15. Text that further supports for the mapping is provided following Table 6.15.

Table 6.15 – Mapping of Threats to Objectives for the PKI Encryption using Key Transfer Algorithms Package

Threat	Objectives
T.Assumed_Identity_WO_En	O.Linkage_Enc_WO
T.Clueless_WO_En	O.Hints_Enc_WO

T.Assumed_Identity_WO_En states that a user may assume the identity of another user in order to perform encryption using Key Transfer algorithms. This threat is addressed by:

- **O.Linkage_Enc_WO**, which states that the TSF shall use the correct user public key for key transfer.

T.Clueless_WO_En states that the user may try only inappropriate certificates in absence of hint for encryption using Key Transfer algorithms. This threat is addressed by:

- **O.Hints_Enc_WO**, which states that the TSF shall provide hints for selecting correct certificates or keys for PKI Encryption using Key Transfer algorithms.

Table 6.16 maps objectives for the PKI Encryption using Key Transfer Algorithms package to threats, demonstrating that every objective is mapped to a threat. The mapping is described in the text above and is not repeated following Table 6.16.

Table 6.16 – Mapping of Objectives to Threats for the PKI Encryption using Key Transfer Algorithms Package

Objective	Threats
O.Hints_Enc_WO	T.Clueless_WO_En
O.Linkage_Enc_WO	T.Assumed_Identity_WO_En

6.1.2.8 PKI Encryption using Key Agreement Algorithms Package Security Objectives Rationale

The mapping of threats to objectives for the PKI Encryption using Key Agreement Algorithms package is shown in Table 6.17. Text that further supports for the mapping is provided following Table 6.17.

Table 6.17 – Mapping of Threats to Objectives for PKI Encryption using Key Agreement Algorithms Package

Threat	Objectives
T.Assumed_Identity_With_En	O.Linkage_Enc_W
T.Clueless_With_En	O.Hints_Enc_W

T.Assumed_Identity_With_En states that a user may assume the identity of another user to perform encryption using Key Agreement Algorithms. This threat is addressed by:

- **O.Linkage_Enc_W**, which states that the TSF shall use the correct user public key for key agreement during encryption.

T.Clueless_With_En states that the user may try only inappropriate certificates for PKI Encryption using Key Agreement algorithms in absence of hint. This threat is addressed by:

- **O.Hints_Enc_W**, which states that the TSF shall provide hints for selecting correct certificates or keys for PKI Encryption using Key Agreement algorithms.

Table 6.18 maps objectives for the PKI Encryption using Key Agreement Algorithms package to threats, demonstrating that every objective is mapped to a threat. The mapping is described in the text above and is not repeated following Table 6.18.

Table 6.18 – Mapping of Objectives to Threats for PKI Encryption using Key Agreement Algorithms Package

Objective	Threats
O.Hints_Enc_W	T.Clueless_With_En
O.Linkage_Enc_W	T.Assumed_Identity_With_En

6.1.2.9 PKI Decryption using Key Transfer Algorithms Package Security Objectives Rationale

The mapping of threats to objectives for the PKI Decryption using Key Transfer Algorithms package is shown in Table 6.19. Text that further supports for the mapping is provided following Table 6.19.

Table 6.19 – Mapping of Threats to Objectives for the PKI Decryption using Key Transfer Algorithms Package

Threat	Objectives
T.Garble_WO_De	O.Correct_KT

T.Garble_WO_De states that the user may not apply the correct key transfer algorithm or private key, resulting in garbled data. This threat is addressed by:

- **O.Correct_KT**, which states that the TSF shall use appropriate private key and key transfer algorithm.

Table 6.20 maps objectives for the PKI Decryption using Key Transfer Algorithms package to threats, demonstrating that every objective is mapped to a threat. The mapping is described in the text above and is not repeated following Table 6.20.

Table 6.20 – Mapping of Objectives to Threats for the PKI Decryption using Key Transfer Algorithms Package

Objective	Threats
O.Correct_KT	T.Garble_WO_De

6.1.2.10 PKI Decryption using Key Agreement Algorithms Package Security Objectives Rationale

The mapping of threats to objectives for the PKI Decryption using Key Agreement Algorithms package is shown in Table 6.21. Text that further supports for the mapping is provided following Table 6.21.

Table 6.21 – Mapping of Threats to Objectives for PKI Decryption using Key Agreement Algorithms Package

Threat	Objectives
T.Assumed_Identity_With_De	O.Linkage_Dec_W
T.Clueless_With_De	O.Hints_Dec_W
T.Garble_With_De	O.Correct_KA

T.Assumed_Identity_With_De states that a user may assume the identity of another user to perform PKI decryption using Key Agreement algorithms. This threat is addressed by:

- **O.Linkage_Dec_W**, which states that the TSF shall use the correct user public key for key agreement during decryption.

T.Clueless_With_De states that the user may try only inappropriate certificates in absence of hint to perform PKI decryption using Key Agreement algorithms. This threat is addressed by:

- **O.Hints_Dec_W**, which states that the TSF shall provide hints for selecting correct certificates or keys for PKI decryption using Key Agreement algorithms.

T.Garble_With_De states that the user may not apply the correct key agreement algorithm or private key, resulting in garbled data. This threat is addressed by:

- **O.Correct_KA**, which states that the TSF shall use appropriate private key and key agreement algorithm.

Table 6.22 maps objectives for the PKI Decryption With DH, ECDH package to threats, demonstrating that every objective is mapped to a threat. The mapping is described in the text above and is not repeated following Table 6.22.

Table 6.22 – Mapping of Objectives to Threats for PKI Decryption using Key Agreement Algorithms Package

Objective	Threats
O.Hints_Dec_W	T.Clueless_With_De
O.Linkage_Dec_W	T.Assumed_Identity_With_De
O.Correct_KA	T.Garble_With_De

6.1.2.11 PKI Based Entity Authentication Package

The mapping of threats to objectives for the PKI Based Entity Authentication package is shown in Table 6.23. Text that further supports the mapping is provided following Table 6.23.

Table 6.23 – Mapping of Threats to Objectives for PKI Based Entity Authentication Package

Threat	Objectives
T.Assumed_Identity_Auth	O.Linkage, O.I&A, O.Limit_Actions_Auth
T.Replay_Entity	O.Single_Use_I&A

T.Assumed_Identity_Auth states that a user may assume the identity of another user to perform entity based authentication. This threat is addressed by:

- **O.Linkage**, which states that the TSF shall use the correct user public for authentication.
- **O.I&A**, which states that the TSF shall uniquely identify all entities, and shall authenticate the claimed identify before granting an entity access to the TOE facilities.
- **O.Limit_Actions_Auth,** which states that the TSF shall restrict the actions an entity may perform before the TSF verifies the identity of the entity.

T.Replay_Entity states that an unauthorized user may replay valid authentication data. This threat is addressed by:

- **O.Single_Use_I&A**, which states that the TSF shall use the I&A mechanism that requires unique authentication information for each I&A.

Table 6.24 maps objectives for the PKI Based Entity Authentication Package to threats, demonstrating that every objective is mapped to a threat. The mapping is described in the text above and is not repeated following Table 6.24.

Table 6.24 – Mapping of Objectives to Threats for PKI Based Entity Authentication Package

Objective	Threats
O.I&A	T.Assumed_Identity_Auth
O.Limit_Actions_Auth	T.Assumed_Identity_Auth
O.Linkage	T.Assumed_Identity_Auth
O.Single_Use_I&A	T.Replay_Entity

6.1.2.12 OCSP Package Security Objectives Rationale

The mapping of threats to objectives for the OCSP package is shown in Table 6.25. Text that further supports the mapping is provided following Table 6.25.

Table 6.25 – Mapping of Threats to Objectives for the OCSP Package

Threat	Objectives
T.DOS_OSCP	O.User_Override_Time_OCSP
T.Replay_OCSP_Info	O.Current_OCSP_Info
T.Wrong_OCSP_Info	O.Accurate_OCSP_Info, O.Auth_OCSP_Info

T.DOS_OSCP states that the OCSP response or access to the OCSP response could be made unavailable, resulting in loss of system availability. This threat is mapped to:

- **O.User_Override_Time_OCSP**, which states that the TSF shall permit the user to override the time checks on the OCSP response.

T.Replay_OCSP_Info states that the user may accept revocation information from well before TOI resulting in accepting revoked certificate for OCSP transactions. This threat is mapped to:

- **O.Current_OCSP_Info**, which states that the TSF accept only OCSP responses current as of TOI .

T.Wrong_OCSP_Info states that the user may accept a revoked certificate or reject a valid certificate due to wrong revocation information. This threat is mapped to:

- **O.Accurate_OCSP_Info**, which states that the TSF shall accept only accurate OCSP responses.
- **O.Auth_OCSP_Info**, which states that the TSF shall accept the OCSP response from an authorized source.

Table 6.26 maps objectives for the OCSP package to threats, demonstrating that every objective is mapped to a threat. The mapping is described in the text above and is not repeated following Table 6.26.

Table 6.26 – Mapping of Objectives to Threats for the OCSP Package

Objective	Threats
O.Accurate_OCSP_Info	T.Wrong_OCSP_Info
O.Auth_OCSP_Info	T.Wrong_OCSP_Info
O.Current_OCSP_Info	T.Replay_OCSP_Info
O.User_Override_Time_OCSP	T.DOS_OCSP

6.1.2.13 CRL Verification Package Security Objectives Rationale

The mapping of threats to objectives for the CRL Verification package is shown in Table 6.27. Text that further supports for the mapping is provided following Table 6.27.

Table 6.27 – Mapping of Threats to Objectives for CRL Verification Package

Threat	Objectives

T.DOS_CRL	O.User_Override_Time_CRL
T.Replay_Revoc_Info_CRL	O.Current_Rev_Info
T.Wrong_Revoc_Info_CRL	O.Accurate_Rev_Info, O.Auth_Rev_Info

T.DOS_CRL states that the CRL or access to the CRL could be made unavailable, resulting in loss of system availability. This threat is mapped to:

- **O.User_Override_Time_CRL**, which states that the TSF shall permit the user to override the time checks on the CRL.

T.Replay_Revoc_Info_CRL states that the user may accept a CRL issued well before TOI resulting in accepting currently revoked certificate. This threat is mapped to:

- **O.Current_Rev_Info**, which states that the TSF shall accept only CRL that are current as TOI.

T.Wrong_Revoc_Info_CRL states that the user may accept a revoked certificate or reject a valid certificate due to wrong revocation information. This threat is mapped to:

- **O.Accurate_Rev_Info**, which states that the TSF shall accept only accurate revocation information.
- **O.Auth_Rev_Info**, which states that the TSF shall accept the revocation information from an authorized source for CRL.

Table 6.28 maps objectives for the CRL Verification package to threats, demonstrating that every objective is mapped to a threat. The mapping is described in the text above and is not repeated following Table 6.28.

Table 6.28 – Mapping of Objectives to Threats for the CRL Verification Package

Objective	Threats
O.Accurate_Rev_Info	T.Wrong_Revoc_Info_CRL
O.Auth_Rev_Info	T.Wrong_Revoc_Info_CRL
O.Current_Rev_Info	T.Replay_Revoc_Info_CRL
O.User_Override_Time_CRL	T.DOS_CRL

6.1.2.14 Audit Package Security Objectives Rationale

The mapping of threats to objectives for the Audit package is shown in Table 6.29. Text that further supports for the mapping is provided following Table 6.29.

Table 6.29 – Mapping of Threats to Objectives for Audit Package

Threat	Objectives
T.PKE_Accountability	O.PKE_Audit

T.PKE_Accountability states that the PKE related audit events cannot be linked to individual actions. This threat is mapped to:

- **O.PKE_Audit**, which states that the TSF shall audit security relevant PKE events. This coupled with the base audit functions provided by the IT Environment mitigate this threat.

Table 6.30 maps objectives for the Audit package to threats, demonstrating that every objective is mapped to a threat. The mapping is described in the text above and is not repeated following Table 6.30.

Table 6.30 – Mapping of Objectives to Threats for Audit Package

Objective	Threats
O.PKE_Audit	T.PKE_Accountability

6.1.2.15 Continuous Authentication Package

The mapping of threats to objectives for the Continuous Authentication package is shown in Table 6.31. Text that further supports the mapping is provided following Table 6.32.

Table 6.31 – Mapping of Threats to Objectives for Continuous Authentication Package

#	Threat	Objectives
1	T.Hijack	O.Continuous_I&A

T.Hijack states that an unauthorized user may hijack an authenticated session. This threat is addressed by:

- **O.Continuous_I&A**, which states that the TSF shall continuously authenticate the entity.

Table 6.32 maps objectives for the Continuous Authentication Package to threats, demonstrating that every objective is mapped to a threat. The mapping is described in the text above and is not repeated following Table 6.24.

Table 6.32 – Mapping of Objectives to Threats for Continuous Authentication Package

#	Objective	Threats
1	O.Continuous_I&A	T.Hijack

6.2 Security Requirements Rationale

In this section, the objectives are mapped to the functional requirements and rationale is provided for the selected assurance level and its components and augmentation.

6.2.1 Functional Security Requirements Rationale

The mapping of all security objectives to functional requirements (components) or to assumptions is provided in Table 6.33. Rationale for the IT Environment functional requirements mapping and for each package are described in separate subsections.

Explicitly stated security functional requirements are IT processing oriented security requirements. These requirements are similar in nature to the security functional requirements in the Common Criteria Part 2. Thus, security assurance requirements from the Common Criteria Part 3 can be used to test the explicitly stated requirements also; no additional assurance requirements beyond those taken from the Common Criteria Part 3 are required.

Table 6.33 – Security Objective to Functional Component Mapping

Objective	Functional Components
Mapping for Objectives for the IT Environment	
OE.AUDIT_GENERATION	FAU_GEN.1-NIAP-0407:1; FAU_GEN.2-NIAP-0410:1; FIA_USB.1; FAU_SEL.1-NIAP-0407
OE.AUDIT_PROTECTION	FAU_SAR.2; FAU_STG.1-NIAP-0429; FAU_STG.NIAP-0429-1; FMT_MOF.1
OE.AUDIT_REVIEW	FAU_SAR.1; FAU_SAR.3
OE.Configuration	Defined in startup and installation guides under ADO_IGS.1
OE.CORRECT_TSF_OPERATION	FPT_TST_SOF_(EXP).1; ATE_COV.2; ATE_DPT.1; ATE_IND.2; ATE_FUN_(EXP).3;
OE.CRYPTOGRAPHY	FCS_CRM_FPS_(EXP).1
OE.DISPLAY_BANNER	FTA_TAB.1
OE.Low	Defined in the SOF analysis and vulnerability assessment.
OE.MANAGE	FMT_MOF.1; FMT_MSA.1; FMT_MSA.3-NIAP-0429; FMT_MTD.1:1; FMT_MTD.1:2; FMT_MTD.1:3; FMT_MTD.1:4; FMT_MTD.1:5; FMT_SMF.1, FMT_SMR.1
OE.MEDIATE	FDP_ACC.1; FDP_ACF.1-NIAP-0407
OE.NO_EVIL	Defined in the Administrator Guide under AGD_ADM.1
OE.PHYSICAL	Defined as part of the physical security policy in AGD_ADM.1 and AGD_USR.1
OE.RESIDUAL_INFORMATION	FDP_RIP.2
OE.SELF_PROTECTION	FPT_SEP.1; FPT_RVM.1
OE.TIME_STAMPS	FPT_STM.1, FMT_SMF.1, FMT_MTD.1:5
OE.TIME_TOE	FPT_STM.1
OE.TOE_ACCESS	FIA_AFL.1; FIA_ATD.1; FIA_UID.2; FIA_UAU.2;

Objective	Functional Components
	FIA_UAU.7; FTA_SSL.1; FTA_SSL.2
OE.TOE_PROTECTION	FPT_SEP_ENV_(EXP).1
Mapping for CPV – Basic Package	
O.Availability	FDP_DAU_CPV_(EXP).1
O.Correct_Temporal	FDP_DAU_CPI_(EXP).1
O.Current_Certificate	FDP_DAU_CPV_(EXP).1
O.Get_KeyInfo	FDP_DAU_CPO_(EXP).1
O.Path_Find	FDP_CPD_(EXP).1
O.Trusted_Keys	FDP_DAU_CPI_(EXP).1
O.User	FDP_DAU_CPV_(EXP).2
O.Verified_Certificate	FDP_DAU_CPV_(EXP).1
O.Valid_Certificate	FDP_DAU_CPV_(EXP).1
Mapping for CPV – Basic Policy Package	
O.Provide_Policy_Info	FDP_DAU_CPI_(EXP).2, FDP_DAU_CPO_(EXP).2
Mapping for CPV – Policy Mapping Package	
O.Map_Policies	FDP_DAU_CPI_(EXP).3, FDP_DAU_CPV_(EXP).3, FDP_DAU_CPO_(EXP).3
O.Policy_Enforce	FDP_DAU_CPI_(EXP).3, FDP_DAU_CPV_(EXP).3, FDP_DAU_CPO_(EXP).3
Mapping for CPV – Name Constraints Package	
O.Authorised_Names	FDP_DAU_CPI_(EXP).4, FDP_DAU_CPV_(EXP).4, FDP_DAU_CPV_(EXP).5
Mapping for PKI Signature Generation Package	
O.Give_Sig_Hints	FDP_ETC_SIG_(EXP).1
Mapping for PKI Signature Verification Package	
O.Use_Sig_Hints	FDP_ITC_SIG_(EXP).1,
O.Linkage_Sig_Ver	FDP_DAU_SIG_(EXP).1
Mapping for PKI Encryption using Key Transfer Algorithms Package	
O.Hints_Enc_WO	FDP_ETC_ENC_(EXP).1
O.Linkage_Enc_WO	FDP_ETC_ENC_(EXP).1, FDP_DAU_ENC_(EXP).1
Mapping for PKI Encryption using Key Agreement Algorithms Package	
O.Hints_Enc_W	FDP_ETC_ENC_(EXP).2
O.Linkage_Enc_W	FDP_ETC_ENC_(EXP).2, FDP_DAU_ENC_(EXP).2
Mapping for PKI Decryption using Key Transfer Algorithms Package	
O.Correct_KT	FDP_ITC_ENC_(EXP).1

Objective	Functional Components
Mapping for PKI Decryption using Key Agreement Algorithms Package	
O.Hints_Dec_W	FDP_ITC_ENC_(EXP).2
O.Linkage_Dec_W	FDP_DAU_ENC_(EXP).3, FDP_ITC_ENC_(EXP).2
O.Correct_KA	FDP_ITC_ENC_(EXP).2
Mapping for PKI Based Entity Authentication Package	
O.I&A	FIA_UAU.1, FIA_UID.1
O.Limit_Actions_Auth	FIA_UAU.1, FIA_UID.1
O.Linkage	FIA_UAU_SIG_(EXP).1
O.Single_Use_I&A	FIA_UAU.4
Mapping for Online Certificate Status Protocol Client Package	
O.Accurate_OCSP_Info	FDP_DAU_OCS_(EXP).1
O.Auth_OCSP_Info	FDP_DAU_OCS_(EXP).1
O.Current_OCSP_Info	FDP_DAU_OCS_(EXP).1
O.User_Override_Time_OCSP	FDP_DAU_OCS_(EXP).1
Mapping for Certificate Revocation List (CRL) Validation Package	
O.Accurate_Rev_Info	FDP_DAU_CRL_(EXP).1
O.Auth_Rev_Info	FDP_DAU_CRL_(EXP).1
O.Current_Rev_Info	FDP_DAU_CRL_(EXP).1
O.User_Override_Time_CRL	FDP_DAU_CRL_(EXP).1
Mapping for Audit Package	
O.PKE_Audit	FAU_GEN.1-NIAP-0407:2; FAU_GEN.2-NIAP-0410:2
Mapping for Continuous Authentication Package	
O.Continuous_I&A	FIA_UAU.6

6.2.1.1 Security Objectives for the IT Environment Requirements

OE.AUDIT_GENERATION state that the IT Environment will provide the capability to detect and create records of security-relevant events associated with users. This objective is satisfied by the following requirements:

- **FAU_GEN.1-NIAP-0407:1**defines the set of events that the IT Environment must be capable of recording. This requirement ensures that the Administrator has the ability to audit any security relevant event that takes place in the TOE. This requirement also defines the information that must be contained in the audit record for each auditable event. This requirement also places a requirement on the level of detail that is recorded on any additional security functional requirements an ST author adds to this PP.

- **FAU_GEN.2-NIAP-0410:1** ensures that the audit records associate a user identity with the auditable event.
- **FIA_USB.1** plays a role is satisfying this objective by requiring a binding of security attributes associated with users that are authenticated with the subjects that represent them in the IT Environment. This only applies to authorized users, since the identity of unauthenticated users cannot be confirmed. Therefore, the audit trail may not always have the proper identity of the subject that causes an audit record to be generated.
- **FAU_SEL.1-NIAP-0407** allows the Administrator to configure which auditable events will be recorded in the audit trail. This provides the administrator with the flexibility in recording only those events that are deemed necessary by site policy, thus reducing the amount of resources consumed by the audit mechanism

OE.AUDIT_PROTECTION states that the IT Environment will provide the capability to protect audit information. This objective is satisfied by the following requirements:

- **FAU_SAR.2** restricts the ability to read the audit trail to the Administrator, thus preventing the disclosure of the audit data to any other user. However, the IT Environment is not expected to prevent the disclosure of audit data if it has been archived or saved in another form (e.g., moved or copied to an ordinary file).
- **FAU_STG.1-NIAP-0429; FAU_STG.NIAP-0429-1:** The **FAU_STG** family dictates how the audit trail is protected. **FAU_STG.1-NIAP-0429** restricts the ability to delete audit records to the administrator. **FAU_STG.NIAP-0429-1** defines the actions that must be available to the administrator, as well as the action to be taken if there is no response. This helps to ensure that audit records are kept until the administrator deems they are no longer necessary. This requirement also ensures that no one has the ability to modify audit records (e.g., edit any of the information contained in an audit record). This ensures the integrity of the audit trail is maintained.
- **FMT_MOF.1** restricts the capability to modify the behavior of the audit function to the administrator. This requirement ensures that only administrator can turn audit on or off, this ensuring users actions are audited according to a site defined policy.

OE.AUDIT_REVIEW states that the IT Environment will provide the capability to selectively view audit information. This objective is satisfied by the following requirements:

- **FAU_SAR.1** provides the administrator with the capability to read all the audit data contained in the audit trail. This requirement also mandates the audit information be presented in a manner that is suitable for the administrator to interpret the audit trail, which is subject to interpretation. It is expected that the audit information be presented in such a way that the administrator can examine an audit record and have the appropriate information (that required by FAU_GEN.2) presented together to facilitate the analysis of the audit review
- **FAU_SAR.3** complements FAU_SAR.1 by providing the administrator the flexibility to specify criteria that can be used to search or sort the audit records residing in the audit trail. FAU_SAR.3 requires the administrator be able to establish the audit review criteria based on a user ID and source subject identity, so that the actions of a user can be readily identified and analyzed.

OE.Configuration states that the TOE shall be installed and configured properly for starting up the TOE in a secure state. This objective covers A.Configuration, an assumption that states that the TOE will be properly installed and configured. This objective is supported by:

- The startup and installation guides required by the ADO_IGS.1 assurance requirement, which states that accurate installation and configuration documentation must be provided that allows the TOE to be properly (i.e., in a secure state) installed and configured.

OE.CORRECT_TSF_OPERATION states that the IT Environment will provide the capability to test the TSF to ensure the correct operation of the TSF at a customer's site.

- **FPT_TST_SOF_(EXP).1** is necessary to ensure the correctness of the TSF configuration files and TSF data and executable. If TSF software is corrupted it is possible that the TSF would no longer be able to enforce the security policies. This also holds true for TSF data, if TSF data is corrupted, the TOE may not correctly enforce its security policies.
- **ATE** security assurance requirements will provide assurance that the TOE has been thoroughly tested to ensure the correct operation of the TSF. Work units for ATE_COV.2, ATE_DPT.1, ATE_IND.2 will demonstrate that the TOE testing contained enough depth and coverage to test TOE TSF functionality.

OE.CRYPTOGRAPHY states that the TOE shall use NIST FIPS 140-2 validated cryptographic services provided by the IT Environment. This objective is satisfied by the following requirements:

- **FCS_CRM_FPS_(EXP).1**, FIPS compliant cryptographic module, which requires that the IT Environment shall provide all cryptographic modules necessary for the TSF and that each cryptographic module shall be FIPS 140 series Level 1 validated.

OE.DISPLAY_BANNER states that the IT Environment will display an advisory warning regarding use of the TOE. This objective is satisfied by the following requirements:

- **FTA_TAB.1** meets this objective by requiring the IT Environment to display an administrator defined banner before a user can establish an authenticated session. This banner is under complete control of the administrator in which they specify any warnings regarding unauthorized use of the TOE and remove any product or version information if they desire.

OE.Low states that the identification and authentication functions in the TOE shall be designed and implemented for a minimum attack potential of low as validated by the vulnerability assessment and strength of function analyses. This objective covers the SOF analysis, which analyzes the strength of function of identification and authentication functions.

OE.MANAGE states that the IT Environment will provide all the functions and facilities necessary to support the administrators in their management of the security of the TOE, and restrict these functions and facilities from unauthorized use. This objective is satisfied by the following requirements:

- **FMT_MOF.1** requires that the ability to use particular TOE capabilities be restricted to the Administrator.

- **FMT_MSA.1** requires that the ability to perform operations on security attributes be restricted to particular roles.
- **FMT_MSA.3-NIAP-0429** requires that default values used for security attributes are restrictive, and that the Administrator has the ability to override those values.
- **FMT_MTD.1:1, FMT_MTD.1:2, FMT_MTD.1:3, FMT_MTD.1:4, and FMT_MFT.1:5** require that the ability to manipulate IT Environment and TOE data is restricted to Administrators and authorized users.
- **FMT_SMF.1** requires that appropriate administrators manage the audit and other functions.
- **FMT_SMR.1** defines the specific security roles to be supported to perform the functions listed in the list above.

OE.MEDIATE states that the IT Environment will protect user data in accordance with its security policy. This objective is satisfied by the following requirements:

- **FDP_ACC.1** defines that an Access Control policy that will be enforced on a list of subjects acting on the behalf of users attempting to gain access to a list of named objects. All the operations between subject and object covered are defined by the policy. The “subjects” are generally the IT Environment's “Agents.” The “named objects” are things that the IT Environment is protecting for itself and for the TOE
- **FDP_ACF.1-NIAP-0407** defines the Security Attribute used to provide Access Control to objects based on the following above Access Control policy and access control rules based on those security attributes.

OE.NO_EVIL states that sites using the TOE will ensure that administrators are non-hostile, appropriately trained and follow all administrator guidance. This objective is supported by:

- The Administrator and User Guides as defined under assurance requirements AGD_ADM.1 and AGD_USR.1, respectively.

OE.PHYSICAL states that the non-IT environment will provide an acceptable level of physical security so that the TOE cannot be tampered with or be subject to side channel attacks such as the various forms of power analysis and timing analysis. This objective is supported by:

- The Administrator and User Guides as defined under assurance requirements AGD_ADM.1 and AGD_USR.1, respectively. The Administrator and User Guides define the security policy for the installation and operation of the TOE.

OE.RESIDUAL_INFORMATION which states that the IT Environment will ensure that any information contained in a protected resource within its Scope of Control is not released when the resource is reallocated. This objective is satisfied by the following requirements:

- **FDP_RIP.2** is used to ensure the contents of resources are not available to subjects other than those explicitly granted access to the data.

OE.SELF_PROTECTION which states that the IT Environment will maintain a domain for its own execution that protects it and its resources from external interference, tampering, or unauthorized disclosure. This objective is satisfied by the following requirements:

- **FPT_RVM.1** ensures that the IT Environment makes policy decisions on all interfaces that perform operations on subjects and objects that are scoped by the policies. Without this non-bypassability requirement, the IT Environment could not be relied upon to completely enforce the security policies, since an interface(s) may otherwise exist that would provide a user with access to TOE resources (including TSF data and executable code) regardless of the defined policies. This includes controlling the accessibility to interfaces, as well as what access control is provided within the interfaces.
- **FPT_SEP.1** was chosen to ensure the IT Environment provides a domain that protects itself from untrusted users. If the IT Environment cannot protect itself it cannot be relied upon to enforce its security policies.

OE.TIME_STAMPS states that the IT Environment will provide reliable time stamps and the capability for the administrator to set the time used for these time stamps. This objective is satisfied by the following requirements:

- **FPT_STM.1** requires that the IT Environment provide time stamps for its own use and for the TOE use.
- **FMT_SMF.1** requires that the IT Environment provide an administrator with the capability to modify system time.
- **FMT_MTD.1:5** requires that the IT Environment restrict the capability to modify system time to an administrator.

OE.TIME_TOE states that The IT Environment will provide reliable time for the TOE use. This objective is satisfied by the following requirements:

- **FPT_STM.1** requires that the IT Environment provide time stamps for its own use and for the TOE use.

OE.TOE_ACCESS states that the IT Environment will provide mechanisms that control a user's logical access to the TOE. This objective is satisfied by the following requirements:

- **FIA_AFL.1** provides a detection mechanism for unsuccessful authentication attempts by the users. The requirement enables an administrator settable threshold that prevents unauthorized users from gaining access to authorized user's account by guessing authentication data by locking the targeted account. Thus, limiting an unauthorized user's ability to gain unauthorized access to the TOE.
- **FIA_ATD.1** defines the attributes of users, including a user ID that is used to by the IT Environment to determine a user's identity and enforce what type of access the user has to the IT Environment (e.g., the IT Environment associates a user ID with any role(s) they may assume).
- **FIA_UID.2** requires that a user be identified to the IT Environment in order to do anything.
- **FIA_UAU.2** requires that a user be authenticated by the IT Environment in order to do anything.
- **FIA_UAU.7** provides that the authentication data provided by the user is not echoed back in plaintext, thus serving to protect that data.

- **FTA_SSL.1 and FTA_SSL.2** components deal with automatic session locking and termination, either initiated by the IT Environment or a user. They protect from an unauthorized entity to use the unattended session.

OE.TOE_PROTECTION states that the IT Environment will protect the TOE and TOE resources from external interference, tampering, or unauthorized disclosure and modification. This objective is satisfied by the following requirements:

- **FPT_SEP_ENV_(EXP).1** ensures that the IT Environment provides a domain that protects TSF from untrusted users. If the TSF cannot be protected, it cannot be relied upon to enforce its security policies.

6.2.1.2 Certification Path Validation – Basic Package Rationale

O.Availability states that the TSF shall continue to provide security services even if revocation information is not available. This objective is met by:

- FDP_DAU_CPV_(EXP).1, Certificate processing – basic, which requires that the TSF bypass the revocation check if the revocation information is not available.

O.Correct_Temporal states that the TSF shall provide accurate temporal validation results. This objective is met by:

- FDP_DAU_CPI_(EXP).1, Certification path initialisation – basic, which requires that the TSF obtain the time of interest called “TOI" from a reliable source.

O.Current_Certificate states that the TSF shall only accept certificates that are not expired as of TOI. This objective is met by:

- FDP_DAU_CPV_(EXP).1, which requires that the TSF accept a certificate only if the specified checks succeed, including that the certificate is not expired as of TOI.

O.Get_KeyInfo states that the TSF shall provide the user public key and related information in order to carry out cryptographic functions. This objective is met by:

- FDP_DAU_CPO_(EXP).1, Certification path output – basic, which requires that the TSF output the subject public key from the certification path and other information specified by the ST author.

O.Path_Find states that the TSF shall be able to find a certification path from a trust anchor to the subscriber. This objective is met by:

- FDP_CPD_(EXP).1, Certification path development, which requires that the TSF shall develop a certification path from a trust anchor to the subscriber.

O.Trusted_Keys states that the TSF shall use trusted public keys in certification path validation. This objective is met by:

- FDP_DAU_CPI_(EXP).1, Certification path initialisation -- basic, which requires that the TSF use trusted public keys in the certification path validation.

O.User states that the TSF shall only accept certificates issued by a CA. This objective is met by:

- FDP_DAU_CPV_(EXP).2, Intermediate certificate processing – basic, which requires that the TSF accept an intermediate certificate only when the certificate is issued by a CA.

O.Verified_Certificate states that the TSF shall only accept certificates with verifiable signatures. This objective is met by:

- FDP_DAU_CPV_(EXP).1, Certificate processing – basic, which requires that the TSF accept certificates only with verifiable signatures.

O.Valid_Certificate states that the TSF shall use certificates that are valid, i.e., not revoked. This objective is met by:

- FDP_DAU_CPV_(EXP).1, Certificate processing – basic, which requires that that the TSF shall use only those certificates that are valid, i.e., revocation status demonstrates that the certificate is not revoked.

6.2.1.3 Certification Path Validation – Basic Policy Package Rationale

O.Provide_Policy_Info states that the TSF shall provide certificate policies for which the certification path is valid. This objective is met by:

- FDP_DAU_CPI_(EXP).2, Certification path initialisation – basic policy, which requires that the TSF shall use the initial-certificate-policies provided by user roles specified by the ST author.
- FDP_DAU_CPO_(EXP).2, Certification path output – basic policy, which requires that The TSF shall output the certificate policies using the following rule: intersection of certificatePolicies extensions in all the certificates in certification path and initial-certificate-policies.

6.2.1.4 Certification Path Validation – Policy Mapping Package Rationale

O.Map_Policies states that the TSF shall map certificate policies in accordance with user and CA constraints. This objective is met by:

- FDP_DAU_CPI_(EXP).3, Certification path initialisation – policy mapping, which requires that the TSF use the explicit-policy-indicator, policy-mapping-inhibit-indicator, inhibit-any-policy-indicator provided by a role specified by the ST author.
- FDP_DAU_CPV_(EXP).3, Intermediate certificate processing – policy mapping, which requires that the TSF use the intermediate certificate to update specified state variables.
- FDP_DAU_CPO_(EXP).3, Certification path output – policy mapping, which requires that the TSF shall map policies in the calculation of the policies intersection according to defined user and CA constraints.

O.Policy_Enforce states that the TSF shall validate a certification path in accordance with certificate policies acceptable to the user. This objective is met by:

- FDP_DAU_CPI_(EXP).3, Certification path initialisation – policy mapping, which requires that the TSF use the explicit-policy-indicator, policy-mapping-inhibit-indicator, inhibit-any-policy-indicator provided by a role specified by the ST author.
- FDP_DAU_CPV_(EXP).3, Intermediate certificate processing – policy mapping, which requires that the TSF use the intermediate certificate to update specified state variables.

- FDP_DAU_CPO_(EXP).3, Certification path output – policy mapping, which requires that the TSF shall map policies in the calculation of the policies intersection according to defined user and CA constraints and that specified policies be enforced.

6.2.1.5 Certification Path Validation – Name Constraints Package Rationale

O.Authorised_Names states that the TSF shall validate a certificate only if the CA is authorized to issue a certificate to the subject. This objective is met by:

- FDP_DAU_CPI_(EXP).4, Certification path initialisation – names, which requires that the TSF initialize the following: permitted-subtrees = ∞, excluded-subtrees = ∅
- FDP_DAU_CPV_(EXP).4, Intermediate certificate processing – name constraints, which requires that the TSF accept a certificate only if the conditions specified by the requirement, including verification of authorization, is satisfied.
- FDP_DAU_CPV_(EXP).5, Intermediate Certificate processing – name constraints, states that the TSF shall use the intermediate certificate to update the following states: permitted-subtrees and excluded-subtrees

6.2.1.6 PKI Signature Generation Package Rationale

O.Give_Sig_Hints states that the TSF shall provide hints for selecting correct certificates for PKI signature verification. This objective is met by:

- FDP_ETC_SIG_(EXP).1 Export of PKI Signature, which requires that the TSF use the user selected private to key perform digital signature and that the TSF include additional information specified by the ST author with the digital signature to facilitate signature verification.

6.2.1.7 PKI Signature Verification Package Rationale

O.Use_Sig_Hints states that the TSF shall use hints for selecting correct certificates for signature verification. This objective is met by:

- FDP_ITC_SIG_(EXP).1, Import of PKI Signature, which requires that the TSF use the following information from the signed data: hashing algorithm, signature algorithm, signer public key certificate, signer DN, signer subject alternative name, signer subject key identifier, or other data during signature verification.

O.Linkage_Sig_Ver states that the TSF shall use the correct user public key for signature verification. This objective is met by:

- FDP_DAU_SIG_(EXP).1, Signature Blob Verification, which requires that the TSF invoke a cryptographic module with the following information from Certification Path Validation to verify digital signature on signed data: subject public key algorithm, subject public key, subject public key parameters and that the TSF perform other verification checks as specified by the ST author.

6.2.1.8 PKI Encryption using Key Transfer Algorithms Package Rationale

O.Hints_Enc_WO states that the TSF shall provide hints for selecting correct certificates or keys for PKI Encryption using Key Transfer algorithms. This objective is met by:

- FDP_ETC_ENC_(EXP).1, Export of PKI Encryption – Key Transfer Algorithms, which requires that the TSF include the information with the encrypted data, such as the public key, as selected or assigned by the ST author and that the TSF invoke a cryptographic module with the following information from Certification Path Validation to create encrypted data: subject public key algorithm, subject public key, subject public key parameters.

O.Linkage_Enc_WO states that the TSF shall use the correct user public key for key transfer.

- FDP_ETC_ENC_(EXP).1, Export of PKI Encryption – Key Transfer Algorithms, which requires that the TSF invoke a cryptographic module with the following information from Certification Path Validation to create encrypted data: subject public key algorithm, subject public key, subject public key parameters.
- FDP_DAU_ENC_(EXP).1, PKI Encryption Verification – Key Transfer, which requires that the TSF apply verification checks for key transfer as selected by the ST author.

6.2.1.9 PKI Encryption using Key Agreement Algorithms Package Rationale

O.Hints_Enc_W states that the TSF shall provide hints for selecting correct certificates or keys for PKI encryption using Key Agreement algorithms. This objective is met by:

- FDP_ETC_ENC_(EXP).2, Export of PKI Encryption – Key Agreement Algorithms, which requires that the TSF include the information with the encrypted data, such as the public key, as selected or assigned by the ST author and that the TSF invoke a cryptographic module with the following information from Certification Path Validation to create encrypted data: subject public key algorithm, subject public key, subject public key parameters.

O.Linkage_Enc_W states that the TSF shall use the correct user public key for key agreement during encryption. This objective is met by:

- FDP_ETC_ENC_(EXP).2, Export of PKI Encryption – Key Agreement Algorithms, which requires that the TSF invoke a cryptographic module with the following information from Certification Path Validation to create encrypted data: subject public key algorithm, subject public key, subject public key parameters.
- FDP_DAU_ENC_(EXP).2, PKI Encryption Verification – Key Agreement, Subject, Decryptor, which requires that the TSF apply verification checks for key agreement as selected by the ST author.

6.2.1.10 PKI Decryption using Key Transfer Algorithms Package Rationale

O.Correct_KT states that the TSF shall use appropriate private key and key transfer algorithm:

- FDP_ITC_ENC_(EXP).1, Import of PKI Encryption – Key Transfer Algorithms, which requires that the TSF invoke a cryptographic module with the information from the encrypted data as selected by the ST author to provide a means to identify an appropriate private key and key transfer algorithm.

6.2.1.11 PKI Decryption using Key Agreement Algorithms Package Rationale

O.Hints_Dec_W states that the TSF shall provide hints for selecting correct certificates or keys for PKI decryption using Key Agreement algorithms. This objective is met by:

- FDP_ITC_ENC_(EXP).2, Import of PKI Encryption – Key Agreement Algorithms, which requires that the TSF use the information from the encrypted data and information from Certification Path Validation to provide hints for selecting correct key agreement algorithm, certificates or keys.

O.Linkage_Dec_W states that the TSF shall use the correct user public key for key agreement during decryption. This objective is met by:

- FDP_ITC_ENC_(EXP).2, Import of PKI Encryption – Key Agreement Algorithms, which requires that the TSF use the information from the encrypted data and information from Certification Path Validation to provide hints for selecting correct key agreement algorithm, certificates or keys.
- FDP_DAU_ENC_(EXP).3, PKI Encryption Verification – Key Agreement, Subject, Encryptor, which requires that the TSF apply checks as selected by the ST author to verify the user public key using certification path validation.

O.Correct_KA states that the TSF shall use appropriate private key and key agreement algorithm. This objective is met by:

- FDP_ITC_ENC_(EXP).2, Import of PKI Encryption – Key Agreement Algorithms, which requires that the TSF use the information from the encrypted data and information from Certification Path Validation to provide hints for selecting correct key agreement algorithm, certificates or keys.

6.2.1.12 PKI Based Entity Authentication Package Rationale

The PKI Based Entity Authentication package may or may not be included in an ST, depending on the functionality of the application.

O.I&A states that the TSF shall uniquely identify all entities, and shall authenticate the claimed identity before granting an entity access to the TOE facilities. This objective is met by:

- FIA_UAU.1;1, Timing of authentication, which requires that the TSF allow the a list of TSF mediated actions, specified by the ST author, to be performed on behalf of the user before the user is authenticated and that TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user. This requirement ensures that all users are authenticated.
- FIA_UID.1;1, Timing of identification, which requires that the TSF allow the a list of TSF mediated actions, specified by the ST author, to be performed on behalf of the user before the user is identified and that TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user. This requirement ensures that all users are identified.

O.Limit_Actions_Auth states that the TSF shall restrict the actions an entity may perform before the TSF verifies the identity of the entity. This objective is met by:

- FIA_UAU.1;1, Timing of authentication, which requires that the TSF allow the a list of TSF mediated actions, specified by the ST author, to be performed on behalf of the user before the user is authenticated and that TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user. This requirement ensures that all users are authenticated.

- FIA_UID.1;1, Timing of identification, which requires that the TSF allow the a list of TSF mediated actions, specified by the ST author, to be performed on behalf of the user before the user is identified and that TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user. This requirement ensures that all users are identified.

O.Linkage states that the TSF shall use the correct user public key for authentication. This objective is met by:

- FIA_UAU_SIG_(EXP).1, Entity authentication, which requires that the TSF invoke a cryptographic module with the following information from Certification Path Validation to verify the signature on signed data: subject public key algorithm, subject public key, subject public key parameters, and that the TSF perform additional checks as specified by the ST author.

O.Single_Use_I&A states that the TSF shall use the I&A mechanism that requires unique authentication information for each I&A. This objective is met by:

- FIA_UAU.4, Single-use authentication mechanisms, which requires that the TSF prevent reuse of authentication data.

6.2.1.13 Online Certificate Status Protocol Package Rationale

O.Accurate_OCSP_Info states that the TSF shall accept only accurate OCSP responses. This objective is met by:

- FDP_DAU_OCS_(EXP).1, Basic OCSP Client, which requires that only accurate revocation information be accepted from the OCSP responder.

O.Auth_OCSP_Info states that the TSF shall accept the OCSP responses from an authorized source. This objective is met by:

- FDP_DAU_OCS_(EXP).1, Basic OCSP Client, which requires that the OCSP responder be verified as an authorized source.

O.Current_OCSP_Info states that the TSF may accept only OCSP responses current as of TOI. This objective is met by:

- FDP_DAU_OCS_(EXP).1, Basic OCSP Client, which requires that only reasonably current as of TOI revocation information may be accepted through a series of policy and parameter checks.

O.User_Override_Time_OCSP states that the TSF shall permit the user to override the time checks on the OCSP response. This objective is met by:

- FDP_DAU_OCS_(EXP).1, Basic OCSP Client, which requires that a role or roles specified by the ST author be able to override the time checks on the OCSP response.

6.2.1.14 Certificate Revocation List (CRL) Validation Package Rationale

O.Accurate_Rev_Info states that the TSF shall accept only accurate revocation information. This objective is met by:

- FDP_DAU_CRL_(EXP).1, Basic CRL checking, which requires that the TSF accept accurate revocation information. Accuracy is determined through a series of verification and policy requirements within this explicitly stated requirement.

O.Auth_Rev_Info states that the TSF shall accept the revocation information from an authorized source for CRL. This objective is met by:

- FDP_DAU_CRL_(EXP).1, Basic CRL checking, which requires that the TSF accept revocation information from an authorized source as selected or assigned by the ST author.

O.Current_Rev_Info states that the TSF shall accept only CRL current as of TOI. This objective is met by:

- FDP_DAU_CRL_(EXP).1, Basic CRL checking, which requires that the TSF accept only reasonably current as of TOI revocation information through a series of policy requirements defined in FDP_DAU_CRL_(EXP).1.

O.User_Override_Time_CRL states that the TSF shall permit the user to override the time checks on the CRL. This objective is met by:

- FDP_DAU_CRL_(EXP).1, Basic CRL checking, which requires that the TSF accept the CRL as current if a role assigned by the ST author overrides time checks.

6.2.1.15 Audit Package Rationale

O.PKE_Audit states that the TSF shall audit security relevant PKE events. This objective is met by:

- **FAU_GEN.1-NIAP-0407**defines the set of events that the TOE must be capable of recording. This requirement ensures that the Administrator has the ability to audit events that take place in the TOE. This requirement also defines the information that must be contained in the audit record for each auditable event. This requirement also places a requirement on the level of detail that is recorded on any additional security functional requirements an ST author adds to this PP.
- **FAU_GEN.2-NIAP-0410** ensures that the audit records associate a user identity with the auditable event.

6.2.1.16 Continuous Authentication Package Rationale

The Continuous Authentication package may or may not be included in an ST, depending on the functionality of the application.

O.Continuous_I&A states that the TSF shall continuously authenticate the entity. This objective is met by:

- FIA_UAU.6, Re-authenticating entity, which requires that the TSF re-authenticate an entity under the conditions specified by the ST author.

6.2.2 Assurance Requirement Rationale

This PP family includes a choice of assurance level for the PP/ST author.

Basic Robustness assurance level is considered sufficient for low threat environments or where compromise of protected information will not have a significant impact on mission objectives. This implies that the motivation of the threat agents will be low in environments that are suitable for TOEs of this robustness. In general, basic robustness results in “good commercial practices” that counter threats based in casual and accidental disclosure or compromise of data protected by the TOE. Basic robustness assurance level should be selected for the threat environments described in this family of PPs.

An EAL 3 with augmentation PP will be selected for TOEs that require a moderate level of independently assured security and require a thorough investigation of the TOE and its development without substantial re-engineering. EAL 3 provides assurance by an analysis of the security functions, using a functional and interface specification, guidance documentation, and the high-level design of the TOE to understand the security behaviour. The analysis is supported by independent testing of the TOE security functions, evidence of developer testing based on the functional specification and high-level design, selective independent confirmation of the developer test results, strength of function analysis, and evidence of a developer search for obvious vulnerabilities. EAL 3 is augmented with ALC_FLR.2 to track and correct the reported and found security flaws in the product and also to provide flaw reporting procedures to the product users.

An EAL 4 with augmentation PP will be selected for TOEs that require a moderate to high level of independently assured security in conventional commodity TOEs and are prepared to incur additional security-specific engineering costs. EAL 4 provides assurance by an analysis of security functions, using a functional and complete interface specification, guidance documentation, the high-level and low-level design of the TOE, and a subset of the implementation, to understand the security behaviour. Assurance is additionally gained through an informal model of the TOE security policy. EAL 4 represents a meaningful increase in assurance from EAL 3 by requiring more design description, a subset of the implementation, and improved mechanisms and/or procedures that provide confidence that the TOE will not be tampered with during development or delivery. EAL 4 is augmented with ALC_FLR.2 to track and correct the reported and found security flaws in the product and also to provide flaw reporting procedures to the product users.

.

6.2.3 Strength of Function Rationale

The TOE is assumed to be designed to protect against “low” attack potential. Thus, based on the CEM Annex B, Table B.2, the minimum strength of function is SOF Basic. The strength of cryptographic algorithms is outside the scope of the CC. Strength of function only applies to non-cryptographic, probabilistic or permutational mechanisms. The SOF requirement applies to the identification and authentication functionality for the TOE. Note that the I&A mechanism used in the TOE is application-specific and SOF analysis must be performed as part of ST development. PPs in this family require a SOF rating of SOF Basic or higher.

A SOF rating reflects the attacker, described in terms of attack potential, against which the probabilistic or permutational security function is designed to protect. To determine a SOF rating for the I&A functionality provided in the TOE, the developer of the ST must calculate the attack potential. One way to calculate the attack potential is to use Table B.3 from the CEM Annex B to calculate a numerical score for attack potential and then use Table B.4 from the CEM Annex B to translate the number into a qualitative attack potential and an SOF rating. For example, using Table B.3, assuming a layman with no knowledge of the TOE and no equipment, with > 1 month elapsed time, and > 1 month access to the TOE results in a score of 17 for attack potential. Note that a brute force attack on the I&A mechanism is obvious and hence the corresponding identifying values are all zero.

Using Table B.4 (duplicated below), the resistance to attack with attack potential score translates to an attack potential of “low”. Again, using Table B.2 or B.4, a SOF rating of SOF Basic is required for attack potential of “low”.

Table B.4 from CEM Annex B

Range of Values	Resistant to attack with attack potential of:	SOF rating
<10	No rating	No rating
10 – 17	Low	Basic
18 – 24	Moderate	Medium
>25	High	High

6.3 Dependency Rationale

Table 6.34 – Functional Requirements Dependencies

Requirement	Dependencies
IT Environment	
FAU_GEN.1-NIAP-0407:1	FPT_STM.1
FAU_GEN.2-NIAP-0410:1	FAU_GEN.1 (met by FAU_GEN.1-NIAP-0407:1) FIA_UID.1 (met by FIA_UID.2)
FAU_SAR.1	FAU_GEN.1 (met by FAU_GEN.1-NIAP-0407:1)
FAU_SAR.2	FAU_SAR.1
FAU_SAR.3	FAU_SAR.1
FAU_SEL.1-NIAP-0407	FAU_GEN.1 (met by FAU_GEN.1-NIAP-0407:1) FMT_MTD.1
FAU_STG.1-NIAP-0429	FAU_GEN.1 (met by FAU_GEN.1-NIAP-0407:1)
FAU_STG.NIAP-0429-1	FAU_STG.1 (met by FAU_STG.1-NIAP-0429) FMT_MTD.1
FCS_CRM_FPS_(EXP).1	None
FDP_ACC.1	FDP_ACF.1 (met by FDP_ACF.1-NIAP-0407)
FDP_ACF.1-NIAP-0407	FDP_ACC.1 FMT_MSA.3 (met by FMT_MSA.3-NIAP-0429)
FDP_RIP.2	None
FIA_AFL.1	FIA_UAU.1 (met by FIA_UAU.2)
FIA_ATD.1	None
FIA_UAU.2	FIA_UID.1 (met by FIA_UID.2)
FIA_UAU.7	FIA_UAU.1 (met by FIA_UAU.2)
FIA_UID.2	None
FIA_USB.1	FIA_ATD.1
FMT_MOF.1	FMT_SMF.1; FMT_SMR.1
FMT_MSA.1	FMT_SMF.1; FMT_SMR.1 [FDP_ACC.1 Subset access control or FDP_IFC Subset information flow control] (satisfied by FDP_ACC.1)
FMT_MSA.3-NIAP-0429	FMT_MSA.1; FMT_SMR.1
FMT_MTD.1:1 through 5	FMT_SMF.1; FMT_SMR.1
FMT_SMF.1	None
FMT_SMR.1	FIA_UID.1 (met by FIA_UID.2)

Requirement	Dependencies
FPT_RVM.1	None
FPT_SEP.1	None
FPT_SEP_ENV_(EXP).1	None
FPT_STM.1	None
FPT_TST_SOF_(EXP).1	None
FTA_SSL.1	FIA_UAU.1 (met by FIA_UAU.2)
FTA_SSL.2	FIA_UAU.1 (met by FIA_UAU.2)
FTA_TAB.1	None
CPV – Basic Package	
FDP_CPD_(EXP).1	None
FDP_DAU_CPI_(EXP).1	FCS_COP.1 (met by FCS_CRM_FPS_(EXP).1) FPT_STM.1
FDP_DAU_CPV_(EXP).1	FCS_COP.1 (met by FCS_CRM_FPS_(EXP).1) FPT_STM.1, [FDP_DAU_OCS_(EXP).1 or FDP_DAU_CRL_(EXP).1]
FDP_DAU_CPV_(EXP).2	FDP_DAU_CPV_(EXP).1
FDP_DAU_CPO_(EXP).1	FDP_DAU_CPV_(EXP).1
CPV – Basic Policy Package	
FDP_DAU_CPI_(EXP).2	FDP_DAU_CPI_(EXP).1 (See Note 1)
FDP_DAU_CPO_(EXP).2	FDP_DAU_CPO_(EXP).1 (See Note 1)
CPV – Policy Mapping Package	
FDP_DAU_CPI_(EXP).3	FDP_DAU_CPI_(EXP).2 (See Note 2)
FDP_DAU_CPV_(EXP).3	FDP_DAU_CPV_(EXP).2 (See Note 3)
FDP_DAU_CPO_(EXP).3	FDP_DAU_CPO_(EXP).2 (See Note 2)
CPV – Name Constraints Package	
FDP_DAU_CPI_(EXP).4	FDP_DAU_CPI_(EXP).1 (See Note 1)
FDP_DAU_CPV_(EXP).4	FDP_DAU_CPV_(EXP).1 (See Note 1)
FDP_DAU_CPV_(EXP).5	FDP_DAU_CPV_(EXP).2 (See Note 1)
PKI Signature Generation Package	
FDP_ETC_SIG_(EXP).1	FCS_CRM_FPS_(EXP).1
PKI Signature Verification Package	
FDP_ITC_SIG_(EXP).1	None
FDP_DAU_SIG_(EXP).1	FCS_CRM_FPS_(EXP).1 FDP_DAU_CPO_(EXP).1 (See Note 1)
PKI Encryption using Key Transfer Algorithms Package	

Requirement	Dependencies
FDP_ETC_ENC_(EXP).1	FCS_CRM_FPS_(EXP).1 FDP_DAU_CPO_(EXP).1 (See Note 1)
FDP_DAU_ENC_(EXP).1	FDP_DAU_CPO_(EXP).1 (See Note 1)
PKI Encryption using Key Agreement Algorithms Package	
FDP_ETC_ENC_(EXP).2	FCS_CRM_FPS_(EXP).1 FDP_DAU_CPO_(EXP).1 (See Note 1)
FDP_DAU_ENC_(EXP).2	FDP_DAU_CPO_(EXP).1 (See Note 1)
PKI Decryption using Key Transfer Algorithms Package	
FDP_ITC_ENC_(EXP).1	FCS_CRM_FPS_(EXP).1
PKI Decryption using Key Agreement Algorithms Package	
FDP_ITC_ENC_(EXP).2	FCS_CRM_FPS_(EXP).1 FDP_DAU_CPO_(EXP).1 (See Note 1)
FDP_DAU_ENC_(EXP).3	FDP_DAU_CPO_(EXP).1 (See Note 1)
PKI Based Entity Authentication Package	
FIA_UAU.1	FIA_UID.1
FIA_UAU.4	None
FIA_UAU_SIG_(EXP).1	FCS_CRM_FPS_(EXP).1 FDP_DAU_CPO_(EXP).1 (see Note 1)
FIA_UID.1	None
Online Certificate Status Protocol Client Package	
FDP_DAU_OCS_(EXP).1	FCS_CRM_FPS_(EXP).1 FPT_STM.1
Certificate Revocation List (CRL) Validation Package	
FDP_DAU_CRL_(EXP).1	FCS_CRM_FPS_(EXP).1 FPT_STM.1
Audit Package	
FAU_GEN.1-NIAP-0407:2	FPT_STM.1
FAU_GEN.2-NIAP-0410:2	FAU_GEN.1 (met by FAU_GEN.1-NIAP-0407:2) FIA_UID.1 (met by FIA_UID.2 in the IT Environment)
Continuous Authentication Package	
FIA_UAU.6	None

Note 1: The dependency is satisfied by including the CPV – Basic Package

Note 2: The dependency is satisfied by including the CPV – Basic Policy Package

Note 3: The dependency is satisfied by including the CPV – Basic Package and the CPV – Basic Policy Package.

6.4 Rationale for not Addressing Consistency Instructions

This section contains the Basic Robustness Consistency requirements that were not addressed.

6.4.1 Software only TOEs

The TOE may consist only of software. The ST author must specify clearly in the TOE description that the TOE will be "Software only". The ST author should follow a set of instructions to validate a software only TOE against a PP in this family of PPs.

- If the TSF provides additional self protection functions, the ST author must use the O.PARTIAL_SELF_PROTECTION objective to supplement the OE.SELF_PROTECTION objective. O.PARTIAL_SELF_PROTECTION was not used in the profile family since the IT Environment will provide the protection for the TOE and some TOE might consist of a toolkit to which this objective is not applicable.
- If the TSF provides additional domain separation functions, the ST author should use explicitly stated requirements (modeled after FPT_SEP.*) for domain separation. This requirement was not levied on all TOE in the profile family since a TOE might consist of a toolkit to which this requirement is not applicable. Requirement FPT_SEP_ENV_(EXP).1 for the IT Environment was added to ensure that the IT Environment provides separation for application TOE executables and data. This is in addition to the FPT_SEP.1 requirement for the IT Environment.
- The requirement FPT_STM.1 should be included in the ST as a requirement on the IT environment. If the TOE is responsible for any part of this requirement, then the requirement, FPT_STM.1, should be iterated and also be placed on the TOE. This requirement was included in the IT Environment since the IT Environment will provide the time. This requirement was not included in the TOE since the TOE is not expected to play any other role except use the time provided by the IT Environment.

6.4.2 Other Requirements

This family of PPs has not addressed the following requirements:

- Many of the requirements listed in the consistency instructions are applicable to the underlying operating system and not to the PK enabled applications. Thus, in order to develop a family of PPs that is useful, is secure, and meets the spirit of the Basic Robustness Consistency Instructions, the requirements were appropriately allocated to the IT Environment, and the family of PPs requires that the PK enabled application be run on an operating system that has been validated to the meet the IT Environment requirements in this family of PPs.
- FCS Class related requirements were not addressed since the family of PPs requires FIPS 140-2 validated cryptographic modules.

- Threats and objectives related to assurance requirements were not included since they do not add any value except restate the assurance requirements in two addition ways (threat and objectives).
- Since the PKI and PKE offers critical security functions via the packages specified in this family of PPs, assurance levels higher than Basic Robustness are included so that the PP/ST author can obtain higher degree of assurance. Since PKE applications and toolkits are not likely to include hardware in the TOE, Medium Robustness requirements and assurance levels could not be selected.

7. Appendices

A. References

Please see the Applicable documents subsection in Section 1 of this document

B. Glossary

Access -- Interaction between an entity and an object that results in the flow or modification of data.

Access Control -- Security service that controls the use of resources (including hardware and software) and the disclosure and modification of data.(including stored and communicated)

Accountability -- Property that allows activities in an IT system to be traced to the entity responsible for the activity.

Administrator -- A user who has been specifically granted the authority to manage some portion or all of the TOE and whose actions may affect the TSP. Administrators may possess special privileges that provide capabilities to override portions of the TSP.

Assurance -- A measure of confidence that the security features of an IT system are sufficient to enforce its' security policy.

Asymmetric Cryptographic System -- A system involving two related transformations; one determined by a public key (the public transformation), and another determined by a private key (the private transformation) with the property that it is computationally infeasible to determine the private transformation (or the private key) from knowledge of the public transformation (and the public key).

Asymmetric Keys-- A pair of keys generated together that have different values such that information encrypted with one key may be decrypted with the other key or the information digitally signed using one key can be verified using the other key. One of the keys called the private key cannot be derived from the other key called the public key without extensive computational complexity.

Attack -- An intentional act attempting to violate the security policy of an IT system.

Authentication -- Security measure that verifies a claimed identity.

Authentication data -- Information used to verify a claimed identity.

Authorization -- Permission, granted by an entity authorized to do so, to perform functions and access data.

Authorized user -- An authenticated user who may, in accordance with the TSP, perform an operation.

Availability – Timely (according to a defined metric), reliable access to IT resources.

Certificate Revocation List (CRL) -- A list of the certificates that relying parties should no longer use or trust because the certificates have been revoked. Normally, the CA that issued the certificates also issues the CRL. The CA may assign responsibility for issuing CRLs to another entity. The CRL is a data structure that the issuer digitally signed.

Compromise -- Violation of a security policy.

Confidentiality -- A security policy pertaining to disclosure of data.

Critical Security Parameters (CSP) -- Security-related information (e.g., cryptographic keys, authentication data such as passwords and pins, and cryptographic seeds) appearing in plaintext or otherwise unprotected form and whose disclosure or modification can compromise the security of a cryptographic module or the security of the information protected by the module.

Cryptographic Administrator -- An authorized user who has been granted the authority to perform cryptographic initialization and management functions. These users are expected to use this authority only in the manner prescribed by the guidance given to them.

Cryptographic boundary -- An explicitly defined contiguous perimeter that establishes the physical bounds (for hardware) or logical bounds (for software) of a cryptographic module.

Cryptographic key (key) -- A parameter used in conjunction with a cryptographic algorithm that determines [7]:

- the transformation of plaintext data into ciphertext data,
- the transformation of cipher text data into plaintext data,
- a digital signature computed from data,
- the verification of a digital signature computed from data, or
- a digital authentication code computed from data.

Cryptographic Module -- The set of hardware, software, firmware, or some combination thereof that implements cryptographic logic or processes, including cryptographic algorithms, and is contained within the cryptographic boundary of the module.

Cryptographic Module Security Policy -- A precise specification of the security rules under which a cryptographic module must operate, including the rules derived from the requirements of this PP and additional rules imposed by the vendor.

Defense-in-Depth (DID) -- A security design strategy whereby layers of protection are utilized to establish an adequate security posture for an IT system.

Digital Envelope -- A collection that consists of data encrypted with a symmetric session key and the session key encrypted for each recipient using the recipient’s public key.

Digital Signature (or Signature) -- A value determined from first computing a hash of the data to be signed and then applying a cryptographic function (the signature algorithm) to a hash value using the private key of the signer.

Digitally Signed Data -- A collection of data (the signed data) and a value (the digital signature) computed from that data. The signature is the result of applying an asymmetric cryptographic algorithm to the data (or an intermediate value derived from the data). The collection may also include information to assist in authenticating the entity that signed the data.

Discretionary Access Control (DAC) -- A means of restricting access to objects based on the identity of subjects and/or groups to which they belong. These controls are discretionary in the sense that a subject with a certain access permission is capable of passing that permission (perhaps indirectly) on to any other subject.

Effective Date -- The date when a digital signature was created. The date includes the calendar date and the time of day. The relying party has to have confidence in the accuracy of the effective date. The date may be either the actual date or a presumed date. The relying party may presume that the effective date is the date of receipt of the document. The relying party knows the signature had to occur prior to receipt.

Embedded Cryptographic Module -- One that is built as an integral part of a larger and more general surrounding system (i.e., one that is not easily removable from the surrounding system).

Enclave -- A collection of entities under the control of a single authority and having a homogeneous security policy. They may be logical, or may be based on physical location and proximity.

Entity -- A subject, object, user or another IT device, which interacts with TOE objects, data, or resources.

Expired Certificate -- A certificate with the **not after** component of its validity field having a value earlier than the current date. Certificates may or may not appear in CRLs issued after their expiration.

External IT entity -- Any trusted Information Technology (IT) product or system, outside of the TOE, which may, in accordance with the TSP, perform an operation.

Hash Algorithm -- An algorithm that maps variable length inputs into a fixed length output value known as the digest or hash. The algorithm is a many-to-one function; multiple inputs may result in the same output. However, discovering an input value that results in a desired or given output is computationally infeasible.

Identity -- A representation (e.g., a string) uniquely identifying an authorized user, which can either be the full or abbreviated name of that user or a pseudonym.

Integrity -- A security policy pertaining to the corruption of data and TSF mechanisms.

Integrity label -- A security attribute that represents the integrity level of a subject or an object. Integrity labels are used by the TOE as the basis for mandatory integrity control decisions.

Integrity level -- The combination of a hierarchical level and an optional set of non-hierarchical categories that represent the integrity of data.

Key Pair -- A set of two keys used in asymmetric cryptography. A key generation algorithm creates the keys.

Mandatory Access Control (MAC) -- A means of restricting access to objects based on subject and object sensitivity labels. The Bell LaPadula model is an example of Mandatory Access Control.

Mandatory Integrity Control (MIC) -- A means of restricting access to objects based on subject and object integrity labels.

Multilevel -- The ability to simultaneously handle (e.g., share, process) multiple levels of data, while allowing users at different sensitivity levels to access the system concurrently. The system permits each user to access only the data to which they are authorized access.

Named Object -- An object that exhibits all of the following characteristics:

- The object may be used to transfer information between subjects of differing user identities within the TSF.
- Subjects in the TOE must be able to request a specific instance of the object.
- The name used to refer to a specific instance of the object must exist in a context that potentially allows subjects with different user identities to request the same instance of the object.

Non-repudiation -- The inability to deny performing an action. Non-repudiation is evidence of the identity of the signer of a message and message integrity, sufficient to prevent a party from successfully denying the origin, submission, or delivery of a message and the integrity of its contents.

Non-Repudiation -- A security policy pertaining to providing one or more of the following:

- To the sender of data, proof of delivery to the intended recipient,
- To the recipient of data, proof of the identity of the user who sent the data.

Object -- An entity within the TSC that contains or receives information and upon which subjects perform operations.

Operating Environment -- The total environment in which a TOE operates. It includes the physical facility and any physical, procedural, administrative and personnel controls.

Operating System (OS) -- An entity within the TSC that causes operations to be performed. Subjects can come in two forms: trusted and untrusted. Trusted subjects are exempt from part or all of the TOE security policies. Untrusted subjects are bound by all TOE security policies.

Operational key -- Key intended for protection of operational information or for the production or secure electrical transmissions of key streams.

Path Processing -- The means employed by a relying party to ensure that the certificates in a path leading from a relying party trust point to subscriber's public key certificate, are all valid. The validation activity includes chaining the subscriber and issuer names, using the subject public key from the parent certificate to verify the signature on a certificate, applying constraints imposed by the various extensions in the certificate, verifying that none of the certificates have expired or been revoked, and other X.509 certification path validation rules.

Peer TOEs -- Mutually authenticated TOEs that interact to enforce a common security policy.

Private Key -- A number, known only to the particular entity, its owner (i.e., the owner keeps the key secret). Owners use private keys to compute signatures on data they send and to decrypt information sent to them.

Public Key Certificate -- A digitally signed statement from one entity, the Certification Authority, binding the public key (and some other information) and the identity of the owner of the corresponding private key. The owner may be an individual, a system or device, an organization, or function.

Public Key–Enabled Application -- A software application that uses PK technology to: authenticate its users (people, systems, and devices), ensure information is not changed or modified either during transmission or storage, hold users responsible and accountable for their actions and representations (i.e., preventing subsequent denial of responsibility), or encrypt information between parties where prior arrangement is neither known nor practical. PK–enabled applications rely on a PKI to create certificates that correctly associate a public key with the name of the owner of the associated private key, to retrieve certificates, and to determine the current validity of certificates (e.g., obtain a Certificate Revocation List [CRL]).

Public Key Infrastructure -- The resources (people, systems, processes, and procedures) that provide services to register and identify new certificate owners, retrieve certificates, and determine the current validity of certificates.

Public Key Owner -- The entity for whom the key pair was generated and who is responsible for the secrecy and protection of the private key. The owner is the same entity as the subscriber associated with a certificate containing the owner's public key.

Public Key Technology -- Techniques and methods to generate related but different (asymmetric) keys for encryption and decryption and to use the keys to provide security services for authentication, confidentiality, integrity, and non-repudiation. The owner retains and keeps secret one of the asymmetric keys, the private key, and openly distributes the other asymmetric key, the public key. Also See **Asymmetric Key**.

Public Object -- An object for which the TSF unconditionally permits all entities "read" access. Only the TSF or authorized administrators may create, delete, or modify the public objects.

Relying Party -- An entity or an organization that depends on a certificate (i.e., uses the public key in the certificate for digital signature and/or encryption) and its association of the subscriber's identity (i.e., subject name) and public key.

Revoked Certificate -- A certificate that relying parties should not trust or use. The CA that issued the certificate (or some similar authority) may revoke the certificate when conditions warrant. Conditions that may warrant revocation include suspected or actual compromise of the key or departure of the subscriber from the organization. CRLs issued by the CA always include all revoked, unexpired certificates (see **Expired Certificate**). Optionally, the CA may include revoked, expired certificates.

Robustness -- A characterization of the strength of a security function, mechanism, service or solution, and the assurance (or confidence) that it is implemented and functioning correctly. DoD has three levels of robustness:

- **Basic:** Security services and mechanisms that equate to good commercial practices.
- **Medium:** Security services and mechanisms that provide for layering of additional safeguards above good commercial practices.
- **High:** Security services and mechanisms that provide the most stringent protection and rigorous security countermeasures.

Root Certificate -- The certificate at the top of the certification authority hierarchy. The certificate is self-signed; that is, the certificate issuer and the subject are the same entity, the Root CA. The certificate is generally a trust point. Since self-signed certificates do not have any trust in them, the root certificate or any other self-signed certificate must be distributed using secure means.

Secure State -- Condition in which all TOE security policies are enforced.

Security attributes -- TSF data associated with subjects, objects, and users that are used for the enforcement of the TSP.

Security level -- The combination of a hierarchical classification and a set of non-hierarchical categories that represent the sensitivity of the information.

Sensitivity label -- A security attribute that represents the security level of an object and that describes the sensitivity (e.g. Classification) of the data in the object. Sensitivity labels are used by the TOE as the basis for mandatory access control decision.

Signature Verification -- The process of verifying a signature that includes the following steps: 1. Certification path validation in order to establish trust in the signer public key, 2. Calculating the hash for the message to be verified, and 3. Using applicable cryptographic algorithm with the signer public key (from step 1), calculated hash (from step 2), and signature to determine if the signature is valid.

Split key -- A variable that consists of two or more components that must be combined to form the operational key variable. The combining process excludes concatenation or interleaving of component variables.

Subject -- An entity within the TSC that causes operations to be performed.

Symmetric Key -- A key that is used to both encrypt and decrypt information. Parties involved in using the key must keep the key secret; anyone with knowledge of the key could either originate or view encrypted information.

Subscriber -- The entity (e.g., an individual) that has possession of the private key corresponding to the public key in a certificate. The certificate’s subject field names the subscriber.

Threat -- Capabilities, intentions and attack methods of adversaries, or any circumstance or event, with the potential to violate the TOE security policy.

Threat Agent - Any human user or Information Technology (IT) product or system, which may attempt to violate the TSP and perform an unauthorized operation with the TOE.

Trust anchor -- A public key that a relying party directly trusts. A trust anchor can be in the form of a self-signed certificate. The self-signed certificate may belong to either a CA or an end-entity. The trust anchor is trusted because the relying party obtained the public key by reliable means outside of the PKI and believes that the trust anchor information (i.e., subject DN, public key, public key algorithms, and public key parameters (if applicable) are accurate. If the trust anchor is a CA, the relying party trusts any certificates the CA issues. This trust is transitive to the extent the X.509 certificate extensions permit; if the CA issues a certificate to another CA, the relying party also trusts the second CA if the X.509 path validation logic succeeds.

Trusted Third Party (TTP) -- An entity that other entities believe reliable, trustworthy and beyond reproach for purposes of performing some service. The TTP generally has no bias and is neutral for purposes of performing the service.

Trusted Timestamp -- A digitally signed collection or other means that provides proof that a document existed at a particular time. The collection includes the date and time and either the document or the hash of the document. Often a TTP provides a timestamp service.

User -- Any entity (human user or external IT entity) outside the TOE that interacts with the TOE.

Vulnerability -- A weakness that can be exploited to violate the TOE security policy.

C. List of Acronyms

CA	Certification Authority
CAC	Common Access Card
CC	Common Criteria
CEM	Common Evaluation Methodology
CPV	Certification Path Validation
CRL	Certificate Revocation List
CRLDP	CRL Distribution Point
DH	Diffie Hellman
DISA	Defense Information Systems Agency
DN	Distinguished Name
DoD	Department of Defense
DSA	Digital Signature Algorithm
EAL	Evaluation Assurance Level
ECDH	Elliptic Curve Diffie Hellman
EFS	Encrypted File System
EKU	Extended Key Usage
FIPS	Federal Information Processing Standard
GMT	Greenwich Mean Time
HMAC	Hash based Message Authentication Code
IDP	Issuing Distribution Point
IEC	International Electrotechnical Committee
IETF	Internet Engineering Task Force
ISO	International Organisation for Standards
IT	Information Technology
JITC	Joint Interoperability Test Center

NSA	National Security Agency
OCSP	On-line Certificate Status Protocol
OS	Operating System
PKCS	Public Key Cryptography Standard
PKE	Public Key Enabled
PKEPP	Public Key Enabled (**PKE**) Protection Profile (**PP**)
PKI	Public Key Infrastructure
PKIX	Public Key Infrastructure Working Group -- **IETF**
PP	Protection Profile
RFC	Request for Comment
RSA	Rivest, Shamir, and Adelman
SCL	Smart Card Logon
SCVP	Simple Certificate Validation Protocol
SFP	Security Function Policy
SOF	Strength of Function
SSL	Secure Socket Layer
ST	Security Target
TLS	Transport Layer Security
TOE	Target of Evaluation
TSC	**TSF** Scope of Control
TSF	**TOE** Security Function
USMC	United States Marine Corps

D. Robustness Environment Characterization

General Environmental Characterization

In trying to specify the environments in which TOEs with various levels of robustness are appropriate, it is useful to first discuss the two defining factors that characterize that environment: **value of the resources** and **authorization of the entities** to those resources.

In general terms, the environment for a TOE can be characterized by the authorization (or lack of authorization) the least trustworthy entity has with respect to the highest value of TOE resources (i.e. the TOE itself and all of the data processed by the TOE).

Note that there are an infinite number of combinations of entity authorization and value of resources; this conceptually “makes sense” because there are an infinite number of potential environments, depending on how the resources are valued by the organization, and the variety of authorizations the organization defines for the associated entities. In the sections below, these two environmental factors will be related to the robustness required for selection of an appropriate TOE.

Value of Resources

Value of the resources associated with the TOE includes the data being processed or used by the TOE, as well as the TOE itself (for example, a real-time control processor). “Value” is assigned by the using organization. For example, in the DoD low-value data might be equivalent to data marked “FOUO”, while high-value data may be those classified Top Secret. In a commercial enterprise, low-value data might be the internal organizational structure as captured in the corporate on-line phone book, while high-value data might be corporate research results for the next generation product. Note that when considering the value of the data one must also consider the value of data or resources that are accessible through exploitation of the TOE. For example, a firewall may have “low value” data itself, but it might protect an enclave with high value data. If the firewall was being depended upon to protect the high value data, then it must be treated as a high-value-data TOE.

Authorization of Entities

Authorization that entities (users, administrators, other IT systems) have with respect to the TOE (and thus the resources of that TOE, including the TOE itself) is an abstract concept reflecting a combination of the trustworthiness of an entity and the access and privileges granted to that entity with respect to the resources of the TOE. For instance, entities that have total authorization to all data on the TOE are at one end of this spectrum; these entities may have privileges that allow them to read, write, and modify anything on the TOE, including all TSF data. Entities at the other end of the spectrum are those that are authorized to few or no TOE resources. For example, in the case of a router, non-administrative entities may have their packets routed by the TOE, but that is the extent of their authorization to the TOE's resources. In the case of an OS, an entity may not be allowed to log on to the TOE at all (that is, they are not valid users listed in the OS’s user database).

It is important to note that authorization **does not** refer to the **access** that the entities actually have to the TOE or its data. For example, suppose the owner of the system determines that no one other than employees was authorized to certain data on a TOE, yet they connect the TOE to the Internet. There are millions of entities that are not

authorized to the data (because they are not employees), but they actually have connectivity to the TOE through the Internet and thus can attempt to access the TOE and its associated resources.

Entities are characterized according to the value of resources to which they are authorized; the extent of their authorization is implicitly a measure of how trustworthy the entity is with respect to compromise of the data (that is, compromise of any of the applicable security policies; e.g., confidentiality, integrity, availability). In other words, in this model the greater the extent of an entity's authorization, the more trustworthy (with respect to applicable policies) that entity is.

Selection of Appropriate Robustness Levels

Robustness is a characteristic of a TOE defining how well it can protect itself and its resources; a more robust TOE is better able to protect itself. This section relates the defining factors of IT environments, authorization, and value of resources to the selection of appropriate robustness levels.

When assessing any environment with respect to Information Assurance the critical point to consider is the likelihood of an attempted security policy compromise, which was characterized in the previous section in terms of entity authorization and resource value. As previously mentioned, robustness is a characteristic of a TOE that reflects the extent to which a TOE can protect itself and its resources. It follows that as the likelihood of an attempted resource compromise increases, the robustness of an appropriate TOE should also increase.

It is critical to note that several combinations of the environmental factors will result in environments in which the likelihood of an attempted security policy compromise is similar. Consider the following two cases:

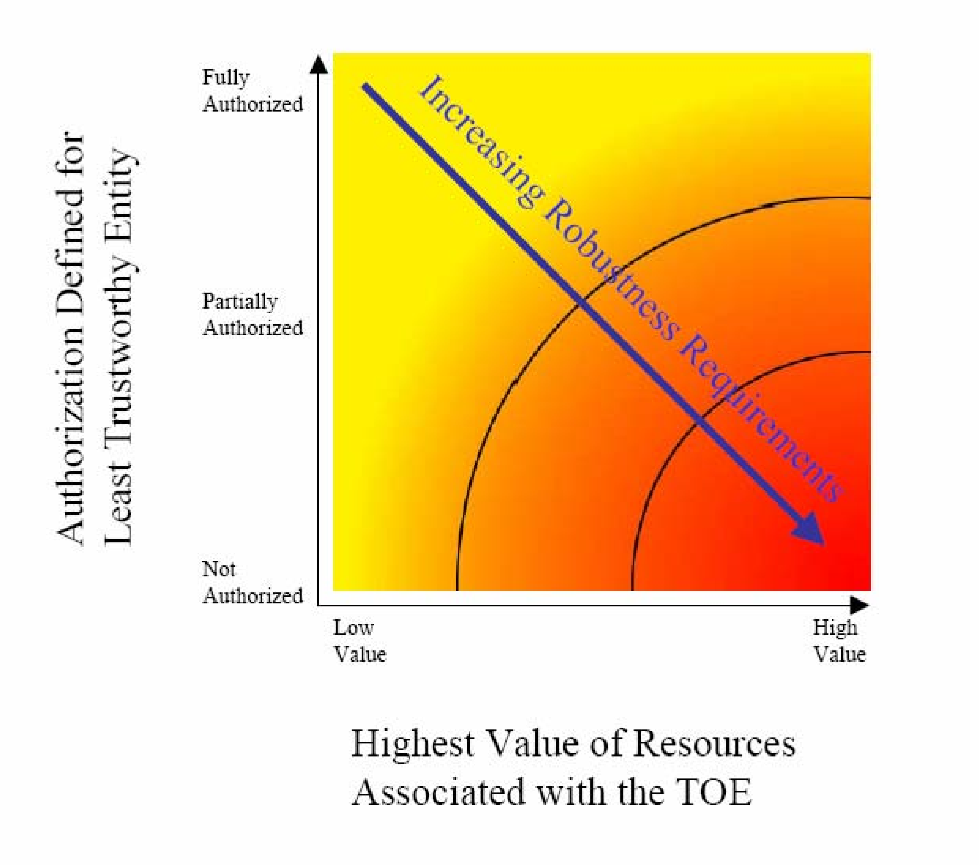
The first case is a TOE that processes only low-value data. Although the organization has stated that only its employees are authorized to log on to the system and access the data, the system is connected to the Internet to allow authorized employees to access the system from home. In this case, the least trusted entities would be unauthorized entities (e.g. non-employees) exposed to the TOE because of the Internet connectivity. However, since only low-value data are being processed, the likelihood that unauthorized entities would find it worth their while to attempt to compromise the data on the system is low and selection of a basic robustness TOE would be appropriate.

The second case is a TOE that processes high-value (e.g., classified) information. The organization requires that the TOE be stand-alone, and that every user with physical and logical access to the TOE undergo an investigation so that they are authorized to the highest value data on the TOE. Because of the extensive checks done during this investigation, the organization is assured that only highly trusted users are authorized to use the TOE. In this case, even though high value information is being processed, it is unlikely that a compromise of that data will be attempted because of the authorization and trustworthiness of the users and once again, selection of a basic robustness TOE would be appropriate.

The preceding examples demonstrated that it is possible for radically different combinations of entity authorization/resource values to result in a similar likelihood of an attempted compromise. As mentioned earlier, the robustness of a system is an indication of the protection being provided to counter compromise attempts. Therefore, a basic robustness system should be sufficient to counter compromise attempts where the likelihood of an attempted compromise is low. The following chart depicts the “universe” of environments characterized by the two factors discussed in the previous section: on one axis is the authorization defined for the least trustworthy entity, and on the other axis is the highest value of resources associated with the TOE.

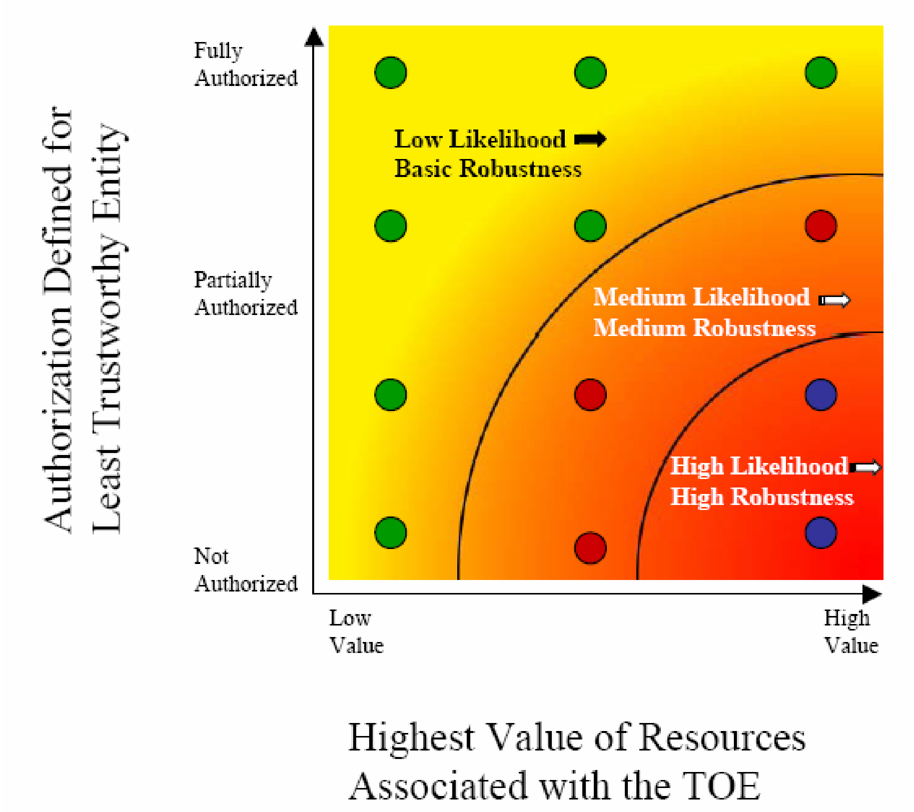
As depicted in the following figure, the robustness of the TOEs required in each environment steadily increases as one goes from the upper left of the chart to the lower right; this corresponds to the need to counter increasingly likely attack attempts by the least trustworthy entities in the environment. Note that the shading of the chart is intended to reflect- the notion that different environments engender similar levels of “likelihood of attempted compromise”, signified by a similar color. Further, the delineations between such environments are not stark, but rather are finely grained and gradual.

While it would be possible to create many different "levels of robustness" at small intervals along the “Increasing Robustness Requirements” line to counter the increasing likelihood of attempted compromise due to those attacks, it would not be practical nor particularly useful. Instead, in order to implement the robustness strategy where there are only three robustness levels: Basic, Medium, and High, the graph is divided into three sections, with each section corresponding to a set of environments where the likelihood of attempted compromise is roughly similar. This is graphically depicted in the following chart


Authorization Defined for Least Trustworthy Entity
Fully Authorized
Partially Authorized
Not Authorized
Increasing Robustness Requirements
Low Value
High Value
Highest Value of Resources Associated with the TOE

In this second representation of environments and the robustness plane below, the “dots” represent given instantiations of environments; like-colored dots define environments with a similar likelihood of attempted compromise. Correspondingly, a TOE with a given robustness should provide sufficient protection for environments characterized by like-colored dots. In choosing the appropriateness of a given robustness level TOE PP for an environment, then, the user must first consider the lowest authorization for an entity as well as the highest value of the resources in that environment. This should result in a “point” in the chart above, corresponding to the likelihood that that entity will attempt to compromise the most valuable resource in the environment. The appropriate robustness level for the specified TOE to counter this likelihood can then be chosen.

The difficult part of this activity is differentiating the authorization of various entities, as well as determining the relative values of resources; (e.g., what constitutes “low value” data vs. “medium value” data). Because every organization will be different, a rigorous definition is not possible. In Section 3 of this PP, the targeted threat level for a Basic robustness TOE is



characterized. This information is provided to help organizations using this PP - ensure that the functional requirements specified by this Basic robustness PP are appropriate for their intended application of a compliant TOE

E. IT Environment Testing

The IT Environment, i.e., the operating system must meet the functional requirements for the IT Environment except for the following requirements:

- FCS_CRM_FPS_(EXP).1
- FPT_SEP_ENV_(EXP).1
- FPT_TST_SOF_(EXP).1

Testing for these requirements is listed separately.

In order to ensure that the IT Environment meets the requirements stated in this PP, a CCTL shall perform the following actions:

1. Verify that the operating systems has obtained a CCEVS certificate for the TOE assurance level (i.e., EAL 3 or EAL 4 augmented by ALC_FLR.2)

2. Examine the operating ST to verify that the operating system assumptions do not contradict the assumptions in this PP.

3. Examine the operating ST and the TOE ST to verify that the operating system assumptions do not contradict the assumptions in the TOE ST.

4. Examine the SFRs in the ST to verify that they provide demonstrable conformance to the SFRs for the IT Environment in this PP, except for the SFRs listed above. These SFRs will be verified using means described herein.

5. During the TOE testing, the operating system shall be configured in accordance with the Guidance Documentation in its evaluated configuration.

6. If the TOE has any privileges with respect to the operating system, the CCTL shall:
   a) Determine that each privilege is required.
   b) Document or review TOE Sponsor document rationale for each privilege use. The rationale shall fall under one or more of the following categories: required to invoke OS security functional requirement; TOE Design; OS – TOE Interface.
   c) Analyze the operating system Guidance document and verify that the TOE uses the privileges as prescribed in the Guidance document.
   d) Analyze that the TOE does not mis-use the privilege.

7. The TOE design document (e.g., Functional Specification) shall describe the composite mechanism, including details of the operating system interfaces used and how the TOE preserves the underlying operating system security.

8. The TOE design document (e.g., ST, architecture or composition document) shall also describe how FCS_CRM_FPS_(EXP).1 requirement is met. The CCTL shall perform the following work units:
   a) The CCTL shall check that the design document lists the cryptographic module(s) used by the TOE. The identification of the module(s) shall be sufficiently detailed for the CCTL to verify FIPS 140 validation
   b) The CCTL shall examine the NIST web site to verify that each cryptographic module listed has been FIPS 140 validated.
   c) During the TOE testing, the CCTL shall use the FIPS 140 security policy for the module and vendor guidance to configure the module in FIPS 140 compliant and validated mode.

9. The TOE design document (e.g., ST, architecture or composition document) shall also describe how FPT_SEP_ENV_(EXP).1 requirement is met. The CCTL shall perform the following work units:
   a) The CCTL shall check that the design document describes how the requirement is met.
   b) The CCTL shall analyze the design document to analyze that the requirement is satisfied. As the application note with this requirement states, the most likely way to meet the requirement is process isolation and discretionary access control for the TOE executable and TOE data and attributes and TOE users' data and attributes. The CCTL shall ensure that the design document specifies these access control protection bits and or lists.

10. The TOE design document (e.g., ST, architecture or composition document) shall also describe how FPT_TST_SOF_(EXP).1 requirement is met. The CCTL shall perform the following work units:
    a) The CCTL shall check that the design document describes how the requirement is met.
    b) If the requirement is allocated to the TOE, the SFR can be analyzed and tested as part of the TOE.
    c) If the requirement is allocated to the IT Environment, the CCTL shall verify the following
       i) The cryptographic mechanism used to check the integrity of the TOE is invoked using one of the cryptographic modules used by the TOE as analyzed in item 8 above.
       ii) The design document describes how the integrity of the integrity verifier (e.g., public key if the mechanism is digital signature; secret key if the mechanism is HMAC or encryption based MAC; hash if the hash is the mechanism) is protected.

11. The following IT Environment requirements shall be considered met, if the CCTL configures the underlying IT Environment (e.g., the Operating System) and performs security functional testing commensurate with evaluation assurance level of the ST. It should be noted that to verify that some of these requirements are met, the

underlying IT Environment may need additional specific configuration changes after putting the IT Environment in evaluated configuration. In such a case, the IT Environment shall be first configured to meet the evaluated configuration requirements. Then, the testing identified in the Appendix and TOE testing shall be conducted: FIA_AFL.1, FMT_MOF.1, FMT_MTD.1:1 through 6, FPT_STM.1, FTA_SSL.1, FTA_SSL.2, and FTA_TAB.1.

12. The FIA_UID.2 and FIA_UAU.2 requirements shall be considered having been met as long as the ST for the underlying operating system claims FIA_UID.1 and FIA_UAU.1 and only permits the following activities prior to identification and authentication: access to Web Services. If the IT Environment contains Web Services, the Web Services shall be configured to provide access to only public folders for unauthenticated users.

F. Demonstrable Conformance Evaluation

This appendix defines demonstrable conformance and describes how demonstrable conformance is achieved.

Definition

Demonstrable conformance requires evidence that the ST is a suitable solution to the generic security problem described in the PP. Demonstrable conformance requires the following to be true:

- The SARs specified in the ST must be a non-strict superset of the SARs specified in the PP; i.e. the ST must claim SARs specified in the PP as a minimum, but could claim more (or hierarchically stronger SARs).
- The ST, although ensuring all requirements specified in the PP are expressed in the ST, is able to use alternative SFRs. A rationale must be provided to explain how the SFRs specified in the ST achieves at least the same as the SFRs specified in the PP.
- Any changes to the security objectives for the environment shall make the description more restrictive (in the sense of refinement), or be as a result of moving an objective specified for the environment in the PP to become an objective for the TOE in the ST. A rationale shall be provided to explain how the environment described in the ST is consistent with that described in the PP.
- The completion of operations on security functional requirements shall be consistent with those in the PP; i.e. the same completion shall be used in the ST as that in the PP or a completion that makes the requirement more restrictive (the rules of refinement apply).

  For example, if the PP author restricts the selection of four items in the component FAU_GEN.1.1b to two items in the PP. The ST can then only choose from the two in the PP, and not the other two. Nevertheless, the ST author may also add some audit events within the assignment in FAU_GEN.1.1c.

Objective

The conformance rationale for demonstrable conformance shall show the following:

1. How each requirement in the PP is represented in the ST. If alternative requirements are expressed in the ST, the rationale shall contain the ST authors understanding of the relevant PP objective(s) and how the alternative requirement(s) still result in achievement of the objective(s).
2. That the statement of objectives for the TOE in the PP is fully expressed in the ST. This may be either:
   - Through equivalent or more restrictive objectives than those in the PP; or
   - Through expression of a TOE requirement that has been introduced in the ST to meet an objective stated for the TOE in the PP.

3. The source of each additional security requirement; how it is necessary to meet the expanded set of security objectives for the TOE, resulting from the expanded security objectives in the ST.

Evaluation Methodology

Security Environment

The evaluator shall examine the conformance claim rationale to determine that it demonstrates that the security environment of the ST is at least equivalent to the security environment in the PP. Specifically, the evaluator shall ensure that:

- The secure usage assumptions in the ST do not attribute trust and security beyond that specified and implied by the secure usage assumptions in the PP. In other words, secure usage assumptions in the ST shall not mitigate more threats or additional aspects of threats then the secure usage assumptions in the PP;
- The threats described in the ST are no less restrictive than those in the PP (e.g., ST could include additional threat agents and/or resources for each threat). Another example is the ST could include additional threats that do not conflict with the threats in the PP; and
- The organization security policies in the ST impose no less security requirements on the TOE than those imposed by the organization security policies in the PP.

Security Objectives

The evaluator shall examine the conformance claim rationale to determine that it demonstrates that the statement of security objectives of the ST is at least equivalent to the statement of security objectives in the PP. It is acceptable for one or more security objectives to be broader (e.g., in terms of resources they cover).

Security Functional Requirements

The evaluator shall examine the conformance claim rationale to determine that it demonstrates that the statement of security functional requirements of the ST is at least equivalent to the statement of security functional requirements in the PP.

The FIA_UID.2 and FIA_UAU.2 requirements shall be considered having been met as long as the ST for the underlying operating system claims FIA_UID.1 and FIA_UAU.1 and only permits the following activities prior to identification and authentication: access to Web Services. If the IT Environment contains Web Services, the Web Services shall be configured to provide access to only public folders for unauthenticated users.

Security Assurance Requirements

The evaluator shall examine the conformance claim rationale to determine that it demonstrates that the statement of SARs specified in the ST is a non-strict superset of the SARs specified in the PP; i.e. the ST must claim SARs specified in the PP as a minimum, but could claim more (or hierarchically stronger SARs).