webMethods Fabric 6.5

EAL2 Common Criteria Evaluation

Security Target V1.0

12 December 2005

Prepared for:

webMethods, Inc.

3877 Fairfax Ridge Road,

Fairfax, VA 22030

http://www.webmethods.com/

Unclassified Controlled

Prepared by:

An Entrust Company

Suite 5200♦ 7925 Jones Branch Drive ♦ McLean, VA 22102-3305 ♦ 703 848-0883 ♦ Fax 703 848-0960

TABLE OF CONTENTS

Tables and Figures

1 Security Target Introduction

1.1 Security Target Identification

TOE Identification:	webMethods Fabric 6.5
ST Title:	webMethods Fabric 6.5 Security Target
ST Version:	Security Target V1.0
ST Authors:	Daniel DePrez
ST Date:	12 December 2005
Assurance Level:	EAL2
Strength of Function:	SOF-basic
Registration:	VID10070-0001-MR
Keywords:	Internet, Intranet, business integration, SSL, TLS, HTTP, FTP, JDBC, JMS

1.2 Security Target Overview

This Security Target (ST) defines the Information Technology (IT) security requirements for webMethods Fabric 6.5.

webMethods Fabric is a client/server application that provides access control of services hosted on the webMethods Integration Server. The product facilitates the secure exchange of data and logic among resources and supports the development and management of complex business processes through web enabled or browser interfaces.

The TOE implements the following security functions: Information Flow, Identification & Authentication, Security Management, and Self Protection. The TOE also supports auditing of security functions.

The TOE is CC Part 2 extended, Part 3 conformant, and meets the requirements of Evaluation Assurance Level (EAL) 2 from the Common Criteria Version 2.2, Rev 256 CCIMB-2004-01-001 January 2004.

1.3 Document Organization

The main sections of an ST are:

- the ST Introduction, Target of Evaluation (TOE) Description, TOE Security Environment, Security Objectives, IT Security Requirements, TOE Summary Specification, and Rationale.
- Section 2, TOE Description, describes the product type and the scope and boundaries of the TOE.
- Section 3, TOE Security Environment, identifies assumptions about the TOE's intended usage and environment and threats relevant to secure TOE operation.
- Section 4, Security Objectives, defines the security objectives for the TOE and its environment.
- Section 5, IT Security Requirements, specifies the TOE Security Functional Requirements (SFR), Security Requirements for the IT Environment, and the Security Assurance Requirements.

- Section 6, TOE Summary Specification, describes the IT Security Functions and Assurance Measures.
- Section 7, Protection Profile (PP) Claims, is not applicable, as this product does not claim conformance to any PP.
- Section 8, Rationale, presents evidence that the ST is a complete and cohesive set of requirements and that a conformant TOE would provide an effective set of IT security countermeasures within the security environment. The Rationale has three main parts:
    - Security Objectives Rationale,
    - Security Requirements Rationale, and
    - TOE Summary Specification Rationale.
- Sections 9 and 10 provide acronym definitions and bibliography.

2 TOE Description

2.1 Product Type

webMethods Fabric is a client/server application that provides access control of services hosted on the webMethods Integration Server. The product facilitates the secure exchange of data and logic among resources and supports the development and management of complex business processes through browser or web enabled interfaces.

2.2 How webMethods Fabric Works

The TOE has two main components:

Integration Server (IS) – Enables access control over the integration logic throughout the integrated applications.

Broker - a high-speed message router. Enables access control over asynchronous messaging.

Other architectural components are included in the Target of Evaluation (TOE) and will be introduced and discussed later.

The TOE implements the following security functions:

Access Control Policy - The flow of information is controlled between the external user and the IS, and between the IS and the Broker in accordance with a Discretionary Access Control (DAC) information flow control policy.

Identification & Authentication - Administrators are identified and authenticated by the Target of Evaluation Security Functions (TSF). Administrator identification and authentication (I&A) is through a user identifier and password. External users may also be I&A by the TSF through a user identifier and password.

Security Management - One administrative role is supported. An Administrator can I&A remotely with the IS and monitor and manage the interaction between external users. An Administrator can also I&A with the Broker Server and manage the interaction between TOE components, or I&A through the Developer interface and configure and develop services or workflows.

Self Protection - The TOE environment ensures all information from a client to a host or a host to a client goes through the TSF. The TSF ensures that all information must flow through the policy enforcement mechanisms.

Security Audit – The TOE supports limited auditing of TOE security functions.

The TOE relies on the IT environment to provide Cryptographic Support. The TSF integrates with and relies upon cryptographic support to protect all data sent between the external users and the IS, mutually authenticate TOE Components, and optionally authenticate external users.

The basic building blocks of an integration solution are documents (data) and steps (services or workflows that carry out work) that an Authorized Administrator can assemble into multi-step processes called flows.

Documents are objects that the webMethods Fabric uses to encapsulate and exchange data among the participating resources and systems in an integration solution. Documents provide the data on which services and workflows operate. In a general sense, a document represents the body of data that a resource passes to the webMethods Fabric (or vice versa). For

example, a service that performs a database query requires a 'document' that contains a query statement. This service would produce a 'document' containing the results of the query.

Documents that the webMethods Fabric handles are associated with a document type. A document type is a named schema-like definition that describes the structure of a particular kind of document. The webMethods Fabric uses document types to validate instances of documents at run time, determine how a document is routed, and for internal access control measures.

Services are method-like units of logic that operate on documents. They are executed on the Integration Server. An Authorized Administrator builds services to carry out work such as extracting data from documents, interacting with back-end resources (through adapters), and publishing documents to the Broker (through Broker Clients). The Integration Server is installed with an extensive library of built-in services for performing common integration tasks. Services may be integrated using the webMethods Flow Language.

2.3 webMethods Fabric Components

The webMethods Fabric is made up of components to design, execute, and manage integration solutions. Components fall into three basic categories: run-time components, design-time components, and administrative interfaces. TOE components are identified in Table 2-1. Excluded components are listed in Table 2-2.

Table 2-1 TOE Components

Component Type	Description
Run-Time	These components execute integration solutions. The TOE run-time components are: • webMethods Integration Server • webMethods Broker • webMethods Adapter (JDBC and JMS)
Design-Time	These components provide tools for developing and testing integration solutions. The TOE design-time component is: • webMethods Developer
Administrative and Monitoring Interface	Only the administrative interface of the TOE components is included in the TOE. Explicitly, no distinct administrative component is included in the evaluated configuration. Rather the TOE will be administered by pointing a browser directly at an Integration Server administrative interface. A specific package[1] on the IS will provide access to the Broker's administrative interface.

2.3.1 webMethods Integration Server

webMethods Integration Server (IS) is the platform's central run-time component and the primary engine for the execution of integration logic. It is the main entry point for the systems and applications. An extensive library of built-in services installed with the IS.

[1] A Java package is a group of types. A type is either a class or an interface.

webMethods Integration Server plays the following key roles:

- **Hosts adapters.** The Integration Server hosts adapters, which are special modules that link the back-end resources to the webMethods Fabric. Adapters interact directly with the integrated applications and systems. A single Integration Server can host zero or more adapters. The TOE includes two adapters, one for JDBC (to Oracle, Microsoft SQL Server, or other database servers) and one for JMS.
- **Serves as a business-to-business gateway.** The Integration Server is the external interface between the webMethods Fabric and systems outside the enterprise. It provides the underlying support for transporting and encoding business documents using the open standards of the Internet. In the evaluated configuration, the TOE supports the application protocols HTTP, HTTPS, FTP, and FTPS. However, in general, it may host modules that provide support for e-commerce standards such as EDI, RosettaNet, and ebXML.
- **Executes integration logic.** Integration logic is housed in units called services. The Integration Server performs the work of retrieving data from one resource and delivering it to another by executing integration logic. Services are also used to perform administrative functions, such as adding a new user or configuring a new HTTP port. Services, other than those built in to the Integration Server, are not part of the TOE.
- **Access Control Mechanisms -** The Integration Server provides access control mechanisms, which collectively restrict access to services, folders, and ports based on the groups to which users belong, their IP address and their credentials. The access control mechanisms are implemented both on the TCP port the user accesses the TOE through and the resource the user is attempting to access. Access to a port is controlled both through the presumed IP address and the Integration Server service. Access to the Integration Server resources is controlled at the group level.
- **webMethods Administrator Interface -** An Authorized Administrator is able to point an HTML-based browser at any Integration Server in order to manage the server and its functions remotely. The webMethods Administrator interface is used by authorized users to:
    - Maintain the Client Certificate store.
    - Create the Integration Server users' accounts (and passwords), user groups and access control lists of user groups. User accounts are assigned to the appropriate groups by the Authorized Administrator.
    - Set IP connection level security. The Authorized Administrator may define a list of IP addresses and domain names that may access the TOE through a configured port. IP addresses and domain names may contain wildcards.
    - Set service connection level security. The Authorized Administrator may define a list of services that are permitted for a given configured port. Services may contain wildcards. This capability does not extend to other Integration Server resources such as server pages.
    - Set service level security. The Authorized Administrator may define a list of users that are permitted access to an installed service.
    - Access the webMethods Broker Administrator interface through the IS if the Broker Administrator package has been installed on the IS.

2.3.2 webMethods Broker

The role of a Broker is to route documents between information producers (publishers) and information consumers (subscribers). In the webMethods Fabric evaluated configuration, Integration Servers function as both publishers and subscribers. They interact with the Broker using a proprietary protocol. In the evaluated configuration, the webMethods Broker is considered to be an internal TOE component having only one external interface: an Authorized Administrator accessing the Broker components configuration files through the IT environment host OS.

webMethods Broker is a high-speed message router. It is the primary component of what is generally referred to as the platform's "message backbone" or "message facility." Along with supporting features provided by the other components, webMethods Broker facilitates asynchronous, message-based solutions using the publish-and-subscribe model.

The publish-and-subscribe model is a specific type of message-based solution in which resources exchange messages (carrying documents) anonymously through a message broker. Under this model, applications that produce information make that information available in specific types of documents that they publish to a Broker. Applications that require information subscribe to the specific types of documents that they need.

The Broker maintains a list of subscribers that are interested in receiving certain types of documents. When a component publishes a document, the Broker queues the document for the subscribers of that particular document type. When a subscriber receives a document from its queue, an action is triggered on the subscriber's system that processes the document.

The evaluated configuration contains a single Broker only. However, a webMethods Fabric can contain multiple Brokers that operate in groups called territories.

The webMethods Broker Server Administrative interface (accessible in the evaluated configuration through the Integration Server) exposes the following security functionality:

- Permit or Deny based on the Client Certificate store Distinguished names.
- Guaranteed transactions - ensures that a document reaches its subscribers queue or it is never published (included in the TOE, but not claimed as security relevant).
- Provides Publish and Subscribe functionality across multiple server instances. (when multiple resources need to obtain information from the same transaction).
- Filter transactions, based on fields within the document (included in the TOE, but not claimed as security relevant).

In the evaluated configuration, the Broker has limited security functionality.

2.3.3 webMethods Adapters

Adapters run on the Integration Server. They connect the back-end resources in an enterprise (for example, a customer database, a Human Resources application, an inventory system) to the webMethods Fabric. The nature of the connection created through an adapter is that the Integration Server plays the role of client and initiates all transactions with the back-end resource.

The adapter handles the low-level work of connecting to the resource, managing communications, encoding and decoding data, and invoking processes via the resource's API, thus allowing the incorporation of a resource in an integration solution without having to build

complex custom code or understand the low-level details of the resource or its application protocol.

An Integration Server requires an adapter for each type of resource with which it interacts. While webMethods provides adapters for a broad range of databases and business systems, the TOE contains adapters for JDBC and JMS only.

2.3.4 webMethods Developer

webMethods Developer is a graphical development tool that an Authorized Administrator uses to build, edit, and test integration logic. It provides an integrated development environment in which to develop the logic and supporting elements that carry out the work of an integration solution. It also provides tools for testing and debugging solutions.

webMethods Developer lets an Authorized Administrator rapidly construct integration logic with an implementation language called the webMethods Flow Language, or flow. Flow provides a set of simple but powerful constructs that an Authorized Administrator uses to specify a sequence of actions (steps) that the Integration Server will execute at run time. Coupled with the Developer's graphical user interface and its drag-and-drop data mapping capability, flow allows an Authorized Administrator to quickly develop integration logic without low-level coding.

Although an Authorized Administrator can implement most integration solutions using flow and the Integration Server's library of built-in services, an Authorized Administrator can also build services using Java, C/C++, or Visual Basic (or any other COM/DCOM-based component) if the solution requires specialized integration logic that the webMethods Fabric does not provide out of the box. User built services are not included in the TOE. Only flow will be used in testing the TOE.

Developer also provides specialized editors for creating logic for specific run-time components and facilities. For example, it supplies the tools an Authorized Administrator uses to create and manage:

- **Adapter services —** Services that invoke specific processes on a back-end resource (for example, query an customer database, post a journal entry to a general ledger application, or delete an item from an inventory system).
- **Adapter notifications —** Alerts that are issued by back-end systems and for which the webMethods Fabric initiates an action after polling.
- **Web service connectors —** Proxy services that allow the webMethods Fabric to invoke Web services located on remote servers (not included in the TOE, except as required to support the Reverse Invoke Server (see Figure 2-4)).
- **Adding Groups to ACL —** The Developer is responsible for assigning ACL to the appropriate resource or resource folder.

The TOE test configuration will include adapter services and adapter notifications created with the supplied tools for the JDBC adapter and the JMS adapter.

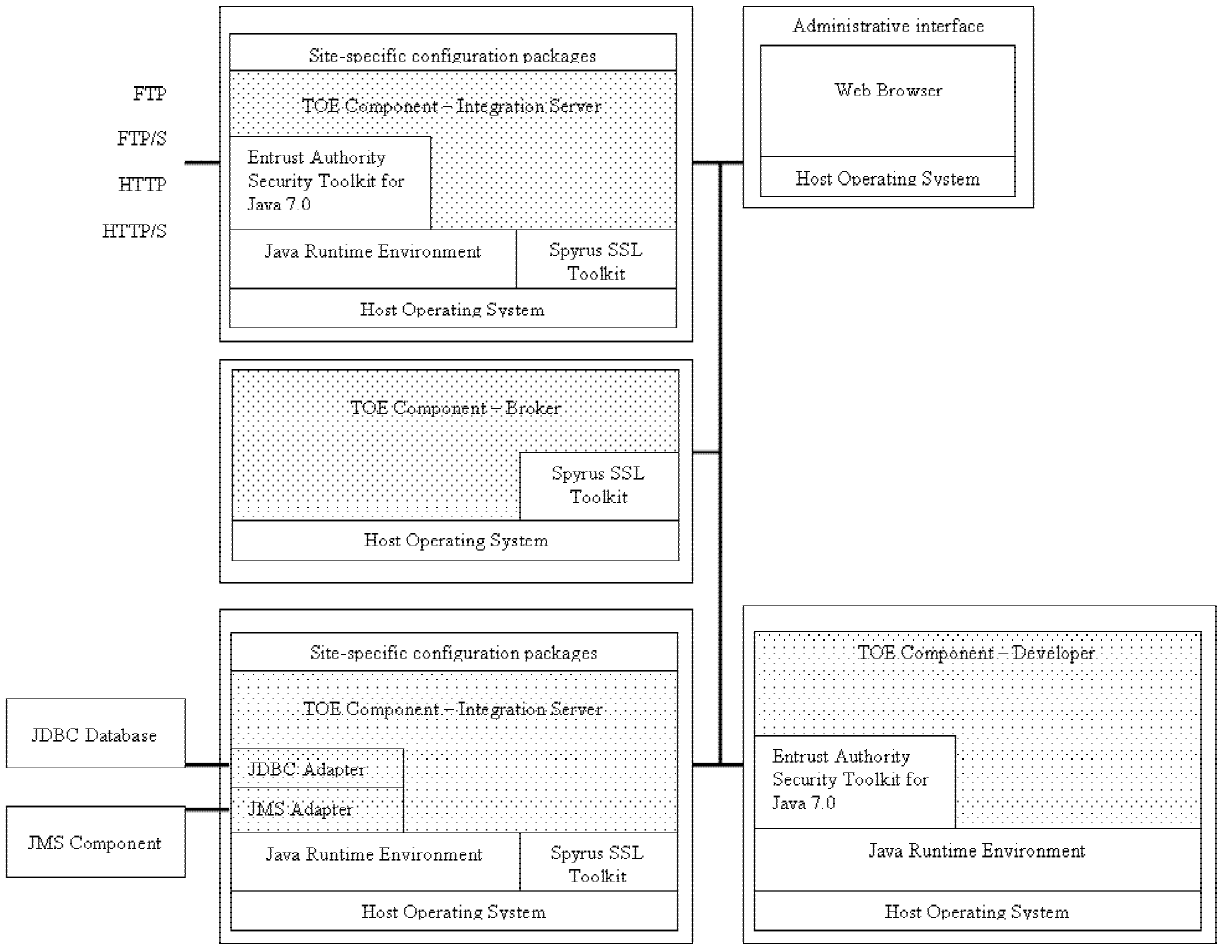

Figure 2-1 – webMethods Architecture Diagram

2.4 TSF Physical Boundary and Scope of the Evaluation

Only the shaded parts of Figure 2-1 are included in the TOE. The TOE includes the following:

- Integration Server
- Broker
- Developer
- Adapters (selected JDBC and JMS)[2]

The evaluated configuration will be tested on the following platforms:

- Two instances of webMethods Integration Server 6.5 and Adapters running on a two separate machines;
- webMethods Broker 6.5 running on a separate machine;

For purposes of testing, all three machines will be Intel or compatible hardware running a MS Windows Operating System (OS). The Integration Server systems will use the IBM 1.3.1 JRE.

[2] Adapters rely on 3rd party software (drivers) which is not part of the TOE.

The webMethods Fabric is based on a distributed architecture, which allows the platform to grow. Because the TOE relies on the IT environment to provide secure communications (OE.ProtectComm), and it is not possible to predict which IT environment the end-user will require, several TOE configurations are discussed.

The evaluated configuration will cover configurations with:

1) Two Integration Servers and a Broker with users physically segregated from the webMethods Fabric platform and its associated servers (Figure 2-2). In this diagram, only the left most Integration Server has an interface to external users, the Broker and right most Integration Server are on a protected LAN.
2) Two Integration Servers and a Broker with users of the webMethods Fabric and its associated servers procedurally segregated (e.g., through encryption or trust) from the TOE (Figure 2-3). In this diagram the Broker and both Integration Servers are externally visible. For this configuration, the IT environment would have to protect network connections.
3) Two Integration Servers and a Broker running procedurally segregated users from behind a VPN (Figure 2-4). The Reverse Invoke server shown in this diagram is not part of the TOE. In this configuration, the TOE is accessed through a firewall and may not be directly connected to by external users.

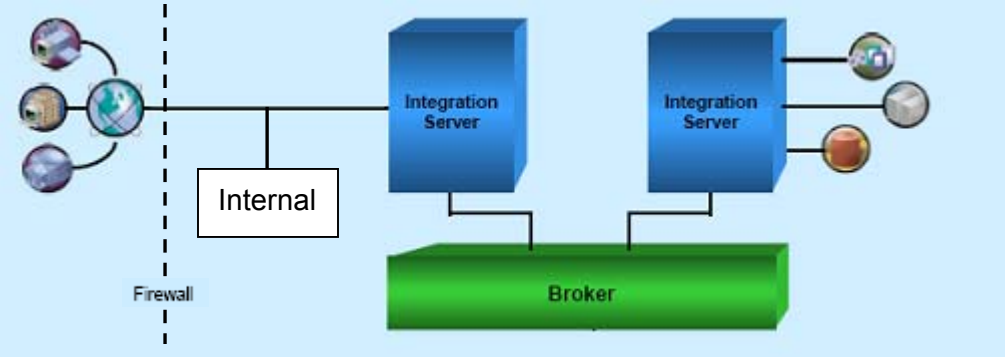

Figure 2-2 – Basic TOE Configuration with Physical Separation

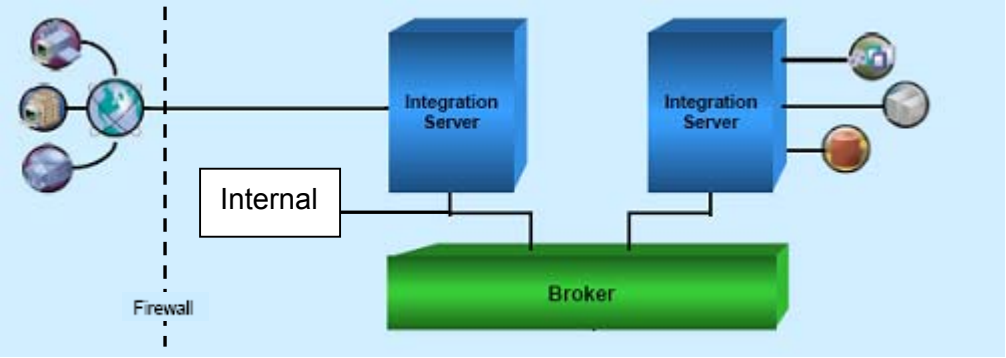

Figure 2-3 – Basic TOE Configuration with Procedural Separation

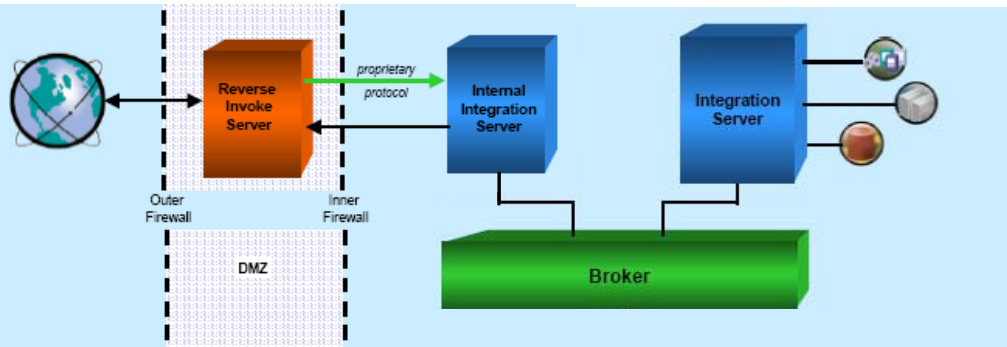

Figure 2-4 – Reverse Invoke TOE Configuration

2.5 Logical Scope and Boundary

webMethods Fabric provides the following security features:

- **Security audit** - webMethods Fabric provides the ability to audit the events. webMethods Fabric provides the ability for an Authorized Administrator to read the audit logs.
- **User data protection** - webMethods Fabric enforces a information flow control policy to control access to services and documents. The webMethods IS access control policy is based on user roles and groups.
- **Identification and Authentication** - The TOE supports user identification and authentication through the use of user accounts and passwords. webMethods Fabric also integrates with the certificate-based authentication of external users provide by the IT environment.
- **Security Management** – An administrator can authenticate with the TOE and monitor and manage the interaction between end-users and TOE components. Two administrative interfaces are provided, one for the Integration Server and one for the Broker, but both interfaces are accessed by connection to the Integration Server.
- **Partial protection of TSF (TOE)** - webMethods Fabric protects its programs and data from unauthorized access through its own interfaces. The TSF ensures that all information that flows through it must flow through the policy enforcement mechanisms.

2.6 TOE Exclusions:

- Third party relational database (Oracle, SQL Server, or DB2 via JDBC, JMS providers)
- The interface (drivers) of the third party relational database
- Transport standard HTTP, HTTPS, FTP and FTPS implementations
- Message format MIME and S/MIME implementations
- Underlying operating system (OS) software and hardware
- Java Virtual Machine (JVM)
- Third-party encryption software that is used to provide a trusted communication path between external users and the TOE;
- Third-party encryption software that is used between TOE Components and the web browser that is used to connect to the Integration Server for administration purposes,

- Web browser that is used to connect to the Integration Server for administration purposes
- Clustered configurations
- Territories
- Reverse Invoke Server

LDAP and NIS servers are explicitly excluded from the evaluated configuration.

Product components that are not within the scope of the evaluation are listed in Table 2-2. Packages corresponding to these components are not loaded in the evaluated configuration.

Table 2-2 Product Components not Included in the TOE

Component Type	Description
Run-Time	Additional product run-time components include webMethods Mainframe webMethods Trading Networks and eStandard Modules webMethods Workflow
Design-Time	Additional product design-time components include webMethods Modeler webMethods Workflow Designer webMethods Trading Networks Console
Administrative and Monitoring	Additional product administrative components include webMethods Administrator (Administrative Server) webMethods Monitor webMethods Manager

Cryptographic Support - All network communications between webMethods Fabric components are encrypted. Since third party software is used to provide confidentiality, the encryption functions are not part of the TOE. The TOE relies on the IT environment to provide cryptographic support. The TOE security environment can be categorized as follows:

- Entrust Authority Security Toolkit for Java 7.0 provides cryptographic services for external users connecting to the IS and for the Developer to connect to the IS.
- Spyrus SSL Toolkit provides cryptographic services connections between TOE components where one of the components is a Broker Server (i.e., between Broker and Integration Server), or when the Authorized Administrator connects to the TOE remotely. This provides encryption to prevent disclosure of TSF data.

webMethods Fabric relies upon support from the IT environment for:

- Reliable time stamps
- Support of Domain separation from the host OS
- Support of Non-bypassability from the host OS
- Protection of the webMethods Fabric hosts from other interference or tampering.

- protection of data transfer between TOE components and for the trusted communication path between authorized account holders and the TOE (encryption software)
- for the web browser functionality needed for the GUI,
- for secure TSF data storage (specifically the TOE relies on the operating system to prevent an attacker from accessing TSF data through the operating system interfaces.)
- access to TOE configuration files.

3 TOE Security Environment

3.1 Introduction

This chapter identifies the following:

- Significant assumptions about the TOE's operational environment.
- IT related threats to the organization countered by the TOE.

Using the above listing, this chapter identifies threats (T), assumptions (A).

3.2 Assumptions

This section contains assumptions regarding the security environment and the intended usage of the TOE. The specific conditions listed in the following subsections are assumed to exist in the TOE environment. These assumptions include both practical realities in the development of the TOE security requirements and the essential environmental conditions on the use of the TOE.

Table 3-1 Assumptions

Identifier	Assumption
A.Admin	The administrator is trusted to correctly configure and operate the TOE according to the instructions provided by the TOE documentation.
A.Manage	It is assumed that one or more administrators are assigned who are competent to manage the TOE and the security of the information it contains, and who can be trusted not to deliberately abuse their privileges so as to undermine security.
A.NoUntrusted	It is assumed that there will be no untrusted software on the webMethods Integration Server and Broker.
A.Physical	The TOE components critical to the security policy enforcement will be protected from unauthorized physical modification.
A.Users	It is assumed that users will protect their authentication data.
A.IT	The TOE relies upon the IT environment to support protected communications, provide audit file protection, support partial domain separation, support non-bypassability, provide reliable time-stamps, and to perform user authentication when configured to do so.

3.3 Threats

Table 3-2 identifies the threats to the TOE. A strength of function level of SOF-basic, consistent with a CC evaluated level of assurance of EAL2, counters an attack level of low; the threat agents are users with public knowledge of how the TOE operates, with access to only standard equipment. Since the TOE includes a public interface, an attacker is considered to have access to the TOE. Mitigation to the threats is through the objectives identified in Section 4, Security Objectives.

Table 3-2 Threats

Identifier	Threat
T.Abuse	An undetected compromise of the TOE may occur as a result of an authorized user of the TOE (intentionally or otherwise) performing actions the individual is authorized to perform.
T.Access	An authorized user of the TOE may access information or resources without having permission from the person who owns, or is responsible for, the information or resource.
T.Bypass	An unauthorised user may attempt to bypass the information flow control policy.
T.Intercept	An unauthorized person on an internal network that connects TOE components may intercept communications between the TOE components and attempt to access and/or modify the data being transmitted.
T.Mismanage	Authorized Administrators may make errors in the management of security functions and TSF data. Administrative errors may allow attackers to gain unauthorized access to resources protected by the TOE.
T.Tamper	An attacker may attempt to modify TSF programs and data.
T.Transmit	TSF data may be disclosed or modified by an attacker while being transmitted between the TOE and its users.
T.Undetect	Attempts by an attacker to violate the security policy may go undetected. If the attacker is successful, TSF data may be lost or altered.

4 Security Objectives

4.1 Security Objectives for the TOE

The security objectives for the TOE are as follows:

Table 4-1 TOE Security Objectives

Identifier	Objective
O.Access	The TOE must allow authorized users to access only appropriate TOE functions and data.
O.Admin	The TOE must provide the functionality to enable authorized user(s) to effectively manage the TOE and its security functions.
O.Attributes	The TOE must be able to maintain user security attributes.
O.Audit	The TOE must provide a means to record a readable audit trail of security-related events, with accurate dates and times.
O.AuditProtection	The TOE must provide the capability to protect audit information.
O.AuditReview	The TOE must provide the functionality to enable authorized user(s) to review the audit logs.
O.IDAuth	The TOE must be able to identify and authenticate the users prior to allowing access to TOE functionality.
O.NonBypass	The TOE must ensure the TOE's security functional policy is invoked and succeeds before allowing another TOE function to proceed.
O.PartialSelfProtection	The TSF will maintain a domain for its own execution that protects itself and its resources from external interference, tampering or unauthorized disclosure, through its own interfaces.
O.RemoteAdministration	An identified and authorized remote administrator may manage identified TSF entities from a remote client.

4.2 Security Objectives for the Environment

The security objectives for the IT environment are as follows:

Table 4-2 Security Objectives for the IT Environment

Identifier	Objective
OE.Audit	The IT environment must provide a means to record a readable audit trail of security-related events, with accurate dates and times.
OE.AuditProtection	The IT environment will provide the capability to protect audit information.
OE.IDAuth[3]	The IT environment must be able to identify and authenticate users prior to allowing access to IT environment functions and data.
OE.NonBypass	The IT environment must ensure the IT environment's security functional policy is invoked and succeeds before allowing another IT environment function to proceed.
OE.PartialSelfProtection	The IT Environment will maintain a domain for its own execution that protects itself and its resources from external interference, tampering, or unauthorized disclosure, through its own interfaces.

[3] This applies only to Administrators who log into the IT environment directly through the system console to perform administrative functions related to the product

Identifier	Objective
OE.ProtectComm[4]	The IT environment must protect communications between the TOE and its users, and between TOE components.
OE.Time	The underlying operating system must provide reliable time stamps.

4.2.1 Security Objectives for Non-IT Security Environment

The Non-IT security objectives are as follows:

Table 4-3 Security Objectives for Non-IT Security Environment

Identifier	Objective
ON.Install	Those responsible for the TOE must ensure that the TOE is delivered and installed in a manner that maintains IT security.
ON.NoUntrusted	The administrator must ensure that there is no untrusted software on the webMethods Integration Server and Broker.
ON.Operations	The TOE will be managed and operated in a secure manner as outlined in the supplied guidance.
ON.Person	Personnel working as authorized administrators shall be carefully selected and trained for proper operation of the system.
ON.Physical	Those responsible for the TOE must ensure that those parts of the TOE critical to the security policy are protected from any physical attack.
ON.ProtectAuth	Users must ensure that their authentication data is held securely and not disclosed to unauthorized persons.

[4] Application Note: This objective may be met by the IT environment in a number of ways: The IS server may be behind a firewall that implements a VPN through which external users communicate; The IS server could require client to connect using HTTPS or FTPS; In a closed environment communication could be protected via physical measures.

5 IT Security Requirements

This section provides the TOE security functional and assurance requirements. In addition, the IT environment security functional requirements on which the TOE relies are described. These requirements consist of functional components from Part 2 of the CC, assurance components from Part 3 of the CC, and CCIMB Final Interpretations.

5.1 Formatting Conventions

The notation, formatting, and conventions used in this security target (ST) are consistent with version 2.2 of the Common Criteria for Information Technology Security Evaluation. Font style and clarifying information conventions were developed to aid the reader.

The CC permits four functional component operations: assignment, iteration, refinement, and selection to be performed on functional requirements. These operations are defined in Common Criteria, Part 1, Section 4.4.1.3.2 as:

- assignment: allows the specification of an identified parameter in a component;
- refinement: allows the addition of details or the narrowing of requirements;
- selection: allows the specification of one or more elements from a list; and
- iteration: allows a component to be used more than once with varying operations.

This ST indicates which text is affected by each of these operations in the following manner:

- *Assignments* and *Selections* specified by the ST author are in **[*italicized bold text*]**.
- *Refinements* are identified with "**Refinement:**" right after the short name. Additions to the CC text are specified in ***italicized bold and underlined text***.
- *Iterations* are identified with a dash number "**-#**". These follow the short family name and allow components to be used more than once with varying operations. "*" refers to all iterations of a component. For clear identification purposes, the base component id is used and a "**: additional information**" is after the component title. For example, **FPT_SEP.1-1 TSF domain separation: TOE**.
- *Application notes* provide additional information for the reader, but do not specify requirements. Application notes are denoted by *italicized text.*
- *NIAP and CCIMB Interpretations* have been reviewed. Relevant Interpretations are included and are noted in Interpretation Notes. Interpretation Notes are denoted by *italicized text.* The original CC text modified by the interpretation is not denoted nor explained.
- *Explicit SFR* are identified by appending _EXP to the SFR name.
- *Comments* are provided as an aid to the reader and evaluation team. These items will be deleted in the final version of the ST.

5.2 TOE Security Functional Requirements

The functional security requirements for the TOE consist of the following components derived from Part 2 of the CC, CC interpretations, explicitly stated SFRs derived from the CC Part 2, and NIAP interpretations as appropriate, summarized in the Table 5-1 below. This section contains the TOE Functional components for the webMethods TOE. The TOE claims a minimum strength level for the TOE security functions of SOF-basic.

Table 5-1 Functional Components

Component	Component Name
FAU_GEN.1	Audit data generation
FAU_SAR.1	Audit review
FAU_SAR.2	Restricted audit review
FAU_SAR.3	Selectable audit review
FAU_SEL.1	Selective audit
FDP_ACC.1	Subset access control
FDP_ACF.1	Security attribute based access control
FIA_ATD.1	User attribute definition
FIA_SOS.1	Verification of secrets
FIA_UAU.5-1	Multiple authentication mechanisms
FMT_MOF.1	Management of Security Functions Behavior
FMT_MSA.1	Management of security attributes
FMT_MSA.3	Static attribute initialisation
FMT_MTD.1	Management of TSF data
FMT_SMF.1	Specification of management functions
FMT_SMR.1	Security roles
FPT_RVM_EXP.1-1	Non-bypassability of the TSP: TOE
FPT_SEP_EXP.1-1	TSF domain separation: TOE

5.2.1 Class FAU: Security Audit

FAU_GEN.1 Audit data generation

Hierarchical to: No other components.

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) **[*the following auditable events:***

- ***Reading of information from the audit records***
- ***Unsuccessful attempts to read information from the audit records***
- ***Modification of the audit configuration that occur while the audit collection functions are operating***
- ***All requests to perform an operation on a package, folder, service, flow service, specification, schema, document type, or trigger***
- ***Rejection by the TSF of any tested secret***
- ***Modification of the behaviour of the functions in the TSF***
- ***Modification of the values of security attributes***
- ***Modification of the default setting of permissive or restrictive rules***
- ***Modification of the initial values of security attributes***

- ***All modification of the values of TSF data***
- ***Use of the management functions***
- ***Modification of the group of users that are part of a role***

]

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: **[*none*]**.

Dependencies: FPT_STM.1 Reliable time stamps

FAU_SAR.1 Audit review

Hierarchical to: No other components.

FAU_SAR.1.1 The TSF shall provide **[*an Authorized Administrator*]** with the capability to read **[*all audit data*]** from the audit records.

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

Dependencies: FAU_GEN.1 Audit data generation

FAU_SAR.2 Restricted audit review

Hierarchical to: No other components.

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

Dependencies: FAU_SAR.1 Audit review

FAU_SAR.3 Selectable audit review

Hierarchical to: No other components.

FAU_SAR.3.1 The TSF shall provide the ability to perform **[*searches*, *ordering*]** of audit data based on **[*the number of audit log entries to be displayed, oldest to newest starting from the beginning, newest to oldest starting from the end*].**

Dependencies: FAU_SAR.1 Audit review

FAU_SEL.1 Selective audit

Hierarchical to: No other components.

FAU_SEL..1.1 The TSF shall be able to include or exclude auditable events from the set of audited events based on the following attributes:

a) **[*level of logging*]**.

Dependencies: FAU_GEN.1 Audit data generation
FMT_MTD.1 Management of TSF data

5.2.2 Class FDP: User Data Protection

FDP_ACC.1 Subset access control

Hierarchical to: No other components.

FDP_ACC.1.1 The TSF shall enforce the **[*webMethods IS Access Control SFP*]** on **[*subjects:***

connections on behalf of external users,
triggers on behalf of adapters

objects:

(port, IS element) pairs,
IS element,
where an IS element is one of: package, folder, service, flow service, specification, schema, document type, or trigger

***operations among subjects and objects covered by the SFP: execute, list, read, and write*.]**

Application Note: This describes the access control mechanism that constrains an IS element (object) to be accessed only via connections to specified ports and for a given operation, without reference to subject security attributes (such as UID). The SFR identified two subjects of this policy. An "external user" is a user that accesses the TOE through a network connection. An adapter is a TOE component that provides a notification when a message arrives from a specific monitored destination. If the notification is configured to publish a document, an IS trigger can monitor for that document and invoke a flow or Java service registered with the trigger. When a trigger associated with an adapter invokes a flow or Java service, it does so with the privilege associated with a configured default UID.

Dependencies: FDP_ACF.1 Security attribute based access control

FDP_ACF.1 Security attribute based access control

Hierarchical to: No other components.

FDP_ACF.1.1 The TSF shall enforce the **[*webMethods IS Access Control SFP*]** to objects based on the following: **[**

Subjects: connections on behalf of external users,
triggers on behalf of adapters:

Subject security attributes:

1. ***Default UID for Triggered objects***
2. ***Real UID of an external user***
3. ***Group Memberships***
4. ***Presumed IP address***
5. ***Port***

Objects: (port, IS element) pairs

Object security attributes:

1. ***set of IP addresses allowed for the port,***

2. ***set of services allowed for the port,***

3. ***Access control list (ACL) on IS element.***

Objects: IS element

Object security attributes:

Access control list (ACL) on IS element].

Application Note: If a user is permitted to invoke an object thru a given port (see FDP_ACC.1), then the access control mechanism constrains an IS element (object) to be accessed for a given subject's security attributes and object security attributes (IP address, service, and ACL).

FDP_ACF.1.2 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: **[**

A. A connection subject may access an object provided:

1. ***Presumed IP address of the subject is in the set of IP addresses allowed for the port, and***
2. ***When the IS element is a service, the service is in the set of services allowed for the Listening Port, and***
3. ***A Group Membership associated with the UID is allowed by the ACL for the IS element, and***
4. ***The operation is permitted by the ACL for the Group Membership on the IS element*]**

B. A trigger subject may access an object provided:

1. ***Default UID for Triggered Services or event has a Group Membership that is allowed by the ACL for the IS element***
2. ***The execute operation is permitted by the ACL for the Group Membership on the IS element*]**

FDP_ACF.1.3 The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: **[*None*]**

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

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

5.2.3 Class FIA: Identification and Authentication

FIA_ATD.1 User attribute definition

Hierarchical to: No other components.

FIA_ATD.1.1 The TSF shall maintain the following list of security attributes belonging to individual users:

- **[*User Name,***
- ***Group Memberships***

- ***Password***
- ***X.509 certificate attributes***].

Dependencies: No dependencies.

FIA_SOS.1 Verification of secrets

Hierarchical to: No other components.

FIA_SOS.1.1 The TSF shall provide a mechanism to verify that secrets meet **[*the rules of the password policy*]**

Dependencies: No dependencies.

FIA_UAU.5-1 Multiple authentication mechanisms

Hierarchical to: No other components.

FIA_UAU.5.1-1 –The TSF shall provide [***password based authentication mechanisms***] to support user authentication.

FIA_UAU.5.2-1 – The TSF shall authenticate any user's claimed identity according to the **[*following multiple authentication mechanism rules:***

a) Reusable password in combination with Server Certificate based authentication mechanism provided by the IT environment shall be used for unauthenticated users sending information to or receiving information from TOE using HTTPS or FTPS such that successful authentication must be achieved before allowing any other TSF-mediated actions on behalf of that user;

b) Reusable password based authentication mechanism shall be used for unauthenticated users sending or receiving information to the TOE through a external VPN gateway using HTTP or FTP such that successful authentication must be achieved before allowing any other TSF-mediated actions on behalf of that human user; **]**

Dependencies: No dependencies

Application Note: Case a) applies to connections made by an external user (or another IS external to the TOE) to the IS where the IT environment did not authenticate the client with certificate based authentication. Case b) applies to clients connecting to the IS through a VPN tunnel.

5.2.4 Class FMT: Security Management (FMT)

FMT_MOF.1 Management of Security Functions Behavior

Hierarchical to: No other components.

FMT_MOF.1.1 The TSF shall restrict the ability to **[*enable, disable*, *determine the behavior of*]** the functions **[*the Operation in Table 5-2*]** to **[*the Authorized Administrator*]*.***

Dependencies: FMT_SMR.1 Security roles
FMT_SMF.1 Specification of Management Functions (CCIMB 065)

Table 5-2 IS Management of Security Functions Behavior

Security attribute	Operation	Nominal Authorized Role[5]
ACL option setting	which direct the server to check or ignore a service's ACL when the service is internally invoked	Developers
element	triggers by enabling or disabling them	Administrators and Developers (Internal ACL)
events	Of services by configuring and scheduling them to run in response to events	Developers
package	assign dependencies of packages	Developers
package	To: reload, enable ,disable, delete, recover and archive a package	Administrators
package	To assign, startup, shutdown replication services and to perform replication of packages	Replicator
package.	To: reload, delete and copy a package	Developers
password	To enable non administrative users to change their passwords	Administrators
scheduled user task	To: Cancel, temporarily suspend, view a list of and update the scheduling options of a scheduled user task	Administrators
server	To connect to a server	Administrators
server	To connect webMethods Developer to the server	Developers
triggers	To configure the user account used to execute services invoked by triggers	Administrators
user	To specify how the server authenticates users	Administrators

FMT_MSA.1 Management of security attributes

Hierarchical to: No other components.

FMT_MSA.1.1 The TSF shall enforce the **[*webMethods IS Access Control SFP*]** to restrict the ability to **[*query, modify, delete,* [*other operations as specified in* Table 5-3**]]** the security attributes **[*as specified in* Table 5-3**]** to **[*the Authorized Administrator*].**

Dependencies: FDP_ACC.1 Subset access control or FDP_IFC.1 Subset information flow control]
FMT_SMR.1 Security roles
FMT_SMF.1 Specification of Management Functions (CCIMB 065)

Table 5-3 IS Management of Security Attributes

Operation	Security attribute	Nominal Identified Role[6]
Associate and disassociate with IS elements an	ACL	Administrators or Developers
Query and edit access specification associated with	ACL.	Developers

[5] The various Administrative role nominally described in the product documentation (Administrator, Broker Administrator, Developer, Replicator) are all treated as a single Administrative role.
[6] The various Administrative role nominally described in the product documentation (Administrator, Broker Administrator, Developer, Replicator) are all treated as a single Administrative role.

Operation	Security attribute	Nominal Identified Role[6]
Query and Edit groups in	ACL.	Administrators
Create and delete ACL that control access to	elements	Administrators
Create, remove, query and edit UID in a	Group	Administrators
Query and edit access mode and services for	listening ports.	Administrators
Query and edit IP access settings for	listening ports.	Administrators
Assign passwords to, create, list, enable, disable and delete	UIDs.	Administrators

FMT_MSA.3 Static attribute initialisation

Hierarchical to: No other components.

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

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

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

FMT_MTD.1-1 Management of TSF data

Hierarchical to: No other components.

FMT_MTD.1.1-1 The TSF shall restrict the ability to **[*query, modify, delete,* [*and other operations as specified in* Table 5-4**]]** the **[*TSF Data as specified in* Table 5-4**]** to **[*the Authorized Administrator*].**

Table 5-4 IS Management of TSF Data

Operation	TSF Data	Nominal Identified Role[7]
Query and edit primary setting of	listening port	Administrators
Configure, associate specific protocol, add, and enable	listening ports	Administrators
Create, assign version numbers	package	Developers
Configure, associate specific directory, add, remove, enable, and disable	file polling ports	Administrators
Change the	primary port	Administrators
Configure, enable, and disable	proxy ports	Administrators
Configure, enable, and disable	registration ports	Administrators
edit	trigger settings.	Developers
Set	Audit level	Administrators
Create, move and delete	Audit logs	Administrators
Create	Document types	Administrators
View	Server Information	Administrators

[7] The various nominative Administrative roles described in the product documentation (Administrator, Broker Administrator, Developer, Replicator) are all treated as a single Administrative role.

Dependencies: FMT_SMR.1 Security roles
FMT_SMF.1 Specification of management functions

FMT_SMF.1 Specification of management functions

Hierarchical to: No other components.

FMT_SMF.1.1 The TSF shall be capable of performing the following security management functions: **[**

- ***Manage ACL,***
- ***Manage Audit logs,***
- ***Manage Sessions,***
- ***Manage documents,***
- ***Manage packages.***
- ***Install folder, service, flow service, specification, document type, and trigger,***
- ***Assign ACL to folder, service, flow service, specification, and document type***
- ***Create document types,***
- ***Manage events,***
- ***Manage listening ports,***
- ***Manage polling ports,***
- ***Manage proxy ports,***
- ***User management,***
- ***Manage user tasks,***
- ***Manage triggers,***
- ***Connect to servers,***
- ***View server information*****]**

Dependencies: No Dependencies

FMT_SMR.1 Security roles

Hierarchical to: No other components.

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.

Dependencies: FIA_UID.1 Timing of identification

Application note: The various Administrative roles described in the product documentation (Administrator, Broker Administrator, Developer, Replicator) are all treated as a single security Administrative role.

5.2.5 Class FPT: Protection of the TOE Security Functions

FPT_RVM_EXP.1-1 Non-bypassability of the TSP: TOE

Hierarchical to: No other components.

FPT_RVM_EXP.1.1-1 The TSF, when invoked by the underlying IT environment, shall ensure that TSP enforcement functions are invoked and succeed before each function within the TSC is allowed to proceed.

Dependencies: No dependencies.

FPT_SEP_EXP.1-1 TSF domain separation: TOE

Hierarchical to: No other components.

FPT_SEP_ EXP.1.1-1 The TSF, when invoked by the underlying host IT environment, shall maintain a security domain that protects it from interference and tampering by untrusted subjects in the TSC.

FPT_SEP_ EXP.1.2-1 The TSF, when invoked by the underlying host IT environment, shall enforce separation between the security domains of subjects in the TSC.

Dependencies: No dependencies.

5.2.6 Strength of Function

The overall strength of function requirement is SOF-basic. The strength of function requirement applies to FIA_SOS.1. The SOF claim for FIA_SOS.1 is SOF-basic. Strength of the "secrets" mechanism is consistent with the objectives of authenticating users (O.IDAuth). Strength of Function shall be demonstrated for the non-certificate based authentication mechanisms to be SOF-basic, as defined in Part 1 of the CC. Specifically, the local authentication mechanism must demonstrate adequate protection against attackers possessing a low-level attack potential.

5.3 IT Environment Security Assurance Requirements

This section contains the Functional components for the webMethods IT environment. The functional security requirements for the IT consist of the following components derived from Part 2 of the CC, CC interpretations, explicitly stated SFRs derived from the CC Part 2, and NIAP interpretations as appropriate, summarized in the Table 5-5 below.

The webMethods Servers require that the IT environment provide reliable time stamps, client and server identification, and support TSF domain separation. The IT environment may optionally be configured to provided certificate based authentication. Any cryptographic functions are part of the IT environment, not part of the TOE.

A typical IT environment supporting an implementation of this TOE could be:

Broker Platform

SSL/TLS Toolkit
Host Operating System

Developer Platform:

Entrust Authority Security Toolkit for Java 7.0
Java Runtime Environment
Host Operating System

Integration Server Platform with Interface to external user

Entrust Authority Security Toolkit for Java 7.0
Java Runtime Environment
SSL/TLS Toolkit

Host Operating System

Integration Sever Platform without interface to external user

Java Runtime Environment
SSL/TLS Toolkit
Host Operating System

Table 5-5 IT Environment Functional Components

Component	Component Name
FAU_STG.1	Protected audit trail storage
FIA_UAU.5-2	Multiple authentication mechanisms
FMT_MTD.1-2	Management of TSF data
FPT_RVM_EXP.1-2	Non-bypassability of the TSP: IT
FPT_SEP_EXP.1-2	TSF domain separation: IT
FPT_STM.1	Reliable time stamps

5.3.1 Class FAU: Security Audit

FAU_STG.1 Protected audit trail storage

Hierarchical to: No other components.

FAU_STG.1.1 **Refinement:** The ***IT Environment*** shall protect the stored audit records from unauthorised deletion.

FAU_STG.1.2 **Refinement:** The ***IT Environment*** shall be able to [***prevent***] unauthorised modifications to the audit records in the audit trail.

Dependencies: FAU_GEN.1 Audit data generation

5.3.2 Class FIA: Identification and authentication

FIA_UAU.5-2 Multiple authentication mechanisms

Hierarchical to: No other components.

FIA_UAU.5.1-2 – **Refinement:** The ***IT environment*** shall provide [***Certificate based authentication mechanisms***] to support user authentication.

FIA_UAU.5.2-2 – **Refinement:** The ***IT environment*** shall authenticate any user's claimed identity according to the **[*following multiple authentication mechanism rules:***

a) Certificate based authentication mechanism, with Client and Server certificates, shall be used for Broker Clients sending or receiving information such that successful authentication must be achieved before allowing any other TSF-mediated actions on behalf of that Broker Client;

b) Certificate based authentication mechanism, with Client and Server certificates, shall be used for users sending information to or receiving information from TOE using HTTPS or FTPS such that successful authentication must be achieved before allowing any other TSF-mediated actions on behalf of that human user;

c) Authentication of the Administrator with UID and reusable password at the system console such that successful authentication must be achieved before allowing any other TSF-mediated actions on behalf of that human user;

Dependencies: No dependencies

Application Note: The webMethods Server relies on the identification that is provided by the IT environment. The Broker Server require each user to be successfully identified via certificate based authentication. The Integration Server minimally relies upon the IP address as a UID, and may be configured to also require certificate based authentication. The webMethods server will accept the UID supplied by the IT environment. Case a) applies to connections made from the Integration Server to the Broker server. Case b) describes mutual certificate based authentication by the client and server. Case c) describes authentication of the authorized administrator.

5.3.3 Class FMT: Security management

FMT_MTD.1-2 Management of TSF data

Hierarchical to: No other components.

FMT_MTD.1.1-2 **Refinement:** The ***IT environment*** shall restrict the ability to **[*query, modify, delete, create, and other operations as specified in*** Table 5-6**]** the **[*TSF Data as specified in*** Table 5-6**]** to **[*the Authorized Administrator].***

Table 5-6 TOE Management of TSF Data

TSF data	Operation	Identified Role
configuration files	Query and edit	Administrators
".access" files	Query and edit	Administrators

Dependencies: FMT_SMR.1 Security roles
FMT_SMF.1 Specification of management functions

Application Note: The IT environment supports editing of TOE data. Some TOE data can only be configured through the IT environment

FPT_RVM_EXP.1-2 Non-bypassability of the TSP

Hierarchical to: No other components.

FPT_RVM_EXP.1.1-2: The security functions of the IT environment shall ensure that IT environment security policy enforcement functions are invoked and succeed before each function within the scope of control of the IT environment is allowed to proceed.

Dependencies: No dependencies.

FPT_SEP_EXP.1-2 TSF domain separation: IT

Hierarchical to: No other components.

FPT_SEP_EXP.1.1-2 The security functions of the IT environment shall maintain a security domain for its own execution that protects it from interference and tampering by untrusted subjects in the scope and control of the IT environment.

FPT_SEP_EXP.1.2-2 The security functions of the IT environment shall enforce separation between the security domains of subjects in the scope of control of the IT environment.

Dependencies: No dependencies.

FPT_STM.1 Reliable time stamps

Hierarchical to: No other components.

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

Dependencies: No dependencies

5.4 TOE Security Assurance Requirements

Table 5-7 EAL2 Assurance Requirements

Assurance Component ID	Assurance Component Name	Dependencies
ACM_CAP.2	Configuration items	None
ADO_DEL.1	Delivery procedures	None
ADO_IGS.1	Installation, generation, and start-up procedures	AGD_ADM.1
ADV_FSP.1	Informal functional specification	ADV_RCR.1
ADV_HLD.1	Descriptive high-level design	ADV_FSP.1, ADV_RCR.1
ADV_RCR.1	Informal correspondence demonstration	None
AGD_ADM.1	Administrator guidance	ADV_FSP.1
AGD_USR.1	User guidance	ADV_FSP.1
ATE_COV.1	Evidence of coverage	ADV_FSP.1, ATE_FUN.1
ATE_FUN.1	Functional testing	None
ATE_IND.2	Independent testing-sample	ADV_FSP.1, AGD_ADM.1, AGD_USR.1, ATE_FUN.1
AVA_SOF.1	Strength of TOE security function evaluation	ADV_FSP.1, ADV_HLD.1
AVA_VLA.1	Developer vulnerability analysis	ADV_FSP.1, ADV_HLD.1 AGD_ADM.1, AGD_USR.1

6 TOE Summary Specification

6.1 IT Security Functions

6.1.1 Security Audit Function

6.1.1.1 AU-1 Audit trail (FAU_GEN.1)

All administrative events audited by the TOE are mediated by built-in services residing on the IS. In order for information to be recorded to support this SFR, the Service Logging Level must be set to **brief** or **verbose** (*see* Table 6-1).

The Service log provides data about flow services and built-in services that run in Integration Server. Log entries include Timestamp, User, Server Id (The ID can be DNSname:port or IPaddress:port.), Service Name, Status (Started, Retried, Ended, or Failed), Duration, Error Message, and Context information Monitor uses to connect related entries from different logs (Root Context, Parent Context, Current Context).

Within each audited event, webMethods Servers records the following information: event, type of event, subject identity, and the outcome

- Date and time of event,
- Type of event (the event type can be equated to the built-in Service called, for example wm.server.query:getPartialLog is used to read audit data),
- Subject identity,
- Success or failure of event

The Integration Server maintains all logging data. By default, Integration Server stores audit data in flat files, and although the product includes a capability to store certain types of logging data into a database, this capability is not utilized in the TOE. Hence, none of the audit data required for the CC evaluation is *required* to be in a database, but rather is stored in flat files.

The logging level setting determines the level of details and types of messages that are seen in the Service Log. The Service Log contains two main types of information: server actions (e.g., packages loaded & unloaded, start up & shutdown messages) and failures (e.g., access failures, login failures). The Audit Log is similar to a traditional security audit trail, with the exception of failure conditions.

Operationally, the audit logs are named auditYYYYMMDD.log while the service logs are named serverYYYYMMDD.log.

Table 6-1 Service Logging Levels

Service Logging	**Setting**
Disable service logging globally.	**off**
Enable service-by-service logging.	**perSvc**
Have the audit subsystem automatically write start and failure or success log entries for every service every time the service is called, either directly (top-level) or by another service (nested).	**brief**
Have the audit subsystem automatically write start and failure or success log entries and the input pipeline every time the service is called, either directly (top-level) or by another service (nested).	**verbose**

6.1.1.2 AU-2 Audit review (FAU_SAR.1)

From the Integration Server Administrative Interface, an authorized administrator[8] can read the service log data (called Audit Log in the GUI) generated by the TOE. Although there are other types of audit logs supported by the product, only the service log is relevant to the auditing functionality claimed in FAU_GEN.1. Hence, while other logs the IS generates do not impede the secure administration of the TOE, they are not relevant to the secure administration of the TOE either. Although the Developer is able to access services hosted on the IS from the Developer platform, the services are audited on the platform where they are hosted and execute (the IS) and not the platform from which they are invoked (in this example the Developer). Hence all security relevant audit data that might be characterized as "developer logging data" is actually recorded on the IS in the service log. The audit records are provided in tabularized text suitable for the user to interpret the information.

6.1.1.3 AU-3 Restricted audit review (FAU_SAR.2)

The TSF prohibits all users read access to the audit records, except those users that have been granted explicit read-access. Unauthorized users are not able to read the audit records in the audit trail. The ACL check performed on the wm.server.query:getPartialLog service supports this functionality.

6.1.1.4 AU-4 Selectable audit review (FAU_SAR.3)

In the Integration Server, the TSF provides the ability to perform the ordering of audit data based on oldest to newest starting from the beginning and newest to oldest starting from the end. The TSF provides the ability to search the audit data by customizing the number of log entries displayed.

Dependencies: FAU_SAR.1 Audit review

6.1.1.5 AU-5 Selective audit (FAU_SEL.1)

The administrator can view the audit logs by going to the **Logs > Session** page to view session logging data. The administrator can change the order of the entries and the number of entries displayed using the Log display controls area on the log page, and include or exclude auditable events from the set of audited events based on the level of logging

6.1.2 Access Control Policy

6.1.2.1 AC-1 Access control function (FDP_ACC.1) (FDP_ACF.1)

The Integration server provides two layers of access control on external users. The first is enforced at the port through which access is granted to the TOE, and the second is enforced at the layer of the object being accessed.

Controlling Access to a Element by Port

A port may provide access to one or more services. It may be configured to deny all services except those allowed (called Deny by Default) or to allow all services except those denied (called Allow by Default).

[8] The Authorized Administrator role includes all of the administrative roles defined in the product - from a security perspective, no distinction is made between the Integration Server Administrator, the Broker Administrator, and the Developer roles provided by the product.

Access to elements through a port is determined as follows:

- If the port does not allow the source IP address (IP addresses may be filtered on using wildcards both in the Domain Name and by wildcarding in dotted decimal notation), then the server rejects the request.
- If the port is configured to require certificate based user authentication, the certificate parameters are verified. Specifically, the issuing authority and certificate currency is checked.
- If the port is configured as Deny by Default and the requested service is not included in the list of services to be allowed, then the server rejects the request.
- If the port is configured as Allow by Default and the requested service is included in the list of services to be denied, then the server rejects the request.

By default, the Integration Server provides a single port that allows all hosts (identified by their IP addresses) to connect to it and allows access to all services through that port (unless prohibited by an ACL). IP addresses may be filtered on using wildcards both in the Domain Name and by wildcarding in dotted decimal notation.

Controlling Access to Resources with ACLs

ACLs control access to packages, folders, and other elements (such as services, document types, and specifications) at the group level for a given UID. The UID is determined either when an external user authenticates, or from the UID configured for a trigger when it executes. An ACL identifies groups that are allowed to access an element (Allowed Groups) and/or groups that are not allowed to access an element (Denied Groups). When identifying Allowed Groups and Denied Groups, the Authorized Administrator selects from groups that have previously defined. This is illustrated in Figure 6-1. and in Figure 6-2..

Access to elements is determined as follows:

- If an element does not have any ACLs specified, the server uses its inherited ACLs; if no parent folder has any ACLs set, then the server uses the Default ACL.
- If the requested element has an ACL, then the server allows the request only if the user is a member of at least one group listed on the ACL's Allowed groups list and is not a member of a group listed on the ACL's Denied groups. The server denies the request in all other cases.

An external user's UID is determined either in the I&A process, or in the case of triggers, from the UID assigned to the trigger.

Notes regarding Dynamic Server Pages (DSP)

If the element being accessed is a DSPs (or other file), access must be controlled for each directory. There is no inheritance or override mechanism as there is with folders and services. Access to DSPs is determined as follows:

- If the server does not allow the source IP address, then the server rejects the request.
- If the user making the request has not been authenticated, then the server processes the request as the Default user. (In the evaluated configuration no unauthenticated users are permitted to access the TOE).
- If the directory containing the DSP has a .access file and the .access file lists the requested DSP, then the server allows the request if the user is a member of at least

one group listed on the ACL's Allowed group and is not a member of any group on the ACL's Denied groups list.

- If the directory containing the DSP does not have a .access file, or if it has an .access SP, then the server applies the Default ACL.

The relationship of users, groups, ACLs, folders, services, and listeners is summarized in Figure 6-1. The relationship of users, groups, ACLs, .access files and DSPs is summarized in Figure 6-2.

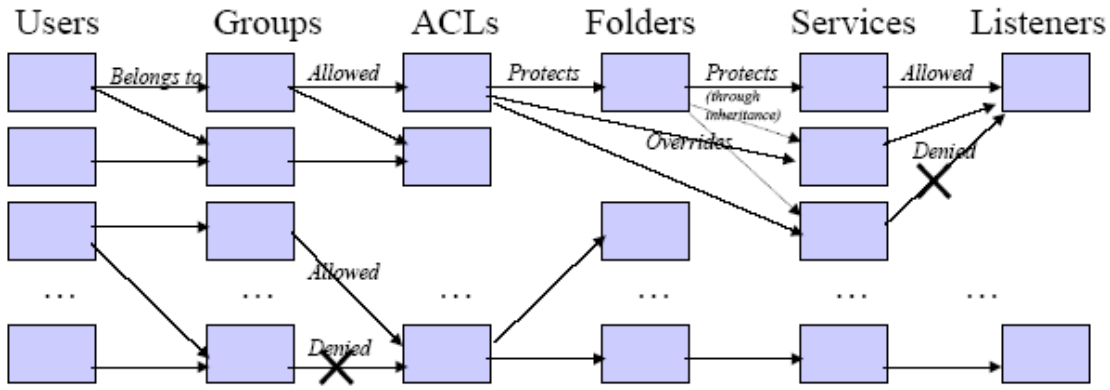

Figure 6-1 – Relationship of Users, Groups, ACLs, Folders, Services, and Ports

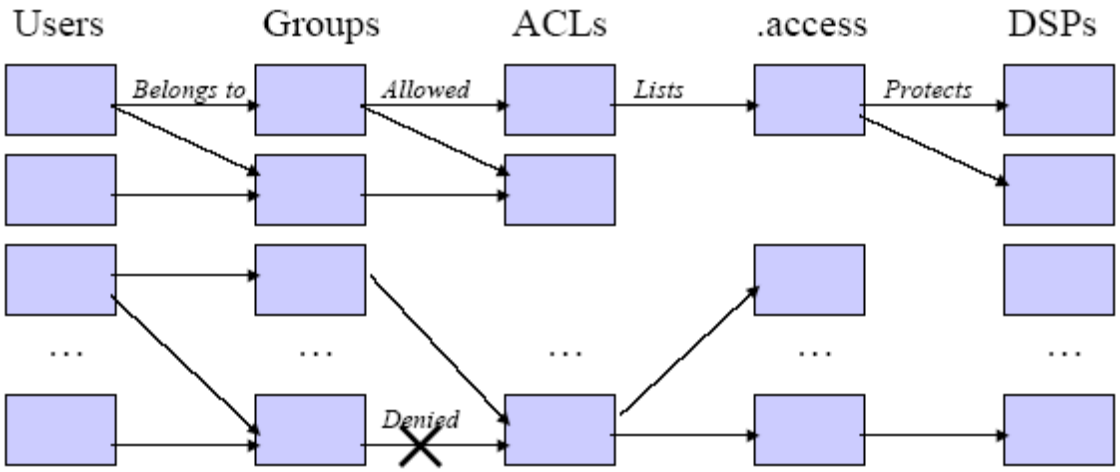

Figure 6-2 – Relationship of Users, Groups, ACLs, .access Files and DSPs

Specifically, an authorized administrator can control access to:

- **Services clients can invoke -** An authorized administrator can control which groups (and therefore which users) can invoke a service. In addition to checking ACLs to determine whether a client can invoke a service, the server performs a number of port level checks.
- **Special tools such as the Server Administrator, the Developer, and replicator functions -** These special abilities are granted by the Administrator, Developer, and Replicator ACLs that are provided with the Integration Server.

- **Elements that developers can see and use -** An authorized administrator can fine tune control over which developers have access to which packages, folders, and other elements. For example, one development group might have access to create, update, and maintain one set of services, while another development group has access to a different set. ACLs can prevent one development group from accidentally updating or damaging the work of another group.
- **Files the server can serve.** The server can serve files (for example .dsp and .htm files) that reside in the pub directory for a package or a subdirectory of the pub directory. An authorized administrator can control access to these files by assigning ACLs to them in .access files.

There are four different kinds of access: List, Read, Write, and Execute. These are defined in Table 6-2.

Table 6-2 Access Privileges

Access Privilege	Description
List	allows a user to see that an element exists. The element will be displayed on screens in the Developer and the Integration Server Administrator. List access also allows you to view an element's metadata.
Read	allows a user to view the main source of an element through the Developer and Integration Server Administrator.
Write	allows a user to edit an element. This access also allows a user to delete or lock an element or to assign an ACL to it.
Execute	allows a user to execute a service. This access also gives the user access to files the server serves, such as .dsp and .htm files.

For an element, an ACL consists of two group lists: a list of groups that are allowed access, and a second list of groups that are denied access. Each element can be associated with four ACLs: one List, one Read, one Write, and one Execute. The same ACL for the List, Read, Write, and Execute ACLs may be reused if desired. List, Read, and Write ACLs are used mostly during development time by developers, and to some extent server administrators, who need access to create, edit, and maintain services and other elements. Execute access is used extensively in production environments. When a user tries to access an element, the server checks the appropriate ACL (List, Read, Write, or Execute) associated with the element.

Privileges to invoke a service or access files are granted and denied by ACLs that an authorized administrator sets up. When an administrator creates ACLs, he or she identifies groups that are allowed to access services and files and groups that are denied access to services and files.

6.1.3 Identification and Authentication

6.1.3.1 IA-1 Security Attributes (FIA_ATD.1)

The TSF maintains the following user security attributes:

- User Name,
- Group Memberships
- Password
- X.509 certificate attributes (Distinguished Name, server private key, user certificate, CA certificate)

6.1.3.2 IA-2 Password Policy (FIA_SOS.1)

The TOE provides a mechanism for specifying that users select password that meet minimum strength criteria of SOF-basic. The default Password Policy is given in Table 6-3.

Table 6-3 Default Password Policy Rules

Requirement	Default
Minimum number of characters (alphabetic characters, digits, and special characters combined) the password must contain.	8
Minimum number of upper case alphabetic characters the password must contain.	2
Minimum number of lower case alphabetic characters the password must contain.	2
Minimum number of digits the password must contain.	1
Minimum number of special characters, such as asterisk (*), period (.), question mark (?), and ampersand (&) the password must contain.	1

6.1.3.3 IA-3 User authentication (FIA_UAU.5-1)

Users must be identified and authenticated before they are allowed to perform any security-relevant actions. The TSF is compatible with:

- external user authentication using UID and reusable password (The IT environment provides protection during transmission for the re-usable password).[9]

The TOE can verify that a certificate is signed by a root certificate trusted by the IS. The IT environment addresses all other aspects of certificate-based authentication in terms of concepts of either the proof component (i.e., proof that the external user has the private key that matches the certificate) and trusted certificate roots.

Once authenticated, the TOE associates the UID and corresponding user attributes (such as, but not limited to, presumed IP address, group memberships and negotiated port) with each network connection. External users are identified by presumed IP address before they are able to access the TOE. If the IT environment authenticates the external user using certificate-based authentication, the IT environment will also identify the user from certificate data, such as Distinguished Name.

6.1.4 Security Management

6.1.4.1 SM-1 Management of Security Functions (FMT_MOF.1)

The Authorized Administrator may choose to configure TSF data through facilities provided by the host OS, or the Authorized Administrator is able to obtain a visual display (HTML format) populated with TSF data that is stored on the system hard drive. The HTML format interface includes a GUI for configuration (external user authentication, allowable users, SSL configuration, package management, system shutdown, logs) and for configuring the security functions related to packages and associated attributes.

Before being able to manage the TOE, the user must be authorized through the identification/ authorization process. Ultimately, the administrator must verify that the security attributes he/she assigns/sets are correct for their security policy.

[9] Integration Server to Integration Server authentication is identical to IS to external user authentication from the perspective of the TOE.

For a complete description of the Administrator Management functions interaction, including configurable attributes and detailed information on managing specific policies see Table 5-2.

6.1.4.2 SM-2 Management of security attributes (FMT_MSA.1)

The Integration Server component of the TOE supports access control mechanism at the port through which an external user accesses the TOE, and also at the level of the object that the external user accesses. These mechanisms allow the administrator to establish parameters of connections (i.e., HTTPS or HTTP), maintain UIDs and authenticators, and specify IP access settings for listening ports. Administrators are also able to maintain ACL and establish the relationship of ACL with elements controlled by the TOE.

The policy is discretionary in that authorized administrators have the capability to change the access control attributes associated with some objects. Those access control attributes determine a user's access privilege.

Defining Groups

Administrator, replicator, and developer privileges are typically granted by adding a user to the Administrators, Replicators, or Developers group, respectively. Alternatively, an authorized administrator can create new groups and add them to the allow lists of the Administrators, Replicators, or Developers ACLs.

Create groups that identify groups of users that will share the same privileges. When an authorized administrator creates a group definition, a group name is specified and the members of the group are assigned.

Group name - A group name is a unique name that identifies the group. Any name can be used, for example, a name that defines a department (Marketing) or job function (Programmers).

Members - List of user names that are members of the group.

Table 6-4 IS Default User to Default Group Assignment

Default user	Group Assignments
Administrator	Everybody, Administrators, and Replicators group.
Default	Everybody and Anonymous group
Developer	Everybody and Developers group
Replicator	Everybody and Replicators group
Anonymous	Anonymous Group
Everybody	Everybody Group

Table 6-5 IS Default Group to ACL Mapping

Group	Description	Members (User)	ACL
Administrators	A user account that has administrator privileges. You can use the Administrator user account to access the Integration Server Administrator to configure and manage the server.	Administrator user	See Administrators ACL See Internal ACL See Replicators ACL
Default	The server uses the information defined for the Default user when the client does not supply a user name and password.	Default user	See Default ACL See Anonymous

Group	Description	Members (User)	ACL
Developers	A user that can connect to the server from the webMethods Developer to create, modify, and delete services that reside on the server.	Developer user	See Developers ACL See Internal ACL
Replicator	The user account that the server uses during package replication.	Administrator user, Replicator user	See Replicator ACL.
Anonymous	This group identifies users that have not been authenticated.	Anonymous user Default user	See Anonymous ACL
Everybody	All authenticated users are a member of this group. Every new user is automatically added to the Everybody group.	Administrator user, Default user, Developer user, Replicator user	

Note: Customized groups can also be created.

Table 6-6 IS Default Access Control Lists

ACL (Security Role)	Description
Administrators	Allows users in the Administrators group, or any other group added to this ACL, access to a package, folder, or other element and denies all other users. In addition, you must make sure the user is not a member of a group that is denied access by this ACL.
Default	Allows all authenticated users access to a package, folder, or other element.
Developers	Allows only users in the Developers group, or any other group added to this ACL, access to a package, folder, or other element and denies all other users. In addition, you must make sure the user is not a member of a group that is denied access by this ACL.
Replicator	Allows users in the Replicator group, or any other group added to this ACL, replication privileges. In addition, you must make sure the user is not a member of a group that is denied access by this ACL.
Anonymous	Provides access to unauthenticated users (those that did not specify a valid UID).
Internal	Allows only users in the Administrators and Developers groups access to a package, folder, or other element and denies all other users. The server assigns this ACL to built-in utility services shipped with the server, such as those in the WmRoot and WmPublic folders. Normally not manually assigned to an element.
BrokerAdministrators	Allows users to perform administrative functions on the Broker Server. (**No user is assigned to this groups by default**)
BrokerUsers	Allows users to be Broker Clients. (**No user is assigned to this groups by default**)

Defining User Accounts

When a user account is created on the Integration Server the following is specified: user name, password, and group membership.

User name - A user name is a unique name that identifies a client. An authorized administrator can specify a user name that represents an actual person (e.g., “JDSmith” for John D. Smith), or you can specify a user name to represent applications, job functions, or organizations. For example, you might set up generically named user names such as “MktgPurchAgent”, “MktgTimeKeeper,” and so forth, to represent job functions.

Group membership - The group membership identifies the groups to which a user belongs. Access to the server's resources is controlled at the group level:

- Only users that are members of the Administrators group can configure and manage the server using the Integration Server Administrator.
- Only users that are members of the Developers group can connect to the server from the webMethods Developer to create, modify, and delete services.
- The server protects access to services and files using Access Control Lists (ACLs). An authorized administrator sets up ACLs that identify groups that are allowed or not allowed to access a resource.

For a complete description reference Table 5-3

6.1.4.3 SM-3 Default Values of Security Attributes (FMT_MSA.3)

The configurable access control mechanisms that govern external users' accessing the Integration Server component of the TOE (via the webMethods IS Access Control SFP) provides for a restrictive default access to all objects. An external user may not access the TOE until the user has been granted access privileges by the Administrator. By default, object access privileges defined in the ACL are restricted to the Administrator and Developer. The TOE does not allow any user to specify alternative initial values for the ACL on an object to override the default value.

webMethods comes to the consumer with an ACL option settings of do not check ACL when internally invoked. Only the Administrator role is authorized to change the security attributes. webMethods comes to the consumer with a default UID of Administrator for services invoked by triggers. Both of these setting may be modified.

6.1.4.4 SM-4 Management of TSF Data (FMT_MTD.1-1)

The TSF restricts the ability to query, modify, delete, create, and other operations as specified in Table 5-4 by restricting the ability to manage all TSF data to the Authorized Administrator.

Capabilities restricted to the Authorized Administrator also include management of identification data, use of audit commands to select auditing levels and to query and review audit records from the audit logs.

6.1.4.5 SM-5 Specification of Management Functions (FMT_SMF.1)

The TOE is capable of performing the following security management functions:

- determine the behavior of the functions listed in Table 5-2 to the authorized roles identified in Table 5-2, or the Broker Administrator.
- query, modify, and delete the security attributes as specified in Table 5-3 to the authorized roles identified in Table 5-3, or the Broker Administrator. (see FMT_MSA.1),
- query, modify, delete, and create TSF Data as specified in Table 5-4 (See FMT_MTD.1-1).

6.1.4.6 SM-6 Security Roles (FMT_SMR.1)

The TOE supports one security role:

- Authorized Administrator

The Authorized Administrator role has the ability to enable, disable, or modify the behavior of all security functions. The TOE maintains this role and supports associating users to this role.

6.1.5 TSF Self-Protection

6.1.5.1 SP-1 Non-bypassability (FPT_RVM_EXP.1-1)

The TOE includes one network connection that must be considered in terms of non-bypassability:

- External User to Integration Server

In combination with the IT environment, the TOE environment ensures all information from an external client to an Integration Server or vice versa goes through the TSF. The TSF ensures that all information must flow through the policy enforcement mechanisms.

In order for an external user to access a host, the client must use an encrypted connection between the External user and the TSF. The encrypted connection is supported by the IT environment. The TSF extracts packet header information revealing user attributes: IP source address fields, negotiated port, and protocol. In addition the TSF may extract certificate based authentication data to identify the external user. If certificate based authentication data is not available, the external user must provide a UID and authenticator (password). Once the external user is identified and authenticated, they cannot act without invoking an object that is protected by the TSF. Based on the user attributes the external users access privilege is determined for the invoked object. The TSF instantiates a separate object unique to the specific external user and negotiated port, to mediate all communication between the External user and the host. There is no communication path that passes the client information directly to the host (or vice versa) except though the object dedicated to the specific authenticated connection. Hence, the TSF ensures that all information must flow through the policy enforcement mechanisms.

In order for a Broker Client to access a Broker, they must mutually authenticate via certificate-based authentication. The TSF extracts packet header information revealing the IP source address fields, negotiated port, and protocol. In addition, the TSF extracts certificate based authentication data to identify the Broker Client. Once the Broker Client is identified and authenticated, they cannot act except to access a document type, which is protected by the TSF. Based upon user attributes and the operation the Broker Clients access privilege is determined for the document type. If access is permitted, documents of the type accessed may be exchanged. There is no communication path that passes the client information directly to the Broker (or vice versa) except though the described process dedicated to the specific authenticated connection. In combination with the IT environment, the TOE environment ensures all information from a Broker Client to a Broker Server or vice versa goes through the TSF. Hence, the TSF ensures that all information must flow through the policy enforcement mechanisms.

6.1.5.2 SP-2 TSF domain separation (FPT_SEP_EXP.1-1)

With support from the IT environment, the TSF maintains a security domain for its own execution that protects it from interference and tampering by untrusted subjects. webMethods Fabric's protected domain includes the webMethods software and all of its software components.

There are two subjects of the TSF policy enforcement mechanisms: external users and Broker Clients. Broker Clients are created by the Authorized Administrator, are exist internal to the TOE, and operate across on the IS-Broker internal interface.

The servers that comprise the TOE reside on different machines. The Integration Server resides on the machine where an external interface to external user services exists, and the Broker Server resides on a machine internal to the TOE where documents are temporarily stored pending delivery to Broker Clients.

No arbitrary code runs on the Integration Server. An extensive library of built-in services installed on the Integration Server by the authorized administrator during system configuration. The Integration Server component of the TOE ensures the isolation and protection of services and arbitrates subject access to each service, which is associated with an ACL and subject to the access control policies. The TSF enforces separation between the security domains of subjects in the TSC by instantiation a separate object for each user connection. The TSF relies on the UID and session parameters (negotiated port) to isolate user sessions and maintain object separation.

No customer specific code resides on the Broker Server. External users are not permitted to directly access the Broker Server; only Broker Clients executing on the Information Server on behalf of the external users may access the Broker Server. Hence an external user has no mechanism available with which to directly access the Brokers installed on the Broker Server.

In addition to the webMethods-specific software, other software files such as configuration files are also stored on disk. For these files webMethods Fabric relies on the IT environment to provide file access permissions and identification and authentication of users at the OS level. An underlying assumption regarding the operation of webMethods Fabric is that it is maintained in a physically secure environment. This is a needed assumption for maintaining a security domain for TSF execution to provide protection from interference from untrusted subjects.

6.2 Assurance Measures

The TOE satisfies CC EAL2 assurance requirements. Table 6-7 identifies the Configuration Management, Delivery and Operation, Development, Guidance Documents, Testing, and Vulnerability Assessment Assurance Measures applied by webMethods to satisfy the CC EAL2 assurance requirements.

Table 6-7 EAL2 Assurance Requirements Measures

Assurance Component	How requirement will be met	Document Version
ACM_CAP.2 Configuration Items	The vendor provided configuration management documents and a Configuration Item list.	webMethods Fabric 6.5 Configuration Management, Version 0.5
ADO_DEL.1 Delivery Procedures	The vendor provided delivery procedures.	webMethods Version 6.5 Delivery Procedures Version 0.3 Final

Assurance Component	How requirement will be met	Document Version
ADO_IGS.1 Installation, Generation and Startup procedures	The vendor provided secure installation, generation and start up procedures.	webMethods Broker Administrator's Guide, Version 6.5, Document ID: BR-AG-65-20050615, webMethods Developer User's Guide, Version 6.5, DEV-UG-65-20050429, webMethods Fabric Integration Platform Error Message Reference, Document ID: PLAT-EM-RF-601-20031201, webMethods Integration Server Administrator's Guide, Version 6.5, Document ID: webM-IS-AG-65-20050429, webMethods Integration Server Built-In Services Reference, Version 6.5, Document ID: IS-BIS-RF-65-20050330, webMethods Installation Guide Version 6.5 and Version 6.5.1 Document ID: WEBM-IG-65-20050930, webMethods Publish-Subscribe Developer's Guide Version 6.5 Document ID: DEV-PS-DG-65-20050429, webMethods Logging Guide Version 6.5 Document ID: WEBM-LG-65-20050511, webMethods JDBC Adapter User's Guide, Document ID: ADAPTER-JDBC-UG-603-20040511, webMethods JMS Adapter Installation Guide, Document ID: ADAPTER-JMS-IG-61-20040213, webMethods JMS Adapter User's Guide, Document ID: ADAPTER-JMS-UG-61-20040213
ADV_FSP.1 Informal functional specification	The vendor provided an informal function specification.	webMethods Fabric 6.5 Proprietary Development Specification, Version 0.6
ADV_HLD.1 Descriptive high-level design	The vendor provided a descriptive high-level design document.	webMethods Fabric 6.5 Proprietary Development Specification, Version 0.6
ADV_RCR.1 Informal correspondence demonstration	The informal correspondence demonstration is provided in the design documentation. ST to FSP in the FSP, FSP to HLD in the HLD.	webMethods Fabric 6.5 Proprietary Development Specification, Version 0.6
AGD_ADM.1 Administrator Guidance	The vendor submitted a system administration manual.	webMethods Fabric 6.5 Guidance Documentation Version 0.3, also see documents listed under ADO_IGS.1
AGD_USR.1 User Guidance	The vendor submitted a user guide.	There is no user role.
ATE_COV.1 Evidence of coverage	The analysis of test coverage was submitted in the evaluation evidence.	webMethods Fabric 6.5 Test coverage analysis, Version 0.4, webMethods Version 6.5 Server Regression Test Cases
ATE_FUN.1 Functional testing	The test evidence was submitted to the CCTL.	webMethods Fabric 6.5 Test coverage analysis, Version 0.4, webMethods Version 6.5 Server Regression Test Cases

Assurance Component	How requirement will be met	Document Version
ATE_IND.2 Independent testing - sample	The laboratory used development evidence submitted by the vendor along with functional testing evidence as a baseline for an independent test plan.	webMethods Fabric 6.5 Test coverage analysis, Version 0.4, webMethods Version 6.5 Server Regression Test Cases
AVA_SOF.1 Strength of TOE security function evaluation[10]	The vendor submitted an analysis of the SOF for the password.	webMethods Version 6.5 Strength of Function Analysis Version 0.3 Final
AVA_VLA.1 Developer vulnerability analysis	The vendor submitted vulnerability analysis was confirmed. The laboratory conducted an independent vulnerability assessment by building on the vendor's. The laboratory conducted penetration testing.	webMethods Fabric 6.5 Vulnerability Analysis Version 0.3 Draft

[10] Cryptographic analysis or certification is not required for the TOE since all cryptographic functions are supported by the IT environment.

7 PP Claims

The Security Target was not written to address any existing Protection Profile.

8 Rationale

8.1 Security Objectives Rationale

8.1.1 Threats to Security

Table 8-1 shows that all the identified threats to security are countered by Security Objectives for the TOE. Rationale is provided for each threat below the table.

Table 8-1 All Threats to Security Countered

Identifier	Threat	Security Objective
T.Abuse	An undetected compromise of the TOE may occur as a result of an authorized user of the TOE (intentionally or otherwise) performing actions the individual is authorized to perform.	O.Access, O.Attributes, O.Audit, O.AuditProtection, O.AuditReview, O.IDAuth, OE.Audit, OE.AuditProtection, OE.IDAuth, OE.Time, ON.Operations, ON.Person
T.Access	An authorized user of the TOE may access information or resources without having permission from the person who owns, or is responsible for, the information or resource.	O.Access, O.Attributes, O.Audit, O.AuditProtection, O.AuditReview, O.IDAuth, O.RemoteAdministration, OE.AuditProtection, OE.IDAuth, OE.Time
T.Bypass	An unauthorised user may attempt to bypass the information flow control policy.	O.NonBypass, O.PartialSelfProtection, OE.NonBypass, OE.PartialSelfProtection
T.Intercept	An unauthorized person on an internal network that connects TOE components may intercept communications between the TOE components and attempt to access and/or modify the data being transmitted.	OE.ProtectComm
T.Mismanage	Authorized administrators may make errors in the management of security functions and TSF data. Administrative errors may allow attackers to gain unauthorized access to resources protected by the TOE.	O.Admin, O.RemoteAdministration, ON.Install , ON.Person
T.Tamper	An attacker may attempt to modify TSF programs and data.	O.PartialSelfProtection, OE.PartialSelfProtection, ON.NoUntrusted, ON.Operations, ON.Person, ON.Physical, ON.ProtectAuth
T.Transmit	TSF data may be disclosed or modified by an attacker while being transmitted between the TOE and its users.	O.PartialSelfProtection, OE.PartialSelfProtection, OE.ProtectComm
T.Undetect	Attempts by an attacker to violate the security policy may go undetected. If the attacker is successful, TSF data may be lost or altered.	O.Audit, O.AuditProtection, O.AuditReview, O.PartialSelfProtection, OE.Audit, OE.AuditProtection, OE.PartialSelfProtection, OE.Time, ON.Physical

Rationale

T.Abuse	Is countered by:
O.Access	This objective counters this threat by providing access controls that limit the actions an individual is authorized to perform.
O.Attributes	This objective counters this threat by requiring the TOE to store and maintain user attributes. These attributes associate users with user roles and access rights, in conformance to the Access Control SFP.
O.Audit	This objective counters this threat by requiring the TOE to audit attempts for data accesses and use of TOE functions.
O.AuditProtection	This objective counters this threat by requiring the TOE to provide protection of the audit interface.
O.AuditReview	This objective counters this threat by giving authorized administrators the ability to review the audit logs. This will help in detecting data accesses and use of TOE functions.
O.IDAuth	This objective provides for authentication of users prior to any TOE data access
OE.Audit	This objective counters this threat by requiring the IT Environment to audit authentication to the OS and therefore *who* had access to TOE configuration files.
OE.AuditProtection	This objective counters this threat by requiring the IT Environment to provide protection of the audit storage.
OE.IDAuth	This objective provides for authentication of users prior to any TOE data access
OE.Time	This objective provides for a reliable way to correlate audit records to reconstruct a potential compromise.
ON.Operations	This objective provides for guidance documentation that explains how to securely manage the TOE.
ON.Person	This objective provides for qualified and trained authorized administrators to manage the TOE.
T.Access	Is countered by:
O.Access	This objective counters this threat by providing access controls that limit the actions an individual is authorized to perform.
O.Attributes	This objective counters this threat by requiring the TOE to store and maintain user attributes. These attributes associate users with user roles and access rights, in conformance to the Access Control SFP.
O.Audit	This objective counters this threat by requiring the TOE to audit attempts for data accesses and use of TOE functions.
O.AuditProtection	This objective counters this threat by requiring the TOE to provide protection of the audit interface.
O.AuditReview	This objective counters this threat by giving authorized administrators the ability to review the audit logs. This will help in detecting data accesses and use of TOE functions.
O.IDAuth	This objective provides for authentication of users prior to any TOE data access
O.RemoteAdministration	This objective provides for authorized administrators to perform TOE functions that are essential to security through the remote administrator client interface
OE.AuditProtection	This objective counters this threat by requiring the IT Environment to provide protection of the audit storage
OE.IDAuth	This objective provides for authentication of users prior to any TOE data access.

OE.Time	This objective provides for a reliable way to correlate audit records to reconstruct a potential compromise.
T.Bypass	Is countered by:
O.NonBypass	This objective counters this threat by ensuring the TOE's protection mechanisms cannot be bypassed, which requires that TSF security functions not be bypassable. This is supported by OE.NonBypass
O.PartialSelfProtection	This objective supports addressing this threat by providing TOE self-protection and separation between users. The TOE will maintain separation between code executing on behalf of different users
OE.NonBypass	This objective supports countering this threat by ensuring the TOE's protection mechanisms cannot be bypassed, which requires that IT environment security functions not be bypassable.
OE.PartialSelfProtection	This objective supports addressing this threat by providing TOE self-protection and separation between users. The TOE will maintain separation between code executing on behalf of different users
T.Intercept	Is countered by:
OE.ProtectComm	This objective supports preventing data from being disclosed or modified when it is being transmitted between client and server components.
T.Mismanage	Is countered by:
O.Admin	This objective counters mismanagement through the functionality that enables authorized user(s) to effectively manage the TOE and its security functions.
O.RemoteAdministration	This objective provides for authorized administrators to perform TOE functions that are essential to security through the remote administrator client interface
ON.Install	This objective provides for secure installation and configuration of the TOE.
ON.Person	This objective provides for qualified and trained authorized administrators to manage the TOE.
T.Tamper	Is countered by:
O.PartialSelfProtection	This objective addresses this threat by providing separation between code executing on behalf of different users. In addition, this objective addresses this threat by providing TOE self-protection and separation between users.
OE.PartialSelfProtection	This objective partially addresses this threat by protecting the IT environment and the TOE and its data.
ON.NoUntrusted	This objective provides for the protection of the TOE from untrusted software.
ON.Operations	This objective provides for guidance documentation that explains how to securely manage the TOE.
ON.Person	This objective provides for qualified and trained authorized administrators to manage the TOE.
ON.Physical	This objective provides for physical protection of the TOE
ON.ProtectAuth	This objective provides for authorized users not sharing their passwords with others

T.Transmit	Is countered by:
O.PartialSelfProtection	This objective addresses this threat by providing separation between code executing on behalf of different users. In addition, this objective addresses this threat by providing TOE self-protection and separation between users.
OE.PartialSelfProtection	This objective supports addressing this threat by protecting the IT environment and the TOE and its data.
OE.ProtectComm	This objective supports preventing data from being disclosed or modified when it is being transmitted between client and server components.
T.Undetect	Is countered by:
O.Audit	This objective counters this threat by requiring the TOE to audit attempts for data accesses and use of TOE functions.
O.AuditProtection	This objective counters this threat by requiring the IT Environment to provide protection of the audit storage.
O.AuditReview	This objective counters this threat by giving authorized administrators the ability to review the audit logs. This will help in detecting data accesses and use of TOE functions.
O.PartialSelfProtection	This objective supports addressing this threat by providing separation between code executing on behalf of different users. In addition, this objective addresses this threat by providing TOE self-protection and separation between users.
OE.Audit	This objective counters this threat by requiring the IT Environment to audit authentication to the OS and therefore *who* had access to TOE configuration files..
OE.AuditProtection	This objective supports countering this threat by requiring the IT Environment to provide protection of the audit storage
OE.PartialSelfProtection	This objective supports addressing this threat by protecting the TOE and its data.
OE.Time	This objective provides for a reliable way to correlate audit records to reconstruct a potential compromise.
ON.Physical	This objective provides for physical protection of the TOE

Table 8-2 Reverse Mapping of TOE Security Objectives to Threats

Identifier	Threat
O.Access	T.Abuse, T.Access
O.Admin	T.Mismanage
O.Attributes	T.Abuse, T.Access
O.Audit	T.Abuse, T.Access, T.Undetect
O.AuditProtection	T.Abuse, T.Access, T.Undetect
O.AuditReview	T.Abuse, T.Access, T.Undetect
O.IDAuth	T.Abuse, T.Access,
O.NonBypass	T.Bypass
O.PartialSelfProtection	T.Bypass, T.Tamper, T.Transmit, T.Undetect
O.RemoteAdministration	T.Access, T.Mismanage
OE.Audit	T.Abuse, T.Undetect
OE.AuditProtection	T.Abuse, T.Access, T.Undetect
OE.IDAuth	T.Abuse, T.Access
OE.NonBypass	T.Bypass

Identifier	Threat
OE.PartialSelfProtection	T.Bypass, T.Tamper, T.Transmit, T.Undetect
OE.ProtectComm	T.Intercept, T.Transmit
OE.Time	T.Abuse, T.Access, T.Undetect
ON.Install	T.Mismanage
ON.NoUntrusted	T.Tamper
ON.Operations	T.Abuse, T.Tamper
ON.Person	T.Abuse, T.Mismanage, T.Tamper
ON.Physical	T.Tamper, T.Undetect
ON.ProtectAuth	T.Tamper

8.1.2 Assumptions

Table 8-3 shows that all of the secure usage assumptions are addressed by either security objectives for the IT environment or Non-IT security objectives. Rationale for each assumption is provided below the table.

Table 8-3 All Assumptions Addressed

Name	Assumption	Objective
A.Admin	The administrator is trusted to correctly configure and operate the TOE according to the instructions provided by the TOE documentation.	ON.Install, ON.Operations
A.IT	The TOE relies upon the IT environment to support protected communications, provide audit file protection, support partial domain separation, support non-bypassability, and to perform user authentication when configured to do so.	OE.AuditProtection, OE.IDAuth, OE.NonBypass, OE.PartialSelfProtection, OE.ProtectComm
A.Manage	It is assumed that one or more authorized administrators are assigned who are competent to manage the TOE and the security of the information it contains, and who can be trusted not to deliberately abuse their privileges so as to undermine security.	ON.Person
A.NoUntrusted	It is assumed that there will be no untrusted software on the webMethods Integration Server and Broker.	ON.NoUntrusted
A.Physical	The TOE components critical to the security policy enforcement will be protected from unauthorized physical modification.	ON.Physical
A.Users	It is assumed that users will protect their authentication data.	ON.ProtectAuth

Rationale:

A.Admin	is covered by:
ON.Install	This objective provides for secure installation and configuration of the TOE.
ON.Operations	This objective provides for operation procedures to be in place.
A.IT	is covered by:
OE.AuditProtection	This objective provides for secure storage of the audit data.

OE.IDAuth	This objective provides for authentication of administrators.
OE.NonBypass	This objective provides support for the TOE's non-bypassability.
OE.PartialSelfProtection	This objective provides support for the TOE's partial self protection.
OE.ProtectComm	This objective provides support for the TOE's protected communications configuration features and also can be configured to provided certificate based authentication.
A.Manage	is covered by:
ON.Person	This objective provides for competent personnel to administer the TOE.
A.NoUntrusted	is covered by:
ON.NoUntrusted	This objective provides for the protection of the TOE from untrusted software.
A.Physical	is covered by:
ON.Physical	This objective provides for the physical protection of the TOE software.
A.Users	is covered by:
ON.ProtectAuth	This objective provides for users protecting their authentication data.

Table 8-4 Environment Objective to Threat or Assumption Mapping

Objective	Threat or Assumption
OE.Audit	T.Abuse, T.Undetect
OE.AuditProtection	A.IT, T.Abuse, T.Access, T.Undetect
OE.IDAuth	A.IT, T.Abuse, T.Access
OE.NonBypass	A.IT, T.Bypass
OE.PartialSelfProtection	A.IT, T.Bypass, T.Tamper, T.Transmit, T.Undetect
OE.ProtectComm	A.IT, T.Intercept, T.Transmit
OE.Time	T.Abuse, T.Access, T.Undetect
ON.Install	A.Admin, T.Mismanage
ON.NoUntrusted	A.NoUntrusted , T.Tamper
ON.Operations	A.Admin, T.Abuse, T.Tamper
ON.Person	A.Manage , T.Abuse, T.Mismanage, T.Tamper
ON.Physical	A.Physical , T.Tamper, T.Undetect
ON.ProtectAuth	A.Users, T.Tamper

8.2 Security Requirements Rationale

8.2.1 Functional Requirements

Table 8-5 shows that all of the security objectives of the TOE are satisfied. Rationale for each objective is included below the table.

Table 8-5 All Objectives Met by Functional Components

Objective	Objective Description	SFR
O.Access	The TOE must allow authorized users to access only appropriate TOE functions and data.	FAU_SAR.2 Restricted audit review, FDP_ACC.1 Subset access control, FDP_ACF.1 Security attribute based access control, FIA_UAU.5-1 Multiple authentication mechanisms, FMT_MOF.1 Management of Security Functions Behavior, FMT_MSA.1 Management of security attributes, FMT_MTD.1-1 Management of TSF data
O.Admin	The TOE must provide the functionality to enable authorized user(s) to effectively manage the TOE and its security functions.	FAU_SAR.1 Audit review, FIA_ATD.1 User attribute definition, FIA_UAU.5-1 Multiple authentication mechanisms, FMT_MOF.1 Management of Security Functions Behavior, FMT_MSA.1 Management of security attributes, FMT_MSA.3 Static attribute initialisation, FMT_MTD.1-1 Management of TSF data, FMT_SMF.1 Specification of management functions, FMT_SMR.1 Security roles
O.Attributes	The TOE must be able to maintain user security attributes.	FIA_ATD.1 User attribute definition
O.Audit	The TOE must provide a means to record a readable audit trail of security-related events, with accurate dates and times.	FAU_GEN.1 Audit data generation, FAU_SAR.3 Selectable audit review, FAU_SEL.1 Selective audit, FIA_ATD.1 User attribute definition, FMT_MOF.1 Management of Security Functions Behavior, FMT_MTD.1-1 Management of TSF data, FMT_SMF.1 Specification of management functions
O.AuditProtection	The TOE must provide the capability to protect audit information.	FAU_SAR.2 Restricted audit review
O.AuditReview	The TOE must provide the functionality to enable authorized user(s) to review the audit logs.	FAU_SAR.1 Audit review, Audit review, FAU_SAR.2 Restricted audit review, FAU_SAR.3 Selectable audit review
O.IDAuth	The TOE must be able to identify and authenticate the users prior to allowing access to TOE functionality.	FIA_ATD.1 User attribute definition, FIA_SOS.1 Verification of secrets, FIA_UAU.5-1 Multiple authentication mechanisms, FMT_MSA.1 Management of security attributes

Objective	Objective Description	SFR
O.NonBypass	The TOE must ensure the TOE's security functional policy is invoked and succeeds before allowing another TOE function to proceed.	FPT_RVM_EXP.1-1 Nonbypassability of the TSP: TOE
O.PartialSelfProtection	The TSF will maintain a domain for its own execution that protects itself and its resources from external interference, tampering or unauthorized disclosure, through its own interfaces.	FPT_SEP_EXP.1-1 TSF domain separation: TOE
O.RemoteAdministration	An identified and authorized remote administrator may manage identified TSF entities from a remote client.	FIA_UAU.5-1 Multiple authentication mechanisms, FMT_SMR.1 Security roles

SFR rationale

O.Access	**The TOE must allow authorized users to access only appropriate TOE functions and data.**
FAU_SAR.2	Restricted audit review (TOE), which requires that access to audit data be restricted to authorized users.
FDP_ACC.1	Subset access control (TOE), ensures there will be a DAC mechanism present to protect objects on the Integration Server.
FDP_ACF.1	Security attribute based access control (TOE), which requires the TSF enforce access controls based on specified security attributes.
FIA_UAU.5-1	Multiple authentication mechanisms (TOE), ensure that the system must perform identification and authentication of all users.
FMT_MOF.1	Management of Security Functions Behavior (TOE), which restricts the ability to disable enable and modify functions to authorized users.
FMT_MSA.1	Management of security attributes (TOE), which enforces the Management of Security Attributes to restrict the ability to create query modify and delete the specified security attributes to the authorized account types.
FMT_MTD.1-1	Management of TSF data (TOE), which specifies the management of TSF Data according to assigned roles.
O.Admin	**The TOE must provide the functionality to enable authorized user (s) to effectively manage the TOE and its security functions.**
FAU_SAR.1	Audit review (TOE), which requires that the authorized administrator be able to read all audit records within the administrator's scope of control.
FIA_ATD.1	User attribute definition (TOE), which requires that the TSF maintain security attributes of users.
FIA_UAU.5-1	Multiple authentication mechanisms (TOE), Multiple authentication mechanisms (TOE), ensure that the system must perform identification and authentication of all users, including Administrators.
FMT_MOF.1	Management of Security Functions Behavior (TOE), which restricts the ability to disable enable and modify functions to authorized users.

FMT_MSA.1	Management of security attributes (TOE), which enforces the Management of Security Attributes to restrict the ability to create query modify and delete the specified security attributes to the authorized account types.
FMT_MSA.3	Static attribute initialisation (TOE), which requires the TSF enforce access control for specified default values of security attributes.
FMT_MTD.1-1	Management of TSF data (TOE), which specifies the management of TSF Data according to assigned roles in the Integration Server.
FMT_SMF.1	Specification of management functions (TOE), which requires the TSF be capable of performing the specified security management functions.
FMT_SMR.1	Security roles (TOE), which requires that the TSF maintain multiple roles.
O.Attributes	**The TOE must be able to maintain user security attributes.**
FIA_ATD.1	User attribute definition (TOE), which requires that the TSF maintain security attributes of users.
O.Audit	**The TOE must provide a means to record a readable audit trail of security-related events, with accurate dates and times.**
FAU_GEN.1	Audit data generation (TOE), which requires the ability to audit specified events.
FAU_SAR.3	Selectable audit review (TOE), ensures that the administrator can perform searches of the audit data
FAU_SEL.1	Selective audit (TOE),
FIA_ATD.1	User attribute definition (TOE), which requires that the TSF maintain security attributes of users.
FMT_MOF.1	Management of Security Functions Behavior (TOE), which restricts the ability to disable enable and modify functions to authorized users.
FMT_MTD.1-1	Management of TSF data (TOE), which specifies the management of TSF Data according to assigned roles.
FMT_SMF.1	Specification of management functions (TOE), which requires the TSF be capable of performing the specified security management functions.
O.AuditProtection	**The TOE must provide the capability to protect audit information.**
FAU_SAR.2	Restricted audit review (TOE), which requires that access to audit data be restricted to authorized users.
O.AuditReview	**The TOE must provide the functionality to enable authorized user (s) to review the audit logs.**
FAU_SAR.1	Audit review (TOE), which requires that the authorized administrator be able to read all audit records within the administrator's scope of control.
FAU_SAR.2	Restricted audit review (TOE), which requires that access to audit data be restricted to authorized users.
FAU_SAR.3	Selectable audit review (TOE), administrator to perform searches of the audit data based on event date.
O.IDAuth	**The TOE must be able to identify and authenticate the users prior to allowing access to TOE functionality.**
FIA_ATD.1	User attribute definition (TOE), which requires that the TSF maintain security attributes of users.
FIA_SOS.1	Verification of secrets (TOE), ensures the strength of secrets used as authenticators in the TOE.
FIA_UAU.5-1	Multiple authentication mechanisms (TOE), Multiple authentication mechanisms (TOE), ensure that the system must perform identification and authentication of all users, including Administrators.

FMT_MSA.1	Management of security attributes (TOE), which enforces the Management of Security Attributes to restrict the ability to create query modify and delete the specified security attributes to the authorized account types.
O.NonBypass	**The TOE must ensure the TOE's security functional policy is invoked and succeeds before allowing another TOE function to proceed.**
FPT_RVM_EXP.1-1	Nonbypassability of the TSP: TOE (TOE), which requires that TSP enforcement functions are invoked and succeed before a security-relevant function is allowed to proceed.
O.PartialSelfProtection	**The TSF will maintain a domain for its own execution that protects itself and its resources from external interference, tampering or unauthorized disclosure, through its own interfaces.**
FPT_SEP_EXP.1-1	TSF domain separation: TOE (TOE), This requires that the TOE provide partial protection to maintain separation between code executing on behalf of different users.
O.RemoteAdministration	**An identified and authorized remote administrator may manage identified TSF entities from a remote client.**
FMT_SMR.1	Security roles (TOE), which requires that the TSF maintain multiple roles.
FIA_UAU.5-1	Multiple authentication mechanisms (TOE), Multiple authentication mechanisms (TOE), ensure that the system must perform identification and authentication of all users, including Administrators.

Table 8-6 All Objectives Met by Functional Components Reversed

SFR	Title	Objective
FAU_GEN.1	Audit data generation	O.Audit
FAU_SAR.1	Audit review	O.Admin, O.AuditReview
FAU_SAR.2	Restricted audit review	O.Access, O.AuditProtection, O.AuditReview
FAU_SAR.3	Selectable audit review	O.Audit, O.AuditReview
FAU_SEL.1	Selective audit	O.Audit
FDP_ACC.1	Subset access control	O.Access
FDP_ACF.1	Security attribute based access control	O.Access
FIA_ATD.1	User attribute definition	O.Admin, O.Attributes, O.Audit, O.IDAuth
FIA_SOS.1	Verification of secrets	O.IDAuth
FIA_UAU.5-1	Multiple authentication mechanisms	O.Access, O.Admin, O.IDAuth, O.RemoteAdministration
FMT_MOF.1	Management of Security Functions Behavior	O.Access, O.Admin, O.Audit
FMT_MSA.1	Management of security attributes	O.Access, O.Admin, O.IDAuth
FMT_MSA.3	Static attribute initialisation	O.Admin
FMT_MTD.1-1	Management of TSF data	O.Access, O.Admin, O.Audit
FMT_SMF.1	Specification of management functions	O.Admin, O.Audit
FMT_SMR.1	Security roles	O.Admin, O.RemoteAdministration

SFR	Title	Objective
FPT_RVM_EXP.1-1	Nonbypassability of the TSP: TOE	O.NonBypass
FPT_SEP_EXP.1-1	TSF domain separation: TOE	O.PartialSelfProtection

8.2.2 Dependencies

Table 8-7 shows the dependencies between the functional requirements. All dependencies are satisfied. Dependencies that are satisfied by a hierarchical component are denoted by an (H) following the dependency reference. If the TOE dependency is met by an SFR in the IT environment an "E" will be next to the reference number.

Table 8-7 TOE Dependencies Satisfied

TOE Security Functional Requirements	Dependencies	How met
Class FAU: Security Audit		
FAU_GEN.1 Audit data generation	FPT_STM.1 Reliable time stamps	FPT_STM.1
FAU_SAR.1 Audit review	FAU_GEN.1 Audit data generation	FAU_GEN.1
FAU_SAR.2 Restricted audit review	FAU_SAR.1 Audit review	FAU_SAR.1
FAU_SAR.3 Selectable audit review	FAU_SAR.1 Audit review	FAU_SAR.1
FAU_SEL.1 Selective audit	FAU_GEN.1 Audit data generation, FMT_MTD.1-1 Management of TSF data	FAU_GEN.1, FMT_MTD.1-1
Class FDP: User Data Protection		
FDP_ACC.1 Subset access control	FDP_ACF.1 Security attribute based access control	FDP_ACF.1-1
FDP_ACF.1 Security attribute based access control	FDP_ACC.1 Subset access control, FMT_MSA.3 Static attribute initialization	FDP_ACC.1, FMT_MSA.3
Class FIA: Identification and Authentication		
FIA_ATD.1 User attribute definition	No dependencies.	N/A
FIA_SOS.1 Verification of secrets	No dependencies.	N/A
FIA_UAU.5-1 Multiple authentication mechanisms	No dependencies.	N/A
Class FMT: Security Management (FMT)		
FMT_MOF.1 Management of Security Functions Behavior	FMT_SMR.1 Security roles, FMT_SMF.1 Specification of management functions.	FMT_SMR.1, FMT_SMF.1
FMT_MSA.1 Management of security attributes	FDP_ACC.1 Subset access control or FDP_IFC.1 Subset information flow control], FMT_SMR.1 Security roles, FMT_SMF.1 Specification of Management Functions (CCIMB 065)	FDP_ACC.1, FMT_SMR.1, FMT_SMF.1
FMT_MSA.3 Static attribute initialisation	FMT_MSA.1 Management of security attributes, FMT_SMR.1 Security roles	FMT_MSA.1, FMT_SMR.1
FMT_MTD.1-1 Management of TSF data	FMT_SMR.1 Security roles, FMT_SMF.1 Specification of management functions	FMT_SMR.1, FMT_SMF.1

TOE Security Functional Requirements	Dependencies	How met
FMT_SMF.1 Specification of management functions	No Dependencies	N/A
FMT_SMR.1 Security roles	FIA_UID.1 Timing of identification	FIA_UAU.5-1, FIA_UAU.5-2
Class FPT: Protection of the TOE Security Functions		
FPT_RVM_EXP.1-1 Non-bypassability of the TSP: TOE	No dependencies	N/A
FPT_SEP_EXP.1-1 TSF domain separation: TOE	No dependencies	N/A

Table 8-8 IT Environment Dependencies are Satisfied

IT Security Functional Requirements	Dependencies	How met
Class FAU: Security Audit		
FAU_STG.1 Protected audit trail storage	FAU_GEN.1 Audit data generation	FAU_GEN.1
Class FIA: Identification and authentication		
FIA_UAU.5-2 Multiple authentication mechanisms	No dependencies.	N/A
Class FMT: Security management		
FMT_MTD.1-1 Management of TSF data	FMT_SMR.1 Security roles, FMT_SMF.1 Specification of management functions	Procedural exclusion of non-administrative users from IT security functions, FMT_SMF.1
Class FPT: Protection of the TSF		
FPT_RVM_EXP.1-2 Non-bypassability of the TSP	No dependencies	N/A
FPT_SEP_EXP.1-2 TSF domain separation	No dependencies	N/A
FPT_STM.1 Reliable time stamps	No dependencies	N/A

8.2.3 Rationale why dependencies are not met

Several SFR have a dependency on FIA_UID.1, which is mapped to FIA_UAU.5. This requirement is iterated in both the IT environment and in the TSF. In combination, both iterations of FIA_UAU.5 cover all users and insure that all users are identified and authenticated before any TSF mediated actions are permitted on the behalf of the user.

8.2.4 Strength of Function Rationale

Part 1 of the CC defines “strength of function” in terms of the minimum efforts assumed necessary to defeat the expected security behavior of a TOE security function. There are three strength of function levels defined in Part 1: SOF-basic, SOF-medium and SOF-high. SOF-basic is the strength of function level chosen for this ST. SOF-basic states, “A level of the TOE strength of function where analysis shows that the function provides adequate protection against casual breach of TOE security by attackers possessing a low attack

potential." The rationale for choosing SOF-basic was to be consistent with the assurance requirements included in this ST; namely the environment is one where the potential attacker is unsophisticated, with access to only standard equipment and public information about the product, consistent with a Common Criteria Level of Evaluation of EAL2. Specifically, AVA_VLA.1 requires that the TOE be resistant to an attacker with a low to moderate attack potential, this is consistent with SOF-basic. Consequently, the metrics (password) chosen for inclusion in this ST for this TOE were determined to be acceptable for SOF-basic and would adequately protect information in a Basic Robustness Environment.

The one security function based on probabilistic methods is identified in Section 6.1.3.2 in Table 6-3 and applies to FIA_SOS.1 (*see* Section 5.2.3). The specific "strength" required of the methods used provide difficult-to-guess passwords.

8.2.5 Assurance Rationale

Evaluation Assurance Level EAL2 was chosen to provide a basic level of assurance due to the value of the information assets and the low level threat of malicious attack potential, including confidence that the TOE will not be tampered with during delivery. . Violation of the information protection policy would cause minor damage to the security, safety, financial posture, or infrastructure of the organization. The most capable threat agents are presumed to be unsophisticated adversaries with only standard equipment and public information about the product who are willing to take little risk, e.g., unsophisticated hackers.

8.2.6 Rationale that IT Security Requirements are internally Consistent

The IT Security Requirements are internally consistent. There are no requirements that conflict with one another. When different IT security requirements apply to the same event, operation, or data there is no conflict between the security requirements. The requirements mutually support each other to apply to the configuration of the TOE, review of audit logs of TOE configuration and IT environment operations, protection of the audit trail, protection of TOE and user data transmitted between TOE components, authentication of users configuring the TOE, and supporting domain separation and non-bypassability.

FAU_STG.1 Protected audit trail storage. The IT environment does not permit the deletion or modification through the IT environment interface of the audit logs, except by an authorized administrator.

FIA_UAU.5 Multiple authentication mechanisms, ensures that that the IT environment minimally sets up an encrypted channel with all users prior before any action. Setting up the encrypted channel also minimally identifies the client by presumed IP address. In the case where the IT environment is configured to verify the user's identify visa certificate based authentication, the IT environment also associates certificate based security attributes with the external user. Administrative users will be identified thorough a direct console connection.

FMT_SMR.1-2 Security roles. The TOE must support the Identification and authentication of Administrative personnel who will configure certain TOE security parameters through the IT environment interfaces.

FPT_RVM_EXP.1-2 Non-bypassability of the TSP. Each communication session from an external client to the Integration server must go through an encryption process. An encrypted connection from the IT Server platform to the client is always required. There is no way for communication to flow between the client and the host without establishing this encrypted connection, since the TOE or IT environment is configured to require this as a minimum. The

client can either be authenticated using certificate based authentication (implemented in the IT environment), or via UID and password (implemented in the TOE). This combination of encryption and user authentication ensures that the TSF functions are always invoked.

FPT_SEP_EXP.1-2 TSF domain separation, ensures that TSF executes in its separated security domain, which is protected from interference and tampering by untrusted subjects. External users are not permitted to access the IT environment components interfaces. Once an external user is identified (and optionally authenticated using certificate based authentication), his/her encrypted connection is passed to the TOE. If the encrypted connection to the TOE is broken (i.e. the TOE crashes), the external user's connection is closed.

FPT_STM.1 Reliable time stamps. The IT environment provides reliable time stamps so that audit logs contain reliable time stamps for each record. This dependency is not satisfied by the TOE because the underlying operating system and hardware will provide the correct and reliable time. The system administrator, among his duties in managing and maintaining the security of the TOE, will keep the operating system time correct.

8.2.7 Requirements for the IT Environment

Table 8-9 shows that all of the security objectives for the IT environment are satisfied. Rationale for each objective is included below the table.

Table 8-9 All Objectives for the IT Environment map to Requirements in the IT environment

Identifier	Objective	SFR	Rationale
OE.Audit	The IT environment must provide a means to record a readable audit trail of security-related events, with accurate dates and times.	FMT_MTD.1-2 Management of TSF data, FPT_STM.1Reliable time stamps	FMT_MTD.1-2 provides that the IT environment will audit authentication through the operating system., FPT_STM.1 provides for reliable time stamps.
OE.AuditProtection	The IT environment will provide the capability to protect audit information.	FAU_STG.1 Protected audit trail storage	FAU_STG.1 provides that the IT environment will prevent unauthorized users from assessing or modifying the audit trail.
OE.IDAuth	The IT environment must be able to identify and authenticate users prior to allowing access to IT environment functions and data.	FIA_UAU.5-2 Multiple authentication mechanisms	FIA_UAU.5-2 provides that users will be identified before being able to invoke the IT environment's functions.
OE.NonBypass	The IT environment must ensure the IT environment's security functional policy is invoked and succeeds before allowing another IT environment function to proceed.	FPT_RVM_EXP.1-2 Non-bypassability of the TSP	FPT_RVM_EXP.1-2 provides that the IT environment will support the TOE's non-bypassability functions

Identifier	Objective	SFR	Rationale
OE.PartialSelfProtection	The IT Environment will maintain a domain for its own execution that protects itself and its resources from external interference, tampering, or unauthorized disclosure, through its own interfaces.	FPT_SEP_EXP.1-2 TSF domain separation	FPT_SEP_EXP.1-2 provides that the IT environment will support the TOE's self protection functions
OE.ProtectComm[11]	The IT environment must protect communications between the TOE and its users, and between TOE components.	FIA_UAU.5-2 Multiple authentication mechanisms	FIA_UAU.5-2 provides that users will be identified as part of setting up protected communication between the IS server and the external user.
OE.Time	The underlying operating system must provide reliable time stamps.	FPT_STM.1 Reliable time stamps	FPT_STM provides that reliable time stamps shall be provided.

8.3 TOE Summary Specification Rationale

8.3.1 IT Security Functions

Table 8-10 shows that the IT Security Functions in the TOE Summary Specification (TSS) address all of the TOE Security Functional Requirements.

Table 8-10 Mapping of Functional Requirements to TOE Summary Specification

Functional Component	Functional Requirement	TSF Ref. No	TSF Ref. Title	Rationale
FAU_GEN.1	Audit data generation	AU-1	Audit trail	Specifies the types of events to be audited
FAU_SAR.1	Audit review	AU-2	Audit review	Specifies who has the capability to read information from the audit records.
FAU_SAR.2	Restricted audit review	AU-3	Restricted audit review	Specifies that only specific users have read access to the audit records.
FAU_SAR.3	Selectable audit review	AU-4	Selectable audit review	Specifies audit selection features for IS Admin.
FAU_SEL.1	Selective audit	AU-5	Selective audit	Specifies audit events that can be included or excluded from the audit logs
FDP_ACC.1	Subset access control	AC-1	Access control function	Specifies the subjects and objects controlled under the Integration Server Access Control SFP

[11] The mapping does not address communication between TOE components. SFR (e.g.: FPT_ITT.1 and FDP_ITT.1) to support the protection of intra-TOE communication channels since the end user could use physical means (as was shown in Figure 2-2), or isolation of the TOE (as was shown in Figure 2-4) as an alternative to IT means to support OE.ProtectComm.

Functional Component	Functional Requirement	TSF Ref. No	TSF Ref. Title	Rationale
FDP_ACF.1	Security attribute based access control	AC-1	Access control function	Specifies the Integration Server Access Control SFP
FIA_ATD.1	User attribute definition	IA-1	Security Attributes	Specifies the security attributes maintained for each user.
FIA_SOS.1	Verification of secrets	IA-2	Password Policy	Specifies the mechanism for secret generation.
FIA_UAU.5-1	Multiple authentication mechanisms	IA-3	User authentication	Specifies the authentication mechanisms and that each user must be authenticated being allowed any other actions.
FMT_MOF.1	Management of Security Functions Behavior	SM-1	Management of Security Functions	Specifies that the ability to manage the behavior of security functions is restricted to the Authorized Administrator.
FMT_MSA.1	Management of security attributes	SM-2	Management of security attributes	Specifies that the ability to query, modify, delete, or create the specified security attributes is restricted to the specified users
FMT_MSA.3	Static attribute initialisation	SM-3	Default Values of Security Attributes	Specifies that restrictive default values for security attributes are provided and only the IS Administrator can specify alternative initial values.
FMT_MTD.1-1	Management of TSF data	SM-4	Management of TSF Data	Specifies that the Integration Server restricts the ability to access TSF data to the Authorized Administrator.
FMT_SMF.1	Specification of management functions	SM-5	Specification of Management Functions	Specifies the security management functions to determine the behaviour of security functions, security attributes, and TSF data.
FMT_SMR.1	Security roles	SM-6	Security Roles	Specifies the roles maintained in the webMethods Servers.
FPT_RVM_EXP.1-1	Non-bypassability of the TSP: TOE	SP-1	Non-bypassability	Specifies the TOE ensures that the webMethods IS Access Control SFP is invoked and succeeds before each function is allowed to proceed.
FPT_SEP_EXP.1-1	TSF domain separation: TOE	SP-2	TSF domain separation	Specifies that the TOE maintains a security domain for its own execution and enforces separation between security domains of users.

8.3.2 Assurance Measures

Table 5-7 of this document identifies the Assurance Measures implemented by webMethods to satisfy the assurance requirements of EAL2 as delineated in the table in Annex B of the CC, Part 3. Table 6-7 maps the Assurance Requirements with the Assurance Requirement's Measures. The assurance measures rationale shows how all assurance requirements are satisfied. The rationale is provided in Table 8-11. SAR dependencies identified in the CC

have been met by this ST as shown in Note: The actual documentation which satisfy the SAR, for example Configuration Management, is referenced in Table 6-7.
Table 8-12.

Table 8-11 Assurance Measures Rationale

Assurance Measure	Configuration Management	Delivery and Operation	Development	Guidance	Test	Vulnerability Assessment
ACM_CAP.2	X					
ADO_DEL.1		X				
ADO_IGS.1		X				
ADV_FSP.1			X			
ADV_HLD.1			X			
ADV_RCR.1			X			
AGD_ADM.1				X		
AGD_USR.1				X		
ATE_COV.1					X	
ATE_FUN.1					X	
ATE_IND.2					X	
AVA_SOF.1						X
AVA_VLA.1						X

Note: The actual documentation which satisfy the SAR, for example Configuration Management, is referenced in Table 6-7.

Table 8-12 EAL2 SAR Dependencies Satisfied

Assurance Component ID	Assurance Component Name	Dependencies	Satisfied
ACM_CAP.2	Configuration items	None	NA
ADO_DEL.1	Delivery procedures	None	NA
ADO_IGS.1	Installation, generation, and start-up procedures	AGD_ADM.1	YES
ADV_FSP.1	Informal functional specification	ADV_RCR.1	YES
ADV_HLD.1	Descriptive high-level design	ADV_FSP.1, ADV_RCR.1	YES
ADV_RCR.1	Informal correspondence demonstration	None	YES
AGD_ADM.1	Administrator guidance	ADV_FSP.1	YES
AGD_USR.1	User guidance	ADV_FSP.1	YES
ATE_COV.1	Evidence of coverage	ADV_FSP.1, ATE_FUN.1	YES
ATE_FUN.1	Functional testing	None	NA

Assurance Component ID	Assurance Component Name	Dependencies	Satisfied
ATE_IND.2	Independent testing-sample	ADV_FSP.1, AGD_ADM.1, AGD_USR.1, ATE_FUN.1	YES
AVA_SOF.1	Strength of TOE security function evaluation	ADV_FSP.1, ADV_HLD.1	YES
AVA_VLA.1	Developer vulnerability analysis	ADV_FSP.1, ATE_HLD.1 AGD_ADM.1, AGD_USR.1	YES

8.4 PP Claims Rationale

Not applicable. There are no PP claims.

8.5 Explicitly Stated Requirements Rationale

The explicitly stated requirements FPT_RVM_EXP.1-1 Non-bypassability of the TSP: TOE, FPT_SEP_EXP.1-1 TSF domain separation: TOE, FPT_RVM_EXP.1-2 Non-bypassability of the TSP: IT environment, and FPT_SEP_EXP.1-2 TSF domain separation: IT environment, were added because the TOE does not completely support either non-bypassability or domain separation with out support from the IT environment.

9 Acronyms

Table 9-1 Acronyms

Acronym	Definition
ACL	Access Control Notes
CC	Common Criteria [for IT Security Evaluation]
CCIMB	Common Criteria Implementation Management Board
CCTL	Common Criteria Testing Lab
DB2	Database 2
DSP	Dynamic Server Page
EAL	Evaluation Assurance Level
ebXML	Electronic Business Extensible Markup Language
EDI	Electronic Data Interchange
FTP	File Transport Protocol
FTPS	File Transport Protocol Secure sockets
GNU	GNU
GUI	Graphical User Interface
HTTP	Hypertext Transfer Protocol (world wide web protocol)
HTTPS	Hyper Text Transfer Protocol Secure sockets
ID	Identifier
IETF	Internet Mail Extensions
IS	Integration Server
I&A	Identification and Authentication
IT	Information Technology
JDBC	Java Database Connectivity
JMS	Java Message Service
JRE	Java Runtime Environment
JSP	Java Server Page
JVM	Java Virtual Machine
LDAP	Light Weight Directory Access Protocol
MIME	Multipurpose Internet Mail Extensions
NIS	Network Information Services
OS	Operating System
PP	Protection Profile
RPC	Remote Procedure Call
S/MIME	Secure/Multipurpose IETF
SF	Security Function
SFP	Security Function Policy
SQL	Structured Query Language (database query language)
SSL	Secure Sockets Layer (web security protocol)
ST	Security Target
TLS	Transaction Layer Security
TOE	Target of Evaluation
TSC	TSF Scope of Control

Acronym	Definition
TSF	TOE Security Functions
TSFI	TOE Security Functions Interface
TSP	TOE Security Policy
TSS	TOE Summary Specification
UID	User Identifier

10 Bibliography

Common Criteria for Information Technology Security Evaluation Parts 1-3, CCIMB-2004-01-001, 002 and 003, Version 2.2, Revision 256, January 2004

Common Methodology for Information Technology Security Evaluation, CCIMB-2004-01-004, Version 2.2, Revision 256, January 2004

webMethods Broker Administrator's Guide, Version 6.5, Document ID: BR-AG-61-20040116BR-AG-65-20050615

webMethods Developer User's Guide, Version 6.5, Document ID: DEV-UG-61-20040116 DEV-UG-65-20050429

webMethods Fabric Integration Platform Error Message Reference, Document ID: PLAT-EM-RF-601-20031201

webMethods Integration Server Administrator's Guide, Version 6.5, Document ID: webM-IS-AG-20040116webM-IS-AG-65-20050429

webMethods Integration Server Built-In Services Reference, Version 6.5, Document ID: IS-BIS-RF-61-20040116IS-BIS-RF-65-20050330

webMethods JDBC Adapter User's Guide, Document ID: ADAPTER-JDBC-UG-603-20040511

webMethods JMS Adapter User's Guide, Document ID: ADAPTER-JMS-UG-61-20040213