Tivoli Security Policy Manager Version 7.1
Security Target

Document Version
Version: 1.24
2013-10-31

Trademarks

Legal Notice

This document is provided AS IS with no express or implied warranties. Use the information in this document at your own risk.

Revision History

Table 1 – Revision History

Version	Date	Author	Changes	Notes
1.24	2013-10-31	G. McIntosh	Section 1.4.2 Corrected a broken link Corrected minor grammatical errors Removed Classification Note Removed revision history from v 0.91 to 1.23 per request from BSI Added Subject and Language properties	

Table of Contents

Tables

Figures

1 Security Target (ST) Introduction

The structure of this document is defined by CC v3.1r3 Part 1 Annex A.2, “Mandatory contents of an ST”:

- Section 1 contains the ST Introduction, including the ST reference, Target of Evaluation (TOE) reference, TOE overview, and TOE description.
- Section 2 contains conformance claims to the Common Criteria (CC) version, Protection Profile (PP) and package claims, as well as rationale for these conformance claims.
- Section 3 contains the security problem definition, which includes threats, Organizational Security Policies (OSP), and assumptions that must be countered, enforced, and upheld by the TOE and its operational environment.
- Section 4 contains statements of security objectives for the TOE, and the TOE operational environment as well as rationale for these security objectives.
- Section 5 contains definitions of any extended security requirements claimed in the ST.
- Section 6 contains the security function requirements (SFR), the security assurance requirements (SAR), as well as the rationale for the claimed SFR and SAR.
- Section 7 contains the TOE summary specification, which includes the detailed specification of the IT security functions

1.1 Security Target Reference

The Security Target reference shall uniquely identify the Security Target.

ST Title:	Tivoli Security Policy Manager Version 7.1 Security Target
ST Version Number:	Version 1.24
ST Author(s):	Gordon McIntosh
ST Publication Date:	2013-10-31

Keywords: Security Policy Management, Access Control

1.2 Target of Evaluation Reference

The Target of Evaluation reference shall identify the Target of Evaluation.

TOE Developer	IBM 11501 Burnet Road IBM Building 101 Austin, Texas, 78758
TOE Name:	Tivoli Security Policy Manager
TOE Version	7.1.0.4 APAR IV44553

1.3 Target of Evaluation Overview

1.3.1 TOE Product Type

The TOE is a software product; the type is Security Management, specifically Security Policy Management.

1.3.2 TOE Usage

The intended usage of the TOE is to manage access to resources by defining and enforcing security policies. In the evaluated configuration, the TOE can manage the following types of policies:

- Authorization policies that protect Web services
    - An authorization policy is a set of conditions that define whether a user should be permitted or denied access to a protected resource[1].
    - These policies are based on extensible access control markup language (XACML).[2]

1.3.3 TOE Major Security Features Summary

- Security Audit
    - Reports security relevant events
- User data protection
    - Provides access control mechanisms to limit access of the management functions to authorized administrators, and limit users' access to protected Web Services based on the policy established by administrators.
        - Web Services are web APIs that can be called using Hypertext Transfer Protocol (HTTP); where the Web Service API is represented as a Web Service Definition Language (WSDL) file which describes the interfaces provided by the web service. Web Service messages are exchanged as Simple Object Access Protocol (SOAP) over HTTP. A service requester (web service client) calls the API provided by the service provider (web service).
- Security Management
    - Provides authorized administrators tools to manage the security features provided by the TOE

1.3.4 TOE IT environment hardware/software/firmware requirement summary

The TOE IT operational environment is required to provide support for the TOE security functions. The TOE is an application running in the runtime environment provided; therefore, it relies on security functions provided by the environment for protection and support of security functions implemented by the TOE. Additionally, the IT Environment is required to provide a management console for the TOE.

1.3.4.1 Required runtime environment

The TOE does not have any direct dependencies on hardware platforms because the TOE's runtime environment provides an abstraction layer from the physical hardware making it unnecessary to specify as part of the evaluated configuration. The runtime environment is comprised of an underlying operating system, an application framework that provides additional services required by the TOE, and supporting services.

The following is the required runtime environment for the TOE:

- Red Hat Enterprise Linux 5.5 Operating System[3]
    - Shown as "RHEL 5.5 OS" in in Figure 1
    - Provides an execution domain for the TOE and IBM WebSphere Application Server (WAS)
- IBM WebSphere Application Server Version 7.0 (WASv7.0); provides support for:

[1] Refer to Section 1.4, Target of Evaluation Description

[2] The OASIS XACML v2.0 specification defines this language [23].

[3] RHEL 64-bit

  - Tivoli Security Policy Manager Server
  - Runtime Security Services Server (RTSS) Server
  - Runtime Security Services Server (RTSS) Client
  - Shown as “WAS” in Figure 1
- IBM Tivoli Integrated Portal Version 1.1 (TIPv1.1)
  - Provides support for the TSPM Management Console component
  - Includes Embedded WebSphere Application Server Version 6.1 (eWASv6.1), shown as “eWAS” in in Figure 1
- IBM DB2® Workgroup Server Edition Version 9.5 or above
  - Shown as the “Policy database” in Figure 1
- IBM Tivoli Directory Server Version 6.2 or above
  - Shown as “Administrator User Registry” in Figure 1
  - Manage the user IDs and passwords of Tivoli® Security Policy Manager administrators
- IBM Tivoli Common Reporting (TCR)
  - TCR is an internal component of TIPv1.1 that an administrative user may use to generate reports; it is not shown Figure 1

1.3.4.1.1 Red Hat Operating System-provided security functionality:

The operating system is responsible for providing the execution domain for the TSPM components and the WebSphere Application Server (WAS) that protects them from interference and tampering by unauthorized subjects.
The OS provides access control mechanisms to limit access to the TOE, eWAS, WAS, and TIP binary files, TSF data stored as configuration files, database files, audit files, etc., to authorized subjects.

1.3.4.1.2 IBM WebSphere Application Server-provided security functionality

TSPM components run in a WebSphere Application Server (eWAS/WAS[4]) environment that is not part of the TOE, but has been subject to a separate Common Criteria evaluation. WAS provides the following relevant security functionality for the TOE:

- Authentication framework. WAS provides an authentication and authorization framework for the TSPM components. Users are authenticated through WAS based on attributes managed on an LDAP server, IBM Tivoli Directory Server Version 6.2. WAS provides user-subject binding for both TIP and TSPM; both obtain the subject name at runtime via calls to eWAS.
- Role-based access control to user interface. WAS allows to restrict access to the individual views implemented by the TSPM Management console roles that are defined during installation of the TSPM Management console. Management of roles and enforcement of access to Web pages based on these roles is performed by WAS.
- Protection of the TSF and TSF data, and domain separation. In the same fashion as the operating systems above are responsible for protecting the native applications running on top of them, WAS is responsible for protecting the TSPM components from unauthorized access and interference.
- TLS connectivity. WAS provides the cryptographic and protocol facilities for TLS connections used by the TSPM components, including the IBMJCEFIPS and IBMJSSEFIPS cryptographic modules.
- Auditing framework. The TSPM components use the auditing facility provided by WAS for recording any TSPM-generated audit records. This requires the configuration of the correct syslog level in WAS.

1.3.4.1.3 IBM Tivoli Integrated Portal -provided security functionality

The TSPM Management Console runs in an IBM Tivoli Integrated Portal (TIP) environment that is not part of the TOE, but that has been subject to a separate Common Criteria evaluation. TIP provides the following relevant security functionality for the TOE:

[4] eWAS – Refers to the embedded version of WAS

- TIP implements rule-based access control that controls the rendering of the GUI views of the TSPM Management console; these roles are defined during installation of the TSPM Management console.

1.3.4.1.4 Required management console

The TOE IT operational environment is required to provide support for the administration of the TOE as follows:

- Remote administrative interface via TLS/HTTPS
    - Java-enabled browser on customer (client) workstation for the GUI
        - Internet explorer 7 and/or
        - Internet explorer 8 in compatibility mode
    - Shown as "Web browser" in Figure 1

1.4 Target of Evaluation Description

This section describes the TOE physical and logical boundaries; the physical boundaries describe the TOE hardware, software and the related guidance documentation; the logical boundary describes what logical security features are included in the TOE.

Figure 1 - Typical TSPM Deployment Diagram, is a simplified diagram showing the TOE components and the TOE's relationship with the required IT environment in a typical deployment.

The TOE is comprised of the following components:

- The TSPM Policy Server, shown as the TSPM Server
    - The TSPM Server component acts as a Policy Administration Point (PAP).
- The TSPM Management Console
    - The TSPM Management Console is the primary management interface for the TOE.
    - To access the policy administration point, the administrators use the TSPM Management console to perform tasks such as authoring, configuring, and distributing policies.
- Runtime Security Services (RTSS) Server
    - The RTSS Server acts as a Policy Decision Point (PDP), and a Policy Distribution Target (PDT)
        - A PDP performs the following tasks during the evaluation of a request:
            - Evaluate an access request against a policy.
            - Decide whether access is to be permitted or denied.
        - A Policy Distribution Target is a location from which PDPs can get policies that have been authored and configured in the Policy Administration Point
- Runtime Security Services (RTSS) Client
    - The RTSS Client acts as a proxy to the Policy Decision Point (PDP) by calling the RTSS server for authorization decision..
        - The TOE supports JAX-WS Policy Enforcement Point (PEP) for WebSphere. JAX-WS PEP for WebSphere is included in the RTSS Client Installation package; JAX-WS is an API for creating web service applications. PEPs perform the following tasks during the evaluation of an access request:
            - Receive an access request.
            - Notify the PDPs of the request.
            - Receive the decision response from the PDP.
            - Enforce the decision by either permitting access or denying access to the request.

In the evaluated configuration, the TOE is able to protect Web Services resources; however, the following types of resources cannot be protected:

- Custom applications
- J2EE applications
- WebSphere Portal resources
- Microsoft SharePoint resources
- Databases

1.4.1 Target of Evaluation Physical Boundaries

The TOE base release, the fix pack, and the APAR[5] fix are delivered via secure download. The TOE base release requires an IBM account to download from the IBM Passport Advantage web portal (http://www.ibm.com/software/passportadvantage/); Fix Pack 4 is downloaded from IBM Fix Central (http://www-933.ibm.com/support/fixcentral/). In both cases, the Download Director applet is used to download the TOE securely. Refer to the TSPM Online Fix Pack 4 Guidance [7].

The APAR fix IV44553 is obtained separately from IBM support via sftp (Secure FTP); IBM support must be contacted to obtain the necessary credentials to authorize the download.

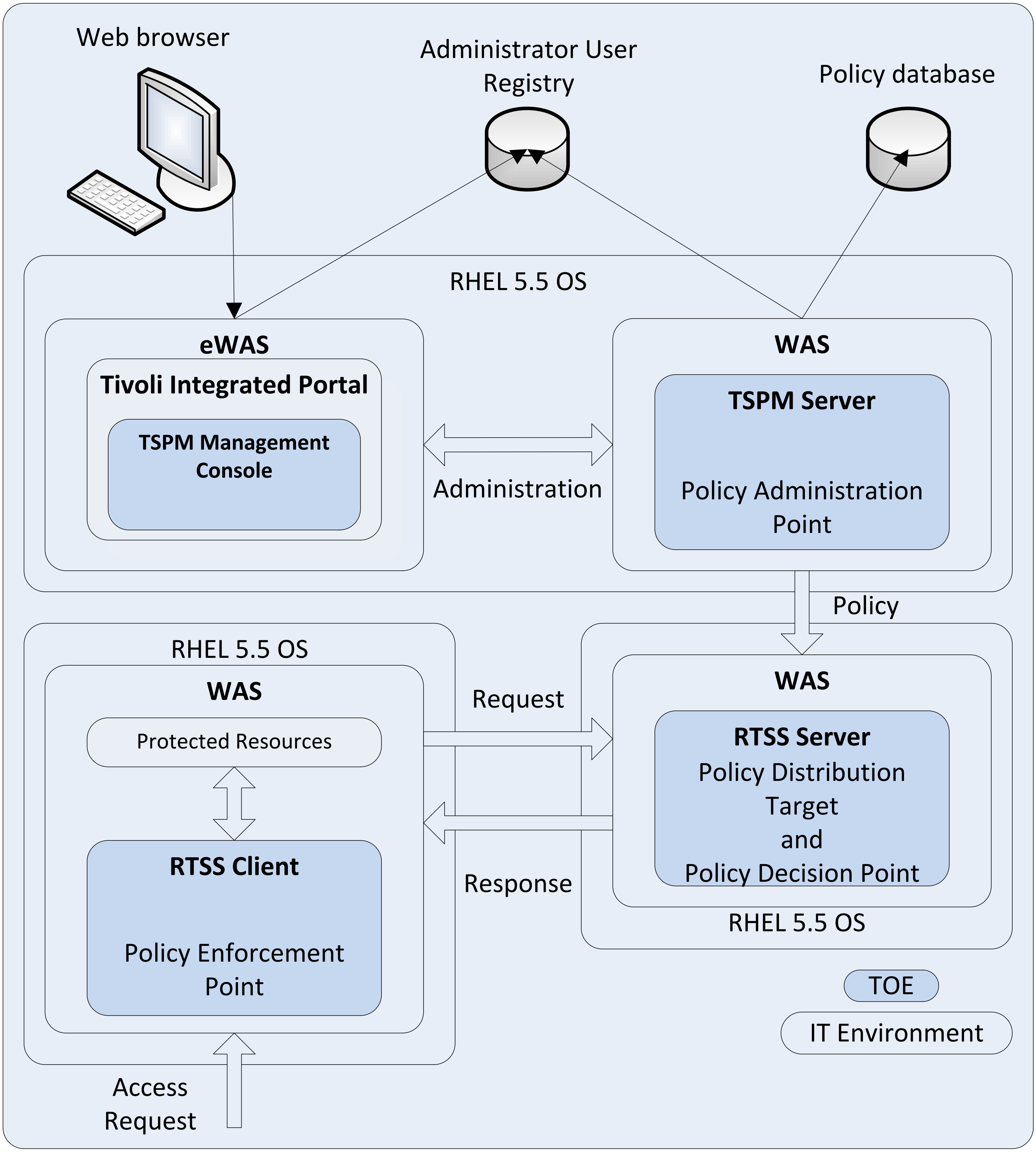
The TOE physical boundary is comprised of the following components:

- IBM Tivoli Security Policy Manager Version 7.1.0.4 APAR IV44553- policy manager, which includes:
    - TSPM Policy Server
    - TSPM Management Console
        - The TSPM Management Console is installed into the Tivoli Integrated Portal (TIP) environment; TIP is not part of the evaluated configuration.
- IBM Tivoli Security Policy Manager Version 7.1.0.4 APAR IV44553 runtime security services, which includes
    - Runtime Security Services Server (RTSS) Server
    - Runtime Security Services Server (RTSS) Client

In the evaluated configuration, the TSPM Policy server and TSPM Management Console run on the same computer.

[5] An Authorized Program Analysis Report, or APAR, is a formal report to IBM development of a problem caused by a suspected defect in a current release of an IBM program.

Figure 1 - Typical TSPM Deployment Diagram



1.4.2 TOE Guidance Documentation

The TOE guidance documentation delivered is listed in Table 24 - TOE Guidance Documentation.

1.4.3 Target of Evaluation Logical Boundaries

The logical boundaries of the TOE include those security functions implemented exclusively by the TOE. These security functions were summarized in Section 1.3.3 above and further described in the following subsections. A more detailed description of the implementation of these security functions is provided in Section 7, "TOE Summary Specification."

1.4.3.1 Audit services

The TOE is capable of auditing internal events by generating audit information for transactions that is stored in files protected by the IT environment. The information in the audit event records conforms to a standard data structure called a Common Base Event[6] (CBE).

The TSPM Management Console enables an authorized administrator to turn audit logging on or off and specify log file settings. Logging can be configured individually for the two Tivoli Security Policy Manager components: policy manager server and runtime security services.

1.4.3.2 User data protection

The TSF implements a rule-based access control mechanism to control access to TSPM management functions to authorized administrative users; additionally, the TSF implements a policy-based access control mechanism to control access to the Web Services protected by the TOE.

Authorized administrative users access the TSPM management functions using the TSPM Management console via a JAVA-enabled Web browser; enforcement of the rule-based access control mechanism is based on a permission attribute associated with the role(s) assigned to administrative users and groups to which users are assigned.

Users attempting to access Web Services protected by the TSF are subject to a policy-based access control mechanism implemented by the RTSS server and RTSS client components.

The RTSS server, acting as the Policy Decision Point, evaluates an access request based on authorization policies that contain custom application roles assigned to Web Services users and groups as well as rules defined by the policy. The application roles and policy rules may be specific to each installation and are created by the Policy Author using the Policy Administration Point. The RTSS Server passes the evaluation results to the JAX-WS PEP within RTSS Client which enforces the decision..

In the evaluated configuration, the authorization policy defines the conditions under which a user or service is permitted to perform an action on a protected Web Services resource. Authorization policies are based on the eXtensible Access Control Markup Language (XACML). Message protection policies are not part of the evaluated configuration.

1.4.3.3 Security Management

Primary management of the TOE's security relevant parameters is performed by an authorized administrator using the TSPM Management console with a JAVA-enabled Web browser via a secure HTTPS/TLS connection. The following management functions are available to administrative users:

- Policy Administration
- Rule Parameter Administration
- Classification Administration

[6] The CBE is a specification based on XML that defines a mechanism for managing events, such as logging, tracing, and management. For more information on the Common Base Event specification, go to the following Web site: http://www.eclipse.org/tptp/platform/documents/index.php

- Distribution Target Administration
- Application Role Administration
- Service Administration
- User Registry Administration
- Administrative Role Administration
- Policy Operations Administration
- General Administration
- Obligations Administration

Offline troubleshooting of the TOE's RTSS Client and Server components is possible using a JAVA-enabled Web browser via a secure HTTPS/TLS connection; although not a part of the evaluated configuration, it is mentioned for completeness.

The following offline troubleshooting functions are available to administrative users following instructions in the installation guide:

- View RTSS configuration.
- View policies on the RTSS server.
- Add and/or delete Policy directly to the RTSS server.
- Download "mustgather" information for support to look at.
- Refresh services configuration, the RTSS services are restarted.
- Refresh Authorization Policy Cache

1.4.4 Protection of the TSF

While in operation, the TOE depends on the underlying IT environment specified in Section 1.3.4, TOE IT environment hardware/software/firmware requirement summary.

The Operating System (OS) is necessary to provide an execution domain for the TSF, eWAS/WAS and TIP that protects them from interference and tampering by unauthorized subjects.

The TSF relies on the eWAS/WAS runtime environment to protect all external interfaces using secure communication protocols and to perform identification, authentication, and user-subject binding; these mechanisms allow only authenticated administrative users to access the TSPM Management Console and RTSS Client and Server management interfaces. These authenticated administrative users are assumed to be trusted and therefore do not represent a threat to the TOE.

The TSF implements rule-based access control mechanisms that requires the eWAS identified and authenticated administrators to be assigned to roles with the permissions required to perform the specific administrative task attempted. This allows multiple administrators with non-overlapping capabilities to implement separation of duties.

The IT environment is required to ensure that no general-purpose code is allowed to run on the computer on which the TSPM Server, the TSPM Management console, or the RTSS Server components execute.

The TOE and underlying hardware and firmware are required to be physically protected from unauthorized access.

The combination of the physical protection, I&A functions, access control mechanisms, secure external interfaces, provide sufficient protections such that the TSF cannot be bypassed, corrupted, or otherwise compromised.

1.5 Roles, User Data, and TSF Data

The TOE supports the following roles:

- TSPM Management Console Administrative roles

  - An administrative role is a collection of permissions that allow access to management functions. The TSF provides named administrative roles with pre-defined permissions; however, roles can be created, modified or deleted. Permissions are necessary to access each individual management function; permissions are maintained as an attribute assigned to each role and permissions cannot be created, modified, or deleted.
  - The TOE provides the following named TSPM Management administrator roles.
    - Application Administrator
    - Application Owner
    - Auditor
    - IT Environment Administrator
    - Policy Author
    - Policy Operator
    - Role Administrator
- Application (User) roles
  - Application roles are user roles created by the Policy Author and used in authorization policies to identify Web Service users and groups of users to whom the policy applies; application roles are mapped to users and groups via the user registry that is configured as part of the Tivoli Security Policy Manager environment.
  - A subject role is the role associated with a user requesting access to Web Services; this is an application role assigned during the policy configuration tasks,

TSF data includes the following:

- System configuration information
- Security attributes belonging to an administrator
- Security attributes belonging to a user
- Deployed Authorization Policy Data
- Audit Configuration
- Audit data

User data includes the following:

- Any data passed through the TOE that does not affect the operation of the TSF
- Authorization Policy Data prior to deployment

1.6 Notation, formatting, and conventions

The notation, formatting, and conventions used in this security target are defined below; these styles and clarifying information conventions were developed to aid the reader.

Where necessary, the ST author has added application notes to provide the reader with additional details to aid understanding; they are italicized and usually appear following the element needing clarification. Those notes specific to the TOE are marked “TOE Application Note;”.

The notation conventions that refer to iterations, assignments, selections, and refinements made in this security target are in reference to SARs and SFRs taken directly from CC Part 2 and Part 3 as well as any SFRs and SARs taken from a protection profile.

The CC permits four component operations: assignment, iteration, refinement, and selection to be performed on requirement components. These operations are defined in Common Criteria, Part 1; paragraph 6.4.1.3.2, “Permitted operations on components” as:

- Iteration: allows a component to be used more than once with varying operations;
- Assignment: allows the specification of parameters;
- Selection: allows the specification of one or more items from a list; and
- Refinement: allows the addition of details.

Iterations made by the ST author are indicated by a number in parenthesis following the requirement number, e.g., FIA_UAU.1.1 (1); the iterated requirement titles are similarly indicated, e.g., FIA_UAU.1 (1).

Assignments made by the ST author are identified with ***bold italics;*** selections are identified with **bold text.**

Refinements made by the ST author are identified with "**Refinement:**" right after the short name; the refined text indicated by underlined text; any refinement that performs a deletion in text is noted in the endnotes sections indicated.

2 Conformance Claims

2.1 Common Criteria Conformance Claims

This Security Target is conformant to the Common Criteria Version 3.1r3, CC Part 2 extended [11], and CC Part 3 [12] conformant.

2.2 Conformance to Protection Profiles

This Security Target does not claim conformance to any protection profile.

2.3 Conformance to Security Packages

This Security Target does not claim conformance to any security function requirements package, neither as package-conformant or package-augmented.

This Security Target claims conformance to Evaluation Assurance Level 2, augmented with ALC_FLR.3.

2.4 Conformance Claims Rationale

Because this security target does not claim conformance to a protection profile, no rationale is presented.

3 Security Problem Definition

3.1 Threats

The following subsections define the security threats for the TOE, characterized by a threat agent, an asset, and an adverse action of that threat agent on that asset.

The threat agents having an interest in obtaining or tampering with these assets can be categorized as either:

- Unauthorized individuals ("attackers") which are unknown to the TOE and its runtime environment
- Users of the TOE (i.e., users who have access to parts of the Web Services assets per the defined Web Services Access Control Policy) who try to access services that they are not authorized to access.

The assets to be protected by the TOE are the TOE's own TSF data, as well as the Web Services the TOE is intended to protect.

This evaluation intends to demonstrate that the TOE is able to withstand attackers with a "Basic" attack potential.

TOE administrators, including administrators of the TOE environment, are assumed to be trustworthy, trained and to follow the instructions provided to them with respect to the secure configuration and operation of the systems under their responsibility. Hence, only inadvertent attempts to manipulate the safe operation of the TOE are expected from this community.

3.1.1 Threats countered by the TOE and TOE IT Environment

Table 2 - Threats countered by the TOE and TOE IT Environment

#	Threat	Description
1	T.UNAUTH_ACCESS	An unauthorized user may gain access to system data because the system does not restrict access.
2	T.UNAUTH_MODIFICATION	An unauthorized user may cause the modification of the security enforcing functions in the system, and thereby gain unauthorized access to system and user resources due to the failure to protect its security enforcing functions

3.2 Organizational Security Policies

3.2.1 Organizational Security Policies for the TOE

An OSP is a set of rules or procedures imposed by an organization upon its operations to protect sensitive data.

Table 3 - Organizational Security Policies for the TOE and TOE IT Environment

#	OSP	Description
1	P.ACCOUNTABILITY	The users of the TOE shall be held accountable for security-relevant actions they have requested.
2	P.POLICY_MANAGER	The system must limit access to protected services based on the security policies defined by an authorized administrator.

3.3 Assumptions on the TOE Operational Environment

This section describes the assumptions that are made on the operational environment in which the TOE is intended to be used in order to be able to provide security functionality. It includes information about the physical, personnel, and connectivity aspects of the environment. The operational environment must be managed in accordance with the provided guidance documentation. The following subsections define specific conditions that are assumed to exist in an environment where the TOE is deployed.

3.3.1 Assumptions on Physical Aspects of the Operational Environment

Table 4 - Assumptions on Physical Aspects of the Operational Environment

Assumption	Description
A.PHYSICAL	It is assumed that the computer(s) providing the runtime environment for the TOE, and the systems in the TOE's operational environment, which provide support of the TOE, are protected against unauthorized physical access and modification.

3.3.2 Assumptions on Personnel Aspects of the Operational Environment

Table 5 - Assumptions on Personnel Aspects of the Operational Environment

Assumption	Description
A.INSTALL	It is assumed that the TOE is configured and operated in its evaluated configuration as defined in this Security Target and the TOE guidance.
A.ADMINISTRATOR	It is assumed that the administrators of the TOE, of the TOE's runtime environment and underlying operating system, and of the systems in the TOE's operational environment on which the TOE depends, in safeguarding TSF data, or providing functionality that the TOE depends on are assumed not to be careless, willfully negligent, or hostile. They will follow and abide by the instructions provided in the administrator guidance that is part of the TOE. They are well trained to securely and trustworthy administer all aspects of the TOE operation in accordance with this Security Target.

3.3.3 Assumptions on Connectivity aspects of the Operational Environment:

Table 6 - Assumptions on Connectivity Aspects of the Operational Environment

Assumption	Description
A.NO_GENERAL_PURPOSE	It is assumed that the machines providing the runtime environment for the server components of the TOE (i.e., all components other than those running on user hosts) are assumed to be used solely for this purpose and not to run other application software except as required for the support of the TOE and for the management and maintenance of the underlying system and hardware. Especially it is assumed that the underlying systems for all TOE components are configured in a way that prevents unauthorized access - either locally or via any network-based connections - to security functions, TSF and user data, including audit records generated by the TSF.
A.COMMUNICATION	It is assumed that communication between remote TOE components, and between the TOE and remote entities in the IT Environment are protected by secure communication protocols where possible, and that the protection of communication over inherently insecure protocols between the TOE and remote IT entities is protected by environmental means as appropriate for the operational environment.
A.RUNTIME	It is assumed the runtime environment provides the following support for the TOE: User identification and authentication

Table 6 - Assumptions on Connectivity Aspects of the Operational Environment

Assumption	Description
	User-subject binding Rule-based access control support for rendering GUI elements Audit support, including audit record formatting, timestamp, and storage

4 Security Objectives

4.1 Security Objectives for the TOE

Table 7 - Security Objectives for the TOE

#	TOE Objective	Description
1	O.AUDIT_GENERATION	The TSF shall offer a mechanism that can be used to hold users of the TOE accountable for specified security-relevant actions.
2	O.ADMIN_ACCESS	The TSF must control the access to management functions to authorized administrative users.
3	O.MANAGE	The TSF will provide functions and facilities necessary to support the administrators in their management of the security of the TOE. The TSF must allow authorized administrators to specify which management function may be accessed by which administrative user.
4	O.SERVICE_ACCESS	The TSF must control the access to protected Web Services to authorized users; access decisions must be based on policies specified by an authorized administrator.

4.1.1 Rationale for the Security Objectives for the TOE

4.1.1.1 Mappings of TOE Security Objectives to Threats and OSP

The following table shows the mapping of security objectives for the TOE to threats countered by that objective and/or the OSP enforced by that objective.

Table 8 - Mapping of TOE Security Objectives to Threats and OSP

		Threats		OSP	
#	TOE Objective	T.UNAUTH_ACCESS	T.UNAUTH_MODIFICATION	P.ACCOUNTABILITY	P.POLICY_MANAGER
1	O.AUDIT_GENERATION			X	
2	O.ADMIN_ACCESS	X	X		
3	O.MANAGE	X	X	X	X
4	O.SERVICE_ACCESS				X

4.1.1.2 Security Objectives Rationale for Threats and OSP

This section presents the rationale that justifies the security objectives for the TOE is suitable to counter those threats to be countered by the TOE and justifies the security objectives are suitable to enforce the OSP.

O.AUDIT_GENERATION

O.AUDIT_GENERATION contributes to satisfying the policy, P.ACCOUNTABILITY, by requiring the TSF generate audit record of security-relevant events.

O.ADMIN_ACCESS

O.ADMIN_ACCESS counters the threats, T.UNAUTH_ACCESS and T.UNAUTH_MODIFICATION by requiring the TOE control the access to management functions to authorized administrative users.

O.MANAGE

O.MANAGE contributes to countering the threats T.UNAUTH_ACCESS and T.UNAUTH_MODIFICATION; the objective requires the TSF provide the functions and facilities necessary to support the administrators in their management of the security of the TOE and requires that the TSF allow authorized administrators to specify which management function may be accessed by which user.

O.MANAGE contributes to satisfying the policies, P.POLICY_MANAGER and P.ACCOUNTABILITY, by ensuring that the TOE provides the functions and facilities necessary to support the administrators in their management of the security of the TOE.

O.SERVICE_ACCESS

O.SERVICE_ACCESS contributes to satisfying the policy, P.POLICY_MANAGER, by ensuring that the TOE control the access to protected Web Services to authorized users, and that access decisions must be based on policies specified by an authorized administrator.

.

4.2 Security Objectives for the TOE Operational Environmental

Table 9 - Security Objectives for the TOE Operational Environmental

#	Objective	Description
1	OE.SECURECOMMS	The runtime environment for the TOE must be able to securely transfer data between the servers and clients that comprise the TOE, and between data sources in the operational environment.
2	OE.AUTHENTICATION	The runtime environment for the TOE shall implement authentication mechanisms commensurate with the level of protection sought by the TOE, and provide authentication decisions for TOE users to the TSF.
3	OE.TIMESOURCE	The runtime environment shall provide a reliable time source for the TOE's use.
4	OE.ADMINISTRATORS	Those responsible for the operation of the TOE must ensure that administrators are not careless, willfully negligent, or hostile, and that they are well trained and will follow the provided administrator guidance to install, configure and operate the TOE and the TOE environment. This includes ensuring that all access credentials are protected against disclosure by the users of the TOE, and that only trusted authentication providers are used.
5	OE.NO_GENERAL_PURPOSE	Those responsible for the operation of the TOE must ensure that the systems hosting the TSPM Administrative Console, TSPM Server, and RTSS Server components are used solely for this purpose and configured

Table 9 - Security Objectives for the TOE Operational Environmental		
#	Objective	Description
		in a way that prevents unauthorized access to the TOE and any TSF and user data, including audit records generated by the TSF.
6	OE.PHYSICAL	The environment provides physical security commensurate with the value of the TOE and the data it contains.
7	OE.RUNTIME	The runtime environment provides the following support for the TOE: Identification, authentication, user-subject binding, rule-based access control, and GUI rendering support for administrative users accessing the TIP Management Console Identification, authentication and user-subject binding of application users for the RTSS Client Audit support, including audit record formatting, timestamp, and storage

4.2.1 Rationale for the Security Objectives for the TOE Operational Environment

4.2.1.1 Mappings of Security Objectives to Threats, OSP and Assumptions

Table 10 - Mapping of TOE Security Objectives to Threats, OSP, and Assumptions, shows the mapping of security objectives for the TOE operational environment to threats countered by that objective, the OSP enforced by that objective, and/or the assumption upheld by that objective.

Table 10 - Mapping of TOE Security Objectives to Threats, OSP, and Assumptions

		Threats		Assumptions						OSP	
#	TOE Objective	T.UNAUTH_ACCESS	T.UNAUTH_MODIFICATION	A.PHYSICAL	A.INSTALL	A.ADMINISTRATOR	A.NO_GENERAL_PURPOSE	A.COMMUNICATION	A.RUNTIME	P.ACCOUNTABILITY	P.POLICY_MANAGER
1	OE.SECURECOMMS		X					X			
2	OE.AUTHENTICATION	X									
3	OE.TIMESOURCE									X	
4	OE.ADMINISTRATORS				X	X					
5	OE.NO_GENERAL_PURPOSE						X				
6	OE.PHYSICAL			X							
7	OE.RUNTIME								X		

4.2.1.2 IT Security Objectives Rationale for Threats and OSP, and Assumptions

This section presents the rationale that justifies the security objectives for the TOE operational environment is suitable to counter those threats to be countered by the TOE operational environment, justifies the security objectives are suitable to enforce the OSP and the assumptions are upheld by that objective.

.

OE.SECURECOMMS

OE.SECURECOMMS addresses the assumption, A.COMMUNICATION, by ensuring the runtime environment for the TOE must be able to securely transfer data between the servers and clients that comprise the TOE, and between data sources in the operational environment.

OE.SECURECOMMS helps mitigate the threat, T.UNAUTH_MODIFICATION, by ensuring all external connections are protected by secure protocols that protect the TOE from attempts to circumvent the TSF.

OE.AUTHENTICATION

OE.AUTHENTICATION helps mitigate the threat, T.UNAUTH_ACCESS, by requiring the runtime environment to implement authentication mechanisms commensurate with the level of protection sought by the TOE, and provide authentication decisions for TOE users to the TSF.

OE.TIMESOURCE

OE.TIMESOURCE contributes to satisfying the policy, P.ACCOUNTABILITY, by ensuring the runtime environment provides a reliable time source for time stamping audit records.

OE.ADMINISTRATORS

OE.ADMINISTRATORS address the assumption, A.ADMINISTRATOR and A.INSTALL, by ensuring that those responsible for the operation of the TOE ensure that administrators are not careless, willfully negligent, or hostile, and that they are well trained and will follow the provided administrator guidance to install, configure and operate the TOE and the TOE environment. This includes ensuring that all access credentials are protected against disclosure by the users of the TOE, and that only trusted authentication providers are used.

OE.NO_GENERAL_PURPOSE

OE.NO_GENERAL_PURPOSE addresses the assumption, A.NO_GENERAL_PURPOSE, by ensuring that those responsible for the operation of the TOE ensure that the systems hosting the TSPM Administrative Console, TSPM Server, and RTSS Server components are used solely for this purpose and configured in a way that prevents unauthorized access to the TOE and any TSF and user data, including audit records generated by the TSF.

OE.PHYSICAL

OE.PHYSICAL addresses the assumption, A.PHYSICAL, by ensuring the environment provides physical security commensurate with the value of the TOE and the data it contains.

OE.RUNTIME

OE.RUNTIME addresses the assumption, A.RUNTIME, by ensuring the runtime environment provides the following:

- Identification, authentication, user-subject binding, rule-based access control, and GUI rendering support for administrative users accessing the TIP Management Console
- Identification, authentication and user-subject binding of application users for the RTSS Client
- Audit support, including audit record formatting, timestamp, and storage

5 Extended Components Definition

This section provides definition of the extended security functional and assurance requirements; the components that are CC Part 2 extended, and CC Part 3 extended, i.e., and extended requirements.

5.1 Extended Security Function Requirements Definitions

This section defines the extended security functional requirements for the TOE. The security functional requirement components defined in this security target are CC Part 2 extended.

Table 11 - TOE Security Functional Requirements CC Part 2 Extended

#	SFR	Description	Dependencies	Hierarchical to
1	FAU_GEN_SUB.1	Subset audit data generation	None	None
2	FDP_RBACC_EXT.1	Rule-based access control policy	FDP_RBACF_EXT.1	None
3	FDP_RBACF_EXT.1	Rule-based access control functions	FDP_RBACC_EXT.1 FMT_MSA.3	None
4	FMT_MOF_EXT.1	Management of functions in TSF	FMT_SMR.1 FMT_SMF.1	None

5.1.1 Class FAU: Security Audit

Security auditing involves recognizing, recording, storing, and analyzing information related to security relevant activities (i.e. activities controlled by the TSF). The resulting audit records can be examined to determine which security relevant activities took place and whom (which user) is responsible for them.

5.1.1.1 FAU_GEN_SUB Subset audit data generation

Family Behavior

This family defines a subset of the component FAU_GEN.1 as defined in part 2 of the CC. This extended component is required because TSPM uses the audit trail interfaces provided by the WebSphere Application Server (WAS) that are available for components that need to store their audit records in the common audit trail provided by WAS. While TSPM collects all the information for its own audit records and formats the audit records, it does not store the time and date into the audit record; this is performed by WAS when a component submits an audit record for inclusion into the WAS audit trail.

The extended component FAU_GEN_SUB.1 is a subset from the component FAU_GEN.1 as listed in part 2 of the CC. The requirement to include the time and date has been dropped. As a consequence of dropping the inclusion of the time and date, the extended component no longer has a dependency on FPT_STM.1

Component leveling

FAU_GEN_SUB Subset audit data generation
1

FAU_GEN_SUB.1 is not hierarchical to any other component.

Management: FAU_GEN_SUB.1
There are no management activities foreseen

Audit: FAU_GEN_SUB.1
There are no auditable events foreseen.

5.1.1.1.1 FAU_GEN_SUB.1 Subset audit data generation

Hierarchical to: None

Dependencies: None

FAU_GEN_SUB.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 [*selection, choose one of: minimum, basic, detailed, not specified*] level of audit; and

c) [*assignment: other specifically defined auditable events*].

FAU_GEN_SUB.1.2 The TSF shall record within each audit record at least the following information:

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

b) For each audit event type, based on the auditable event definitions of the functional components included in the ST, [*assignment: other audit relevant information*].

5.1.2 Class FDP: User data Protection

This class contains families specifying requirements related to the protection of user data.

5.1.2.1 FDP_RBACC_EXT Rule-based access control policy

Family Behavior

This family provides requirements defining a rule-based access control mechanism between subjects and functions; the functions perform operations on objects based on the subject's role and the permissions assigned to roles, permissions assigned explicitly to a user or group, or rules associated with users that can be evaluated to provide an access decision.

This is fundamentally different from traditional DAC or MAC as these mechanisms control access to objects, which are passive entities that contain or receive information; whereas, a function is an active entity that provides services to a subject. The subject only has to have permission to execute the function, the subject has no need to understand what underlying objects or functions are accessed, or the nature of the operations performed; all relationships between the subject and underlying objects or functions are hidden in the implementation of the function. In this model, it is the responsibility of the function's author to define and implement the necessary permissions for the correct operation of the function.

Component leveling

FDP_RBACC_EXT.1 Rule-based access control policy specifies the scope of control of the policy to require the TSF to control execution access to the functions that operate on objects based on the subject's role and the permissions assigned to roles, permissions assigned explicitly to a user or group, or rules associated with users that can be evaluated to provide an access decision.

Management: FDP_RBACC_EXT.1

There are no management activities foreseen

Audit: FDP_RBACC_EXT.1
There are no auditable events foreseen.

5.1.2.1.1 FDP_RBACC_EXT.1 Rule-based access control policy

Hierarchical to: None

Dependencies: FDP_RBACF_EXT.1 Rule-based access control functions

FDP_RBACC_EXT.1.1 The TSF shall enforce the [assignment: *access control SFP*] on [assignment: *list of subjects, functions, and required permissions]* for subjects to access functions covered by the SFP.

Application Note: *The assignment [*list of subjects, functions, and required permissions*], is asking for a list in the form {subject, function, required permission} such that each entry lists a function, the required permission the subject must have to access the function.*

5.1.2.2 FDP_RBACF_EXT Rule-based access control functions

Family Behavior

This family describes the rules for the specific functions that can implement an access control policy named in Rule-based access control policy (FDP_RBACC_EXT). Rule-based access control policy (FDP_RBACC_EXT) specifies the scope of control of the policy.

This is fundamentally different from traditional DAC or MAC as these mechanisms control access to objects, which are passive entities that contain or receive information; whereas, a function is an active entity that provides services to a subject. The subject only has to have permission to execute the function, the subject has no need to understand what underlying objects or functions are accessed, or the nature of the operations performed; all relationships between the subject and underlying objects or functions are hidden in the implementation of the function. In this model, it is the responsibility of the function's author to define and implement the necessary permissions for the correct operation of the function.

Component leveling

FDP_RBACF_EXT.1 Rule-based access control provides for the functionality to require the TSF to control execution access to the functions that operate on objects based on the permissions assigned to subjects, or permissions assigned to users and/or groups to which a subject is associated. Permissions may be assigned to subjects using a role assignment, where a role is a collection of permissions.

Management: FDP_RBACF_EXT.1
The following actions could be considered for the management functions in FMT:

- Assignment of default roles to users
- Explicit assignment of permissions to users/groups
- Explicit removal of permissions from users/groups

Audit: FDP_RBACF_EXT.1
The following actions should be auditable if FAU_GEN Security audit data generation is included in the ST:

a) Minimal: Successful requests to perform an operation on a function covered by the SFP
b) Basic: All requests to perform an operation on a function covered by the SFP
c) Detailed: The specific security attributes used in making an access check

5.1.2.2.1 FDP_RBACF_EXT.1 Rule-based access control functions

Hierarchical to: None

Dependencies: FDP_RBACC_EXT.1 Rule-based access control policy
FMT_MSA.3 Static attribute initialization

FDP_RBACF_EXT.1.1 The TSF shall enforce the [assignment: *access control SFP*] to authorize access of subjects to functions based on the [assignment: *permissions assigned to the roles or derived from evaluation of rules*] assigned to the subject.

FDP_RBACF_EXT.1.2 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled functions is allowed: [*assignment: rules governing access among controlled subjects and controlled functions*].

FDP_RBACF_EXT.1.3 The TSF shall explicitly authorize access of subjects to functions based on [assignment: *rules governing access among controlled subjects and controlled functions*].

FDP_RBACF_EXT.1.4 The TSF shall explicitly deny access of subjects to functions based on [assignment: *rules governing access among controlled subjects and controlled functions*].

5.1.3 Class FMT: Security management

This class is intended to specify the management of several aspects of the TSF: security attributes, TSF data and functions. The different management roles and their interaction, such as separation of capability, can be specified.

5.1.3.1 FMT_MOF Management of functions in TSF

Family Behavior
This family allows authorized users control over the management of functions in the TSF. Examples of functions in the TSF include the audit functions.

Component leveling

FMT_MOF_EXT Management of functions in TSF
1

FMT_MOF_EXT.1 Management of security functions behavior allows the authorized users (roles) to manage the behavior of functions in the TSF that use rules or have specified conditions that may be manageable.

Management: FMT_MOF_EXT.1
The following actions could be considered for the management functions in FMT:

a) Managing the group of roles that can interact with the functions in the TSF;

Audit: FMT_MOF_EXT.1
The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

a) Basic: All modifications in the behavior of the functions in the TSF

5.1.3.1.1 FMT_MOF_EXT.1 Management of security functions behavior

Hierarchical to: None

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

FMT_MOF_EXT.1.1 The TSF shall restrict the ability to [assignment: list of behaviors and functions] to [assignment: the authorized identified roles].

5.2 Extended Security Assurance Requirement Definitions

There are no extended Security Assurance Requirements defined in this Security Target.

5.3 Rationale for Extended Security Requirements

This section presents the rationale for the inclusion of the extended requirements found in this Security Target.

5.3.1 Rationale for Extended Security Function Requirements

The extended component FAU_GEN_SUB.1 is required because TSPM uses the audit trail interfaces provided by WAS that is available for components that need to store their audit records in the common audit trail provided by WAS. While TSPM collects all the information for its own audit records and formats the audit records, it does not store the time and date into the audit record; this is performed by WAS when a component submits an audit record for inclusion into the WAS audit trail.

The extended component FAU_GEN_SUB.1 is a subset from the component FAU_GEN.1 as listed in part 2 of the CC. The requirement to include the time and date has been dropped. Because dropping the inclusion of the time and date, the extended component no longer has a dependency on FPT_STM.1

The FDP_RBACC_EXT.1 and FDP_RBACF_EXT.1 SFRs listed in Table 11 are extended requirements; Part 2 of the Common Criteria does not include SFRs that describe the requirements for Rule-based access control. This is fundamentally different from traditional DAC or MAC as these mechanisms control access to objects, which are defined in CC Part 1 as passive entities that contain or receive information; whereas, a function is an active entity that provides services to a subject. The subject only has to have permission to execute the function, the subject has no need to understand what underlying objects or functions are accessed, or the nature of the operations performed; all relationships between the subject and underlying objects or functions are hidden in the implementation of the function. In this model, it is the responsibility of the function's author to define and implement the necessary permissions for the correct operation of the function.

FMT_MOF_EXT.1 is an extended requirement; it is necessary because the selections in FMT_MOF.1 require multiple iterations that may be confusing to the readers, especially when there are multiple security functions. Additionally, the extended requirement allows an assignment that allow better specification of the actions than "determine the behaviour".

5.3.2 Rationale for Extended Security Assurance Requirements

There are no extended Security Assurance Requirements defined in this ST; therefore, no rationale is presented.

6 Security requirements

This section describes the security functional and assurance requirements for the TOE; those that are CC Part 2 conformant, CC Part 2 extended, and CC Part 3 conformant.

6.1 Security Function Requirements

This section describes the functional requirements for the TOE. The security functional requirement components in this security target are CC Part 2 conformant or CC Part 2 extended as defined in Section 5, Extended Components Definition. Table 12 - TOE Security Functional Requirements, lists the SFRs included in this Security Target.

Table 12 - TOE Security Functional Requirements

#	SFR	Description	Operations
1	FAU_GEN_SUB.1	Subset audit data generation	A - S
2	FAU_GEN.2	User identity association	---
3	FDP_RBACC_EXT.1 (1)	Rule-based access control policy (TSPM Console, EJB)	A
4	FDP_RBACC_EXT.1 (2)	Rule-based access control policy (RTSS Client)	A
5	FDP_RBACF_EXT.1 (1)	Rule-based access control functions (TSPM Console, EJB)	A
6	FDP_RBACF_EXT.1 (2)	Rule-based access control functions (RTSS Client)	A
7	FMT_MOF_EXT.1	Management of security functions behavior (Audit, policy, roles)	A
8	FMT_MSA.1 (1)	Management of security attributes (Permissions)	A - S
9	FMT_MSA.1 (2)	Management of security attributes (Policy)	A - S
10	FMT_MSA.3 (1)	Static attribute initialization (Permissions)	A - S
11	FMT_MSA.3 (2)	Static attribute initialization (Policy)	A - S
12	FMT_MTD.1 (1)	Management of TSF data (Authorization Policy)	A - S
13	FMT_MTD.1 (2)	Management of TSF data (Audit Configuration)	A - S
14	FMT_SMF.1	Specification of Management Functions	A
15	FMT_SMR.1	Security roles	A

6.1.1 Class FAU: Security Audit

6.1.1.1 FAU_GEN Security audit data generation

6.1.1.1.1 FAU_GEN_SUB.1 Subset audit data generation

FAU_GEN_SUB.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:***

- ***RTSS access decision requests,***
- ***RTSS admin command received***
- ***RTSS notifications: update notifications, removal notification***
- ***RTSS policy: policy updated, removed***
- ***TSPM classification: Add, modify or delete a classification, classify or declassify a service, associate or disassociate a policy with a classification.***
- ***TSPM Policy: Add, modify, delete policies***
- ***TSPM Policy Association: Add association, delete an association***
- ***TSPM Service: Add a service, modify service attribute, or delete a service***
- ***TSPM PDT (PDT): Add PDT, modify PDT attribute, set PDT property, delete PDT***
- ***TSPM Policy Distribution: Distribute policy to PDT, redistribute policy to PDT, remove policy from PDT***
- ***TSPM Role: Add, modify, delete, add user to role, remove user from role, add group to role, delete group from role***
- ***TSPM Role Mapping: Add, Modify, Delete, add user to role mapping, remove user from role mapping, add group to role mapping, delete group from role mapping***
- ***TSPM Rule: Add, modify or delete Rule***
- ***TSPM Configured Rule - Add, Modify or delete a configured Rule***
- ***TSPM Admin Role: Add, modify, or delete administrative role;***
- ***TSPM Admin Role User: Use of admin role to add, remove, delegate, or undelegate user***
- ***TSPM Admin Role Group: Use of admin role to add, remove, delegate, or undelegate group***

FAU_GEN_SUB.1.2 The TSF shall record within each audit record at least the following information:

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

b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, ***no additional information.***

6.1.1.1.2 FAU_GEN.2 User identity association

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

6.1.2 Class FDP

6.1.2.1 FDP_RBACC_EXT Rule-based access control policy

6.1.2.1.1 FDP_RBACC_EXT.1 (1) Rule-based access control policy (TSPM Console, EJB))

FDP_RBACC_EXT.1.1 (1) The TSF shall enforce the ***TSPM Console Access Control (TCAC) SFP*** on ***processes acting on the behalf of administrative users (or groups) as subjects, the functions and the associated permissions listed in Table 13 – TSPM Functions and Permissions,*** for subjects to access functions covered by the SFP.

Table 13 – TSPM Functions and Permissions

Function	Permissions
Policy Administration	Create, delete, modify, and view policy
Rule Parameter Administration	Create, delete, and modify rule parameters
Classification Administration	Classify and declassify services, and create, modify, and delete classifications
Distribution Target Administration	Create, delete, and modify distribution targets
Application Role Administration	Create, delete, and modify application roles
Service Administration	Attach policy to services, detach policy from services, and create, delete, modify, and view services
User Registry Administration	Create, delete, and modify user registries
Administrative Role Administration	Assign, create, delete, and modify administrative roles
Policy Operations Administration	View, configure, and distribute policy
General Administration	Enable, disable, or modify the settings for runtime security services auditing and Tivoli Security Policy Manager auditing. Enable or disable reloading of the configuration for the runtime security services
Obligations Administration	Create, modify and delete obligations

6.1.2.1.2 FDP_RBACC_EXT.1 (2) Rule-based access control policy (RTSS Client)

FDP_RBACC_EXT.1.1 (2) The TSF shall enforce the ***RTSS Policy Access Control (RPAC) SFP*** on ***processes acting on the behalf of application users (or groups) as subjects, Web services under the control of this policy as functions, and permissions derived from the evaluation of Authorization Policies for that Service,*** for subjects to access functions covered by the SFP.

6.1.2.2 FDP_RBACF_EXT Rule-based access control functions

6.1.2.2.1 FDP_RBACF_EXT.1 (1) Rule-based access control functions (TSPM Console, EJB)

FDP_RBACF_EXT.1.1 (1) The TSF shall enforce the ***TCAC SFP*** to authorize access of subjects to functions based on the ***permissions assigned to the TSPM Administrative Roles, listed in Table 14 – TSPM Administrative Roles and Assigned Permissions,*** assigned to the subject.

Table 14 – TSPM Administrative Roles and Assigned Permissions

Administrative Role	Assigned Permissions
Application Administrator	Create classifications
	Classify services
	Create services
	View services
	Import Services
	Modify Services
	Declassify Services
	View policy
Application Owner	Attach policy to services
	Delete services
	Detach policy from services

	Modify services
	View services
	Create classification
	Classify services
	Declassify services
	View policy
Auditor	View policy
	View services
IT Environment Administrator	Create, delete, import, and modify registries
	Create, delete, and modify user registries
	Create, delete, and modify distribution targets
	Add, modify and delete policy information points.
	Enable, disable, or modify the settings for Runtime Security Services auditing
	Enable, disable, or modify the settings for Tivoli Security Policy Manager auditing
	Enable or disable reloading of the configuration for the Runtime Security Services, and general administration, which includes audit runtime security services, and reload runtime security services audit configuration.
Policy Author	Create, delete, modify, and view policy
	Create rule parameters
	Create application roles
	Create obligations
	View Services
Policy Operator	View policy
	Configure policy
	Distribute policy
	Configure distribution
	View services
	Add, delete and modify Policy Information Points.
Role Administrator	Assign, create, delete, and modify administrative roles.

FDP_RBACF_EXT.1.2 (1) The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled functions is allowed: ***If the role that is assigned to the subject possesses the required permissions to access a functions, access is allowed, otherwise, access is denied.***

FDP_RBACF_EXT.1.3 (1) The TSF shall explicitly authorize access of subjects to functions based on ***none***.

FDP_RBACF_EXT.1.4 (1) The TSF shall explicitly deny access of subjects to functions based on ***none*****.**

6.1.2.2.2 FDP_RBACF_EXT.1 (2) Rule-based access control functions (RTSS Client)

FDP_RBACF_EXT.1.1 (2) The TSF shall enforce the ***RPAC SFP*** to authorize access of subjects to functions based on the ***access decision resulting from the rule evaluation performed in FDP_RBACF_EXT.1.2 (2),*** assigned to the subject.

Application Note: *The rules to be evaluated are contained in an Authorization Policy associated with the function(s) that are attempting to be accessed. Authorization policies are based on extensible access control markup language (XACML).*

FDP_RBACF_EXT.1.2 (2) The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled functions is allowed:

1. ***If the subject role[7] is a match for the application role filter,***
    a. ***Evaluate #2, else deny access***

[7] A subject role is associated with the user requesting access; during the policy configuration tasks, the application roles are mapped to users and groups via the user registry that is configured as part of the Tivoli Security Policy Manager environment.

Application Note: *If the application role filter yields a match, the process continues with the next step. If the application role filter does not yield a match, this policy is not applicable because the user making the request is not a member of the role to which this policy applies. If another policy is attached to the service or service element, that policy is then evaluated for applicability beginning with this step. If no other attached policies yield a match for the application role filter, by default, the access request is denied.*

.

2. ***Evaluates the policy rules in the order specified, until all the conditions of a single rule are met:***
   a. ***If any of the conditions in a rule evaluate to false, the rule evaluates to not applicable and the next rule is evaluated.***
   b. ***If all the conditions of a rule evaluate to true, the access decision for the policy is returned and enforced.***
   c. ***If none of the specified rules yields an exact match, and all the rules are therefore considered "not applicable" to the request, the outcome of this policy is not applicable and access is denied.***
3. ***Returns either a Permit or Deny access decision based on the evaluation of the policy.***
4. ***If there are multiple policies attached to the service, then the next policy is evaluated.***
5. ***When all policies have been evaluated, their results are examined.***
   a. ***If any policy evaluates to Deny, the access request is denied.***
   b. ***If no individual policy evaluates to Deny, and at least one policy results in Permit, the access request is permitted.***
   c. ***If no individual policy evaluates to either Permit or Deny, and all attached policies result in "not applicable," by default, the access request is denied.***
6. ***The final access decision (either Permit or Deny) is then returned to the Policy Enforcement Point.***

FDP_RBACF_EXT.1.3 (2) The TSF shall explicitly authorize access of subjects to functions based on ***none***.

FDP_RBACF_EXT.1.4 (2) The TSF shall explicitly deny access of subjects to functions based on ***none*****.**

6.1.3 Class FMT Security Management

6.1.3.1 FMT_MOF Management of functions in TSF

6.1.3.1.1 FMT_MOF_EXT.1 Management of security functions behavior (Audit, policy, roles)

FMT_MOF_EXT.1.1 The TSF shall restrict the ability to ***perform the actions listed in column "Actions" in Table 15 – Administrative Roles and Actions*** to ***the associated role in column "Administrative Role"***.

Table 15 – Administrative Roles and Actions

Administrative Role	Actions
IT Environment Administrator	Enable, disable, or modify the settings for Runtime Security Services auditing
	Enable, disable, or modify the settings for Tivoli Security Policy Manager auditing
	Enable or disable reloading of the configuration for the Runtime Security Services, and general administration, which includes audit runtime security services, and reload runtime security services audit configuration.
Policy Author	Create, delete, modify, and view policy

	Create rule parameters
	Create application roles
	View policy
Policy Operator	View policy
	Configure policy
	Distribute policy
	Add policy information points
	Modify policy information points
	Delete policy information points
	Configure distribution
	View services
Role Administrator	Assign, create, delete, and modify administrative roles.

Application Note: *When Tivoli Security Policy Manager is installed and configured, two WAS administrative user groups are created; tspm_admin and tapm_user.*

The tspm_admin group is used for TSPM administrators who require all Tivoli Security Policy Manager permissions. Members of this group are considered to be the "super users" of TSPM. All TSPM permissions are assigned by default to users in this group.

The tapm_users group is used for TSPM administrators who are or will be assigned to specific, limited permissions in TSPM. No TSPM permissions are assigned by default to users in this group. No users are assigned to this group by default.

6.1.3.2 FMT_MSA Management of security attributes

6.1.3.2.1 FMT_MSA.1 (1) Management of security attributes (Permissions)

FMT_MSA.1.1 (1) The TSF shall enforce the ***TCAC SFP*** to restrict the ability to **modify, delete, *assign and create*** the security attributes ***permissions associated with administrative roles*** to ***Role Administrator***.

6.1.3.2.2 FMT_MSA.1 (2) Management of security attributes (Policy)

FMT_MSA.1.1 (2) The TSF shall enforce the ***TCAC SFP*** to restrict the ability to **modify, delete, *assign and create*** the security attributes ***Authorization Policy*** to ***Policy Author.***

6.1.3.2.3 FMT_MSA.3 (1) Static attribute initialization (Permissions)

FMT_MSA.3.1 (1) The TSF shall enforce the ***TCAC SFP*** to provide **permissive** default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2 (1) The TSF shall allow the ***Role Administrator*** to specify alternative initial values to override the default values when an object or information is created.

6.1.3.2.4 FMT_MSA.3 (2) Static attribute initialization (Policy)

FMT_MSA.3.1 (2) The TSF shall enforce the ***TCAC SFP*** to provide **restrictive** default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2 (2) The TSF shall allow the ***Policy Author*** to specify alternative initial values to override the default values when an object or information is created.

6.1.3.3 FMT_MTD Management of TSF data

6.1.3.3.1 FMT_MTD.1 (1) Management of TSF data (Authorization Policy)

FMT_MTD.1.1 (1) The TSF shall restrict the ability to **modify, *distribute*** the ***Authorization Policy*** to ***Policy Operator.***

6.1.3.3.2 FMT_MTD.1 (2) Management of TSF data (Audit Configuration)

FMT_MTD.1.1 (2) The TSF shall restrict the ability to **modify, *reload*** the ***Audit Configuration*** to ***IT Environment Administrator.***

6.1.3.4 FMT_SMF Specification of Management Functions

6.1.3.4.1 FMT_SMF.1 Specification of Management Functions

FMT_SMF.1.1 The TSF shall be capable of performing the following management functions listed in ***the "Assigned Permissions" column of*** **Table 14 – TSPM Administrative Roles and Assigned Permissions.**

6.1.3.5 FMT_SMR Security management roles

6.1.3.5.1 FMT_SMR.1 Security roles

FMT_SMR.1.1 The TSF shall maintain the roles

- ***TSPM Management Console Administrative Roles***[8]
  - ***Named administrative roles***
    - ***Application Administrator***
    - ***Application Owner***
    - ***Auditor***
    - ***IT Environment Administrator***
    - ***Policy Author***
    - ***Policy Operator***
    - ***Role Administrator***
  - ***Custom administrative roles created by the Role Administrator***
- ***Application (User)***[9]
  - ***Custom application roles created by the Policy Author***

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

[8] A TSPM Management Console administrative role is a collection of permissions that allow access to management functions when using the Management Console. The TSF provides named administrative roles with pre-defined permissions; however, roles can be created, modified or deleted. Permissions are necessary to access each individual management function; permissions are maintained as an attribute assigned to each role and permissions cannot be created, modified, or deleted.

[9] Application roles are user roles created by the Policy Author and used in authorization policies to identify Web Service users and groups of users to whom the policy applies

6.2 Security Assurance Requirements for the TOE

This Security Target is Evaluation Assurance Level 2 (EAL 2) augmented with ALC_FLR.3 as shown in Table 16 – Assurance Requirements below. The security assurance requirements for the TOE consist of the following components that are CC Part 3 conformant as summarized in Table 16 below and detailed in the following subsections. These requirements are included by reference.

Table 16 – Assurance Requirements

Assurance Class	Assurance Component	Assurance Components Description
Development	ADV_ARC.1	Security architecture description
	ADV_FSP.2	Security-enforcing functional specification
	ADV_TDS.1	Basic design
Guidance Documents	AGD_OPE.1	Operational user guidance
	AGD_PRE.1	Preparative User guidance
Life-cycle Support	ALC_CMC.2	Use of a CM system
	ALC_CMS.2	Parts of the TOE CM coverage
	ALC_DEL.1	Delivery procedures
	ALC_FLR.3[10]	Systematic flaw remediation
Security Target	ASE_CCL.1	Conformance claims
	ASE_ECD.1	Extended components definition
	ASE_INT.1	ST introduction
	ASE_OBJ.2	Security objectives
	ASE_REQ.2	Derived security requirements
	ASE_SPD.1	Security problem definition
	ASE_TSS.1	TOE summary specification
Tests	ATE_COV.1	Analysis of coverage
	ATE_FUN.1	Functional testing
	ATE_IND.2	Independent testing - sample
Vulnerability Assessment	AVA_VAN.2	Vulnerability analysis

[10] ALC_FLR.3 is an augmentation over EAL2

6.3 Security Requirements Rationale

6.3.1 Security Function Requirements Rationale

Table 17 - TOE SFR to Objective Mapping satisfies the requirement to trace each SFR back to the security objectives for the TOE.

Table 17 - TOE SFR to Objective Mapping

		TOE Objective			
#	SFR/SAR	O.AUDIT_GENERATION	O.ADMIN_ACCESS	O.MANAGE	O.SERVICE_ACCESS
1	FAU_GEN_SUB.1	x			
2	FAU_GEN.2	x			
3	FDP_RBACC_EXT.1 (1)		X		
4	FDP_RBACC_EXT.1 (2)				X
5	FDP_RBACF_EXT.1 (1)		X		
6	FDP_RBACF_EXT.1 (2)				X
7	FMT_MOF_EXT.1			X	
8	FMT_MSA.1 (1)			X	
9	FMT_MSA.1 (2)			X	
10	FMT_MSA.3 (1)			X	
11	FMT_MSA.3 (2)			X	
12	FMT_MTD.1 (1)			X	
13	FMT_MTD.1 (2)			X	
14	FMT_SMF.1			X	
15	FMT_SMR.1			X	

6.3.1.1 Security Function Requirements Rationale

The following paragraphs present the rationale that demonstrates that the SFRs meet all security objectives for the TOE.

O.AUDIT_GENERATION

FAU_GEN_SUB.1 defines the set of events that the TOE must be capable of recording. This requirement ensures that the administrator has the ability to audit any security relevant event that takes place in the TOE. This requirement also defines the information that must be contained in the audit record for each auditable event. There is a minimum of information that must be present in every audit record and this requirement defines that, as well as the additional information that must be recorded for each auditable event.

FAU_GEN.2 ensures that the audit records associate a user identity with the auditable event. In the case of authorized users, the association is accomplished with the user ID. In all other cases, the association is based on the source network identifier, which is presumed to be the correct identity, but cannot be confirmed since these subjects are not authenticated.

O.ADMIN_ACCESS

FDP_RBACC_EXT.1 (1) and FDP_RBACF_EXT.1 (1) play a role in satisfying this objective by ensuring that the TOE implements access controls to restrict access to the management functions to the roles having correct permissions.

O.MANAGE

The FMT requirements are used to satisfy this management objective, as well as other objectives that specify the control of functionality. The requirements' rationale for this objective focuses on the administrator's capability to perform management functions in order to control the behavior of security functions.

FMT_MOF_EXT.1 ensures that the administrator has the ability manage the audit, policy, and role functions.

FMT_MSA.1 (1) and (2) ensures the TOE provides secure initialization for security attributes that are restrictive.

FMT_MSA.3 (1) and (2) provides the administrator the ability to supply alternative initial values to override the default restrictive values for access control and policy security attributes.

FMT_MTD.1 (1) and (2) ensure that the administrator can manage TSF data.

FMT_SMR.1 defines the specific security roles supported.

FMT_SMF.1 support this objective by identifying all management functions supported by the TOE.

O.SERVICE_ACCESS

FDP_RBACC_EXT.1 (2) and FDP_RBACF_EXT.1 (2) play a role in satisfying this objective by ensuring that the TOE implements access controls to restrict access to protected Web Services to users that meet the defined policies.

6.3.1.2 Security requirement dependency analysis

Table 18 - SFR Component Dependency Mapping maps the dependencies that exist for each SFR. If the column labeled "satisfied" shows a dependency that has not been resolved, the rationale is provided in the text following the table as why this dependency does not apply for the TOE.

Table 18 - SFR Component Dependency Mapping

#	Component	Dependencies	Satisfied [Component #]
1	FAU_GEN_SUB.1	None	None
2	FAU_GEN.2	FAU_GEN.1 FIA_UID.1	FAU_GEN_SUB.1 None
3	FDP_RBACC_EXT.1(1)	FDP_RBACF_EXT.1	FDP_RBACF_EXT.1 (1)
4	FDP_RBACC_EXT.1(2)	FDP_RBACF_EXT.1	FDP_RBACF_EXT.1 (2)
5	FDP_RBACF_EXT.1 (1)	FDP_RBACC_EXT.1 FMT_MSA.3	FDP_RBACC_EXT.1(1) FMT_MSA.3 (1)
6	FDP_RBACF_EXT.1 (2)	FDP_RBACC_EXT.1 FMT_MSA.3	FDP_RBACC_EXT.1(2) None
7	FMT_MOF_EXT.1	FMT_SMF.1 FMT_SMR.1	FMT_SMF.1 FMT_SMR.1
8	FMT_MSA.1 (1)	[FDP_ACC.1 or FDP_IFC.1] FMT_SMR.1 FMT_SMF.1	FDP_RBACC_EXT.1 (1) FMT_SMR.1 FMT_SMF.1
9	FMT_MSA.1 (2)	[FDP_ACC.1 or FDP_IFC.1] FMT_SMR.1 FMT_SMF.1	FDP_RBACC_EXT.1 (1) FMT_SMR.1 FMT_SMF.1
10	FMT_MSA.3 (1)	FMT_MSA.1 FMT_SMR.1	FMT_MSA.1 (1) FMT_SMR.1
11	FMT_MSA.3 (2)	FMT_MSA.1 FMT_SMR.1	FMT_MSA.1 (2) FMT_SMR.1
12	FMT_MTD.1 (1)	FMT_SMR.1 FMT_SMF.1	FMT_SMR.1 FMT_SMF.1
13	FMT_MTD.1 (2)	FMT_SMR.1 FMT_SMF.1	FMT_SMR.1 FMT_SMF.1
14	FMT_SMF.1	None	None
15	FMT_SMR.1	FIA_UID.1	None

Rationale for unsatisfied dependencies:

The dependency on FIA_UID.1 by FAU_GEN.2 and FMT_SMR.1 are not required because the runtime environment provides all identification and authentication of users; this information is passed to the TOE.

The dependency on FMT_MSA.3 by FDP_RBACF_EXT.1 (2) is not required because the role is assigned to the subject by the runtime environment and the role is tested against a fixed role, there is no management function necessary.

6.3.2 Security Assurance Requirements Rationale

This ST contains the assurance requirements from the Common Criteria EAL2 assurance package augmented with ALC_FLR.3. The Common Criteria allows assurance packages to be augmented, which allows the addition of assurance components from the Common Criteria not already included in the EAL.

Augmentation was chosen to provide the added assurance that is provided by defining flaw remediation procedures and correcting security flaws (ALC_FLR.3). The EAL chosen is based on the statement of the security environment (threats, organizational policies, assumptions) and the security objectives defined in this ST. The sufficiency of the EAL chosen (EAL2 augmented) is justified based on those aspects of the environment that have impact upon the assurance needed in the TOE

Given the amount of assurance deemed necessary to meet the security environment and objectives of the TOE and the intent of EAL 2, EAL 2 is an appropriate level of assurance for the TOE described in this ST. Therefore, EAL2 augmented is an appropriate level of assurance for the TOE.

Table 19 shows the matrix of Security Assurance requirements; the ST assurance levels are shown in **BOLD** text, which clearly demonstrates that this Security Target meets EAL2+.

Table 19 - Evaluation assurance level summary

Assurance Class	Assurance Family	Assurance Components by Evaluation Assurance Level						
		EAL1	EAL2	EAL3	EAL4	EAL5	EAL6	EAL7
Development	ADV_ARC		**1**	1	1	1	1	1
	ADV_FSP	1	**2**	3	4	5	5	6
	ADV_IMP				1	1	2	2
	ADV_INT					2	3	3
	ADV_SPM						1	1
	ADV_TDS		**1**	2	3	4	5	6
Guidance Documents	AGD_OPE	1	**1**	1	1	1	1	1
	AGD_PRE	1	**1**	1	1	1	1	1
Life-cycle Support	ALC_CMC	1	**2**	3	4	4	5	5
	ALC_CMS	1	**2**	3	4	5	5	5
	ALC_DEL		**1**	1	1	1	1	1
	ALC_DVS			1	1	1	2	2
	ALC_FLR							
	ALC_LCD			1	1	1	1	2
	ALC_TAT				1	2	3	3
Security Target Evaluation	ASE_CCL	1	**1**	1	1	1	1	1
	ASE_ECD	1	**1**	1	1	1	1	1
	ASE_INT	1	**1**	1	1	1	1	1
	ASE_OBJ	1	**2**	2	2	2	2	2
	ASE_REQ	1	**2**	2	2	2	2	2
	ASE_SPD		**1**	1	1	1	1	1
	ASE_TSS	1	**1**	1	1	1	1	1
Tests	ATE_COV		**1**	2	2	2	3	3
	ATE_DPT			1	1	3	3	4
	ATE_FUN		**1**	1	1	1	2	2
	ATE_IND	1	**2**	2	2	2	2	3
Vulnerability Assessment	AVA_VAN	1	**2**	2	3	4	5	5

Table 20 - SAR Component Dependency Mapping, maps the dependencies that exist for each SAR to demonstrate all SAR dependencies are satisfied.

Table 20 - SAR Component Dependency Mapping

Component	Dependencies	Satisfied
ADV_ARC.1	ADV_FSP.1 ADV_TDS.1	Yes – ADV_FSP.2 Yes – ADV_TDS.1
ADV_FSP.2	ADV_TDS.1	Yes – ADV_TDS.1
ADV_TDS.1	ADV_FSP.2	Yes - ADV_FSP.2
AGD_OPE.1	ADV_FSP.1	Yes - ADV_FSP.2
AGD_PRE.1	None	--
ALC_CMC.2	ALC_CMS.1	Yes – ALC_CMS.2
ALC_CMS.2	None	--
ALC_DEL.1	None	--
ALC_FLR.3	None	--
ATE_COV.2	ADV_FSP.2 ATE_FUN.1	Yes – ADV_FSP.2 Yes - ATE_FUN.1
ATE_FUN.1	ATE_COV.1	Yes - ATE_COV.1
ATE_IND.2	ADV_FSP.2 AGD_OPE.1 AGD_PRE.1 ATE_COV.1 ATE_FUN.1	Yes – ADV_FSP.2 Yes – AGD_OPE.1 Yes – AGD_PRE.1 Yes – ATE_COV.1 Yes - ATE_FUN.1
AVA_VAN.2	ADV_ARC.1 ADV_FSP.2 ADV_TDS.1 AGD_OPE.1 AGD_PRE.1	Yes - ADV_ARC.1 Yes - ADV_FSP.2 Yes - ADV_TDS.1 Yes – AGD_OPE.1 Yes - AGD_PRE.1

Rationale for unsatisfied dependencies:

There are no unsatisfied dependencies; therefore, no rationale is presented.

7 TOE Summary Specification

7.1 Implementation description of TOE SFRs

This section provides evaluators and potential consumers of the TOE with a high-level description of each SFR, thereby enabling them to gain a general understanding of how the TOE is implemented. These descriptions are intentionally not overly detailed, thereby disclosing no proprietary information. This section refers to SFRs defined in Section 6, Security requirements.

7.2 TOE Security Functions

The TFS supports the following security functions:

- Security Audit
- User data protection
- Security Management

7.2.1 Security Audit

The TOE audit functionality is limited to the generation of audit records; the TOE requires support from the IT environment (WebSphere Application Server (WAS)) to add date and time, store and review audit records as described in Section 7.2.1.1, TSPM Security Audit Support Requirements. **FAU_GEN_SUB.1, FAU_GEN.2**

The TOE generates TIVOLI_RTSS (runtime security services) and TIVOLI_TSPM (policy manager) events and hands them to WebSphere for storage. TSPM generates the following record types:

- TIVOLI_RTSS Audit Records
  - RTSS access decision requests
  - RTSS admin command received
  - RTSS notifications: update notifications, removal notification
  - RTSS policy: policy updated, removed
- TIVOLI_TSPM Audit Records
  - TSPM classification: Add, modify or delete a classification, classify or declassify a service, associate or disassociate a policy with a classification.
  - TSPM Policy: Add, modify, delete policies
  - TSPM Policy Association: Add association, delete an association
  - TSPM Service: Add a service, modify service attribute, or delete a service
  - TSPM PDT (PDT): Add PDT, modify PDT attribute, set PDT property, delete PDT
  - TSPM Policy Distribution: Distribute policy to PDT, redistribute policy to PDT, remove policy from PDT
  - TSPM Role: Add, modify, delete, add user to role, remove user from role, add group to role, delete group from role
  - TSPM Role Mapping: Add, Modify, Delete, add user to role mapping, remove user from role mapping, add group to role mapping, delete group from role mapping
  - TSPM Rule: Add, modify or delete Rule
  - TSPM Configured Rule - Add, Modify or delete a configured Rule
  - TSPM Admin Role: Add, modify, or delete administrative role;
  - TSPM Admin Role User: Use of admin role to add, remove, delegate, or undelegate user
  - TSPM Admin Role Group: Use of admin role to add, remove, delegate, or undelegate group

TSPM audit event records are a standard data structure called a Common Base Event (CBE). The CBE is a specification based on XML that defines a mechanism for managing events, such as logging, tracing, and management. Events that start with TIVOLI_RTSS are generated by the runtime security services component; events that start with TIVOLI_TSPM are generated by the policy manager component.

The CBE provides for common properties and attributes that can be used for all events, as well as the extendedDataElement, which enables definition of additional properties and attributes that are required to audit specific events. These properties and attributes are too detailed to describe in this Security Target; they are fully described in [1] Chapter 12, Auditing.

The TSPM Administration console enables an administrative user with the IT Environment Administrator role to modify the audit configuration settings and reload the RTSS audit configuration. Normally, the audit configuration is read at startup; however, the reload allows runtime modification of the audit settings.

The audit configuration settings control enabling/disabling audit and specify log file settings; however, these settings are passed to WAS, where the log file settings are enforced. An authorized administrator can configure logging individually for the two primary Tivoli Security Policy Manager components: policy manager server and runtime security services. (Refer to Section 7.2.3.1, Management of audit functions)

The audit configuration settings options for both RTSS and policy manager components are described in Table 21 - Audit Configuration Settings below.

Table 21 - Audit Configuration Settings

Option	Description
Enable /Disable Audit	Select Enable Audit to begin capturing audit records generated by the policy manager component or runtime security services component. Logging is enabled by default.
Audit file path and name	Specify the directory and file for storing event records, or accept the default log file location. The default for policy manager is logs/tspm.audit.log. The default for RTSS is logs/rtss.audit.log. The specified log directory locations are relative to the WebSphere profile root directory.
Enable rollover	Specifies that all of the files specified in Maximum number of auditing files are to be retained. When the last file in the specified file group fills with entries, the entries in the oldest file (first file in the chain of files that is filled with entries) are overwritten. This selection is enabled by default.
Maximum audit file size before rollover	Specify the maximum amount of data that is stored in each file. The default value is 5 megabytes. The minimum size is 1 megabyte; the maximum size is 2 gigabytes.
Maximum number of audit files to retain	Specify the number of files used to store log entries before the oldest file is overwritten. The default value is 5 files. The minimum number of files is 1; the maximum number of files is 1000000. This field is required if auditing is enabled. Coupled with the maximum audit file size, this option sets an upper limit on the amount of audit data that is captured and maintained.

7.2.1.1 TSPM Security Audit Support Requirements

TSPM requires support from the IT environment to store, manage, and review audit records. As mentioned above, after TSPM generates an audit record, it is passed to Websphere for storage; WebSphere controls the storage and the rollover of log files based on information contained in the log configuration file.

Tivoli Security Policy Manager provides the audit report templates in a report package. A report package is a .zip file that contains the definitions, designs, and the report set hierarchy for running the predefined reports. To generate reports, an authorized administrator must import the Tivoli Security Policy Manager report package into the Tivoli Common Reporting environment; however, this function is a part of TCR and therefore not in the evaluated configuration.

TSPM uses Tivoli Common Reporting (TCR) to generate, format, view, and print report data. TCR is a reporting environment that provides a consistent approach across IBM products for viewing and administering reports. TCR is installed as part of the Tivoli Integrated Portal (TIP) component; therefore, it is not included in the evaluated configuration. TCR provides the following support necessary for the TSPM audit function:

- Data store for storing and organizing report designs, reports, and supporting resources
- Web-based user interface for generating and viewing formatted reports
  - Use the interface to specify report parameters and other report properties.
- Command-line interface for working with objects in the data store and for performing additional administrative functions

7.2.2 User Data Protection

The TSF implements multiple rule-based access control mechanisms for users and groups of users[11] to control access to the TIP management functions, the TSPM management functions, and to the Web Services protected by the TOE.

7.2.2.1 Access Control Mechanisms

As previously stated, The TSF implements multiple Rule-based access control mechanisms for users and groups to control access to the TSPM management functions, and to the services protected by the TOE.

The access mechanisms implemented for administrative users needing access to management functions via console are hierarchical; to access the TSPM console, an administrative user must have access to TIP; to access the TIP console, an administrative user must have access to eWAS. In each layer of the hierarchy, the user must have a role assigned allowing access to the next layer.

7.2.2.1.1 TSPM Management Console Access Control

The TSPM management console access control mechanisms provide fine-grained control for the administrative users and groups of TSPM functionality. This access control mechanism implements a simple test, if the roles assigned a user has the correct permissions for the operation, access is allowed, otherwise, access is denied.

The permissions associated with the predefined administrative roles are as follows.

- Policy Administration
  - Create, delete, modify, and view policy
- Rule Parameter Administration
  - Create, delete, and modify rule parameters
- Classification Administration
  - Classify and declassify services, and create, modify, and delete classifications
- Distribution Target Administration
  - Create, delete, and modify distribution targets
- Application Role Administration
  - Create, delete, and modify application roles
- Service Administration
  - Attach policy to services, detach policy from services, and create, delete, modify, and view services
- User Registry Administration

[11] A user may derive permissions associated with the group the user is a member; in FDP_ACC_EXT and FDP_ACF_EXT the term user allows this association.

  - Create, delete, and modify user registries
- Administrative Role Administration
  - Assign, create, delete, and modify administrative roles
- Policy Operations Administration
  - View, configure, and distribute policy
- General Administration
  - Enable, disable, or modify the settings for runtime security services auditing and Tivoli Security Policy Manager auditing.
  - Enable or disable reloading of the configuration for the runtime security services
- Obligations Administration
  - Create Obligations
  - Delete Obligations
  - Modify Obligations

FDP_RBACC_EXT.1 (1), FDP_RBACF_EXT.1 (1)

7.2.2.1.2 RTSS Access Control

The RTSS Server and Client implement the access control mechanisms that control access to the protected Web Services resources; this mechanism uses the Authorization Policies that are created by the Policy Author using the Policy Administration Point and deployed to the Policy Distribution Target (PDT). After distribution to the PDT, the policies are distributed to the Policy Decision Point (PDP) where the policy is evaluated when a user requests access to a service protected by the policy.

The Authorization Policy is made up of an application role filter and the rules that are used to specify conditions; an administrative user having the Policy Author role creates policies.

The application role filter must match the user requesting access, otherwise, the policy is not applicable because the user making the request is a not a member of the role to which this policy applies.
In the evaluated configuration, rules must be created using a collection of predefined operators, predefined and custom rule parameters, and custom roles; external rules, e.g., custom code that makes an authorization decision based on appropriate logic, perhaps calling an external system during the evaluation process are not allowed in the evaluated configuration.

Rules are used to specify conditions under which access by specific users should be permitted or denied; rules consist of one or more If-Then condition statements. The “If” half of each statement presents a condition or a context that the request is evaluated against. The “Then” half of each statement specifies whether to permit or deny access based on the evaluation.

For example, the following If-Then statement permits access to a protected resource, if the access request is made by a specific role at 8:00 AM or after:

> *If current time is greater than or equal to 8:00 AM; Then Permit access.*

In this example, “*current time*” is referred to as a rule parameter, and “*is greater than or equal*” is referred to as operators. A rule parameter specifies the context of a request that is to be evaluated as part of an access decision. The following rule parameters are available by default:

- current time
  - Specifies that the current time of the request is evaluated.
- current date
  - Specifies that the current date of the request is evaluated.
- current date and time
  - Specifies that the current date of the request is evaluated
- risk score
- registered device

Additionally, new rule parameters can be created that represent unique information, called attributes, about a request. At policy configuration time, an authorized administrator must specify the type of attribute that each new rule parameter represents:

- Subject attributes
    - Information about the user that is making the access request.
- Resource attributes
    - Information about the resource, service, or system that is being requested.
- Action attributes
    - The action that the user wants to take on the resource.
- Environment attributes
    - Information about the environment that the request is in, such as geographical location, the time, the date, or other such elements.

The following default types are required to be assigned to each rule parameter to define the data type for comparison:

- String
- Boolean
- Integer
- Double
- Time
- Date
- Date Time
- URI

The following default comparison operators are provided; no custom operators are allowed:

=	(is equals)
<=	(is greater than or equal)
<=	(is less than or equal)
>	(is greater than)
<	(is less than)

When a user requests access to a service, Tivoli Security Policy Manager initiates the following evaluation process for each authorization policy that is attached to the service:

7. Evaluates the application role filter for a match, if an application role is used. If the application role filter yields a match, the process continues with the next step. If the application role filter does not yield a match, this policy is not applicable because the user making the request is a not a member of the role to which this policy applies. If another policy is attached to the service or service element, that policy is then evaluated for applicability beginning with this step. If no other attached policies yield a match for the application role filter, by default, the access request is denied.
8. Evaluates the action requested by the user with the actions selected in the policy:
    a. Because in the evaluated configuration, only Web Services are protected resources, the invoke action is always selected.
9. Evaluates the policy rules in the order specified, until *all* the conditions of a single rule are met:
    a. If *any* of the conditions in a rule evaluate to false, the rule evaluates to not applicable and the next rule is evaluated.
    b. If all the conditions of a rule evaluate to true, the access decision for the policy is returned and enforced.
    c. If none of the specified rules yields an exact match, and all the rules are therefore considered "not applicable" to the request, the outcome of this policy is not applicable and access is denied.
10. Returns either a Permit or Deny access decision based on the evaluation of the policy.
11. If there are multiple policies attached to the service, then the next policy is evaluated.

12. When all policies have been evaluated, their results are examined.
    a. If any policy evaluates to Deny, the access request is denied.
    b. If no individual policy evaluates to Deny, and at least one policy results in Permit, the access request is permitted.
    c. If no individual policy evaluates to either Permit or Deny, and all attached policies result in "not applicable," by default, the access request is denied.
13. The final access decision (either Permit or Deny) is then returned to the Policy Enforcement Point

FDP_RBACC_EXT.1 (2), FDP_RBACF_EXT.1 (2)

7.2.2.1.3 TSPM EJB Access Control

The TSPM Server access control mechanisms provide fine-grained control for the methods within each EJB running in the TSPM server; the EJBs contain the backend code that is invoked to perform product operations. Each method within the EJBs implements a specific management function; the subject is required to have the permission for each method to be executed.

When a TSPM Server EJB method is called, the method performs these actions:

1. The method will query the TSPM Server for the Subject name
2. The method will then query the TSPM Server for the Subject's role,
3. The method will then obtain the role/permissions mapping
4. Once the permissions are obtained an access decision is made.
5. If the Subject has the correct permissions, the request to access the method in the EJB is granted.
6. Else, return error.

FDP_RBACC_EXT.1 (1), FDP_RBACF_EXT.1 (1)

The following are a list of available EJBs, the methods within the EJBs beyond the scope of this security target and therefore not listed.

- Security Administrator EJBs
  - AdminManagerService
  - AuditManagerService
  - ClassificationManagerService
  - DistributionManagerService
  - PluginUIManagerService
  - PolicyManagerService
  - RTSSManagerService
  - RegistryManagerService
  - RoleManagerService
  - RuleParameterManagerService
  - SecurityManagerService
  - ServiceManagerService
  - StatusManagerService
  - UserRegistryManagerService
  - WorkflowManagerService
- The Policy Reader EJBs
  - DistributionManagerService
  - PolicyManagerService
  - SecurityManagerService
  - ServiceManagerService
  - StatusManagerService
- The PDT Register EJBs
  - DistributionManagerService
  - SecurityManagerService

7.2.3 Security Management

Primary management of the TOE's security relevant parameters is performed by an authorized administrator using the TSPM Management console with a JAVA-enabled Web browser via a secure HTTPS/TLS connection. The following management functions are available to administrative users:

- Policy Administration
  - Policies are sets of conditions (rules) that define the circumstances under which a request for access is permitted or denied.
- Rule Parameter Administration
  - A rule parameter is an operand to a conditional expression that is associated with a rule that specifies the context of a request to be evaluated as part of an access decision. The value of a rule parameter can be bound to an attribute in the authorization request, or, it can be bound to a value in the IT environment (example: an LDAP user attribute).
- Classification Administration
  - A classification is a means of associating one or more security policies to one or more services. Adding a security policy to a classification attaches the policy to all the services in the classification. Adding a service to a classification attaches all the policies in the classification to the service
- Distribution Target Administration
  - Policies created in Tivoli Security Policy Manager can be made available to policy decision points through the Policy Distribution Targets (PDT). In the evaluated configuration, an instance of the RTSS server acts as a PDT.
- Application Role Administration
  - Application roles can be used in policies to specify groups of users who are affected by the policy.
- Service Administration
  - Services are resources that are to be protected by the TOE; before a service can be protected, the service must be create or acquired in Tivoli Security Policy Manager.
- User Registry Administration
  - A user registry is used to store the identities that are used for various TSPM components to communicate to each other. In the evaluated configuration, IBM TDS 6.2 will be used as the User Registry.
- Administrative Role Administration
  - Administrative roles provide fine-grained control of Tivoli Security Policy Manager functionality to administrators.
- Policy Operations Administration
  - Policy operations allow configuration and distribution of policies, and the administration of policy information points. Policy Information Points (PIPs) are a source of attribute data that is available for use in authorization decisions; however, PIPs are not a part of the evaluated configuration.
- General Administration
  - General administration allows management of RTSS audit functions, TSPM audit functions and controls the ability to reload RTSS configuration.
- Obligations Administration
  - Obligations Administration allows management to define actions that must be performed before a policy enforces an access decision. An obligation is an optional element attached to a policy. Obligation attributes, defined when an obligation is created, are assigned values within the policy. When a Policy Enforcement Point cannot perform an obligation, access is denied.

7.2.3.1 Management of audit functions

The Tivoli Security Policy Management console enables an administrative user with the IT Environment Administrator role to turn audit logging on or off, specify log file settings, and reload the configuration. Logging can be configured individually for the two primary Tivoli Security Policy Manager components: policy manager server and runtime security services. These functions control audit using a configuration file that is read on startup or restart of the TSPM components and is modified by the following management functions:

- Enable Audit or Disable Audit
  - Select Enable Audit to begin capturing audit records generated by the policy manager component or runtime security services component. Logging is enabled by default.
- Audit file path and name
  - Specify the directory and file for storing event records, or accept the default log file location. The default for policy manager is logs/tspm.audit.log. The default for runtime security services is logs/rtss.audit.log. The specified log directory location is relative to the WebSphere profile root directory.
- Enable rollover
  - Specifies that all of the files specified in Maximum number of auditing files are to be retained. When the last file in the specified file group fills with entries, the entries in the oldest file (first file in the chain of files that is filled with entries) are overwritten. This selection is enabled by default.
- Maximum audit file size before rollover
  - Specify the maximum amount of data that is stored in each file before the next file in the file chain is used to store entries. The default value is 5 megabytes of information. The minimum size is 1 megabyte. The maximum size is 2 gigabytes.
- Maximum number of audit files to retain
  - Specify the number of files used to store log entries before the oldest file in the file chain begins to be overwritten. The default value is 5 files. The minimum number of files is 1; the maximum number of files is 1000000. This field is required if auditing is enabled. Coupled with the maximum audit file size, this option sets an upper limit on the amount of audit data that is captured and maintained.

FMT_MOF_EXT.1 (Audit), FMT_MTD.1 (2)

7.2.3.2 Management of RTSS Access Control functions (Policy)

The Tivoli Security Policy Management console enables an administrative user with the Policy Author role to create, delete, modify and view authorization policies, create rule parameters, and to create application roles. Because access to protected resources is not allowed until a policy is created having rules that allow access; it is considered the default values are restrictive and can be overridden by an administrative user with the Policy Author role.

The Tivoli Security Policy Management console enables an administrative user with the Policy Operator role can view, configure and distribute authorization policies as well as configure distribution. **FMT_MTD.1 (1)**

As explained in Section 7.2.2.1.2, RTSS Access Control, these roles act together to allow a policy to be created, and distributed so the authorization policy can be used to control access to the protected services.

FMT_MOF_EXT.1 (Policy), FMT_MSA.1 (2), FMT_MSA.3 (2)

7.2.3.3 Management of Administrative (Roles)

The Tivoli Security Policy Management console enables an administrative user with the Role Administrator role to assign create, delete, and modify administrative roles; this includes adding and/or deleting permissions from the default and/or custom roles. Because default permissions are defined that allow access, it is considered the default values are permissive and can be overridden by an administrative user with the Role Administrator role.

FMT_MOF_EXT.1 (Role), FMT_MSA.1 (1), FMT_MSA.3 (1)

7.2.4 TSPM Roles

7.2.4.1 TSPM Console Management Roles

During TSPM installation and configuration, a set of named administrative roles are created. These roles represent responsibilities that might be common among administrators who are working with a TSPM environment. Each role consists of a collection of permissions that are needed to perform related tasks as defined in Table 14 – TSPM Administrative Roles and Assigned Permissions. Although these roles are predefined, an authorized administrator can modify or delete them. **FMT_SMF.1.1, FMT_SMR.1.1**

- Named administrative roles
  - Application Administrator
  - Application Owner
  - Auditor
  - IT Environment Administrator
  - Policy Author
  - Policy Operator
  - Role Administrator
- Custom administrative roles created by the Role Administrator

The TSF relies on the runtime environment to maintain role/user relationships; however, the TSF is able to query the runtime environment to obtain the required association between a user and the user's role.

FMT_SMR.1.2

7.2.4.2 TSPM Application Roles

The TSF allows the Policy Author to define application roles used in authentication policies; these roles are part of the access control decision used to protected resources as explained in Section 7.2.2.1.2, RTSS Access Control.

FMT_SMR.1.1

8 Abbreviations / Acronyms

Table 22 - TOE Related Abbreviations and Acronyms

Abbreviation / Acronym	Description
CBE	Common Base Event
DOD	Department of Defense
DoD	See DOD
EJB	Enterprise JavaBeans
IBMJCEFIPS	IBM Java JCE (Java Cryptographic Extension) FIPS
IBMJSSEFIPS	IBM Java JSSE (Java Secure Sockets Extension) FIPS
J2EE	Java 2 Platform, Enterprise Edition
RTSS	RunTime Security Services
TIVOLI_RTSS	Tivoli RunTime Security Services
WAS	WebSphere Application Server

Table 23 - CC Related Abbreviations / Acronyms

Abbreviation / Acronym	Acronym Description
CAP	Composed Assurance Package
CC	Common Criteria
CCRA	Arrangement on the Recognition of Common Criteria Certificates in the field of IT Security
DAC	Discretionary Access Control
EAL	Evaluation Assurance Level
IT	Information Technology
OSP	Organizational Security Policy
PP	Protection Profile
SAR	Security Assurance Requirement
SF	Security Function
SFR	Security Functional Requirement
SFP	Security Function Policy
ST	Security Target
TOE	Target of Evaluation
TSF	TOE Security Functionality
TSFI	TSF Interface

9 References

Table 24 - TOE Guidance Documentation		
Reference	Description	Control Number
[1]	Security Policy Manager Version 7.1 Administration Guide	SC23-9476-01
[2]	Security Policy Manager Version 7.1 Installation Guide	GC27-2712-00
[3]	Security Policy Manager Version 7.1 Configuration Guide	GC27-2713-00
[4]	Security Policy Manager Version 7.1 Error Message Reference	GC23-9477-01
[5]	Security Policy Manager Version 7.1 Troubleshooting Guide	GC27-2711-00
[6]	Security Policy Manager Version 7.1 Common Criteria Guide	SC27-5627-00
[7]	TSPM Online Fix Pack 4 Guidance http://pic.dhe.ibm.com/infocenter/tivihelp/v2r1/index.jsp?topic=%2Fcom.ibm.tspm.doc_7.1%2Fwelcome.html	None

Table 25 - Common Criteria v3.1 References			
Reference	Description	Version	Date
[10]	Common Criteria for Information Technology Security Evaluation Part 1: Introduction and general model CCMB-2009-07-001	V3.1 R3	July 2009
[11]	Common Criteria for Information Technology Security Evaluation Part 2: Security functional components CCMB-2009-07-002	V3.1 R3	July 2009
[12]	Common Criteria for Information Technology Security Evaluation Part 3: Security assurance components CCMB-2009-07-003	V3.1 R3	July 2009
[13]	Common Criteria for Information Technology Security Evaluation Evaluation Methodology CCMB-2009-07-004	V3.1 R3	July 2009

Table 26 – Supporting Documents			
Reference	Description	Version	Date
[22]	WS-SecurityPolicy OASIS Standard	1.2	1 July 2007
[23]	eXtensible Access Control Markup Language 3 (XACML) OASIS Standard	2.0	1 Feb 2005