# IBM WebSphere Business Integration Message Broker Security Target

Version 1.0

November 21, 2005

**Prepared for:**

IBM UK Labs Limited
Mail Point 208
Hursley Park
Winchester
Hampshire
SO21 2JN
United Kingdom

**Prepared By:**


**Common Criteria Testing Laboratory**

7125 Columbia Gateway Drive, Suite 300
Columbia, MD 21046

**TABLE OF CONTENTS**

## LIST OF FIGURES



## LIST OF TABLES

# 1 Security Target Introduction

This section identifies the Security Target (ST) and Target of Evaluation (TOE) identification, ST conventions, ST conformance claims, and the ST organization. The TOE is WBI Message Broker provided by IBM. WBI Message Broker enables information, packaged as messages to flow between different business applications, ranging from large legacy systems through to unmanned devices such as sensors on pipelines.

The Security Target contains the following additional sections:

- TOE Description (Section 2)
- Security Environment (Section 3)
- Security Objectives (Section 4)
- IT Security Requirements (Section 5)
- TOE Summary Specification (Section 6)
- Protection Profile Claims (Section 7), and
- Rationale (Section 8).

## 1.1 Security Target, TOE and CC Identification

**ST Title –** IBM WebSphere Business Integration Message Broker Security Target

**ST Version** – Version 1.0

**ST Date** – 11/21/05

**TOE Identification** – IBM WebSphere Business Integration Message Broker, Version 5.0, Fix Pack 4

**EAL** – Evaluation Assurance Level (EAL) 3 augmented with ALC_FLR.2

**CC Identification** – Common Criteria for Information Technology Security Evaluation, Version 2.2, Revision 256, January 2004.

## 1.2 Conformance Claims

This TOE is conformant to the following CC specifications:

- Common Criteria for Information Technology Security Evaluation Part 2: Security Functional Requirements, Version 2.2, Revision 256, January 2004.
    - Part 2 Conformant
- Common Criteria for Information Technology Security Evaluation Part 3: Security Assurance Requirements, Version 2.2, Revision 256, January 2004.
    - Part 3 Conformant
    - EAL 3 augmented with ALC_FLR.2

## 1.3 Conventions, Terminology, Acronyms

This section specifies the formatting information used in the Security Target.

### *1.3.1 Conventions*

The following conventions have been applied in this document:

- Security Functional Requirements – Part 2 of the CC defines the approved set of operations that may be applied to functional requirements: iteration, assignment, selection, and refinement.
    - Iteration: allows a component to be used more than once with varying operations. In the ST, iteration is indicated by a letter in parenthesis placed at the end of the component. For example FDP_ACC.1a and FDP_ACC.1b indicate that the ST includes two iterations of the FDP_ACC.1 requirement, a and b.
    - Assignment: allows the specification of an identified parameter. Assignments are indicated using bold and are surrounded by brackets (e.g., [**assignment**]).
    - Selection: allows the specification of one or more elements from a list. Selections are indicated using bold italics and are surrounded by brackets (e.g., [***selection***]).
    - Refinement: allows the addition of details. Refinements are indicated using bold, for additions, and strike-through, for deletions (e.g., "… **all** objects …" or "… ~~some~~ **big** things …").
- Other sections of the ST – Other sections of the ST use bolding to highlight text of special interest, such as captions.

### *1.3.2 Terminology*

This section provides an explanation of each unique term used throughout this ST.

**Ancestor** A topic that exists at a higher level within a hierarchy or topic tree.

**Message** A string of bytes that is meaningful to the applications that use it. Messages are used to transfer information from one application program to another (or between different parts of the same application).

**Queue Manager** Responsible for maintaining the queues it owns and for storing all the messages it receives onto the appropriate queues.

### *1.3.3 Acronyms*

This section provides a definition of each acronym used throughout this ST.

**ACL** Access Control List

**CC** Common Criteria

**CM** Configuration Management

**EAL** Evaluation Assurance Level

**FIPS** Federal Information Processing Standard

**GUI** Graphical User Interface

**ID** Identification/Identifier

**IT** Information Technology

**MQ** Message Queue

**OS** Operating System

**SAR** Security Assurance Requirements

**SF** Security Function

**SFP** SF Policy

**SFR** Security Functional Requirements

**SOF** Strength of Function

| | |
|---|---|
| **SSL** | Secure Socket Layer |
| **ST** | Security Target |
| **TCP** | Transmission Control Protocol |
| **TOE** | Target of Evaluation |
| **TSC** | TSF Scope of Control |
| **TSF** | TOE Security Functions |
| **TSFI** | TSF Interface |
| **TSP** | TOE Security Policy |
| **TSS** | TOE Summary Specification |
| **UNS** | User Name Server |
| **WBI** | WebSphere Business Integration |
| **WMB** | WBI Message Broker |

# 2 TOE Description

The Target of Evaluation (TOE) is IBM WebSphere Business Integration Message Broker, Version 5.0, Fix Pack 4.

WBI Message Broker (WMB) enables information, packaged as messages to flow between different business applications, ranging from large legacy systems through to unmanned devices such as sensors on pipelines.

There are two ways in which WMB can act on messages:

In real-time mode, all communications between parts of the TOE are performed over a SSL session. The SSL session will protect the data transmitted from unauthorized modification or disclosure. In the non-real-time mode, communications between the client and Broker are via the MQ transport. Non-real time mode is included in the evaluation but the MQ transport is outside the scope of this evaluation (but encrypted nonetheless) [1].

WMB provides authentication services between client applications that use the WebSphere MQ Real-Time Transport. The message flow connection determines whether the real-time or the non-real-time MQ transport will be used.

1) **Message routing** from sender to recipient based on the content of the message where WMB can be configured for message routing via message flows that can be designed. A message flow describes the operations to be performed on the incoming message, and the sequence in which they are carried out. Each message flow consists of: A series of steps used to process a message, as defined in message flow nodes, and connections between the nodes that define the routes through the processing. Connections are made using message flow node connections. WMB provides built-in nodes and samples for numerous common functions. Additional functions can be built using a simple Graphical User Interface (GUI) to create user-defined nodes.
2) **Message transformation** before being delivered from one format to another by modifying, combining, adding or removing data fields, perhaps involving the use of information stored in a database provided by the environment where information can be mapped between messages and databases or by the use of Extended Structured Query Language (ESQL). Transformations can be made by various nodes in a message flow but before a message flow node can operate on an incoming message, it must understand the structure of that message such as: some messages contain a definition of their own structure and format known as *self-defining messages* that can be handled without the need for additional information about the

[1] If the non-real-time MQ transport is used, the IT environment provides message encryption; if the real-time transport is employed, the TOE provides encryption.

structure and format; and, other messages do not contain information about their structure and format. To process the later, a *message definition* of their structure must be created and made available. The message definitions defined are created within a *message set* which contains one or more message definitions. Like message flows, message definitions are built using GUI actions that contain two types of information: the *logical structure*—the abstract arrangement and characteristics of the data, represented as a tree structure; and, one or more *physical formats*—the way the data is represented and delimited in the physical bitstream.

## 2.1 TOE Overview

The TOE is comprised of the WMB components created by IBM. The TOE architecture consists of five subsystems functional components, which are placed at key points within the Enterprise architecture: Message Brokers Toolkit, Broker, Configuration Manager, User Name Server, and Application (client). Figure 1 below shows these components, their location, and interaction in the architecture.

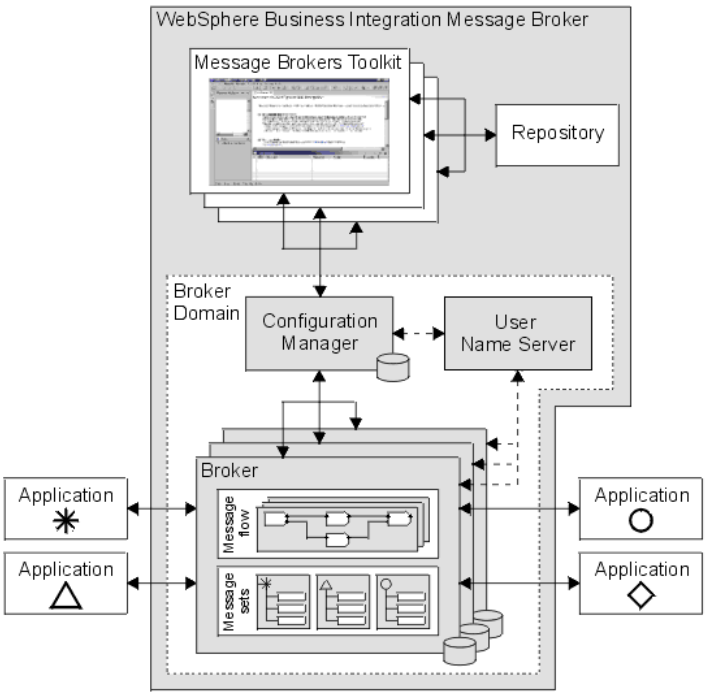


Figure 1 – WMB Components

## 2.2 TOE Architecture

Four of the five components that are provided by the TOE, run within a specific Broker Domain environment and work together to enforce the overall security policies. The fifth component, the Message Brokers Toolkit, provides the business integration repository for the WMB. The databases used by the TOE are outside the scope of the evaluation.

### *2.2.1 Physical Boundaries*

Each of the TOE components is a software application designed to execute within an operating system context provided by the environment. WMB is designed to operate on Windows 2000 (this includes all combinations of Advanced Server and Server with recommended Service Pack and hotfixes) or Windows Server 2003 (this includes all combinations of Standard and Enterprise with recommended Service Pack and hotfixes); AIX Version 5.1 (maintenance level 3) or AIX Version 5.2 (maintenance level 2); HP-UX, V11.11 (December 2002 Quality Pack); Sun Solaris 2.8 ( with the SunSolve recommended patch level); Red Hat Enterprise Linux AS 3.0 (for Linux Intel); SuSE Linux Enterprise Server (SLES) 8 (for Linux Intel). The TOE also requires a database that uses the ODBC protocol and WebSphere MQ.

#### 2.2.1.1 The Message Brokers Toolkit

The Message Brokers Toolkit is an integrated development environment and graphical user interface used for management. The Message Brokers Toolkit also communicates with one or more Configuration Managers, and is used to manage broker domains. In the evaluated configuration, the Message Brokers Toolkit must be installed on the same platform as the Configuration Manager.

When the Message Brokers Toolkit is started, a single window is displayed called the workbench window and it displays one or more perspectives. A perspective is a collection of views and editors that help users complete a specific task, or work with specific types of resource. The Toolkit is aware of the user identity that initiates the workbench window display. However, the Toolkit relies upon the IT environment to establish the identification and authentication for this user.

#### 2.2.1.2 The Broker

A broker is a system service on Windows platforms or a daemon process on Unix platforms that controls processes that run message flows.

Applications send messages to the broker using WebSphere Message Queue (MQ) queues and connections. The broker routes each message using the rules defined in message flows and message sets, and transforms the data into the structure required by the receiving application.

The broker uses sender and receiver channels to communicate with the Configuration Manager and other brokers in the broker domain.

The broker is created using command line instructions on the machine where the component is installed. Brokers can be installed and created on one or more machines.

The broker depends on a broker database to hold broker information that includes control data for resources defined to the broker (i.e., deployed message flows). A database is defined and authorized access for specific users is created before creating the Broker Administration since creating the broker creates tables within the database (i.e., also known as the broker's local persistent store).

The broker connects to the database using an Open DataBase Connectivity (ODBC) connection.

When creating the Broker Administration, a unique name within the broker domain must be identified. Broker names are case-sensitive on all supported platforms, except Windows platforms; and the same name must be used when creating a reference to the broker in the broker domain topology in the workbench. The reference to the broker is a representation of the physical broker in the configuration repository.

After creating the broker reference, changes are deployed to the broker domain. Deployment starts communications between the broker and the Configuration Manager. The broker receives configuration information from the Configuration Manager, and stores it in the configuration repository. Deployment also initializes the broker to make it ready to execute message flows.

When a broker is created, a set of database tables for storing the information used by the broker to process messages at runtime is also defined and created.

##### 2.2.1.2.1 Configuration Manager

The Configuration Manager is the runtime component that acts as an intermediary between the toolkit and the runtime broker domain. It is able to police which Windows users are able to perform actions within the domain. The Configuration Manager is only supported on Windows 2000.

In order to support both group level and user level security the following design is used by the Configuration Manager:

1) The object type, unique identifier and requested action are flowed in a WebSphere MQ message from the Toolkit to the Configuration Manager.

2) The Configuration Manager retrieves the object type, name and requested action from the MQ message and passes the information onto one of the Configuration Manager's internal components (the *RoleManager* class) where the logic to grant or deny the requested operation is performed.

3) The group membership for the supplied user is discovered from the Operating System (OS), and if the user is a member of the relevant broker (e.g., mq-type) group for the requested operation, the operation is granted.

4) If the user is not a member of the special groups, the Access Control List (ACL) table in the Configuration Manager database is queried to see whether, based on the permission set configured by the administrator, permission is to be granted.

#### 2.2.1.2.2 User Name Server

The User Name Server is a runtime component that provides authentication of users and groups performing publish/subscribe operations.

If the operating environment has applications that use the publish/subscribe services of a broker, an additional level of security can be applied to the topics on which messages are published and subscribed. This additional security, known as topic-based security, is managed by the User Name Server that provides administrative control over who can publish and who can subscribe. For example, if a client application publishes messages containing sensitive company finance information, or personnel details, the User Name Server can be used to restrict access to those messages, on a per topic basis.

The User Name Server interfaces with operating system facilities to provide information about valid users and groups in a broker domain.

The User Name Server can be installed, created and started in any supported operating environments.

The User Name Server can share a host queue manager (located in the environment) with the Configuration Manager and one broker in the broker domain. To communicate with other brokers in the broker domain, the User Name Server requires sender and receiver channels.

#### 2.2.1.2.3 Applications (Clients)

To provide data protection applications send messages to the broker using WebSphere MQ queues and connections. The broker routes each message using the rules defined in message flows and message sets, and transforms the data into the structure required by the receiving application.

### *2.2.2 Logical Boundaries*

The TOE logically supports the following security functions at its interfaces:

- Security audit
- Communication
- User data protection
- Identification and authentication
- Security Management, and
- Protection of the TSF.

#### 2.2.2.1 Authentication

WMB provides authentication services between client applications that use the WebSphere MQ Real-Time Transport and WebSphere MQ Message Broker Real-Time Input and Real-timeOptimizedFlow nodes. The WMB Message Broker identification and authentication services verify that a broker and a client application are who they claim they are, and can participate in a publish/subscribe session; where each participant uses an authentication protocol to prove to the other that they are who they say they are and are not an intruder impersonating a valid participant.

The WMB supports the following three protocols:

- M – mutual challenge-response password authentication;
- S – asymmetric Secure Socket Layer (SSL); and,
- R – symmetric SSL.

#### 2.2.2.2 Communication

WMB provides the ability to verify the sender and receiver of messages. Support for the authenticity of the sender and/or receiver is proved through the use of SSL.

#### 2.2.2.3 Access Control

WMB uses topic-based security to control which applications in the environments publish/subscribe system can access information on which topics. For each topic for restricted access, the principals (i.e., groups of user IDs) can be specified where this information can be published to determine which principals can subscribe to a given topic. Principals can also be specified on persistent messages (i.e., stored messages).

WMB also has an access policy to control who can create the topology of the domain. Like the topic-based access policy, access decisions are based on groups IDs.

#### 2.2.2.4 Security Audit

WMB performs security auditing for all authentication attempts made to the TOE. Audit records are generated when audit events occur, including the responsible user, date, time, and other details. The audit data is recorded into the operating system for protection.

#### 2.2.2.5 Security Management

WMB provides security management functionality for the management of the access control policies. Management is performed from the Broker Toolkit and the command line.

#### 2.2.2.6 Protection of the TSF

WMB protects itself and ensures that its policies are enforced in a number of ways. First, WMB interacts with users through well-defined interfaces designed to ensure that the WMB security policies are always enforced. Next, WMB encrypts all communications between physically separate parts of the TOE to ensure that no data is disclosed or modified.

## 2.3 TOE Documentation

IBM offers a series of documents that describe the installation process for WMB as well as guidance for subsequent use and administration of the applicable security features. Refer to Section 6 for information about these and other documents associated with WMB.

# 3 Security Environment

The security environment for the functions addressed by this specification includes threats and assumptions, as discussed below.

## 3.1 Threats

| | |
|---|---|
| T.ADMIN_ERROR | An authorized administrator may incorrectly install or configure the TOE resulting in ineffective security mechanisms. |
| T.MESSAGE_DENIAL | The sender or receiver of a message denies sending or receiving the message to avoid accountability for subsequent action or inaction |
| T.NETWORK | Data transferred between workstations is disclosed to or modified by unidentified users or processes monitoring network traffic. |
| T.ROLE | A non-privileged user may gain administrative privileges and bypasses the security policy. |
| T.TSF_COMPROMISE | A malicious user or process may cause configuration data to be inappropriately accessed (viewed, modified or deleted) by changing access to the configuration data. |
| T.UNAUTH_ACCESS | A user may gain unauthorized access (view, modify, delete) to user data by bypassing access controls. |
| T.UNDETECTED_ACTIONS | Failure of the IT system to detect and record unauthorized actions may occur. |

## 3.2 Assumptions

| | |
|---|---|
| A.INSTALL_REQ | The Message Brokers Toolkit must be installed on the same platform as the Configuration Manager because the two components do not encrypt traffic when communicating via a network. |
| A.NO_EVIL | Authorized administrators are non-hostile, appropriately trained and follow all administrator guidance. |
| A.PHYSICAL | It is assumed that appropriate physical security is provided within the domain for the value of the IT assets protected by the TOE and the value of the stored, processed, and transmitted information. |
| A.PLATFORM | The IT Environment underlying the TOE is assumed to fulfill the requirements for the IT Environment described in this ST. It is also assumed that the IT Environment will provide a suitable operational environment for the TOE where the TOE will be able to properly execute and the dependencies that the TOE has upon the IT Environment are properly fulfilled. |

# 4 Security Objectives

This section defines the security objectives for the TOE and its environment. These objectives are suitable to counter all identified threats and cover all identified assumptions. The TOE security objectives are identified with 'O.'

inserted at the beginning of the name and the environment objectives are identified with 'OE.' inserted at the beginning of the name.

## 4.1 Security Objectives for the TOE

| | |
|---|---|
| O.ACCESS | The TOE will ensure that users gain only authorized access to it and to the resources that it controls. |
| O.ADMIN_ROLE | The TOE will provide authorized administrator roles to isolate administrative actions. |
| O.AUDIT_GENERATION | The TOE will provide the capability to create records of security relevant events associated with users. |
| O.MANAGE | The TOE will provide all the functions and facilities necessary to support the authorized administrators in their management of the security of the TOE. |
| O.NON_REPUDIATION | The TOE will prevent users from avoiding accountability for sending and receiving a message by providing evidence that the user sent or received the message. |
| O.SECURE_TRANS | The TOE will secure transmission within the TOE so that unauthorized user or process are unable eavesdrop on communications between separate parts of the TOE allowing them to intercept and modify transmitted data |
| O.TOE_PROTECTION | The TOE will protect itself and its assets from external interference or tampering. |
| O.USER_IDENTIFICATION | The TOE will uniquely identify users. |

## 4.2 Security Objectives for the IT Environment

| | |
|---|---|
| OE.AUDIT_STORAGE | The IT environment will provide the capability to protect audit information. |
| OE.USER_AUTHENTICATION | The IT environment will verify the claimed identity of users. |
| OE.USER_IDENTIFICATION | The IT environment will uniquely identify users. |
| OE.TIME | The IT environment will provide a time source that provides reliable time stamps. |
| OE.TOE_PROTECTION | The IT environment will provide protection to the TOE and its assets from external interference or tampering. |
| OE.PLATFORM | The IT Environment underlying the TOE must fulfill the requirements for the IT Environment described in this ST. The IT Environment must provide a suitable operational environment for the TOE where the TOE is able to properly execute and the dependencies that the TOE has upon the IT Environment are properly fulfilled. |

## 4.3 Security Objectives for the Non-IT Environment

| | |
|---|---|
| OE.ADMIN_GUIDANCE | The TOE will provide authorized administrators with the necessary information for secure management of the TOE. |

| | |
|---|---|
| OE.CONFIG | The TOE will be installed, configured, managed and maintained in accordance with its guidance documentation and applicable security policies and procedures. |
| OE.INSTALL | The TOE will be delivered with the appropriate installation guidance to establish and maintain TOE security. |
| OE.PHYSICAL | Physical security will be provided within the domain for the value of the IT assets protected by the TOE and the value of the stored, processed, and transmitted information. |
| OE.SELF_PROTECTION | IT environment and its assets will be protected from external interference, tampering or unauthorized disclosure. |

# 5 IT Security Requirements

The TOE makes no strehgth of function claim. Note that some of the TOE security functional requirements (FCS_COP.1 and FPT_ITT.1) are based on cryptography, the strength of which is outside the scope of the evaluation..

## 5.1 TOE Security Functional Requirements

The following table describes the Security Functional Requirements (SFRs) that are candidates to be satisfied by WBI Message Broker.

**Table 1 – TOE Security Functional Components**

| REQUIREMENT CLASS | REQUIREMENT COMPONENT |
|---|---|
| **FAU: Security Audit** | FAU_GEN.1: Audit Data Generation |
| | FAU_GEN.2: User Identity Association |
| **FCO: Communication** | FCO_NRO.1: Selective Proof of Origin |
| | FCO_NRR.1: Selective Proof of Receipt |
| **FCS: Cryptographic Support** | FCS_COP.1: Cryptographic Operation |
| **FDP: User Data Protection** | FDP_ACC.2a: Complete Access Control |
| | FDP_ACF.1a: Security Attribute Based Access Control |
| | FDP_ACC.2b: Complete Access Control |
| | FDP_ACF.1b: Security Attribute Based Access Control |
| **FIA: Identification and Authentication** | FIA_UID.2a: User Identification Before any Action |
| **FMT: Security Management** | FMT_MSA.1a: Management of Security Attributes |
| | FMT_MSA.1b: Management of Security Attributes |
| | FMT_MSA.3a: Static Attribute Initialization |
| | FMT_MSA.3b: Static Attribute Initialization |
| | FMT_SMF.1: Specification of Management Functions |
| | FMT_SMR.1: Security Roles |
| **FPT: Protection of the TSF** | FPT_ITT.1a: Basic Internal TSF Data Transfer Protection |
| | FPT_RVM.1a: Non-bypassability of the TSP |

### *5.1.1 Security Audit (FAU)*

#### 5.1.1.1 Audit Data Generation (FAU_GEN.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 audit functions; b) All auditable events for the [***not specified***] level of audit; and c) [**authentication attempts, Broker Policy accesses, and Topic Policy accesses for Websphere MQ transport.**].

**FAU_GEN.1.2** The TSF shall record within each audit record at least the following information: 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, [**no additional details**].

#### 5.1.1.2 User Identity Association (FAU_GEN.2)

**FAU_GEN.2.1** The TSF shall be able to associate each auditable event with the identity of the user that caused the event.

### *5.1.2 Communication (FCO)*

#### 5.1.2.1 Selective Proof of Origin (FCO_NRO.1)

**FCO_NRO.1.1** The TSF shall be able to generate evidence of origin for transmitted [**messages**] at the request of the [***originator, recipient,*** [**broker manager**]].

**FCO_NRO.1.2** The TSF shall be able to relate the [**UserIdentifer**] of the originator of the information, and the [**message text**] of the information to which the evidence applies.

**FCO_NRO.1.3** The TSF shall provide a capability to verify the evidence of origin of information to [***originator, recipient,*** [**broker manager**]] given [**the UserIdentifer is authentic**].

#### 5.1.2.2 Selective Proof of Receipt (FCO_NRR.1)

**FCO_NRR.1.1** The TSF shall be able to generate evidence of receipt for received [**messages**] at the request of the [***originator, recipient,*** [**broker manager**]].

**FCO_NRR.1.2** The TSF shall be able to relate the [**UserIdentifer**] of the recipient of the information, and the [**message text**] of the information to which the evidence applies.

**FCO_NRR.1.3** The TSF shall provide a capability to verify the evidence of receipt of information to [***originator, recipient,*** [**broker manager**]] given [**the UserIdentifer is authentic**].

### *5.1.3 Cryptographic Support (FCS)*

#### 5.1.3.1 Cryptographic Operation (FCS_COP.1)

**FCS_COP.1.1** The TSF shall perform [**encryption and decryption**] in accordance with a specified cryptographic algorithm [**IBM CryptoLite[2] and IBM Java Secure Sockets Extension (JSSE)[3] cryptographic libraries**] and cryptographic key sizes [**20 - 2048 bits**] that meet the following: [**ANSI X9.31 1998, FIPS 46-3, FIPS PUB 140-2, FIPS PUB 180-1, FIPS PUB 186-2 and FIPS PUB 197**].

**Application Notes:** (1) CryptoLite supports SHA-1, DES, DES-CBC, 3DES, 3DES-CBC, AES, AES-CBC, AES 256, HMAC SHA-1, RSA Sign/Verify, DSA Sign/Verify and PSEUDO Random Number Generator approved algorithms;

[2] FIPS 140-2 validation available at http://csrc.nist.gov/cryptval/140-1/140sp/140sp409.pdf
[3] FIPS 140-2 validation available at http://csrc.nist.gov/cryptval/140-1/140sp/140sp354.pdf

and, (2) JSSE supports SHA, DES, DES-CBC, 3DES, 3DES-CBC, AES, AES-CBC, RSA Sign/Verify and PSEUDO Random Number Generator approved algorithms.

### *5.1.4 User Data Protection (FDP)*

#### 5.1.4.1 Complete Access Control (FDP_ACC.2a)

**FDP_ACC.2.1a** The TSF shall enforce the [**Toolkit Access SFP**] on [**subjects: users; objects: topology, broker, execution group, root topic and subscription objects] and all operations among subjects and objects covered by the SFP**.

**FDP_ACC.2.2a** The TSF shall ensure that all operations between any subject in the TSC and any object within the TSC are covered by an access control SFP.

#### 5.1.4.2 Security Attribute Based Access Control (FDP_ACF.1a)

**FDP_ACF.1.1a** The TSF shall enforce the [**Toolkit Access SFP**] to objects based on the following: [**subject security attributes: user identifier; object security attributes: access control list (ACL)**].

**FDP_ACF.1.2a** The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [**if the user identity is a member of a group defined for the object and the ACL grants the group the requested access, the requested access is allowed].**

**FDP_ACF.1.3a** The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [**there are no explicit authorization rules**].

**FDP_ACF.1.4a** The TSF shall explicitly deny access of subjects to objects based on the [no explicit denial].

#### 5.1.4.3 Complete Access Control (FDP_ACC.2b)

**FDP_ACC.2.1b** The TSF shall enforce the [**Topic Access SFP**] on [**subjects: users; objects: topics] and all operations among subjects and objects covered by the SFP**.

**FDP_ACC.2.2b** The TSF shall ensure that all operations between any subject in the TSC and any object within the TSC are covered by an access control SFP.

#### 5.1.4.4 Security Attribute Based Access Control (FDP_ACF.1b)

**FDP_ACF.1.1b** The TSF shall enforce the [**Topic Access SFP**] to objects based on the following: [**subject security attributes: user identifier; object security attributes: access control list (ACL)**].

**FDP_ACF.1.2b** The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [(a) **if the ACL grants the requesting user identity the requested access, the requested access is allowed; or (b) if the user identity is a member of a group defined for the object and the ACL grants the group the requested access, the requested access is allowed; or (c) an explicit user ACL always takes priority**].

**FDP_ACF.1.3b** The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: **[(a.) if no explicit user ACL is present on the topic but an explict user ACL is present against an ancestor, the closest is used; or (b) if there is no explicit user ACL then:**

**1. if the user is a member of a group with an explict ACL, the explicit ACL is used;**
**2. if the user is a member of a group which does not have an explicit ACL at the topic level, but has an explicit ACL present against an ancestor, the closest is used;**
**3. if the user is contained in more than one group with an explicit ACL, permission is granted if any specifications are positive, otherwise access is denied ]**.

**FDP_ACF.1.4b** The TSF shall explicitly deny access of subjects to objects based on the **[(a) if the ACL entry explicitly denies access for a user, no access is granted; (b) if the ACL entry explicitly denies access for a group associated with the user identity, no access is granted**].

### *5.1.5 Identification and Authentication (FIA)*

#### 5.1.5.1 User Identification before any action (FIA_UID.2.1a)

**FIA_UID.2.1a** The TSF shall require each user to identify itself before allowing any other TSF-mediated actions on behalf of that user.

### *5.1.6 Security Management (FMT)*

#### 5.1.6.1 Management of security attributes (FMT_MSA.1a)

**FMT_MSA.1.1a** The TSF shall enforce the **[Toolkit Access SFP]** to restrict the ability to **[manage]** the security attributes **[of toolkit objects, namely ACLs]** to **[authorized administrators]**.

#### 5.1.6.2 Management of security attributes (FMT_MSA.1b)

**FMT_MSA.1.1b** The TSF shall enforce the **[Topic Access SFP]** to restrict the ability to **[manage]** the security attributes **[of topic objects, namely ACLs]** to **[authorized administrators]**.

#### 5.1.6.3 Static attribute initialization (FMT_MSA.3a)

**FMT_MSA.3.1a** The TSF shall enforce the **[Toolkit Access SFP]** to provide **[*restrictive*]** default values for security attributes that are used to enforce the SFP.

**FMT_MSA.3.2a** The TSF shall allow the **[no one]** to specify alternative initial values to override the default values when an object or information is created.

#### 5.1.6.4 Static attribute initialization (FMT_MSA.3b)

**FMT_MSA.3.1b** The TSF shall enforce the **[Topic Access SFP]** to provide **[*permissive*]** default values for security attributes that are used to enforce the SFP.

**FMT_MSA.3.2b** The TSF shall allow the **[authorized administrator]** to specify alternative initial values to override the default values when an object or information is created

#### 5.1.6.5 Specification of Management Functions (FMT_SMF.1)

**FMT_SMF.1.1** The TSF shall be capable of performing the following security management functions: [**management of Toolkit and Topic Access SFPs object attributes**].

#### 5.1.6.6 Security Roles (FMT_SMR.1)

**FMT_SMR.1.1** The TSF shall maintain the roles [**authorized administrator**].

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

### *5.1.7 Protection of the TSF (FPT)*

#### 5.1.7.1 Basic Internal TSF Data Transfer Protection (FPT_ITT.1a)

**FPT_ITT.1.1a** The TSF shall protect TSF data from [***disclosure* and *modification***] when it is transmitted between separate parts of the TOE.

#### 5.1.7.2 Non-bypassability of the TSP (FPT_RVM.1a)

**FPT_RVM.1.1a** The TSF shall ensure that TSP enforcement functions are invoked and succeed before each function within the TSC is allowed to proceed.

## 5.2 IT Environment Security Functional Requirements

The following table describes the SFRs that are candidates to be satisfied by the IT environment of WBI Message Broker.

**Table 2 – IT Environment Security Functional Components**

| REQUIREMENT CLASS | REQUIREMENT COMPONENT |
|---|---|
| **FAU: Security Audit** | FAU_STG.1: Protected Audit Trail Storage |
| **FIA: Identification and Authentication** | FIA_UAU.2: User Authentication Before any Action |
| | FIA_UID.2b: User Identification Before any Action |
| **FPT: Protection of the TSF** | FPT_ITT.1b: Basic Internal TSF Data Transfer Protection |
| | FPT_RVM.1b: Non-bypassability of the TSP |
| | FPT_SEP.1: TSF Domain Separation |
| | FPT_STM.1: Reliable Time Stamps |

### *5.2.1 Security Audit (FAU)*

#### 5.2.1.1 Protected Audit Trail Storage (FAU_STG.1)

**FAU_STG.1.1** The TSF shall protect the stored audit records from unauthorised deletion.

**FAU_STG.1.2** The TSF shall be able to [***prevent***] unauthorised modifications to the audit records in the audit trail.

### *5.2.2 Identification and Authentication (FIA)*

#### 5.2.2.1 User Authentication Before any Action (FIA_UAU.2)

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

#### 5.2.2.2 User Identification before any Action (FIA_UID.2b)

**FIA_UID.2.1b** The TSF shall require each user to identify itself before allowing any other TSF-mediated actions on behalf of that user.

### *5.2.3 Protection of the TSF (FPT)*

#### 5.2.3.1 Basic Internal TSF Data Transfer Protection (FPT_ITT.1b)

**FPT_ITT.1.1a** The TSF shall protect TSF data from [***disclosure*** **and** ***modification***] when it is transmitted between separate parts of the TOE.

#### 5.2.3.2 Non-bypassability of the TSP (FPT_RVM.1b)

**FPT_RVM.1.1b** The TSF shall ensure that TSP enforcement functions are invoked and succeed before each function within the TSC is allowed to proceed.

#### 5.2.3.3 TSF Domain Separation (FPT_SEP.1)

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

**FPT_SEP.1.2** The TSF shall enforce separation between the security domains of subjects in the TSC.

#### 5.2.3.4 Reliable Time Stamps (FPT_STM.1)

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

## 5.3 TOE Security Assurance Requirements

The security assurance requirements for the TOE are the EAL 3 augmented with ALC_FLR.2 as specified in Part 3 of the Common Criteria. No operations are applied to the assurance components.

**Table 3 – EAL 3 Assurance Components**

| REQUIREMENT CLASS | REQUIREMENT COMPONENT |
|---|---|
| **ACM: Configuration Management** | ACM_CAP.3: Authorization Controls |
| | ACM_SCP.1: TOE CM Coverage |
| **ADO: Delivery and Operation** | ADO_DEL.1: Delivery Procedures |
| | ADO_IGS.1: Installation, Generation, and Start-up Procedures |
| **ADV: Development** | ADV_FSP.1: Informal Functional Specification |
| | ADV_HLD.2: Security Enforcing High-level Design |
| | ADV_RCR.1: Informal Correspondence Demonstration |
| **AGD: Guidance Documents** | AGD_ADM.1: Administrator Guidance |
| | AGD_USR.1: User Guidance |
| **ALC: Life Cycle Support** | ALC_DVS.1: Identification of Security Measures |
| | ALC_FLR.2: Flaw reporting procedures |
| **ATE: 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 |
| **AVA: Vulnerability Assessment** | AVA_MSU.1: Examination of Guidance |
| | AVA_SOF.1: Strength of TOE Security Function Evaluation |
| | AVA_VLA.1: Developer Vulnerability Analysis |

### *5.3.1 Configuration Management (ACM)*

#### 5.3.1.1 Authorization Controls (ACM_CAP.3)

**ACM_CAP.3.1d** The developer shall provide a reference for the TOE.
**ACM_CAP.3.2d** The developer shall use a CM system.
**ACM_CAP.3.3d** The developer shall provide CM documentation.
**ACM_CAP.3.1c** The reference for the TOE shall be unique to each version of the TOE.
**ACM_CAP.3.2c** The TOE shall be labelled with its reference.
**ACM_CAP.3.3c** The CM documentation shall include a configuration list and a CM plan.
**ACM_CAP.3.4c** The configuration list shall uniquely identify all configuration items that comprise the TOE.
**ACM_CAP.3.5c** The configuration list shall describe the configuration items that comprise the TOE.
**ACM_CAP.3.6c** The CM documentation shall describe the method used to uniquely identify the configuration items.
**ACM_CAP.3.7c** The CM system shall uniquely identify all configuration items.
**ACM_CAP.3.8c** The CM plan shall describe how the CM system is used.
**ACM_CAP.3.9c** The evidence shall demonstrate that the CM system is operating in accordance with the CM plan.
**ACM_CAP.3.10c** The CM documentation shall provide evidence that all configuration items have been and are being effectively maintained under the CM system.
**ACM_CAP.3.11c** The CM system shall provide measures such that only authorised changes are made to the configuration items.

**ACM_CAP.3.1e** The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

#### 5.3.1.2 TOE CM Coverage (ACM_SCP.1)

**ACM_SCP.1.1d** The developer shall provide a list of configuration items for the TOE.
**ACM_SCP.1.1c** The list of configuration items shall include the following: implementation representation and the evaluation evidence required by the assurance components in the ST.
**ACM_SCP.1.1e** The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

### *5.3.2 Delivery and Operation (ADO)*

#### 5.3.2.1 Delivery Procedures (ADO_DEL.1)

**ADO_DEL.1.1d** The developer shall document procedures for delivery of the TOE or parts of it to the user.
**ADO_DEL.1.2d** The developer shall use the delivery procedures.
**ADO_DEL.1.1c** The delivery documentation shall describe all procedures that are necessary to maintain security when distributing versions of the TOE to a user's site.
**ADO_DEL.1.1e** The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

#### 5.3.2.2 Installation, Generation, and Start-up Procedures (ADO_IGS.1)

**ADO_IGS.1.1d** The developer shall document procedures necessary for the secure installation, generation, and start-up of the TOE.
**ADO_IGS.1.1c** The installation, generation and start-up documentation shall describe all the steps necessary for secure installation, generation and start-up of the TOE.
**ADO_IGS.1.1e** The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
**ADO_IGS.1.2e** The evaluator shall determine that the installation, generation, and start-up procedures result in a secure configuration.

### *5.3.3 Development (ADV)*

#### 5.3.3.1 Informal Functional Specification (ADV_FSP.1)

**ADV_FSP.1.1d** The developer shall provide a functional specification.
**ADV_FSP.1.1c** The functional specification shall describe the TSF and its external interfaces using an informal style.
**ADV_FSP.1.2c** The functional specification shall be internally consistent.
**ADV_FSP.1.3c** The functional specification shall describe the purpose and method of use of all external TSF interfaces, providing details of effects, exceptions and error messages, as appropriate.
**ADV_FSP.1.4c** The functional specification shall completely represent the TSF.
**ADV_FSP.1.1e** The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
**ADV_FSP.1.2e** The evaluator shall determine that the functional specification is an accurate and complete instantiation of the TOE security functional requirements.

#### 5.3.3.2 Security Enforcing High-level Design (ADV_HLD.2)

**ADV_HLD.2.1d** The developer shall provide the high-level design of the TSF.
**ADV_HLD.2.1c** The presentation of the high-level design shall be informal.
**ADV_HLD.2.2c** The high-level design shall be internally consistent.
**ADV_HLD.2.3c** The high-level design shall describe the structure of the TSF in terms of subsystems.

**ADV_HLD.2.4c** The high-level design shall describe the security functionality provided by each subsystem of the TSF.
**ADV_HLD.2.5c** The high-level design shall identify any underlying hardware, firmware, and/or software required by the TSF with a presentation of the functions provided by the supporting protection mechanisms implemented in that hardware, firmware, or software.
**ADV_HLD.2.6c** The high-level design shall identify all interfaces to the subsystems of the TSF.
**ADV_HLD.2.7c** The high-level design shall identify which of the interfaces to the subsystems of the TSF are externally visible.
**ADV_HLD.2.8c** The high-level design shall describe the purpose and method of use of all interfaces to the subsystems of the TSF, providing details of effects, exceptions and error messages, as appropriate.
**ADV_HLD.2.9c** The high-level design shall describe the separation of the TOE into TSPenforcing and other subsystems.
**ADV_HLD.2.1e** The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
**ADV_HLD.2.2e** The evaluator shall determine that the high-level design is an accurate and complete instantiation of the TOE security functional requirements.

#### 5.3.3.3 Informal Correspondence Demonstration (ADV_RCR.1)

**ADV_RCR.1.1d** The developer shall provide an analysis of correspondence between all adjacent pairs of TSF representations that are provided.
**ADV_RCR.1.1c** For each adjacent pair of provided TSF representations, the analysis shall demonstrate that all relevant security functionality of the more abstract TSF representation is correctly and completely refined in the less abstract TSF representation.
**ADV_RCR.1.1e** The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

### *5.3.4 Guidance Documents (AGD)*

#### 5.3.4.1 Administrator Guidance (AGD_ADM.1)

**AGD_ADM.1.1d** The developer shall provide administrator guidance addressed to system administrative personnel.
**AGD_ADM.1.1c** The administrator guidance shall describe the administrative functions and interfaces available to the administrator of the TOE.
**AGD_ADM.1.2c** The administrator guidance shall describe how to administer the TOE in a secure manner.
**AGD_ADM.1.3c** The administrator guidance shall contain warnings about functions and privileges that should be controlled in a secure processing environment.
**AGD_ADM.1.4c** The administrator guidance shall describe all assumptions regarding user behaviour that are relevant to secure operation of the TOE.
**AGD_ADM.1.5c** The administrator guidance shall describe all security parameters under the control of the administrator, indicating secure values as appropriate.
**AGD_ADM.1.6c** The administrator guidance shall describe each type of security-relevant event relative to the administrative functions that need to be performed, including changing the security characteristics of entities under the control of the TSF.
**AGD_ADM.1.7c** The administrator guidance shall be consistent with all other documentation supplied for evaluation.
**AGD_ADM.1.8c** The administrator guidance shall describe all security requirements for the IT environment that are relevant to the administrator.
**AGD_ADM.1.1e** The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

#### 5.3.4.2 User Guidance (AGD_USR.1)

**AGD_USR.1.1d** The developer shall provide user guidance.
**AGD_USR.1.1c** The user guidance shall describe the functions and interfaces available to the non-administrative users of the TOE.

**AGD_USR.1.2c** The user guidance shall describe the use of user-accessible security functions provided by the TOE.
**AGD_USR.1.3c** The user guidance shall contain warnings about user-accessible functions and privileges that should be controlled in a secure processing environment.
**AGD_USR.1.4c** The user guidance shall clearly present all user responsibilities necessary for secure operation of the TOE, including those related to assumptions regarding user behaviour found in the statement of TOE security environment.
**AGD_USR.1.5c** The user guidance shall be consistent with all other documentation supplied for evaluation.
**AGD_USR.1.6c** The user guidance shall describe all security requirements for the IT environment that are relevant to the user.
**AGD_USR.1.1e** The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

### *5.3.5 Life Cycle Support (ALC)*

#### 5.3.5.1 Identification of Security Measures (ALC_DVS.1)

**ALC_DVS.1.1d** The developer shall produce development security documentation.
**ALC_DVS.1.1c** The development security documentation shall describe all the physical, procedural, personnel, and other security measures that are necessary to protect the confidentiality and integrity of the TOE design and implementation in its development environment.
**ALC_DVS.1.2c** The development security documentation shall provide evidence that these security measures are followed during the development and maintenance of the TOE.
**ALC_DVS.1.1e** The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
**ALC_DVS.1.2e** The evaluator shall confirm that the security measures are being applied.

#### 5.3.5.2 Flaw reporting procedures (ALC_FLR.2)

**ALC_FLR.2.1d** The developer shall provide flaw remediation procedures addressed to TOE developers. *(per International Interpretation #94)*
**ALC_FLR.2.2d** The developer shall establish a procedure for accepting and acting upon all reports of security flaws and requests for corrections to those flaws. *(per International Interpretation #62)*
**ALC_FLR.2.3d** The developer shall provide flaw remediation guidance addressed to TOE users. *(per International Interpretation #94)*
**ALC_FLR.2.1c** The flaw remediation procedures documentation shall describe the procedures used to track all reported security flaws in each release of the TOE.
**ALC_FLR.2.2c** The flaw remediation procedures shall require that a description of the nature and effect of each security flaw be provided, as well as the status of finding a correction to that flaw.
**ALC_FLR.2.3c** The flaw remediation procedures shall require that corrective actions be identified for each of the security flaws.
**ALC_FLR.2.4c** The flaw remediation procedures documentation shall describe the methods used to provide flaw information, corrections and guidance on corrective actions to TOE users.
**ALC_FLR.2.5c** The flaw remediation procedures documentation shall describe a means by which the developer receives from TOE users reports and enquiries of suspected security flaws in the TOE.
**ALC_FLR.2.6c** The procedures for processing reported security flaws shall ensure that any reported flaws are corrected and the correction issued to TOE users.
**ALC_FLR.2.7c** The procedures for processing reported security flaws shall provide safeguards that any corrections to these security flaws do not introduce any new flaws.
**ALC_FLR.2.8c** The flaw remediation guidance shall describe a means by which TOE users report to the developer any suspected security flaws in the TOE.
**ALC_FLR.2.1e** The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

### *5.3.6 Tests (ATE)*

#### 5.3.6.1 Analysis of Coverage (ATE_COV.2)

**ATE_COV.2.1d** The developer shall provide an analysis of the test coverage.
**ATE_COV.2.1c** The analysis of the test coverage shall demonstrate the correspondence between the tests identified in the test documentation and the TSF as described in the functional specification.
**ATE_COV.2.2c** The analysis of the test coverage shall demonstrate that the correspondence between the TSF as described in the functional specification and the tests identified in the test documentation is complete.
**ATE_COV.2.1e** The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

#### 5.3.6.2 Testing: High-level Design (ATE_DPT.1)

**ATE_DPT.1.1d** The developer shall provide the analysis of the depth of testing.
**ATE_DPT.1.1c** The depth analysis shall demonstrate that the tests identified in the test documentation are sufficient to demonstrate that the TSF operates in accordance with its high-level design.
**ATE_DPT.1.1e** The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

#### 5.3.6.3 Functional Testing (ATE_FUN.1)

**ATE_FUN.1.1d** The developer shall test the TSF and document the results.
**ATE_FUN.1.2d** The developer shall provide test documentation.
**ATE_FUN.1.1c** The test documentation shall consist of test plans, test procedure descriptions, expected test results and actual test results.
**ATE_FUN.1.2c** The test plans shall identify the security functions to be tested and describe the goal of the tests to be performed.
**ATE_FUN.1.3c** The test procedure descriptions shall identify the tests to be performed and describe the scenarios for testing each security function. These scenarios shall include any ordering dependencies on the results of other tests.
**ATE_FUN.1.4c** The expected test results shall show the anticipated outputs from a successful execution of the tests.
**ATE_FUN.1.5c** The test results from the developer execution of the tests shall demonstrate that each tested security function behaved as specified.
**ATE_FUN.1.1e** The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

#### 5.3.6.4 Independent Testing - Sample (ATE_IND.2)

**ATE_IND.2.1d** The developer shall provide the TOE for testing.
**ATE_IND.2.1c** The TOE shall be suitable for testing.
**ATE_IND.2.2c** The developer shall provide an equivalent set of resources to those that were used in the developer's functional testing of the TSF.
**ATE_IND.2.1e** The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
**ATE_IND.2.2e** The evaluator shall test a subset of the TSF as appropriate to confirm that the TOE operates as specified.
**ATE_IND.2.3e** The evaluator shall execute a sample of tests in the test documentation to verify the developer test results.

### *5.3.7 Vulnerability Assessment (AVA)*

#### 5.3.7.1 Examination of Guidance (AVA_MSU.1)

**AVA_MSU.1.1d** The developer shall provide guidance documentation.
**AVA_MSU.1.1c** The guidance documentation shall identify all possible modes of operation of the TOE (including operation following failure or operational error), their consequences and implications for maintaining secure operation.

**AVA_MSU.1.2c** The guidance documentation shall be complete, clear, consistent and reasonable.
**AVA_MSU.1.3c** The guidance documentation shall list all assumptions about the intended environment.
**AVA_MSU.1.4c** The guidance documentation shall list all requirements for external security measures (including external procedural, physical and personnel controls).
**AVA_MSU.1.1e** The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
**AVA_MSU.1.2e** The evaluator shall repeat all configuration and installation procedures to confirm that the TOE can be configured and used securely using only the supplied guidance documentation.
**AVA_MSU.1.3e** The evaluator shall determine that the use of the guidance documentation allows all insecure states to be detected..

#### 5.3.7.2 Strength of TOE Security Function Evaluation (AVA_SOF.1)

**AVA_SOF.1.1d** The developer shall perform a strength of TOE security function analysis for each mechanism identified in the ST as having a strength of TOE security function claim.
**AVA_SOF.1.1c** For each mechanism with a strength of TOE security function claim the strength of TOE security function analysis shall show that it meets or exceeds the minimum strength level defined in the PP/ST.
**AVA_SOF.1.2c** For each mechanism with a specific strength of TOE security function claim the strength of TOE security function analysis shall show that it meets or exceeds the specific strength of function metric defined in the PP/ST.
**AVA_SOF.1.1e** The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
**AVA_SOF.1.2e** The evaluator shall confirm that the strength claims are correct.

#### 5.3.7.3 Developer Vulnerability Analysis (AVA_VLA.1)

**AVA_VLA.1.1d** The developer shall perform a vulnerability analysis.
**AVA_VLA.1.2d** The developer shall provide vulnerability analysis documentation.
**AVA_VLA.1.1c** The vulnerability analysis documentation shall describe the analysis of the TOE deliverables performed to search for obvious ways in which a user can violate the TSP.
**AVA_VLA.1.2c** The vulnerability analysis documentation shall describe the disposition of obvious vulnerabilities.
**AVA_VLA.1.3c** The vulnerability analysis documentation shall show, for all identified vulnerabilities, that the vulnerability cannot be exploited in the intended environment for the TOE.
**AVA_VLA.1.1e** The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.
**AVA_VLA.1.2e** The evaluator shall conduct penetration testing, building on the developer vulnerability analysis, to ensure obvious vulnerabilities have been addressed.

# 6 TOE Summary Specification

This chapter describes the security functions and associated assurance measures.

## 6.1 TOE Security Functions

### *6.1.1 Security Audit*

WMB provides its own audit mechanism. It writes all of its audit records to the underlying operating system. In Unix, all records are stored in the SYSLOG and in Windows all records are stored in the Event Log.

The auditable actions are authentication attempts, Broker Policy accesses, and Topic Policy accesses. Each audit record identifies the event type, responsible user (i.e. hostname), data and time of the event, an indication of success or failure, and other information specific to each audit event. Starting and stopping of audit function is implicitly audited as a result of stopping and starting product.

The Security audit function is designed to satisfy the following security functional requirement:

- FAU_GEN.1: The audit events as well as the audit record content enumerated above represent the set of required events and information.
- FAU_GEN.2: Each audit record contains the responsible user identifier.

### *6.1.2 Communication*

WMB has the capability to verify the sender and receiver of all messages. The recipient, sender, and Broker Manager all have the ability to request proof that the sender is authentic and they also have the ability to request proof the receiver's identity. The proof the TSF generates is the ensuring the UserIdentifier field in the messages is correct. This field is protected using SSL.

The Communication security function is designed to satisfy the following security functional requirements:

- FCO_NRO.1: Originator identification requests are verified using SSL protections.
- FCO_NRR.1: Receiver identification requests are verified using SSL protections.

### *6.1.3 User data protection*

WMB implements two access control policies – a Toolkit Policy and a Topic Policy. The Toolkit Policy controls access between users and the following objects:

- Topology - The brokers and collectives (and connections between them) in the broker domain.
- Broker - A set of execution processes that host one or more message flows.
- Execution group - A named process or set of processes within a broker in which message flows are executed. The broker is guaranteed to enforce some degree of isolation between message flows in distinct execution groups because it ensures that they execute in separate address spaces, or as unique processes.
- Root Topic - A character string that describes the nature of the data that is published in a publish/subscribe system. The root topic may contain sub-topics.
- Subscription - A record that contains the information that a subscriber passes to its local broker to describe the publications that it wants to receive.

Access to all Toolkit objects is controlled by an access control list (ACL). An ACL entry can contain a group and can grant or deny permission. Access is determined by the corresponding permission on the first entry in the ACL that the group matches.

The Topic Access Policy controls access between users and topics. The purpose of this policy is to control who can publish on, and subscribe to, which topics. Like the Broker Policy, the Topic Access Policy is controlled by ACLs. ACLs can contain entries for users or groups and can grant or deny permission. The permissions associated with topics are:

- Publish - Permits or denies the principal to publish messages on this topic.
- Subscribe - Permits or denies the principal to subscribe to messages on this topic.
- Persistent - Specifies whether the principal can receive messages persistently. If the principal is not permitted, all messages are sent non-persistently. Each individual subscription indicates whether the subscriber requires persistent messages.
- QoP - Specifies the level of message protection that is enforced. One of the following four values can be chosen:
  - None
  - Channel Integrity
  - Message Integrity
  - Encrypted – (In the evaluated configuration QoP must always be set to 'Encrypted'.)

The default value is 'None'

The User Data Protection function is designed to satisfy the following security functional requirements:

- FDP_ACC.2a: All users are subject to the Toolkit access policy for all available operations on topology, broker, execution group, root topic and subscription objects.
- FDP_ACF.1a: Toolkit objects have ACLs. ACLs are compared against user identities and group memberships for that user in order to determine whether the request operation should be allowed.
- FDP_ACC.2b: All users are subject to the Topic access policy for all available operations on topic objects.
- FDP_ACF.1b: Topic objects have ACLs. ACLs are compared against user identities and group memberships for that user in order to determine whether the request operation should be allowed. Alternately, a user may have an identity has access on an ancestor ACL; if the ancestor ACL grants permission, then permission is permitted. If both of these checks fail, access will be refused

### *6.1.4 Identification and Authentication*

WMB requires all users to perform identification and authentication before permitting any other actions to occur. WMB supports three modes of authentication in the evaluated configuration:

1. M - mutual challenge-response password authentication
2. A - asymmetric SSL
3. S - symmetric SSL

In the mutual challenge-response password authentication method, the client must satisfy the server's challenge before the server satisfies the client's challenge. This means that an attacker impersonating a client can gather no information to mount an "offline" password guessing attack. The broker must have access to user and password information. Information about the user ID and password is distributed by the User Name Server to all the brokers in the domain. The User Name Server extracts user and password information from an operating system file.

Each client application must know its own user ID and keep its password secret. When creating a connection, a client specifies its credentials as a name/password combination. The encryption algorithms supporting this authentication method are provided by the IBM CryptoLite package, which is FIPS compliant.

The asymmetric SSL authentication method is commonly used in most web browsers. In this protocol, only the brokers have public/private key pairs and clients know the brokers' public keys. The SSL protocol establishes a secure connection in which the broker is authenticated to the client using public key cryptography after which the client can send its password, encrypted by a secure session key, to authenticate itself to the broker.

The symmetric SSL authentication method is where both participants have public/private key pairs. The SSL protocol uses public key cryptography to accomplish mutual authentication. The encryption algorithms used for both Asymmetric and Symmetric SSL are provided by the JSSE Library (IBMJSSE.jar) stored within the broker directory. The libraries themselves are provided by the IBM JSSE package, which is FIPS compliant.

The Identification and authentication function is designed to satisfy the following security functional requirements:

- FIA_UID.2: The TOE offers no TSF-mediated functions until the user is identified.
- FCS_COP.1: FIPS approved encryption algorithm are used to support the authentication protocols.

### *6.1.5 Security Management*

The TSF provides the ability to manage the security functions of the TOE. The authorized administrator is the only user permitted to perform management functions. An authorized administrator is anyone assigned to the administrator group.

The management functions include the ability to manage the Toolkit and Broker access policies, including the ACLs associated with the controlled objects. Management occurs via a GUI interface using the Toolkit or using a command line interface. By default, all topic policy objects are created such that everyone has access. By default, all Toolkit policy objects are created to deny access to everyone. The authorized administrator can change this default to a more restrictive value during creation.

The Security management function is designed to satisfy the following security functional requirements:

- FMT_MSA.1a and b: The ability to manage controlled objects attributes is restricted to an authorized administrator.
- FMT_MSA.3a and b: By default every object is created with permissive access. The authorized administrator can specify access other than the default during creation.
- FMT_SMF.1: Administrators are able to perform all management functions, including management of Toolkit and Broker object attributes.
- FMT_SMR.1: Any user account that is assigned to the administrators group is considered an 'authorized administrator' and other user accounts are considered simply 'users'.

### *6.1.6 TSF Self Protection*

In real-time mode, all communications between parts of the TOE are performed over a SSL session. The SSL session will protect the data transmitted from unauthorized modification or disclosure. In the non-real-time mode, communications between the client and Broker are via the MQ transport. Non-real time mode is included in the evaluation but the MQ transport is outside the scope of this evaluation (but encrypted nonetheless). All encryption is performed using FIPS-validated algorithms. Furthermore, WMB has been carefully designed to offer well-defined interfaces that ensure that access to protected resources is subject to the applicable WMB security policies.

The Protection of the TSF function is designed to satisfy the following security functional requirements:

- FPT_RVM.1a: This requirement is met because the TOE security policy functions are invoked and succeed before each function within the TSC is allowed to proceed
- FDP_ITT.1a - All communications between parts of the TOE are performed over a encrypted session which protects the data transmitted from unauthorized modification or disclosure[4].
- FCS_COP.1: FIPS approved encryption algorism are used to protect TOE communications.

## 6.2 TOE Security Assurance Measures

### *6.2.1 Configuration management*

The configuration management measures applied by IBM ensure that configuration items are uniquely identified, and that documented procedures are used to control and track changes that are made to the TOE. IBM ensures changes to the implementation representation are controlled. IBM performs configuration management on the TOE implementation representation, design documentation, tests and test documentation, user and administrator guidance, delivery and operation documentation, life-cycle documentation, vulnerability analysis documentation, and configuration management documentation and all of these items are identified in the CM Plan as configuration items.

These activities are documented in:

- WBI Message Broker - Configuration Management Plan

The Configuration management assurance measure satisfies the following EAL 3 augmented with ALC_FLR.2 assurance requirements:

[4] This requirement applies for real-time mode. FDP_ITT.1b applies to the MQ transport.

- ACM_CAP.3
- ACM_SCP.1

### 6.2.2 Delivery and operation

IBM provides delivery documentation and procedures to identify the TOE, secure the TOE during delivery, and provide necessary installation and generation instructions. IBM's delivery procedures describe all applicable procedures to be used to prevent in appropriate access to the TOE. IBM also provides documentation that describes the steps necessary to install WBI Message Broker in accordance with the evaluated configuration.

These activities are documented in:

- WBI Message Broker - Installation and Delivery Guide

The Delivery and operation assurance measure satisfies the following EAL 3 augmented with ALC_FLR.2 assurance requirements:

- ADO_DEL.1
- ADO_IGS.1

### 6.2.3 Development

IBM has numerous documents describing all facets of the design of the TOE. In particular, they have a functional specification that describes the accessible TOE interfaces; a high-level design that decomposes the TOE architecture into subsystems and describes each subsystem and their interfaces; and, correspondence documentation that explains how each of the design abstractions correspond from the TOE summary specification in the Security Target to the subsystems.

These activities are documented in:

- WBI Message Broker - Functional Specification
- WBI Message Broker - High-level Design
- WBI Message Broker - Design Correspondence Analysis

The Development assurance measure satisfies the following EAL 3 augmented with ALC_FLR.2 assurance requirements:

- ADV_FSP.1
- ADV_HLD.2
- ADV_RCR.1

### 6.2.4 Guidance documents

IBM provides administrator and user guidance on how to utilize the TOE security functions and warnings to administrators and users about actions that can compromise the security of the TOE.

These activities are documented in:

- WBI Message Broker - Administration Guide
- WBI Message Broker - User Guide

The Guidance documents assurance measure satisfies the following EAL 3 augmented with ALC_FLR.2 assurance requirements:

- AGD_ADM.1
- AGD_USR.1

### 6.2.5 Life cycle support

IBM ensures the adequacy of the procedures used during the development and maintenance of the TOE through its life-cycle. IBM includes security controls on the development environment that are adequate to provide the confidentiality and integrity of the TOE design and implementation that is necessary to ensure the secure operation of the TOE. In addition, IBM identifies and tracks reported flaws, ensuring that they are addressed and corrections and corrective measures are made available as applicable.

These activities are documented in:

- WBI Message Broker - Life-cycle Plan

The Life cycle support assurance measure satisfies the following EAL 3 augmented with ALC_FLR.2 assurance requirements:

- ALC_DVS.1
- ALC_FLR.2

### 6.2.6 Tests

IBM has a test plan that describes how each of the necessary security functions is tested, along with the expected test results. IBM has documented each test as well as an analysis of test coverage demonstrating that the security aspects of the design evident from the functional specification and high level design are appropriately tested. Actual test results are provided that demonstrate that the tests have been applied and that the TOE operates as designed. The test documentation consists of the following documents:

- WBI Message Broker - Test Plan
- WBI Message Broker - Test Coverage Analysis
- WBI Message Broker - Test Results

The Tests assurance measure satisfies the following EAL 3 augmented with ALC_FLR.2 assurance requirements:

- ATE_COV.2
- ATE_DPT.1
- ATE_FUN.1
- ATE_IND.2

### 6.2.7 Vulnerability assessment

The TOE administrator and user guidance documents describe the operation of WBI Message Broker and how to maintain a secure state. These guides also describe all necessary operating assumptions and security requirements outside the scope of control of the TOE. They have been developed to serve as complete, clear, consistent, and reasonable administrator and user references.

IBM performs regular vulnerability analyses of the entire TOE (including documentation) to identify obvious weaknesses that can be exploited in the TOE.

These activities are documented in:

- WBI Message Broker - Vulnerability Analysis Report

The Vulnerability assessment assurance measure satisfies the following EAL 3 augmented with ALC_FLR.2 assurance requirements:

- AVA_MSU.1
- AVA_SOF.1
- AVA_VLA.1

# 7 Protection Profile Claims

There are no Protection Profile claims.

# 8 Rationale

This section provides the rationale for completeness and consistency of the Security Target. The rationale addresses the following areas:

- Security Objectives;
- Security Functional Requirements;
- Security Assurance Requirements;
- Strength of Functions;
- Security Functional Requirement Dependencies;
- Explicitly Stated Requirements;
- TOE Summary Specification; and
- PP Claims.

## 8.1 Security Objectives Rationale

This section shows that all secure usage assumptions and threats are completely covered by security objectives. In addition, each objective counters or addresses at least one assumption or threat.

### *8.1.1 Security Objectives Rationale for the TOE and Environment*

This section provides evidence demonstrating the coverage of threats and usage assumptions by the security objectives.

| | T.ADMIN_ERROR | T.MESSAGE_DENIAL | T.NETWORK | T.ROLE | T.TSF_COMPROMISE | T.UNAUTH_ACCESS | T.UNDETECTED_ACTIONS | A.INSTALL_REO | A.NO_EVIL | A.PHYSICAL | A.PLATFORM |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **O.ACCESS** | | | | | | X | | | | | |
| **O.ADMIN_ROLE** | | | | X | | | | | | | |
| **O.AUDIT_GENERATION** | | | | | | | X | | | | |
| **O.MANAGE** | X | | | | | | | | | | |
| **O.NON_REPUDIATION** | | X | | | | | | | | | |
| **O.SECURE_TRANS** | | | X | | | | | | | | |
| **O.TOE_PROTECTION** | | | | | X | | | | | | |
| **O.USER_IDENTIFICATION** | | | | X | | | | | | | |
| **OE.AUDIT_STORAGE** | | | | | X | | | | | | |
| **OE.TIME** | | | | | | | X | | | | |
| **OE.TOE_PROTECTION** | | | X | | X | | | | | | |
| **OE.ADMIN_GUIDANCE** | X | | | | | | | | | | |

| | T.ADMIN_ERROR | T.MESSAGE_DENIAL | T.NETWORK | T.ROLE | T.TSF_COMPROMISE | T.UNAUTH_ACCESS | T.UNDETECTED_ACTIONS | A.INSTALL_REO | A.NO_EVIL | A.PHYSICAL | A.PLATFORM |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **OE.CONFIG** | X | | | | | | | | X | | |
| **OE.INSTALL** | X | | | | | | | X | | | |
| **OE.PHYSICAL** | | | | | X | | X | | | X | |
| **OE.SELF_PROTECTION** | | | | | | X | | | | | |
| **OE.PLATFORM** | | | | | | | | | | | X |
| **OE.USER_AUTHENTICATION** | | | | X | | | | | | | |
| **OE.USER_IDENTIFICATION** | | | | X | | | | | | | |

**Table 4 Environment to Objective Correspondence**

#### 8.1.1.1 T.ADMIN_ERROR

*An authorized administrator may incorrectly install or configure the TOE resulting in ineffective security mechanisms.*

This Threat is satisfied by ensuring that:

- O.MANAGE: Improper administration could result if the TOE does not provide the proper administration tools. There is always the possibility that the administrator will make an honest mistake. This threat should be mitigated as long as the TOE provides the necessary administrator support.
- OE.ADMIN_GUIDANCE: Improper administration could result if the authorized administrator is unknowledgeable. There is always the possibility that the administrator will make an honest mistake. This threat should be mitigated as long as the authorized administrator is provided with knowledge necessary to carry out administrative duties.
- OE.INSTALL: The authorized administrator is provided with necessary installation instructions from the developer that details how to securely install the TOE.
- OE.CONFIG: The authorized administrator is provided with necessary instructions to securely configure the TOE.

#### 8.1.1.2 T.MESSAGE_DENIAL

*The sender or receiver of a message denies sending or receiving the message to avoid accountability for subsequent action or inaction.*

This Threat is satisfied by ensuring that:

- O.NON-REPUDIATION: This requires the TOE to produce evidence of origin and receipt of messages.

#### 8.1.1.3 T.NETWORK

*Data transferred between workstations is disclosed to or modified by unidentified users or processes monitoring network traffic.*

This Threat is satisfied by ensuring that:

- O.SECURE_TRANS : The TOE will secure transmission within the TOE so that unauthorized user or process are unable eavesdrop on communications between separate parts of the TOE allowing them to intercept and modify transmitted data

- O.TOE_PROTECTION: The IT environment will provide protection of the TSS data from disclosure and modification.

#### 8.1.1.4 T.ROLE

*A non-privileged user may gain administrative privileges and bypasses the security policy.*

This Threat is satisfied by ensuring that:

- O.USER_IDENTIFICATION: This requires users to claim their unique identity prior to accessing the TOE.
- O.ADMIN_ROLE: This requires the TOE to support the concept of an Administrator to manage the TOE.
- OE.USER_AUTHENTICATION: This requires users to authenticate their identity prior to accessing the TOE.
- OE.USER_IDENTIFICATION: This requires users to claim their unique identity prior to accessing the TOE.

#### 8.1.1.5 T.TSF_COMPROMISE

*A malicious user or process may cause configuration data to be inappropriately accessed (viewed, modified or deleted) by changing access to the configuration data).*

This Threat is satisfied by ensuring that:

- OE.AUDIT_STORAGE: The IT environment will protect the audit data.
- O.TOE_PROTECTION: The TSF data and executable code is protected under the TOE objective for TOE protection.
- OE.TOE_PROTECTION: The TSF data and executable code is protected under the environmental objective for TOE protection.
- OE.PHYSICAL: The IT environment will protect the TSF data and executable code from a compromise through physical means.

#### 8.1.1.6 T.UNAUTH_ACCESS

*A user may gain unauthorized access (view, modify, delete) to user data by bypassing access controls.*

This Threat is satisfied by ensuring that:

- O.ACCESS: The TOE must satisfy the objective of ensuring that only authorized users may gain access to the TOE and the resources it protects, and that users are not allowed to access protected data for which they are not authorized.
- OE_SELF_PROTECTION: The IT environment and its assets are protected under the environmental objective for self-protection.

#### 8.1.1.7 T.UNDETECTED_ACTIONS

*Failure of the IT operating system to detect and record unauthorized actions may occur.*

This Threat is satisfied by ensuring that:

- O.AUDIT_GENERATION: Non-physical security relevant actions are detected and a record is made.
- OE.TIME: All audit records include reliable timestamps.
- OE.PHYSICAL: The threat of undetected physical manipulation of the TOE is addressed by the physical protection in the environment.

#### 8.1.1.8 A.INSTALL_REQ

*The Message Brokers Toolkit must be installed on the same platform as the Configuration Manager because the two components do not encrypt traffic when communicating via a network.*

This Assumption is satisfied by ensuring that:

- OE.CONFIG: Authorized administrators are trusted to properly configure the IT environment so it enforces its security policies
- OE.INSTALL: Authorized administrators have the proper documentation to install configure and manage the TOE and will follow that documentation.

#### 8.1.1.9 A.NO_EVIL

*Authorized administrators are non-hostile, appropriately trained and follow all administrator guidance.*

This Assumption is satisfied by ensuring that:

- OE.CONFIG: Authorized administrators are trusted to properly configure the TOE and IT environment so it enforces its security policies.

#### 8.1.1.10 A.PHYSICAL

*It is assumed that appropriate physical security is provided within the domain for the value of the IT assets protected by the TOE and the value of the stored, processed, and transmitted information.*

This Assumption is satisfied by ensuring that:

- OE.PHYSICAL: The TOE, the TSF data, and protected user data is assumed to be protected from physical attack (e.g., theft, modification, destruction, or eavesdropping). Physical attack could include unauthorized intruders into the TOE environment, but it does not include physical destructive actions that might be taken by an individual that is authorized to access the TOE environment.

#### 8.1.1.11 A.PLATFORM

*The IT Environment underlying the TOE is assumed to fulfill the requirements for the IT Environment described in this Security Target. It is also assumed that the IT Environment will provide a suitable operational environment for the TOE where the TOE will be able to properly execute and the dependencies that the TOE has upon the IT Environment are properly fulfilled.*

This Assumption is satisfied by ensuring that:

- O.PLATFORM: This objective basically reiterates the assumption to expect the IT Environment to provide a suitable and effective environment for the operation of the TOE.

## 8.2 Security Requirements Rationale

This section provides evidence supporting the internal consistency and completeness of the components (requirements) in the Security Target. Note that **Table 5** indicates the requirements that effectively satisfy the individual objectives. .

### *8.2.1 Security Functional Requirements Rationale*

All Security Functional Requirements (SFR) identified in this Security Target are fully addressed in this section and each SFR is mapped to the objective for which it is intended to satisfy.

| | O.ACCESS | O.ADMIN_ROLE | O.AUDIT_GENERATION | O.MANAGE | O.NON_REPUDIATION | O.SECURE_TRANS | O.TOE_PROTECTION | O.USER_IDENTIFICATION | OE.AUDIT_STORAGE | OE.TIME | OE.TOE_PROTECTION | OE..USER_AUTHENTICATION | OE.USER_IDENNTIFICATION |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **FAU_GEN.1** | | | X | | | | | | | | | | |

| | O.ACCESS | O.ADMIN_ROLE | O.AUDIT_GENERATION | O.MANAGE | O.NON_REPUDIATION | O.SECURE_TRANS | O.TOE_PROTECTION | O.USER_IDENTIFICATION | OE.AUDIT_STORAGE | OE.TIME | OE.TOE_PROTECTION | OE..USER_AUTHENTICATION | OE.USER_IDENNTIFICATION |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **FAU_GEN.2** | | | X | | | | | | | | | | |
| **FCO_NRO.1** | | | | | X | | | | | | | | |
| **FCO_NRR.1** | | | | | X | | | | | | | | |
| **FCS_COP.1** | | | | | | | X | | | | | | |
| **FDP_ACC.2a** | X | | | | | | | | | | | | |
| **FDP_ACF.1a** | X | | | | | | | | | | | | |
| **FDP_ACC.2b** | X | | | | | | | | | | | | |
| **FDP_ACF.1b** | X | | | | | | | | | | | | |
| **FIA_UID.2a** | | | | | | | | X | | | | | |
| **FMT_MSA.1a** | | | | X | | | | | | | | | |
| **FMT_MSA.1b** | | | | X | | | | | | | | | |
| **FMT_MSA.3a** | | | | X | | | | | | | | | |
| **FMT_MSA.3b** | | | | X | | | | | | | | | |
| **FMT_SMF.1** | | | | X | | | | | | | | | |
| **FMT_SMR.1** | | X | | | | | | | | | | | |
| **FPT_ITT.1a** | | | | | | X | | | | | | | |
| **FPT_ITT.1b** | | | | | | | | | | | X | | |
| **FPT_RVM.1a** | | | | | | | X | | | | | | |
| **FAU_STG.1** | | | | | | | | | X | | | | |
| **FPT_RVM.1b** | | | | | | | | | | | X | | |
| **FPT_SEP.1** | | | | | | | | | | | X | | |
| **FPT_STM.1** | | | | | | | | | | X | | | |
| **FIA_UAU.2** | | | | | | | | | | | | X | |
| **FIA_UID.2b** | | | | | | | | | | | | | X |

**Table 5 Objective to Requirement Correspondence**

#### 8.2.1.1 O.ACCESS

*The TOE will ensure that users gain only authorized access to it and to the resources that it controls.*

This TOE Security Objective is satisfied by ensuring that:

- FDP_ACC.2a and b: The subjects and objects within the TOE are under the enforcement of an access control policy. All operations between the subjects and objects are controlled by the access policies.
- FDP_ACF.1a and b: The subjects and objects under the access control policies have certain rules that apply to all accesses between them. In the case of both access policies, all accesses are controlled by decisions based on user identities and ACLS on objects.

#### 8.2.1.2 O.ADMIN_ROLE

*The TOE will provide authorized administrator roles to isolate administrative actions.*

This TOE Security Objective is satisfied by ensuring that:

- FMT_SMR.1: The TOE will establish an authorized administrator role. The authorized administrator will be given privileges to perform certain tasks that other users will not be able to perform. These privileges include, but are not limited to, access to managing the security policies.

#### 8.2.1.3 O.AUDIT_GENERATION

*The TOE will provide the capability to create records of security relevant events associated with users.*

This TOE Security Objective is satisfied by ensuring that:

- FAU_GEN.1: This objective is satisfied in part by the requirement that the TOE generate audit records.
- FAU_GEN.2: Each audit record written must be descriptive of the event that caused a record to be generated, and must be associated with the unique identity of the user that caused the event.

#### 8.2.1.4 O.MANAGE

*The TOE will provide all the functions and facilities necessary to support the authorized administrators in their management of the security of the TOE.*

This TOE Security Objective is satisfied by ensuring that:

- FMT_MSA.1a and b: Only authorized administrators may manipulate the security attributes of policy objects, namely ACLs.
- FMT_MSA.3a and b: Only authorized administrators may manipulate the security attributes of policy objects.
- FMT_SMF.1: The authorized administrator will be able to manage the access policies object attributes.

#### 8.2.1.5 O.NON_REPUDIATION

*The TOE will prevent users from avoiding accountability for sending and receiving a message by providing evidence that the user sent or received the message.*

This TOE Security Objective is satisfied by ensuring that:

- FCO_NRO.1 – This objective is satisfied in part by the requirement that the TOE generate evidence of origin.
- FCO_NRR.1 – This objective is satisfied in part by the requirement that the TOE generate evidence of receipt.

#### 8.2.1.6 O.SECURE_TRANS

*The TOE will secure transmission within the TOE so that unauthorized user or process are unable eavesdrop on communications between separate parts of the TOE allowing them to intercept and modify transmitted data.*

This TOE Security Objective is satisfied by ensuring that:

- FPT_ITT.1a: The TOE will protect all TSF data while it is in transit among distributed portions of the TOE.

#### 8.2.1.7 O.TOE_PROTECTION

*The TOE will protect itself and its assets from external interference or tampering.*

This TOE Security Objective is satisfied by ensuring that:

- FPT_RVM.1a: The TOE is required to allow access to protected objects only after it makes informed access decisions.
- FCS_COP.1: When sending encrypted data among distributed portions of the TOE, the data is encrypted using a FIPS-complaint algorithm.

#### 8.2.1.8 O.USER_IDENTIFICATION

*The TOE will uniquely identify users.*

This TOE Security Objective is satisfied by ensuring that:

- FIA_UID.2a: Users must be identified to the TOE before they can perform any TSF-mediated functions.

#### 8.2.1.9 OE.AUDIT_STORAGE

*The IT environment will provide the capability to protect audit information.*

This IT Environment Security Objective is satisfied by ensuring that:

- FAU_STG.1: The IT environment is required to protect the audit records from deletion.

#### 8.2.1.10 OE.USER_AUTHENTICATION

*The IT environment will verify the claimed identity of users.*

This TOE Security Objective is satisfied by ensuring that:

- FIA_UAU.2: Users must be authenticated before they can perform any TSF-mediated functions.

#### 8.2.1.11 OE.USER_IDENTIFICATION

*The IT environment will uniquely identify users.*

This TOE Security Objective is satisfied by ensuring that:

- FIA_UID.2b: Users must be identified to the TOE before they can perform any TSF-mediated functions.

#### 8.2.1.12 OE.TIME

*The IT environment will provide a time source that provides reliable time stamps.*

This IT Environment Security Objective is satisfied by ensuring that:

- FPT_STM.1: The IT environment is required to provide a reliable time source.

#### 8.2.1.13 OE.TOE_PROTECTION

*The IT environment will provide protection to the TOE and its assets from external interference or tampering.*

This IT Environment Security Objective is satisfied by ensuring that:

- FPT_RVM.1b: The IT environment is required to access to protected objects only after it makes informed access decisions.
- FPT_ITT.1b: The IT environment will protect all TSF data while it is in transit among distributed portions of the TOE.
- FPT_SEP.1: The IT environment is required to protect itself and separate the contexts of its users.

## 8.3 Security Assurance Requirements Rationale

WMB is targeted at a generalized IT environment with good physical access security and competent administrators. Within such environments it is assumed that attackers will have a low to moderate attack potential. As such, EAL 3 (augmented with ALC_FLR.2) is appropriate to provide the assurance necessary to counter the potential for attack.

## 8.4 Strength of Function Rationale

The TOE makes no strength of function claim. Note that some of the TOE security functional requirements (FCS_COP.1 and FPT_ITT.1) are based on cryptography, the strength of which is outside the scope of the evaluation.

## 8.5 Requirement Dependency Rationale

The following table represents an analysis of the dependencies of the security functional requirements (SFRs) in this security target. The first column identifies all of the SFRs in this security target. The TOE SFRs are highlighted in bold, unlike the IT environment SFRs. The second column identifies the minimum dependencies defined in the Common Criteria v2.2 and associated interpretations[5]. The third column identifies the actual requirements in this security target that correspond to the identified dependencies. Notice that this table demonstrates that all of the identified dependencies are satisfied with the exception of the dependencies for FCS_COP.1. The rationale for the missing dependencies is included after the table.

| ST Requirement | CC Dependencies | ST Dependencies |
|---|---|---|
| **FAU_GEN.1** | FPT_STM.1 | *FPT_STM.1* |
| **FAU_GEN.2** | FAU_GEN.1 and FIA_UID.1 | FAU_GEN.1 and FIA_UID.2 |
| **FCO_NRO.1** | FIA_UID.1 | FIA_UID.2 |
| **FCO_NRR.1** | FIA_UID.1 | FIA_UID.2 |
| **FCS_COP.1** | (FDP_ITC.1 or (FCS_CKM.1, FCS_CKM.4, and FMT_MSA.2) | none |
| **FDP_ACC.2a and b** | FDP_ACF.1 | FDP_ACF.1 |
| **FDP_ACF.1** | none | none |
| **FIA_UID.2a** | none | none |
| **FMT_MSA.1a and b** | FMT_SMR.1 and FMT_SMF.1 and (FDP_ACC.1 or FDP_IFC.1) | FMT_SMR.1 and FMT_SMF.1 and FDP_ACC.2a and b |
| **FMT_MSA.3a and b** | FMT_MSA.1 and FMT_SMR.1 | FMT_MSA.1a and b and FMT_SMR.1 |
| **FMT_SMF.1** | none | none |
| **FMT_SMR.1** | FIA_UID.1 | FIA_UID.2 |
| **FPT_ITT.1** | none | none |
| **FPT_RVM.1a** | none | none |
| **FAU_STG.1** | FAU_GEN.1 | FAU_GEN.1 |
| **FPT_RVM.1b** | none | none |
| **FPT_SEP.1** | none | none |
| **FPT_STM.1** | none | none |
| **FAU_UAU.2** | FIA_UID.1 | FIA_UID.2 |
| **FIA_UID.2b** | none | none |

**Table 6 Requirement Dependencies**

- FMT_MSA.2 – The FMT_MSA.2 requirement is a dependency of the FCS_COP.1 requirement to support the management of the cryptographic security function. In WMB, the administrator does not enter any values related to the cryptographic security function so this requirement is not applicable.
- FCS_CKM.1 and FCS_CKM.4 – These requirements addresses key creation and destruction. The keys associated with the encryption are automatically created and deleted upon connection and disconnection and are never exposed at the user interface. Therefore, these requirements are not necessary to satisfy the FCS_COP.1 requirement

## 8.6 Explicitly Stated Requirements Rationale

This security target includes no explicitly stated requirements.

[5] No International Interpretations are applicable to any SFR or SAR in this Security Target.

## 8.7 TOE Summary Specification Rationale

Each subsection in Section 6, the TOE Summary Specification, describes a security function of the TOE. Each description is followed with rationale that indicates which requirements are satisfied by aspects of the corresponding security function. The set of security functions work together to satisfy all of the security functions and assurance requirements. Furthermore, all of the security functions are necessary in order for the TSF to provide the required security functionality.

This Section in conjunction with Section 6, the TOE Summary Specification, provides evidence that the security functions are suitable to meet the TOE security requirements. The collection of security functions work together to provide all of the security requirements. The security functions described in the TOE summary specification are all necessary for the required security functionality in the TSF. **Table 7 Security Functions vs. Requirements Mapping** demonstrates the relationship between security requirements and security functions.

| | **Security audit** | **Communications** | **User data protection** | **Identification and authentication** | **Security management** | **TSF Self Protection** |
|---|---|---|---|---|---|---|
| **FAU_GEN.1** | X | | | | | |
| **FAU_GEN.2** | X | | | | | |
| **FCO_NRO.1** | | X | | | | |
| **FCO_NRR.1** | | X | | | | |
| **FCS_COP.1** | | | | | | X |
| **FDP_ACC.2a and b** | | | X | | | |
| **FDP_ACF.1** | | | X | | | |
| **FIA_UID.2a** | | | | X | | |
| **FMT_MSA.1a and b** | | | | | X | |
| **FMT_MSA.3a and b** | | | | | X | |
| **FMT_SMF.1** | | | | | X | |
| **FMT_SMR.1** | | | | | X | |
| **FPT_ITT.1** | | | | | | X |
| **FPT_RVM.1a** | | | | | | X |

**Table 7 Security Functions vs. Requirements Mapping**

## 8.8 PP Claims Rationale

See Section 7, Protection Profile Claims.