

BITS Product Certification Program



Common Criteria Package of Requirements for Application Products

Table of Contents

1. INTRODUCTION	5
1.1 Package Identification	5
1.2 Acronyms and Glossary.....	5
1.3 Application Product Package Overview.....	5
1.4 Application Product Description.....	6
2. REQUIREMENTS FOR THE ADMINISTRATION AND OPERATION OF THE PRODUCT...	7
2.1 Class FAU: Security Audit.....	10
2.1.1 <i>SECURITY AUDIT AUTOMATIC RESPONSE (FAU_ARP)</i>	10
2.1.2 <i>SECURITY AUDIT DATA GENERATION (FAU_GEN)</i>	10
2.1.3 <i>SECURITY AUDIT ANALYSIS (FAU_SAA)</i>	11
2.1.4 <i>SECURITY AUDIT REVIEW (FAU_SAR)</i>	11
2.1.5 <i>SECURITY AUDIT EVENT STORAGE (FAU_STG)</i>	12
2.2 Class FCO: Communication.....	13
2.2.1 <i>NON-REPUDIATION OF ORIGIN (FCO_NRO)</i>	13
2.3 Class FCS: Cryptographic support	13
2.3.1 <i>CRYPTOGRAPHIC KEY MANAGEMENT (FCS_CKM)</i>	13
2.3.2 <i>CRYPTOGRAPHIC OPERATION (FCS_COP)</i>	14
2.4 Class FDP: User data protection.....	14
2.4.1 <i>ACCESS CONTROL POLICY (FDP_ACC)</i>	15
2.4.2 <i>ACCESS CONTROL FUNCTIONS (FDP_ACF)</i>	15
2.4.3 <i>DATA AUTHENTICATION (FDP_DAU)</i>	16
2.4.4 <i>INTERNAL TOE TRANSFER (FDP_ITT)</i>	16
2.4.5 <i>RESIDUAL INFORMATION PROTECTION (FDP_RIP)</i>	17
2.4.6 <i>STORED DATA INTEGRITY (FDP_SDI)</i>	17
2.4.7 <i>INTER-TSF USER DATA CONFIDENTIALITY TRANSFER PROTECTION</i> <i>(FDP_UCT)</i>	17
2.4.8 <i>INTER-TSF USER DATA INTEGRITY TRANSFER PROTECTION</i> <i>(FDP_UIT)</i>	17
2.5 Class FIA: Identification and Authentication.....	18
2.5.1 <i>AUTHENTICATION FAILURES (FIA_AFL)</i>	18
2.5.2 <i>USER ATTRIBUTE DEFINITION (FIA_ATD)</i>	18
2.5.3 <i>SPECIFICATION OF SECRETS (FIA_SOS)</i>	19
2.5.4 <i>USER AUTHENTICATION (FIA_UAU)</i>	20
2.5.5 <i>USER IDENTIFICATION (FIA_UID)</i>	21
2.5.6 <i>USER-SUBJECT BINDING (FIA_USB)</i>	21
2.6 Class FMT: Security management	21
2.6.1 <i>MANAGEMENT OF FUNCTIONS IN TSF (FMT_MOF)</i>	21
2.6.2 <i>MANAGEMENT OF SECURITY ATTRIBUTES (FMT_MSA)</i>	22
2.6.3 <i>MANAGEMENT OF TSF DATA (FMT_MTD)</i>	23
2.6.4 <i>REVOCAION (FMT_REV)</i>	23
2.6.5 <i>SECURITY ATTRIBUTE EXPIRATION (FMT_SAE)</i>	24

- 2.6.6 SECURITY MANAGEMENT ROLES (FMT_SMR)..... 24
- 2.7 Class FPR: Privacy 24
 - 2.7.1 UNOBSERVABILITY (FPR_UNO)..... 24
- 2.8 Class FPT: Protection of the TSF 25
 - 2.8.1 UNDERLYING ABSTRACT MACHINE TEST (FPT_AMT)..... 25
 - 2.8.2 FAIL SECURE (FPT_FLS)..... 25
 - 2.8.3 CONFIDENTIALITY OF EXPORTED TSF DATA (FPT_ITC) 25
 - 2.8.4 INTEGRITY OF EXPORTED TSF DATA (FPT_ITI)..... 25
 - 2.8.5 INTERNAL TOE TSF DATA TRANSFER (FPT_ITT) 26
 - 2.8.6 TRUSTED RECOVERY (FPT_RCV) 26
 - 2.8.7 REPLAY DETECTION (FPT_RPL)..... 27
 - 2.8.8 REFERENCE MEDIATION (FPT_RVM)..... 28
 - 2.8.9 TIME STAMPS (FPT_STM) 28
 - 2.8.10 TSF SELF-TEST (FPT_TST)..... 28
- 2.9 Class FTA: TOE access 28
 - 2.9.1 LIMITATION ON MULTIPLE CONCURRENT SESSIONS (FTA_MCS).... 28
 - 2.9.2 SESSION LOCKING (FTA_SSL)..... 29
 - 2.9.3 TOE ACCESS BANNERS (FTA_TAB)..... 29
 - 2.9.4 TOE ACCESS HISTORY (FTA_TAH) 29
 - 2.9.5 TOE SESSION ESTABLISHMENT (FTA_TSE) 30
- 2.10 Class FTP: Trusted path/channels 30
 - 2.10.1 INTER-TSF TRUSTED CHANNEL (FTP_ITC)..... 30
 - 2.10.2 TRUSTED PATH (FTP_TRP) 30
- 3. APPLICATION PRODUCT CLASS REQUIREMENTS 32**
- 4. ASSURANCE REQUIREMENTS 32**
- 5. PACKAGE APPLICATION NOTES 36**
- 6. APPENDIX A – INDUSTRY STANDARDS..... 37**
- 7. APPENDIX B - GLOSSARY 38**
- 8. APPENDIX C – BITS PRODUCT CERTIFICATION PROGRAM OVERVIEW..... 41**

Technical Contact Information

If further information regarding technical content is required, please contact:

BITSlab@fsround.org

Tel.: 202.289.4322

Fax: 202.289.3562

Document Feedback

If you have any comments (technical or otherwise) regarding this document, please send an email to BITSlab@fsround.org. Please include the document name along with your name, email address, telephone, and fax number, and include whether you would like to be contacted. *Please note: BITS will take all comments under advisement, but reserves the right to include or exclude suggested changes.*

Application Products Package – Document Version Control History

*Note: **Bold** in Version/Date column indicates a public release.*

Version	Date	Changes	Author	Reviewer
.1 Draft	17 Jul 02	Initial draft	Terrie L. Diaz, SAIC	Bob Williamson, SAIC Cynthia Reese, SAIC
1.0	28 Oct 02	FINAL	Laura Lundin, BITS	The BITS Lab Governance Committee

1. INTRODUCTION

The purpose of the Application Products Package is to identify the set of security requirements for application products used by the Financial Services industry. The **BITS Product Certification Program** has developed the security requirements identified in this Application Products Package. These security requirements are designed to be used within security specifications for application products articulated by Financial Service providers in a Protection Profile (PP) and by vendors providing application products to Financial Service providers through a Security Target (ST).

1.1 PACKAGE IDENTIFICATION

Package Title – Application Products Package
Package Version – Version 1.0

1.2 ACRONYMS AND GLOSSARY

The acronyms used in this Application Products Package are specified in Appendix B – Glossary

1.3 APPLICATION PRODUCT PACKAGE OVERVIEW

This Application Products Package defines the minimum security requirements that must be implemented in order to receive the ***BITS Tested Mark*** and the ***NIAP Certification of CC Compliance***. The Application Product requirements have been derived and expanded from the Common Criteria – Master Security Requirements (CC-MSR). It is recommended that readers of this document review the CC-MSR and use it as a reference document to this Application Product Package.

The framework of requirements is designed to be hierarchical, from defining the overall security attributes and features expected in the administration and operation of products, to the specific Application Product class and related sub-class level features.

Requirements identified in the Administration and Operation section relate to the capabilities of the product itself to be secured (i.e., administrative interfaces, logging, authentication to the product, etc.). It is permissible for requirements outlined in this section to be fulfilled by the environment (underlying platform or supporting components as defined in boundaries of the test environment in the Security Target) when the TOE does not provide a required feature.

Requirements identified in the Product Class and Sub-class sections relate to the “security functionality” expected to be provided by this specific type of product. However, any product feature that does not support a security functional requirement is considered non-security product functionality and therefore would not be included within the Common Criteria syntax. Often, it is difficult to identify the difference between security features and non-security product functionality, especially when the product’s primary functionality is related to security. An example of product functionality that is not a

security functional requirement is: “the product should integrate with third-party systems management tools in order to facilitate centralized proactive alert monitoring and administration simplicity/consistency, etc.”

To earn the ***BITS Tested Mark***, the Application Product must meet all the mandatory requirements. A product will not merit a ***BITS Tested Mark*** if it misses any one mandatory requirement.

Some requirements are identified within this document as “desired.” These requirements are not necessary to obtain the ***BITS Tested Mark***, but compliance with them will be noted in the final test report. Desired requirements are recognized by the financial services industry as advantageous and may become mandatory requirements for certification in the future.

1.4 APPLICATION PRODUCT DESCRIPTION

The basic model of an application system is a client-server arrangement with the application interfacing with a browser or thin client, the underlying operating system, and other entities using application program interfaces (APIs). The Application Products Package lists the security-related criteria that apply to the features and functionality normally found in application products. Examples of applications at a financial institution include: electronic bill presentment and payment systems, funds transfer systems, personal finance managers, groupware and messaging systems, Web-based transaction systems and more.

2. REQUIREMENTS FOR THE ADMINISTRATION AND OPERATION OF THE PRODUCT

This section lists the security criteria that are common to all application products. The Common Criteria (CC) security functional requirements (SFRs) are the refinement of the security objectives into security requirements for IT products.

The SFRs are organized by CC class. The CC permits four functional component operations to refine an SFR to make the SFR more specific to the type of product or implementation required by the consumer. The four operations are assignment, refinement, selection, and iteration, to be performed on security functional requirements. The four operations are applied in the following manner:

- **Assignment:** Allows the specification of an identified parameter (indicated with bold)
- **Refinement:** Allows the addition of details (indicated with bold italics text)
- **Selection:** Allows the specification of one or more elements from a list (indicated with underlined text)
- **Iteration:** Allows a component to be used more than once with varying operations (indicated by a letter in parentheses placed at the end of the element names)

The following table lists Security Functional Requirement components organized by CC functional security class

Table 1 – Security Functional Requirements

Functional Security Class	Security Functional Requirement Components	Required/Desired
Security audit (FAU)	FAU_ARP.1 - Security Alarms	Required
	FAU_GEN.1 - Audit Data Generation	Required
	FAU_GEN.2 - User Identity Association	Required
	FAU_SAA.1 - Potential Violation Analysis	Required
	FAU_SAR.1 - Audit Review	Required
	FAU_SAR.2 - Restricted Audit Review	Required
	FAU_SAR.3 - Selectable Audit Review	Required
	FAU_STG.1 - Protected Audit Trail Storage	Required
	FAU_STG.2 - Guarantees of Audit Data Availability	Required
	FAU_STG.3 - Action In Case of Possible Audit Data Loss	Required
	FAU_STG.4 - Prevention of Audit Data Loss	Required
Communications (FCO)	FCO_NRO.2 - Enforced Proof of Origin	Required

Functional Security Class	Security Functional Requirement Components	Required/Desired
	FCO_NRR.2 - Enforce Proof of Receipt	Required
Cryptographic support (FCS)	FCS_CKM.1 - Cryptographic Key Generation	Required
	FCS_CKM.2 - Cryptographic Key Distribution	Required
	FCS_CKM.3 - Cryptographic Key Access	Required
	FCS_CKM.4 - Cryptographic Key Destruction	Required
	FCS_COP.1 - Cryptographic Operation	Required
User data protection (FDP)	FDP_ACC.2 - Complete Access Control	Required
	FDP_ACF.1 - Security Attribute-Based Access Control	Required
	FDP_DAU.2 - Data Authentication with Identification of Guarantor	Required
	FDP_ITT.1 - Basic Internal Transfer Protection	Required
	FDP_ITT.3 - Integrity Monitoring	Required
	FDP_RIP.1 - Subset Residual Information Protection	Required
	FDP_SDI.1 - Stored Data Integrity	Required
	FDP_UCT.1 - Basic Data Exchange Confidentiality	Required
	FDP_UIT.1 - Data Exchange Integrity	Required
Identification and authentication (FIA)	FIA_AFL.1 - Authentication Failure Handling	Required
	FIA_ATD.1 - User Attribute Definition	Required
	FIA_SOS.1 - Verification of Secrets	Required
	FIA_SOS.2 - TSF Generation of Secrets	Required
	FIA_UAU.1 - Timing of Authentication	Required
	FIA_UAU.3 - Unforgeable Authentication	Required
	FIA_UAU.5 - Multiple Authentication Mechanisms	Required – if claimed, otherwise this is a desired feature
	FIA_UAU.6 - Re-Authenticating	Required
	FIA_UAU.7 - Protected Authentication Feedback	Required
	FIA_UID.2 - User Identification Before Any Action	Required
	FIA_USB.1 - User-Subject Binding	Required
Security management (FMT)	FMT_MOF.1 - Management of Security Functions Behavior	Required
	FMT_MSA.1 - Management of Security Attributes	Required
	FMT_MSA.2 - Secure Security Attributes	Required

Functional Security Class	Security Functional Requirement Components	Required/Desired
	FMT_MSA.3 - Static Attribute Initialization	Required
	FMT_MTD.1 - Management of TSF Data	Required
	FMT_MTD.2 - Management Limits on TSF Data	Required
	FMT_MTD.3 - Secure TSF Data	Required
	FMT_REV.1 - Revocation	Required
	FMT_SAE.1 - Time-Limited Authorization	Required
	FMT_SMR.1 - Security Roles	Required
Privacy (FPR)	FPR_UNO.4 - Authorized User Observability	Required
Protection of the TSF (FPT)	FPT_AMT.1 - Abstract Machine Testing	Required
	FPT_FLS.1 - Failure with Preservation of Secure State	Required
	FPT_ITC.1 - Inter-TSF Confidentiality During Transmission	Required Dependency of FDP_UIT
	FPT_ITI.1 - Inter-TSF Detection of Modification	Required
	FPT_ITT.2 - TSF Data Transfer Separation	Required
	FPT_ITT.3 - TSF Data Integrity Monitoring	Required
	FPT_RCV.1 - Manual Recovery	Required
	FPT_RCV.3 - Automated Recovery Without Undue Loss	Required
	FPT_RCV.4 - Function Recovery	Required
	FPT_RPL.1 - Replay Detection	Required
	FPT_RVM.1 - Non-bypassability of the TSP	Required
	FPT_STM.1 - Reliable Time Stamps	Required Dependency of FAU_GEN
	FPT_TST.1 - Testing	Required
TOE access (FTA)	FTA_MCS.1 - Basic Limitation on Multiple Concurrent Sessions	Required
	FTA_SSL.3 - TSF-initiated Termination	Required
	FTA_TAB.1 - Default TOE Access Banners	Required
	FTA_TAH.1 - TOE Access History	Required
	FTA_TSE.1 - TOE Session Establishment	Required
Trusted path/channel (FTP)	FTP_ITC.1 - Inter-TSF Trusted Channel	Required Dependency of FDP_UIT
	FTP_TRP.1 - Trusted Path	Required

2.1 CLASS FAU: SECURITY AUDIT

2.1.1 SECURITY AUDIT AUTOMATIC RESPONSE (FAU_ARP)

2.1.1.1 FAU_ARP.1 Security Alarms

2.1.1.1.1 FAU_ARP.1.1

The TSF shall **have the capability to generate a real-time alarm and/or send an email notification to the administrator** in the event that a potential security violation or audit log malfunction is detected.

2.1.2 SECURITY AUDIT DATA GENERATION (FAU_GEN)

2.1.2.1 FAU_GEN.1 Audit Data Generation

2.1.2.1.1 FAU_GEN.1.1

The TSF shall be able to generate an audit record of the following auditable events:

- a) Start-up and shutdown of the TSF functions and equipment;
- b) All auditable events for the minimal level of audit; and
- c) The following events:
 - **All sessions established**
 - **Failed user authentication attempts**
 - **Failed attempts to access resources**
 - **Administrator actions**
 - **Administrator disabling of audit logging**
 - **Changes to user's security profile and/or attributes**
 - **Changes to security profile and/or attributes of system interfaces**
 - **Changes in permission levels needed to access a resource**
 - **Changes to system security configuration**
 - **Modifications to system software**
 - **Changes to critical system resources**

2.1.2.1.2 FAU_GEN.1.2

The TSF shall record within each audit the following information, at a minimum:

- a) Date and time of the event, type of event, subject identity, 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:
 - **User ID**
 - **Host name of system generating the log record**

- **Names of resources accessed**
- **Host name of system that initiated the attempted event**

2.1.2.2 FAU_GEN.2 User Identity Association

2.1.2.2.1 FAU_GEN.2.1

The TSF shall be able to associate each auditable event with the identity of the user that caused the event.

2.1.3 SECURITY AUDIT ANALYSIS (FAU_SAA)

2.1.3.1 FAU_SAA.1 Potential Violation Analysis

2.1.3.1.1 FAU_SAA.1.1

The TSF shall apply a set of rules in monitoring the audited events and based upon these rules indicate a known or suspected violation of the TSP.

2.1.3.1.2 FAU_SAA.1.2

The TSF shall enforce the following rules for monitoring audited events:

- a) Accumulation or combination of **administrator-specified set of auditable events** known to indicate a *known or suspected* security violation; and
- b) **No other rules**

2.1.4 SECURITY AUDIT REVIEW (FAU_SAR)

2.1.4.1 FAU_SAR.1 Audit Review

2.1.4.1.1 FAU_SAR.1.1

The TSF shall provide the **authorized administrator** with the capability to read, **retrieve, print, and copy the contents of the audit log** from the collected audit records *to a long-term storage device*.

2.1.4.1.2 FAU_SAR.1.2

The TSF shall provide the audit records in a manner suitable for the user to interpret the information.

2.1.4.2 FAU_SAR.2 Restricted Audit Review

2.1.4.2.1 FAU_SAR.2.1

The TSF shall prohibit all users *to* read, **write, modify, and/or delete** access to the audit records, except those users that have been granted explicit read, **write, modify, and/or delete** access.

2.1.4.3 FAU_SAR.3 Selectable Audit Review

2.1.4.3.1 FAU_SAR.3.1

The TSF shall provide the ability to perform **selective retrieval** of audit data based on **criteria with logical relations, such as a user ID and time-of-day or machine name and port-of-entry to perform functions such as producing reports and establishing audit trails.**

2.1.5 SECURITY AUDIT EVENT STORAGE (FAU_STG)

2.1.5.1 FAU_STG.1 Protected Audit Trail Storage

2.1.5.1.1 FAU_STG.1.1

The TSF shall protect the stored audit records from unauthorized deletion.

2.1.5.1.2 FAU_STG.1.2

The TSF shall be able to prevent modifications to the audit records.

2.1.5.2 FAU_STG.2 Guarantees of audit data availability

2.1.5.2.1 FAU_STG.2.1

The TSF shall protect the stored audit records from unauthorized deletion.

2.1.5.2.2 FAU_STG.2.2

The TSF shall be able to prevent *any* modifications to the audit records.

2.1.5.2.3 FAU_STG.2.3

The TSF shall ensure that **all** audit records will be maintained when the following conditions occur: audit storage exhaustion, failure, and through system restarts.

2.1.5.3 FAU_STG.3 Action in Case of Possible Audit Data Loss

2.1.5.3.1 FAU_STG.3.1

The TSF shall *have the capability to generate a real-time alarm and/or send an email notification to the administrator* if the audit trail exceeds **the storage capacity or there is a failure of the storage mechanism.**

2.1.5.4 FAU_STG.4 Prevention of Audit Data Loss

2.1.5.4.1 FAU_STG.4.1

The TSF shall prevent auditable events, except those taken by the authorized user with special rights and **provide the capability for the administrator to shut down or continue processing** if the audit trail is full.

2.2 CLASS FCO: COMMUNICATION

2.2.1 NON-REPUDIATION OF ORIGIN (FCO_NRO)

2.2.1.1 FCO_NRO.2 Enforced proof of origin

2.2.1.1.1 FCO_NRO.2.1

The TSF shall enforce the generation of evidence of origin for transmitted **information from a user or another system that is being replicated** at all times.

2.2.1.1.2 FCO_NRO.2.2

The TSF shall be able to relate the **certificate** of the originator of the information, and the **digital signature and other characteristics such as date and time** of the information to which the evidence applies.

2.2.1.1.3 FCO_NRO.2.3

The TSF shall provide a capability to verify the evidence of origin of information to recipient, given **the originator's certificate is authentic**.

2.2.1.2 FCO_NRR.2 Enforced proof of receipt

2.2.1.2.1 FCO_NRR.2.1

The TSF shall enforce the generation of evidence of receipt for received information from a user or another system that is being replicated.

2.2.1.2.2 FCO_NRR.2.2

The TSF shall be able to relate the **certificate** of the recipient of the information, and the **digital signature** and other characteristics such as date and time of the information to which the evidence applies.

2.2.1.2.3 FCO_NRR.2.3

The TSF shall provide a capability to verify the evidence of receipt of information to originator given **the recipient's certificate is authentic**.

2.3 CLASS FCS: CRYPTOGRAPHIC SUPPORT

2.3.1 CRYPTOGRAPHIC KEY MANAGEMENT (FCS_CKM)

2.3.1.1 FCS_CKM.1 Cryptographic key generation

2.3.1.1.1 FCS_CKM.1.1

The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm **3DES, IDEA, RC4, RC5, or RIPEM**

and specified cryptographic key sizes **1024 bit** that meet the following: **ANS X9, CMP, PKCS #7, #10, or IETF PKIX.**

2.3.1.2 FCS_CKM.2 Cryptographic key distribution

2.3.1.2.1 FCS_CKM.2.1

The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method **3DES, IDEA, RC4, RC5, or RIPEM** that meets the following: **ANS X9, CMP, PKCS #7, #10, or IETF PKIX.**

2.3.1.3 FCS_CKM.3 Cryptographic key access

2.3.1.3.1 FCS_CKM.3.1

The TSF shall perform **key assignment, key access** to include prevention of use of keys where the administrator-specified time period has expired, and key recovery in accordance with a specified cryptographic key access method **3DES, IDEA, RC4, RC5, or RIPEM** that meets the following: **ANS X9, CMP, PKCS #7, #10, or IETF PKIX.**

2.3.1.4 FCS_CKM.4 Cryptographic key destruction

2.3.1.4.1 FCS_CKM.4.1

The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method **3DES, IDEA, RC4, RC5, or RIPEM**, which also includes the immediate revocation of a user and the associated keying material when requested by an authorized administrator that meets the following: **FIPS 140-2**

2.3.2 CRYPTOGRAPHIC OPERATION (FCS_COP)

2.3.2.1 FCS_COP.1 Cryptographic operation

2.3.2.1.1 FCS_COP.1.1

The TSF shall perform **data encryption services** in accordance with a specified cryptographic algorithm **3DES, IDEA, RC4, RC5, or RIPEM** and cryptographic key sizes **1024** that meet the following: **ANS X9, CMP, PKCS #7, #10, or IETF PKIX.**

2.4 CLASS FDP: USER DATA PROTECTION

2.4.1 ACCESS CONTROL POLICY (FDP_ACC)

2.4.1.1 FDP_ACC.2 Complete Access Control

2.4.1.1.1 FDP_ACC.2.1

The TSF shall enforce the **Access Control Security Policy** on all **users, groups, resources, and interfaces** and all operations among subjects and objects covered by the SFP.

2.4.1.1.2 FDP_ACC.2.2

The TSF shall ensure that all operations between any subject in the ***TSF Scope of Control (TSC)*** and any object within the TSC are covered by an access control SFP.

2.4.2 ACCESS CONTROL FUNCTIONS (FDP_ACF)

2.4.2.1 FDP_ACF.1 Security Attribute-based Access Control

2.4.2.1.1 FDP_ACF.1.1

The TSF shall enforce the **Access Control Security Policy** to objects based on:

- **The user identity and group membership(s) associated with a subject;**
- **The ability to associate users with groups; and**
- **The following access control attributes associated with an object. The access control attributes must provide attributes with:**
 - **The ability to associate allowed or denied operations with one or more user identities**
 - **The ability to associate allowed or denied operations with one or more group identities**
 - **Defaults for allowed or denied operations (such as the ability to back-up files and time-of-day and port-of-entry)**

2.4.2.1.2 FDP_ACF.1.2

The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed:

- **The system shall deny the access unless a user has permission to access a resource.**
- **Unless a port has explicit permission to access a resource, the system shall deny the access to all users who log in to that interface.**

2.4.2.1.3 FDP_ACF.1.3

The TSF shall explicitly authorize access of subjects to objects based on the following additional rules:

- **No additional rules**

2.4.2.1.4 FDP_ACF.1.4

The TSF shall explicitly deny access of subjects to objects based on:

- **If a process's access control attribute is explicitly listed in the user identity attribute without access, the process is denied access, regardless of the group identity attribute**
- **Explicitly configured settings and/or controls such as damaging commands as delete all files.**

2.4.3 DATA AUTHENTICATION (FDP_DAU)

2.4.3.1 FDP_DAU.2 Data Authentication with Identity of Guarantor

2.4.3.1.1 FDP_DAU.2.1

The TSF shall provide a capability to generate evidence that can be used as a guarantee of the validity of **any information received from a network interface or entered via a user interface.**

2.4.3.1.2 FDP_DAU.2.2

The TSF shall provide **authorized administrator** with the ability to verify evidence of the validity of the indicated information and the identity of the user that generated the evidence.

2.4.4 INTERNAL TOE TRANSFER (FDP_ITT)

2.4.4.1 FDP_ITT.1 Basic Internal Transfer Protection

2.4.4.1.1 FDP_ITT.1.1

The TSF shall enforce the **Access Control Security Policy** to prevent the **disclosure_and/or modification**, of user data when it is transmitted between physically separated parts of the TOE.

2.4.4.2 FDP_ITT.3 Integrity Monitoring

2.4.4.2.1 FDP_ITT.3.1

The TSF shall enforce the **Access Control Security Policy** to monitor user data transmitted between physically separated parts of the TOE for the following errors:

- **Any integrity errors such as checksums or secure hashes and replay**

2.4.4.2.2 FDP_ITT.3.2

Upon detection of a data integrity error, the TSF shall **generate an alarm and/or send e-mail notification to authorized administrator.**

2.4.5 RESIDUAL INFORMATION PROTECTION (FDP_RIP)

2.4.5.1 FDP_RIP.1 Subset Residual Information Protection

2.4.5.1.1 FDP_RIP.1.1

The TSF shall ensure that any previous information content of a resource is made unavailable upon the **allocation of the resource to** the following objects:

- **Memory and disk storage space that has been identified as being available for allocation.**

2.4.6 STORED DATA INTEGRITY (FDP_SDI)

2.4.6.1 FDP_SDI.1 Stored Data Integrity Monitoring

2.4.6.1.1 FDP_SDI.1.1

The TSF shall monitor user data, *system files, and application software* stored within the TSC for **any integrity errors** on all objects, based on the following attributes:

- **Checksums**
- **Synchronization points**

2.4.7 INTER-TSF USER DATA CONFIDENTIALITY TRANSFER PROTECTION (FDP_UCT)

2.4.7.1 FDP_UCT.1 Basic Data Exchange Confidentiality

2.4.7.1.1 FDP_UCT.1.1

The TSF shall enforce the **Access Control Security Policy** to be able to **transmit *and* receive** objects in a manner protected from unauthorized disclosure.

2.4.8 INTER-TSF USER DATA INTEGRITY TRANSFER PROTECTION (FDP_UIT)

2.4.8.1 FDP_UIT.1 Data Exchange Integrity

2.4.8.1.1 FDP_UIT.1.1

The TSF shall enforce the **Access Control Security Policy** to be able to transmit *and* receive user data in a manner protected from modification, deletion, insertion, *and* replay errors.

2.4.8.1.2 FDP_UIT.1.2

The TSF shall be able to determine on receipt of user data, whether modification, deletion, insertion, or replay has occurred.

2.5 CLASS FIA: IDENTIFICATION AND AUTHENTICATION

2.5.1 AUTHENTICATION FAILURES (FIA_AFL)

2.5.1.1 FIA_AFL.1 Authentication Failure Handling

2.5.1.1.1 FIA_AFL.1.1

The TSF shall detect when **administrator specified number (maximum default number is four) of unsuccessful authentication attempts occur related to all authentication attempts.**

2.5.1.1.2 FIA_AFL.1.2

When the defined number of unsuccessful authentication attempts has been met or surpassed, the TSF shall:

- **Lock out the account for an administrator-specified threshold or until the administrator intervenes**
- **Notify authorized administrator (via alarm and/or e-mail)**

2.5.2 USER ATTRIBUTE DEFINITION (FIA_ATD)

2.5.2.1 FIA_ATD.1 User Attribute Definition

2.5.2.1.1 FIA_ATD.1.1

The TSF shall maintain the following list of security attributes belonging to individual users:

- **Unique user IDs**
- **Specific security characteristics as configured by an authorized administrator**
- **Autonomous processes running on behalf of a user, such as a print spooler shall be associated with an identifier code**

2.5.3 SPECIFICATION OF SECRETS (FIA_SOS)

2.5.3.1 FIA_SOS.1 Verification of Secrets

2.5.3.1.1 FIA_SOS.1.1

The TSF shall provide a mechanism to verify that secrets (*identification and authentication data, digital signatures, digital certificates, encryption keys, etc.*) meet *the following specifications*:

- **Encryption key lengths and algorithms must be compliant with public and widely accepted algorithms or financial services industry standards as listed in Appendix A.**
- **Transmission and storage of secrets must be secure; secrets shall not be transmitted in clear text or stored in clear text.**
- **Secrets shall not be displayed in clear text to any user, including the administrator.**
- **Users shall be able to change their own secrets; user must authenticate first in order to change the secret.**
- **Users shall be required to change initial secret; access is denied if user does not comply.**
- **Predefined secret expiration dates must be configurable by authorized administrator by user ID.**
- **Secrets must have redefined expiration dates with a notification warning of upcoming secret expiration date.**
- **Secrets may not be reused within an administrator-defined period.**
- **Secrets must have a predefined character length, minimum alphabetic character, minimum numeric character, and minimum special character.**
- **Secrets shall not be trivial or predictable; the use of traditional multiple use passwords or weak authentication mechanisms are unacceptable.**
- **Secrets shall not be disclosed if inadvertently chosen by another (unique) user ID.**

2.5.3.2 FIA_SOS.2 TSF Generation of Secrets

2.5.3.2.1 FIA_SOS.2.1

The TSF shall provide a mechanism to generate secrets that meet:

- **Defined quality metric as indicated in FIA_SOS.1.1**

2.5.3.2.2 FIA_SOS.2.2

The TSF shall be able to enforce the use of TSF-generated secrets for:

- **All network access**
- **All network and interface monitoring**
- **All configuration changes**
- **All access to security incident data**

2.5.4 USER AUTHENTICATION (FIA_UAU)

2.5.4.1 FIA_UAU.1 Timing of Authentication

2.5.4.1.1 FIA_UAU.1.1

The TSF shall allow **user identification** on behalf of the user to be performed before the user is authenticated.

2.5.4.1.2 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.

2.5.4.2 FIA_UAU.3 Unforgeable Authentication

2.5.4.2.1 FIA_UAU.3.1

The TSF shall prevent use of authentication data that has been forged by any user of the TSF.

2.5.4.2.2 FIA_UAU.3.2

The TSF shall prevent use of authentication data that has been copied from any other user of the TSF.

2.5.4.3 FIA_UAU.5 Multiple Authentication Mechanisms

2.5.4.3.1 FIA_UAU.5.1

The TSF shall provide **multiple (system- or user-generated secrets (passwords), PIN numbers, token seeds, smart card seeds, and/or biometrics) authentication mechanisms** to support user authentication.

2.5.4.3.2 FIA_UAU.5.2

The TSF shall authenticate any user's claimed identity according to the *following rules*:

- **General requirements that apply to all types of authentication mechanisms to minimize the compromise of the authenticator**
- **Knowledge- and possession-based requirements that address mechanisms that support security information known and possessed by the user and submitted for validation to verify the user's identity**
- **Personal characteristic-based requirements that securely capture the physical characteristics of the user and provides that data to the authentication process for validating the identity of the user**
- **System requirements including the system authenticating itself to the user and/or another system**

2.5.4.4 FIA_UAU.6 Re-authenticating

2.5.4.4.1 FIA_UAU.6.1

The TSF shall re-authenticate the user or process under the *following* conditions:

- **Pre-configured system requirement as defined by an authorized administrator, which includes the capability of random re-authentication during any active session**

2.5.4.5 FIA_UAU.7 Protected Authentication Feedback

2.5.4.5.1 FIA_UAU.7.1

The TSF shall provide only *an invalid response* (i.e., **the system shall not reveal which part of the authentication procedure is incorrect**) to the user while the authentication is in progress.

2.5.5 USER IDENTIFICATION (FIA_UID)

2.5.5.1 FIA_UID.2 User Identification Before any Action

2.5.5.1.1 FIA_UID.2.1

The TSF shall require each user to identify *his or herself* before allowing any other TSF-mediated actions on behalf of that user.

2.5.6 USER-SUBJECT BINDING (FIA_USB)

2.5.6.1 FIA_USB.1 User-Subject Binding

2.5.6.1.1 FIA_USB.1.1

The TSF shall associate the appropriate user security attributes with subjects acting on behalf of that user.

2.6 CLASS FMT: SECURITY MANAGEMENT

2.6.1 MANAGEMENT OF FUNCTIONS IN TSF (FMT_MOF)

2.6.1.1 FMT_MOF.1 Management of Security Functions Behavior

2.6.1.1.1 FMT_MOF.1.1

The TSF shall restrict the ability to disable, enable, or modify the behavior of the functions **administrator-configured confidentiality mechanisms** to **authorized administrators**.

2.6.2 MANAGEMENT OF SECURITY ATTRIBUTES (FMT_MSA)

2.6.2.1 FMT_MSA.1 Management of Security Attributes

2.6.2.1.1 FMT_MSA.1.1

The TSF shall enforce the **Access Control Security Policy** to restrict the ability to change default, query, modify, delete, create, and/or bypass the *following* security attributes: **administrator-configured data integrity controls, security-related attributes of users, interfaces, and software and data elements to authorized administrators.**

2.6.2.2 FMT_MSA.2 Secure Security Attributes

2.6.2.2.1 FMT_MSA.2.1

The TSF shall ensure that only secure values are accepted for security attributes.

Application Note: This component applies to security attributes that are used to maintain the TSP. Other user attributes may be specified in the ST, other attributes such as users, subjects and objects have associated security attributes that will affect the behavior of the TSF. Examples of such security attributes are the groups to which a user belongs, the roles he/she might assume, the priority of a process (subject), and the rights belonging to a role or a user. These security attributes might need to be managed by the user, a subject or a specific authorized user (a user with explicitly given rights for this management). Additionally, this component contains requirements on the values that can be assigned to security attributes. The assigned values should be such that the TOE will remain in a secure state. The definition of what 'secure' means is not answered in this component but is left to the development of the TOE (specifically ADV_SPM.1 Informal TOE security policy model) and the resulting information in the guidance. An example could be that if a user account is created, it should have a non-trivial password. A further example could be that the TOE shall perform validity checks on the entered data so that it only accepts data that is within acceptable ranges and proper lengths.

2.6.2.3 FMT_MSA.3 Static Attribute Initialization

2.6.2.3.1 FMT_MSA.3.1

The TSF shall enforce the **Access Control Security Policy** to provide restrictive default values for security attributes that are used to enforce the SFP.

2.6.2.3.2 FMT_MSA.3.2

The TSF shall allow the **authorized administrators** to specify alternative initial values to override the default values when an object or information is created.

2.6.3 MANAGEMENT OF TSF DATA (FMT_MTD)

2.6.3.1 FMT_MTD.1 Management of TSF Data

2.6.3.1.1 FMT_MTD.1.1

The TSF shall restrict the ability to change default, query, modify, delete, or clear the **administrator configurable security enforcing functions of the TSF data to authorized administrators.**

2.6.3.2 FMT_MTD.2 Management of Limits on TSF Data

2.6.3.2.1 FMT_MTD.2.1

The TSF shall restrict the specification of the limits for **all administrator-configurable security enforcing functions of the TSF data to authorized administrators.**

2.6.3.2.2 FMT_MTD.2.2

The TSF shall take the following actions, if the TSF data are at, or exceed, the indicated limits:

- **Generate an alarm**
- **Send e-mail to the authorized administrators**

2.6.3.3 FMT_MTD.3 Secure TSF Data

2.6.3.3.1 FMT_MTD.3.1

The TSF shall ensure that only secure values are accepted for TSF data.

2.6.4 REVOCATION (FMT_REV)

2.6.4.1 FMT_REV.1 Revocation

2.6.4.1.1 FMT_REV.1.1

The TSF shall restrict the ability to revoke security attributes associated with the users, subjects, objects, and other additional resources within the TSC to **authorized administrators.**

2.6.4.1.2 FMT_REV.1.2

The TSF shall enforce the rules:

- **Access rights based on user and interface privileges**
- **Immediate revocation of attributes**
- **No other rules**

Application Note: Many security-relevant authorizations could have serious consequences if misused, so an immediate revocation method must exist, although it need not be the usual method. (For example, the usual method may be editing the trusted users profile, but the change doesn't take effect until the user logs off and logs back on. The method for immediate revocation might be to edit the trusted users profile and

“force” the trusted user to log off). The immediate method must be specified in the ST and in administrator guidance. In a distributed environment the developer must provide a description of how the “immediate” aspect of this requirement is met.

2.6.5 SECURITY ATTRIBUTE EXPIRATION (FMT_SAE)

2.6.5.1 FMT_SAE.1 Time-Limited Authorization

2.6.5.1.1 FMT_SAE.1.1

The TSF shall restrict the capability to specify an expiration time, *such as, three months* for **account inactivity (active accounts that are dormant), to authorized administrators.**

2.6.5.1.2 FMT_SAE.1.2

For each of these security attributes, the TSF shall be able to **automatically disable and lock the account and send notification to the authorized administrators** after the expiration time for the indicated security attribute has passed.

2.6.6 SECURITY MANAGEMENT ROLES (FMT_SMR)

2.6.6.1 FMT_SMR.1 Security Roles

2.6.6.1.1 FMT_SMR.1.1

The TSF shall maintain the roles:

- **Authorized users with privileges to modify their own authentication data (secrets/passwords)**
- **Authorized administrators**

2.6.6.1.2 FMT_SMR.1.2

The TSF shall be able to associate users with roles.

2.7 CLASS FPR: PRIVACY

2.7.1 UNOBSERVABILITY (FPR_UNO)

2.7.1.1 FPR_UNO.4 Authorized User Observability

2.7.1.1.1 FPR_UNO.4.1

The TSF shall provide **authorized administrators** with the capability to observe the usage of:

- **All terminals, ports, and network addresses**

- **All interfaces**
- **All users currently logged on**

2.8 CLASS FPT: PROTECTION OF THE TSF

2.8.1 UNDERLYING ABSTRACT MACHINE TEST (FPT_AMT)

2.8.1.1 FPT_AMT.1 Abstract Machine Testing

2.8.1.1.1 FPT_AMT.1.1

The TSF shall run a suite of tests periodically during normal operation *and* at the request of an authorized user to demonstrate the correct operation of the security assumptions provided by the abstract machine that underlies the TSF.

2.8.2 FAIL SECURE (FPT_FLS)

2.8.2.1 FPT_FLS.1 Failure with Preservation of Secure State

2.8.2.1.1 FPT_FLS.1.1

The TSF shall preserve a secure state when the following types of failures occur:

- **Buffer overflow**

2.8.3 CONFIDENTIALITY OF EXPORTED TSF DATA (FPT_ITC)

2.8.3.1 FPT_ITC.1 Inter-TSF Confidentiality During Transmission

2.8.3.1.1 FPT_ITC.1.1

The TSF shall protect all TSF data transmitted from the TSF to a remote, trusted IT product from unauthorized disclosure during transmission.

2.8.4 INTEGRITY OF EXPORTED TSF DATA (FPT_ITI)

2.8.4.1 FPT_ITI.1 Inter-TSF Detection of Modification

2.8.4.1.1 FPT_ITI.1.1

The TSF shall provide the capability to detect modification of all TSF data during transmission between the TSF and a remote, trusted IT product within the following metrics:

- **Data integrity checks**
- **Verification of checksums**

- **Various tools used by authorized administrators**

2.8.4.1.2 FPT_ITI.1.2

The TSF shall provide the capability to verify the integrity of all TSF data transmitted between the TSF and a remote trusted IT product and **have the ability to generate an alarm and/or send e-mail notification to the authorized administrators** if modifications are detected.

2.8.5 INTERNAL TOE TSF DATA TRANSFER (FPT_ITT)

2.8.5.1 FPT_ITT.2 TSF Data Transfer Separation

2.8.5.1.1 FPT_ITT.2.1

The TSF shall protect TSF data from disclosure **and** modification when it is transmitted between separate parts of the TOE.

2.8.5.1.2 FPT_ITT.2.2

The TSF shall separate user data from TSF data when such data is transmitted between separate parts of the TOE.

2.8.5.2 FPT_ITT.3 TSF Data Integrity Monitoring

2.8.5.2.1 FPT_ITT.3.1

The TSF shall be able to detect modification of data, substitution of data, re-ordering of data, and deletion of data for TSF data transmitted between separate parts of the TOE.

2.8.5.2.2 FPT_ITT.3.2

Upon detection of a data integrity error, the TSF shall take the following actions:

- **Generate an alarm**
- **Send e-mail notification to authorized administrators**
- **Reject data**

2.8.6 TRUSTED RECOVERY (FPT_RCV)

2.8.6.1 FPT_RCV.1 Manual Recovery

2.8.6.1.1 FPT_RCV.1.1

After a failure or service discontinuity, the TSF shall enter a maintenance mode where the ability to return the TOE to a secure state is provided.

2.8.6.2 FPT_RCV.3 Automated Recovery Without Undue Loss

2.8.6.2.1 FPT_RCV.3.1

When automated recovery from a failure or service discontinuity is not possible, the TSF shall enter a maintenance mode where the ability to return the TOE to a secure state is provided.

2.8.6.2.2 FPT_RCV.3.2

For **any system crash or shutdown** the TSF shall ensure the return of the TOE to a secure state using automated procedures.

2.8.6.2.3 FPT_RCV.3.3

The functions provided by the TSF to recover from failure or service discontinuity shall ensure that the secure initial state is restored without exceeding **any loss or unauthorized disclosure** of TSF data, **authentication information, and/or** objects within the TSC.

2.8.6.2.4 FPT_RCV.3.4

The TSF shall provide the capability to determine the objects that were or were not capable of being recovered.

2.8.6.3 FPT_RCV.4 Function recovery

2.8.6.3.1 FPT_RCV.4.1

The TSF shall ensure that **any security function, such as the audit log, that encounters a failure, of size limit exceeded**, have the property that the SF either completes successfully, or for the indicated failure scenarios, recovers to a consistent and secure state.

2.8.7 REPLAY DETECTION (FPT_RPL)

2.8.7.1 FPT_RPL.1 Replay Detection

2.8.7.1.1 FPT_RPL.1.1

The TSF shall detect replay for the following entities:

- **Transmitted authentication information**
- **Authentic messages**

2.8.7.1.2 FPT_RPL.1.2

The TSF shall **have the ability to generate an alarm and/or send e-mail notification to the authorized administrators** when replay is detected.

2.8.8 REFERENCE MEDIATION (FPT_RVM)

2.8.8.1 FPT_RVM.1 Non-bypassability of the TSP

2.8.8.1.1 FPT_RVM.1.1

The TSF shall ensure that TSP enforcement functions are invoked and succeed before each function within the TSC is allowed to proceed.

2.8.9 TIME STAMPS (FPT_STM)

2.8.9.1 FPT_STM.1 Reliable Time Stamps

2.8.9.1.1 FPT_STM.1.1

The TSF shall be able to provide reliable time stamps for its own use.

2.8.10 TSF SELF-TEST (FPT_TST)

2.8.10.1 FPT_TST.1 TSF Testing

2.8.10.1.1 FPT_TST.1.1

The TSF shall run a suite of self-tests periodically during normal operation **and** at the request of the authorized *administrators*, and at the conditions as deemed necessary to demonstrate the correct operation of the TSF.

2.8.10.1.2 FPT_TST.1.2

The TSF shall provide authorized *administrators* with the capability to verify the integrity of TSF data.

2.8.10.1.3 FPT_TST.1.3

The TSF shall provide authorized *administrators* with the capability to verify the integrity of stored TSF-executable code.

2.9 CLASS FTA: TOE ACCESS

2.9.1 LIMITATION ON MULTIPLE CONCURRENT SESSIONS (FTA_MCS)

2.9.1.1 FTA_MCS.1 Basic Limitation on Multiple Concurrent Sessions

2.9.1.1.1 FTA_MCS.1.1

The TSF shall restrict the maximum number of concurrent sessions that belong to the same user.

2.9.1.1.2 FTA_MCS.1.2

The TSF shall enforce, by default, a limit of **as specified by the authorized administrator** sessions per user.

2.9.2 SESSION LOCKING (FTA_SSL)

2.9.2.1 FTA_SSL.3 TSF-Initiated Termination

2.9.2.1.1 FTA_SSL.3.1

The TSF shall terminate an interactive session after **an authorized administrator-specified period of time**.

2.9.3 TOE ACCESS BANNERS (FTA_TAB)

2.9.3.1 FTA_TAB.1 Default TOE Access Banners

2.9.3.1.1 FTA_TAB.1.1

Before establishing a user session, the TSF shall display an advisory warning message regarding unauthorized use of the TOE.

2.9.4 TOE ACCESS HISTORY (FTA_TAH)

2.9.4.1 FTA_TAH.1 TOE Access History

2.9.4.1.1 FTA_TAH.1.1

Upon successful session establishment, the TSF shall display the date, time, and location of the last successful session establishment to the user.

2.9.4.1.2 FTA_TAH.1.2

Upon successful session establishment, the TSF shall display the date, time, and location of the last unsuccessful attempt at session establishment and the number of unsuccessful attempts since the last successful session establishment.

2.9.4.1.3 FTA_TAH.1.3

The TSF shall not erase the access history information from the user interface without giving the user an opportunity to review the information.

2.9.5 TOE SESSION ESTABLISHMENT (FTA_TSE)

2.9.5.1 FTA_TSE.1 TOE Session Establishment

2.9.5.1.1 FTA_TSE.1.1

The TSF shall be able to deny session establishment based on:

- **Time of day**
- **Day of week**
- **Calendar date of login**
- **Source of connection**
- **User access rights**
- **As deemed necessary by an authorized administrator**

2.10 CLASS FTP: TRUSTED PATH/CHANNELS

2.10.1 INTER-TSF TRUSTED CHANNEL (FTP_ITC)

2.10.1.1 FTP_ITC.1 Inter-TSF Trusted Channel

2.10.1.1.1 FTP_ITC.1.1

The TSF shall provide a communication channel between itself and a remote, trusted IT product that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.

2.10.1.1.2 FTP_ITC.1.2

The TSF shall permit the TSF to initiate communication via the trusted channel.

2.10.1.1.3 FTP_ITC.1.3

The TSF shall initiate communication via the trusted channel for **the validity of all traffic and transmissions.**

2.10.2 TRUSTED PATH (FTP_TRP)

2.10.2.1 FTP_TRP.1 Trusted Path

2.10.2.1.1 FTP_TRP.1.1

The TSF shall provide a communication path between itself, remote, **and** local users that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from modification or disclosure.

2.10.2.1.2 FTP_TRP.1.2

The TSF shall permit the TSF to initiate communication via the trusted path.

2.10.2.1.3 FTP_TRP.1.3

The TSF shall require the use of the trusted path for initial user authentication, **user-defined information, and all security-related information.**

3. APPLICATION PRODUCT CLASS REQUIREMENTS

There are no additional security functional requirements relative to application products beyond the general requirements listed in Section 2. Master Level Requirements.

4. ASSURANCE REQUIREMENTS

To establish the appropriate assurance level for the product to be evaluated against, the author of the Protection Profile (PP) and/or the author of the Security Target (ST) have to take into account:

- The product's physical environment that identifies all aspects of the product's operating environment relevant to the product's security, including known physical and personnel security arrangements
- The assets requiring protection by the element of the product to which security requirements or policies will apply; this may include assets that are directly referred to, such as files and databases, as well as assets that are indirectly subject to security requirements, such as authorization credentials and the IT implementation itself
- The product's purpose that would address the product type and the intended usage of the product.

Furthermore, the author of the PP and/or ST would have to take into account any assumptions, threats, and organizational security policies.

Assumptions are to be met by the environment of the product in order for the product to be considered secure. An example of an assumption is that the product is physically secure and system administrators are not careless, willfully negligent, or hostile, and will follow and abide by the instructions provided by the administrator documentation. The threats identify the threats to security of the assets. The threats would identify all threats perceived by the security analysis as relevant to the product. An assessment of risks to security would qualify each threat with an assessment of the likelihood of such a threat developing into an actual attack, the likelihood of such an attack proving successful, and the consequences of any damage that may result. An example of a threat is an unauthorized user may gain access to system data due to failure of the system to restrict access.

Organizational security policies are policies that would identify relevant policies and rules.

The CC has established assurance levels from EAL1 to EAL7. Following is a brief overview of the requirements for each EAL.

EAL1 provides an evaluation of the product as made available to the customer, including independent testing against a specification, and an examination of the guidance documentation provided.

EAL2 provides assurance by an analysis of the security functions, using a functional and interface specification, guidance documentation, and the high-level design of the product, to understand the security behavior. The analysis is supported by independent testing of the TOE security functions, evidence of developer testing based on the functional specification, selective independent confirmation of the developer test results, strength of function analysis, and evidence of a developer search for obvious vulnerabilities (e.g. those in the public domain). EAL2 also provides assurance through a configuration list for the product, and evidence of secure delivery procedures. This EAL represents a meaningful increase in assurance from EAL1 by requiring developer testing, a vulnerability analysis, and independent testing based upon more detailed product specifications.

EAL3 provides assurance by an analysis of the security functions, using a functional and interface specification, guidance documentation, and the high-level design of the product, to understand the security behavior. The analysis is supported by independent testing of the product 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 (e.g. those in the public domain). EAL3 also provides assurance through the use of development environment controls, product configuration management, and evidence of secure delivery procedures. This EAL represents a meaningful increase in assurance from EAL2 by requiring more complete testing coverage of the security functions and mechanisms and/or procedures that provide some confidence that the product will not be tampered with during development.

EAL4 provides assurance by an analysis of the security functions, using a functional and complete interface specification, guidance documentation, the high-level and low-level design of the product, and a subset of the implementation, to understand the security behavior. Assurance is additionally gained through an informal model of the product security policy. The analysis is supported by independent testing of the product 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, evidence of a developer search for vulnerabilities, and an independent vulnerability analysis demonstrating resistance to penetration attackers with a low attack potential. EAL4 also provides assurance through the use of development environment controls and additional product configuration management including automation, and evidence of secure delivery procedures. This EAL represents a meaningful increase in assurance from EAL3 by requiring more design description, a subset of the implementation, and improved mechanisms and/or procedures that provide confidence that the product will not be tampered with during development or delivery.

EAL5 provides assurance by an analysis of the security functions, using a functional and complete interface specification, guidance documentation, high-level and low-level design of the product, and all of the implementation, to understand the security behavior. Assurance is additionally gained through a formal model of the product security policy

and a semiformal presentation of the functional specification and high-level design and a semiformal demonstration of correspondence between them. A modular product design is also required. The analysis is supported by independent testing of the product security functions, evidence of developer testing based on the functional specification, high-level design and low-level design, selective independent confirmation of the developer test results, strength of function analysis, evidence of a developer search for vulnerabilities, and an independent vulnerability analysis demonstrating resistance to penetration attackers with a moderate attack potential. The analysis also includes validation of the developer's covert channel analysis. EAL5 also provides assurance through the use of a development environment controls, and comprehensive product configuration management including automation, and evidence of secure delivery procedures. This EAL represents a meaningful increase in assurance from EAL4 by requiring semiformal design descriptions, the entire implementation, a more structured (and hence analyzable) architecture, covert channel analysis, and improved mechanisms and/or procedures that provide confidence that the product will not be tampered with during development.

EAL6 provides assurance by an analysis of the security functions, using a functional and complete interface specification, guidance documentation, the high-level and low-level design of the of the product, and a structured presentation of the implementation, to understand the security behavior. Assurance is additionally gained through a formal model of the product security policy, a semiformal presentation of the functional specification, high-level design, and low-level design and a semiformal demonstration of correspondence between them. A modular and layered product design is also required. The analysis is supported by independent testing of the product security functions, evidence of developer testing based on the functional specification, high-level design and low-level design, selective independent confirmation of the developer test results, strength of function analysis, evidence of a developer search for vulnerabilities, and an independent vulnerability analysis demonstrating resistance to penetration attackers with a high attack potential. The analysis also includes validation of the developer's systematic covert channel analysis. EAL6 also provides assurance through the use of a structured development process, development environment controls, and comprehensive product configuration management including complete automation, and evidence of secure delivery procedures. This EAL represents a meaningful increase in assurance from EAL5 by requiring more comprehensive analysis, a structured representation of the implementation, more architectural structure (e.g. layering), more comprehensive independent vulnerability analysis, systematic covert channel identification, and improved configuration management and development environment controls.

EAL7 provides assurance by an analysis of the security functions, using a functional and complete interface specification, guidance documentation, the high-level and low-level design of the product, and a structured presentation of the implementation, to understand the security behavior. Assurance is additionally gained through a formal model of the product security policy, a formal presentation of the functional specification and high-level design, a semiformal presentation of the low-level design, and formal and semiformal demonstration of correspondence between them, as appropriate. A modular, layered and simple product design is also required. The analysis is supported by

independent testing of the product security functions, evidence of developer testing based on the functional specification high-level design, low-level design and implementation representation, complete independent confirmation of the developer test results, strength of function analysis, evidence of a developer search for vulnerabilities, and an independent vulnerability analysis demonstrating resistance to penetration attackers with a high attack potential. The analysis also includes validation of the developer's systematic covert channel analysis. EAL7 also provides assurance through the use of a structured development process, development environment controls, and comprehensive product configuration management including complete automation, and evidence of secure delivery procedures. This EAL represents a meaningful increase in assurance from EAL6 by requiring more comprehensive analysis using formal representations and formal correspondence, and comprehensive testing.

5. PACKAGE APPLICATION NOTES

This chapter describes the additional supporting information.

A TOE that requires some of the IT security requirements to be met by the TOE IT environment is permissible. For Security Functional Requirements (SFR) that are attributed to the environment are therefore a SFR of the environment and not of the TOE. Should this be the case, the Security Target author must explain the partition of security requirements between the TOE and its IT environment and demonstrate that the TOE in its IT environment satisfies all of the security requirements and is compliant with the Common Criteria and the ***BITS Tested Mark***.

The method of identification is not specified in this Package, but should be specified in the ST and it should specify how this relates to user identifiers maintained by the TSF. In addition, the method of authentication is not specified in this Package, but should be specified in the ST. The method that is used must show a low probability that authentication data cannot be forged or guessed. Furthermore, a Security Target wishing to claim conformance with this Package must state which authentication package is being implemented, externally by the underlying operating system or within the TOE itself.

Furthermore, the TOE is required to implement a DAC policy. The rules that govern the policy must be specified in the ST. The mechanism must be able to specify access rules that apply to a single user and membership of at least a single group. The ST must also list the attributes that are used by the DAC policy for access decisions. These attributes may include permission bits, access control list, and object ownership.

6. APPENDIX A – INDUSTRY STANDARDS

For the purposes of these criteria the security functional requirements, the terms “public and widely used” and “financial industry standards” shall refer to those standards, algorithms, and protocols listed below as well as other relevant standards approved by the following standards organizations: IETF, ANSI X9, ITU-T, ISO, NIST, and IEEE.

Symmetric encryption algorithms	<ul style="list-style-type: none"> • 3DES (ANS X9.52, X9.66) • IDEA • RC4 • RC5 • RIPEM
Asymmetric algorithms (for symmetric key agreement or key transport)	<ul style="list-style-type: none"> • RSA (ANS X9.44) • D-H (minimum 1024-bit modulus – ANSI X9.42) • ECDH (ANS X9.63) • Elliptic Curve
Digital hashing algorithms	<ul style="list-style-type: none"> • SHA-1 (ANS X9.30-2) • MD5
Digital signature algorithms	<ul style="list-style-type: none"> • DSA (ANS X9.30-1) • rDSA (ANS X9.31) (includes RSA) • EC-DSA (ANS X9.62)
Key management standards and protocols	<ul style="list-style-type: none"> • ANS X9.70, ANS X9.73, ANS X9.69, ANS X9.24, ANS X9.77 • CMP • PKCS #7, #10 • IETF PKIX standards
Random number generators	<ul style="list-style-type: none"> • ANS X9.82
Prime number generators	<ul style="list-style-type: none"> • ANSI X9.80
Cryptographic device security	<ul style="list-style-type: none"> • ANS X9.66 • FIPS 140-2
Peer entity authentication	<ul style="list-style-type: none"> • ANS X9.72 • FIPS 196
PIN security	<ul style="list-style-type: none"> • ANS X9.8, ANS X9.86, ANS X9.87
Biometrics management and security	<ul style="list-style-type: none"> • ANS X9.84
Directory standards	<ul style="list-style-type: none"> • X.500 • LDAP v3
TCP/IP integrity	<ul style="list-style-type: none"> • IPsec

The system product shall use any of the algorithms listed above or those that are supported by any of the standards organizations listed above. If the system uses any other cryptographic algorithm, then it shall be configurable to also allow the use of an acceptable algorithm in place of the unlisted algorithm.

7. APPENDIX B - GLOSSARY

Assets — Information or resources to be protected by the countermeasures of a TOE.

Assignment — The specification of an identified parameter in a component.

Assurance — Grounds for confidence that an entity meets its security objectives.

Augmentation — The addition of one or more assurance component(s) from Part 3 to an EAL or assurance package.

Authentication Data — Information used to verify the claimed identity of a user.

Authorized User — A user who may, in accordance with the TSP, perform an operation.

Class — A grouping of families that share a common focus.

Common Criteria Protection Profile (CC-PP) — a Protection Profile as defined in Part 1 of the CC. For Protection Profile definition, refer to www.commoncriteria.org

Component — The smallest selectable set of elements that may be included in a PP, an ST, or a package.

Dependency — A relationship between requirements such that the requirement that is depended upon must normally be satisfied for the other requirements to be able to meet their objectives.

Element — An indivisible security requirement.

Evaluation — Assessment of a PP, an ST, or a TOE against defined criteria.

Evaluation Assurance Level (EAL) — A package consisting of assurance components from Part 3 of CC that represents a point on the CC predefined assurance scale.

Evaluation Scheme — The administrative and regulatory framework under which the CC is applied by an evaluation authority within a specific community.

Extension — The addition to an ST or PP of functional requirements not contained in Part 2 and/or assurance requirements not contained in Part 3 of the CC.

Family — A grouping of components that share security objectives but may differ in emphasis or rigor.

Internal Communication Channel — A communication channel between separated parts of TOE.

Internal TOE Transfer — Communicating data between separated parts of the TOE.

Inter-TSF Transfers — Communicating data between the TOE and the security functions of other trusted IT products.

Iteration — The use of a component more than once with varying operations.

Object — An entity within the TSC that contains or receives information and upon which subjects perform operations.

Organizational Security Policies — One or more security rules, procedures, practices, or guidelines imposed by an organization upon its operations.

Package — A reusable set of either functional or assurance components (e.g., an EAL) combined together to satisfy a set of identified security objectives.

Product — A package of IT software, firmware, and/or hardware, providing functionality designed for use or incorporation within a multiplicity of systems.

Product Class — the name typically used to describe a specific IT product (e.g., biometric authentication device, firewall, or smart card).

Protection Profile (PP) — An implementation-independent set of security requirements for a category of TOEs that meet specific consumer needs.

PP Evaluation — An evaluation of a CC-PP according to the requirements identified in Part 1 of the CC and the Common Evaluation Methodology.

Reference Monitor — The concept of an abstract machine that enforces TOE access control policies.

Reference Validation Mechanism — An implementation of the reference monitor concept that possesses the following properties: it is tamperproof, always invoked, and simple enough to be subjected to thorough analysis and testing.

Refinement — The addition of details to a component.

Role — A predefined set of rules establishing the allowed interactions between a user and the TOE.

Security Assurance Requirements (SARs) — Assurances associated with Part 3 of the CC; often grouped in a package called an Evaluation Assurance Level (EAL), e.g., EAL2, EAL – Medium Robustness.

Secret — Information that must be known only to authorized users and/or the TSF in order to enforce a specific SFP.

Security Attribute — Information associated with subjects, users, and/or objects that is used for the enforcement of the TSP.

Security Function (SF) — A part or parts of the TOE that have to be relied upon for enforcing a closely related subset of the rules from the TSP.

Security Function Policy (SFP) — The security policy enforced by an SF.

Security Functional Requirements (SFRs) — Security functions drawn from Part 2 of the CC.

Security Objective — A statement of intent to counter identified threats and/or satisfy identified organization security policies and assumptions.

Security Target (ST) — A set of security requirements and specifications to be used as the basis for evaluation of an identified TOE.

Selection — The specification of one or more items from a list in a component.

Strength of Function (SOF) — A qualification of a TOE security function expressing the minimum efforts assumed necessary to defeat its expected security behavior by directly attacking its underlying security mechanisms.

SOF-basic — A level of the TOE strength of function where analysis shows that the function provides adequate protection against casual breach of TOE security by attackers possessing a low attack potential.

SOF-medium — A level of the TOE strength of function where analysis shows that the function provides adequate protection against straightforward or intentional breach of TOE security by attackers possessing a moderate attack potential.

SOF-high — A level of the TOE strength of function where analysis shows that the function provides adequate protection against deliberately planned or organized breach of TOE security by attackers possessing a high attack potential.

Subject — An entity within the TSC that causes operations to be performed.

System — A specific IT installation with a particular purpose and operational environment.

Target of Evaluation (TOE) — An IT product or system and its associated administrator and user guidance documentation that is the subject of an evaluation.

TOE Security Functions (TSF) — A set consisting of all hardware, software, and firmware of the TOE that must be relied upon for the correct enforcement of the TSP.

TOE Security Functions Interface (TSFI) — A set of interfaces, whether interactive (man-machine interface) or programmatic (application programming interface), through which TOE resources are accessed, mediated by the TSF, or information is obtained from the TSF.

TOE Security Policy (TSP) — A set of rules that regulate how assets are managed, protected, and distributed within a TOE.

TOE Security Policy Model — A structured representation of the security policy to be enforced by the TOE.

Transfers Outside TSF Control — Communicating data to entities not under control of the TSF.

Trusted Channel — A means by which a TSF and a remote, trusted IT product can communicate with necessary confidence to support the TSP.

Trusted Path — A means by which a user and a TSF can communicate with necessary confidence to support the TSP.

TSF Data — Data created by and for the TOE, which might affect the operation of the TOE.

TSF Scope of Control (TSC) — The set of interactions that can occur with or within a TOE and are subject to the rules of the TSP.

User — Any entity (human user or external IT entity) outside the TOE that interacts with the TOE.

User Data — Data created by and for the user, which does not affect the operation of the TSF.

8. APPENDIX C – BITS PRODUCT CERTIFICATION PROGRAM OVERVIEW

BITS, the Technology Group for The Financial Services Roundtable, was created in 1996 to foster the growth and development of secure e-commerce in an open environment for the benefit of financial institutions and their customers. BITS serves as a business and technology strategy consortium for approximately 100 of the largest financial services

companies in the US. BITS is governed by a Board of Directors comprised of the Chairmen and CEOs of the member institutions, as well as representatives of the American Bankers Association (ABA) and the Independent Community Bankers of America (ICBA).

A hallmark focus for BITS is information security. The mission of the BITS Security and Risk Assessment (SRA) Committee is to strengthen the safety and soundness of financial institutions through enhancing e-commerce and payments security, sharing knowledge of successful strategies for development of secure infrastructures, products and services; and working with government agencies and regulators supervisory guidance and regulations. In recognition of the important role of security as a fundamental building block for all aspects of information technology use, the senior officers responsible for information security at the nation's leading financial services firms that comprise the BITS SRA Committee, developed the concept of the **BITS Product Certification Program** and the **BITS Tested Mark**.

The **BITS Product Certification Program** was designed to assure that software products contain the necessary security features to serve the financial services industry, to provide an opportunity to leverage independent testing efforts, to provide tangible evidence of implementing a best practice within the industry. Minimum baseline security criteria for six categories of commercial software products was established through a collaborative, three-year development effort, involving the work of 32 BITS financial services member companies, 23 outside organizations and over 100 security professionals from technology vendors, government and financial regulatory agencies and leading financial services firms.

In recognition of the common goals of the **BITS Product Certification Program** and the internationally recognized Common Criteria Certification, the financial services industry represented by BITS became one of the first private-sector "user communities" to use Common Criteria to define product security requirements. With the help of SAIC, BITS translated its "plain English" security criteria profile documents into packages of security requirements under the Common Criteria schema. Technology vendors are now able to test against the BITS criteria through an independent testing facility recognized by BITS or through their Common Criteria testing efforts at a Common Criteria Testing Lab.

While the criteria were created for testing purposes, many financial institutions have adopted the testing criteria as internal standards for product development. Financial institutions are including language about the **BITS Tested Mark** in their procurement policies, RFPs and vendor contracts. BITS members are committed to making the **BITS Tested Mark** a major part of their technology purchasing process

Certification helps build confidence in software products, leads to more widespread use of technology, and promotes the growth of e-commerce. The BITS Certification process is an objective means of evaluating and testing for compliance with industry-set minimum security criteria. It minimizes testing redundancy, can make product testing more efficient and reduces the cost and time-to-market industry wide. Certification with a **BITS Tested Mark** will not only be a key product differentiator but also a move towards aligning with current and proposed financial regulation and cyber-security

legislation during a time of evolving liability issues. The ***BITS Tested Mark*** demonstrates to customers that your company is committed to addressing security issues and the security needs of the financial services industry.

After achieving a ***BITS Tested Mark***, BITS will promote the certified products within the financial services industry through activities such as:

- Issuing joint press release and arranging joint press interviews,
- Including an announcement and article in the BITS Bulletin (bi-monthly newsletter with a 5000+ readership), the BITS Brief (monthly update to the BITS member company's CIOs/CTOs) and in a member wide e-mail communication,
- Post the announcement on the BITS Web site and add the product to the Certified Product List,
- Mention the accomplishment during conferences, seminars and other industry presentations,

In addition, vendors are free to use the ***BITS Tested Mark*** in accordance to the terms of the Seal Usage Agreement. Uses can include:

- Financial services industry sales proposals
- Advertising campaigns
- Product package design
- Vendor Web site posting
- Product marketing collateral / company brochure
- Display for trade show exhibit booth

For more information about the BITS Product Certification Program including the program operating rules and certification seal use terms and conditions please visit the BITS website at www.bitsinfo.org/fslab.html

BITS

FINANCIAL SERVICES
ROUNDTABLE

THE FINANCIAL SERVICES ROUNDTABLE



FSR/BITS MEMBER COMPANIES

COMPANY	CITY
ABN-AMRO North America, Inc.	Chicago
AEGON USA, Inc.	Baltimore
Allfirst Financial, Inc.	Baltimore
Allied Capital Corporation	Washington, DC
AMCORE Financial, Inc.	Rockford
American General	Houston
AmSouth Bancorporation	Birmingham
Aon Corporation	Chicago
Associated Banc-Corp	Green Bay
AXA Financial Inc.	New York
BancorpSouth, Inc.	Tupelo
BancWest Corporation	Honolulu
Bank of America Corporation	Charlotte
Bank of New York Company, Inc., The	New York
Bank of Tokyo-Mitsubishi Trust Company	New York
BANK ONE CORPORATION	Chicago
BB&T Corporation	Winston-Salem
Capital One Financial Corporation	Falls Church
Charles Schwab Corporation, The	San Francisco
Charter One Financial, Inc.	Cleveland
Chubb Corporation, The	Warren
Citigroup Inc.	New York
Citizens Financial Group, Inc.	Providence
City National Corporation	Beverly Hills
Comerica Incorporated	Detroit
Commerce Bancshares, Inc.	Kansas City
Compass Bancshares, Inc.	Birmingham
Countrywide Credit Industries, Inc.	Calabasas
Credit Suisse First Boston	New York
Cullen/Frost Bankers, Inc.	San Antonio
Edward Jones Investments	St. Louis
F.N.B. Corporation	Naples
FMR Corp. (Fidelity Investments)	Boston
Fifth Third Bancorp	Cincinnati
First Commonwealth Financial Corporation	Indiana
First National of Nebraska, Inc.	Omaha
First Tennessee National Corporation	Memphis

COMPANY	CITY
First Virginia Banks, Inc.	Falls Church
FleetBoston Financial Corporation	Boston
Ford Financial	Dearborn
Fortis, Inc./Assurant Group	New York/Atlanta
Fulton Financial Corporation	Lancaster
General Motors Acceptance Corporation	Detroit
Goldman Sachs Group, Inc., The	New York
Guaranty Financial Services	Austin
Harris Bankcorp, Inc.	Chicago
Hartford Financial Services Group, Inc., The	Hartford
Hibernia Corporation	New Orleans
Household International, Inc.	Prospect Heights
HSBC USA Inc.	New York
Hudson United Bancorp	Mahwah
Huntington Bancshares Incorporated	Columbus
ING Americas	Atlanta
Jefferson-Pilot Corporation	Greensboro
J.P. Morgan Chase & Co.	New York
KeyCorp	Cleveland
Legg Mason, Inc.	Baltimore
M&T Bank Corporation	Buffalo
Marshall & Ilsley Corporation	Milwaukee
MassMutual Financial Group	Springfield
MBNA Corporation	Wilmington
Mellon Financial Corporation	Pittsburgh
Mercantile Bankshares Corporation	Baltimore
Merrill Lynch & Co., Inc.	New York
Minnesota Mutual	St. Paul
National City Corporation	Cleveland
National Commerce Financial Corporation	Memphis
Nationwide	Columbus
Northern Trust Corporation	Chicago
Old National Bancorp	Evansville
Pacific Century Financial Corporation	Honolulu
PNC Financial Services Group, Inc., The	Pittsburgh
Provident Bankshares Corporation	Baltimore
Provident Financial Group, Inc.	Cincinnati
Providian Financial Corporation	San Francisco
Prudential Insurance Company of America, The	Newark
Raymond James Financial, Inc.	St. Petersburg
RBC Centura Banks, Inc.	Rocky Mount
Regions Financial Corporation	Birmingham
Riggs National Corporation	Washington, D.C.
Sky Financial Group, Inc.	Bowling Green

COMPANY	CITY
St. Paul Companies, Inc., The	St. Paul
State Farm Insurance Companies	Bloomington
State Street	Boston
SunTrust Banks, Inc.	Atlanta
Synovus	Columbus
UBS Warburg LLC	Stamford
Union Planters Corporation	Memphis
U.S. Bancorp	Minneapolis
United Bankshares, Inc.	Parkersburg
USAA	San Antonio
Wachovia Corporation	Charlotte
Waddell & Reed Financial, Inc.	Overland Park
Washington Mutual, Inc.	Seattle
Wells Fargo & Company	San Francisco
Whitney Holding Corporation	New Orleans
Zions Bancorporation	Salt Lake City
Zurich North America	Schaumburg

BITS ONLY MEMBER COMPANIES

COMPANY	CITY
SouthTrust Bank	Birmingham

BITS AFFILIATE ORGANIZATIONS

ORGANIZATION	CITY
American Bankers Association (ABA)	Washington, D.C.
America's Community Bankers (ACB)	Washington, D.C.
Association for Payment Clearing Services (APACS)	London
Canadian Bankers Association (CBA)	Toronto
Canadian Payments Association (CPA)	Ottawa
CUNA	Washington, D.C.
ECCHO	Dallas
Independent Community Bankers of America (ICBA)	Washington, D.C.
NACHA	Herndon
Spectrum EBP, L.L.C.	Union
VISA USA	San Francisco

BITS STRATEGIC ALLIANCES

ORGANIZATION	CITY
US Department of the Navy	Reston
Financial Services Technology Consortium (FSTC)	Chicago