# Marimba Client and Server Management from BMC Software Release 6.0.3

**Security Target Version 2.3.0**

4 June, 2007

**Prepared by:**

**BMC Software, Inc.**
2101 City West Blvd.
Houston, Texas 77042

## TABLE OF CONTENTS

**1. Security Target Introduction ..... 5**
1.1 SECURITY TARGET, TOE AND CC IDENTIFICATION ..... 5
1.2 CONFORMANCE CLAIMS ..... 6
1.3 CONVENTIONS, TERMINOLOGY AND ACRONYMS ..... 6
*1.3.1 Conventions ..... 6*
*1.3.2 Operations ..... 6*
*1.3.3 Naming Conventions ..... 6*
*1.3.4 Terminology ..... 7*
*1.3.5 Acronyms ..... 8*
1.4 SECURITY TARGET ORGANIZATION ..... 9
**2. TOE Description ..... 9**
2.1 PRODUCT TYPE ..... 10
2.2 PRODUCT DESCRIPTION ..... 10
2.3 PRODUCT SECURITY FEATURES ..... 13
2.4 SECURITY ENVIRONMENT TOE BOUNDARY ..... 13
*2.4.1 Physical Boundaries ..... 13*
*2.4.2 Logical Boundaries ..... 17*
**3. Security Environment ..... 18**
3.1 THREATS TO SECURITY ..... 19
3.2 ORGANIZATIONAL SECURITY POLICIES ..... 19
3.3 SECURE USAGE ASSUMPTIONS ..... 19
*3.3.1 Physical Assumptions ..... 20*
*3.3.2 Personnel Assumptions ..... 20*
*3.3.3 System Assumptions ..... 20*
**4. Security Objectives ..... 20**
4.1 IT SECURITY OBJECTIVES FOR THE TOE ..... 20
4.2 IT SECURITY OBJECTIVES FOR THE ENVIRONMENT ..... 21
4.3 SECURITY OBJECTIVES OF THE NON-IT ENVIRONMENT ..... 22
**5. IT Security Requirements ..... 22**
5.1 TOE SECURITY FUNCTIONAL REQUIREMENTS ..... 22
*5.1.1 Security Audit (FAU) ..... 23*
*5.1.2 Cryptographic support (FCS) ..... 24*
*5.1.3 User Data Protection (FDP) ..... 25*
*5.1.4 Identification and Authentication (FIA) ..... 27*
*5.1.5 Security management (FMT) ..... 27*
*5.1.6 TOE access (FTA) ..... 30*
*5.1.7 Trusted Path/Channels ..... 30*
5.2 SECURITY FUNCTIONAL REQUIREMENTS FOR THE IT ENVIRONMENT ..... 31
*5.2.1 User Data Protection (FDP) ..... 31*
*5.2.2 Identification and Authentication (FIA) ..... 31*
*5.2.3 Protection of the TSF (FPT) ..... 32*
5.3 TOE SECURITY ASSURANCE REQUIREMENTS ..... 32
*5.3.1 Configuration Management (ACM) ..... 33*
*5.3.2 Delivery and Operation (ADO) ..... 34*
*5.3.3 Development (ADV) ..... 35*
*5.3.4 Guidance Documents (AGD) ..... 36*
*5.3.5 Life Cycle Support (ALC) ..... 38*
*5.3.6 Security Testing (ATE) ..... 38*
*5.3.7 Vulnerability Assessment (VLA) ..... 40*

**6. TOE Summary Specification ..... 42**
6.1 TOE SECURITY FUNCTIONS ..... 42
*6.1.1 Cryptographic Support ..... 42*
*6.1.2 Security Audit ..... 43*
*6.1.3 User Data Protection ..... 44*
*6.1.4 Identification and Authentication ..... 46*
*6.1.5 Security Management ..... 46*
*6.1.6 TOE Access ..... 48*
*6.1.7 Trusted Path/Channel ..... 48*
6.2 TOE SECURITY ASSURANCE MEASURES ..... 49
*6.2.1 Process Assurance ..... 49*
*6.2.2 Delivery and Operation ..... 50*
*6.2.3 Design Documentation ..... 50*
*6.2.4 Guidance Documentation ..... 51*
*6.2.5 Test Documentation ..... 52*
*6.2.6 Vulnerability Assessment ..... 52*
**7. Protection Profile Claims ..... 53**
**8. Rationale ..... 53**
8.1 SECURITY OBJECTIVES RATIONALE ..... 53
*8.1.1 Security Objective for the TOE Rationale ..... 54*
*8.1.2 Security Objectives for Environment Rationale ..... 55*
8.2 SECURITY REQUIREMENTS RATIONALE ..... 57
*8.2.1 Security Functional Requirements Rationale ..... 57*
8.3 SECURITY ASSURANCE REQUIREMENTS RATIONALE ..... 61
8.4 SECURITY REQUIREMENTS DEPENDENCIES RATIONALE ..... 61
8.5 TOE SUMMARY SPECIFICATION RATIONALE ..... 62
8.6 INTERNAL CONSISTENCY AND SUPPORT RATIONALE ..... 66
8.7 STRENGTH OF FUNCTION (SOF) RATIONALE ..... 67

## LIST OF FIGURES

**Figure 1: Client and Server Management TOE Physical Boundaries** ..... 17

## LIST OF TABLES

**Table 1: Identification of Administrative Interfaces** ..... 12

# 1. Security Target Introduction

Marimba® Client and Server Management from BMC Software offers a policy-based change and configuration management solution that automates the discovery, packaging, provisioning, configuration, patching, and repair of software (Operating systems, Patches, Applications, Content, & Configurations) across heterogeneous operating systems.

BMC's Marimba software provisioning and distribution products enable enterprises to rapidly respond to changing business requirements by re-purposing, re-provisioning, and updating IT resources to achieve required IT configurations.

Marimba configuration discovery and tracking products enable enterprises to track both the state and usage of their hardware and software assets. The data improves the customer's service model as well as facilitates better decision making with a more accurate view of the environment.

This section identifies the Security Target (ST) and Target of Evaluation (TOE), specifies ST conventions and conformance claims, and describes how the ST is organized.

## 1.1 Security Target, TOE and CC Identification

**ST Title –** Marimba Client and Server Management from BMC Software

**ST Version –** Version 2.3.0

**ST Date** –4 June 2007

**TOE Identification –** The TOE is composed of the following product modules:

- Marimba® Control Center by BMC Software 6.0.3 SP2, with SSL enabled, and Publisher and Channel Copier versions 4.6.2, Logging Service 5.0.1 and Policy Service 5.1
- Marimba® Patch Management by BMC Software 6.5
- Marimba® Content Management by BMC Software with Content Replicator 6.5
- Marimba® Desktop/Mobile Application Management by BMC Software with Application Packager 6.5
- Marimba® Server Application Management by BMC Software with Application Packager 6.5
- Marimba® Desktop OS Management by BMC Software 6.0.3
- Marimba® Server OS Management for Unix and Linux by BMC Software 6.0.3

**Evaluation Assurance Level (EAL) –** EAL 3

**CC Identification –** Common Criteria for Information Technology Security Evaluation, Version 2.2, January 2004, ISO/IEC 15408

## 1.2 Conformance Claims

This TOE is conformant to the following CC specifications:

- Common Criteria for Information Technology Security Evaluation Part 2: Security functional requirements, Version 2.2, January 2004, ISO/IEC 15408-2.
    - Part 2 Conformant
- Common Criteria for Information Technology Security Evaluation Part 3: Security assurance requirements, Version 2.2, January 2004, ISO/IEC 15408-3.
    - Part 3 Conformant
    - Evaluation Assurance Level 3 (EAL3)

## 1.3 Conventions, Terminology and Acronyms

The following conventions have been applied in this document:

### 1.3.1 Conventions

All requirements in this ST are reproduced relative to the requirements defined in CC v2.2.

### 1.3.2 Operations

Security Functional Requirements – Part 2 of the CC defines the approved set of operations that may be applied to functional requirements: iteration, assignment, selection, and refinement.

- Iteration: allows a component to be used more than once, with varying operations. In the ST, iteration is indicated by a letter in parenthesis, placed at the end of the component. For example, FDP_ACC.1(a) and FDP_ACC.1(b) indicate that the ST includes two iterations of the FDP_ACC.1 requirement: a and b.
- Assignment: allows the specification of an identified parameter. Assignments are indicated using bold and are surrounded by brackets (e.g., [**assignment**]).
- Selection: allows the specification of one or more elements from a list. Selections are indicated using underlined text and surrounded by brackets (e.g., [selection]).
- Refinement: allows the addition of details. Refinements are indicated using bold, for additions, and strike-through, for deletions (e.g., "… **all** objects …" or "… ~~some~~ **big** things …").
- Assignment within a selection: allows an assignment as an element in a selection. The operation is enclosed in brackets, and then if an assignment is used as an element, it is also enclosed in brackets. If an assignment is the only element of a selection, then the words 'no selection' are included as one of the elements to add clarity (e.g., [no selection,[**assignment**]].

Other sections of the ST use bolding and italics to highlight text of special interest, such as captions.

### 1.3.3 Naming Conventions

**Assumptions:** TOE security environment assumptions are given names beginning with "A." and are presented in alphabetical order.

Example:

A.MANAGE There will be one or more competent individuals assigned to manage the TOE and the security of the information it contains.

**Threats:** TOE security threats for the TOE and for the environment are given names beginning with "T." , and are presented in alphabetical order.

Example:

T.ACCESS An unauthorized user may gain access to the TOE and exploit system privileges to gain access to TOE security functions and data.

**Policies:** TOE security environment policies are given names beginning with "P." and are presented in alphabetical order.

Example:

P.ACCOUNTABILITY The Administrators and users of the system shall be held accountable for their security relevant actions within the system.

**Objectives:** Security objectives for the TOE and for the environment are given names beginning with "O." and "OE." Respectively, and are presented in alphabetical order.

Examples:

O.AUTHORIZATION The TSF must ensure that only authorized users gain access to the TOE and its resources.

OE.AUTH_ACCESS The TOE operating environment must ensure that only authorized users gain access to the TOE.

### 1.3.4 Terminology

In the Common Criteria, many terms are defined in Section 2.3 of Part 1. The following terms are a subset of those definitions. They are listed here to aid the reader of the Security Target.

| TERM | DEFINITION |
|---|---|
| **User** | Any entity (human user or external IT entity) outside the TOE that interacts with the TOE. |
| **Human user** | Any person who interacts with the TOE. |
| **Authorized User** | A user that, in accordance with the TOE Security Policy (TSP) may perform an action. (As identified by group membership.) |
| **Role** | A predefined set of rules establishing the allowed interactions between a user and the TOE. |
| **Authentication Date** | Information used to verify the claimed identity of a user. |
| **Component** | The smallest selectable set of elements that may be included in a PP, an ST, or a package. (Components map to Marimba *Channels*) |
| **Guidance Documentation** | Guidance documentation describes the delivery, installation, configuration, operation, management and use of the TOE as these activities apply to the users, administrators, and integrators of the TOE. The requirements on the scope and contents of guidance documents are defined in a PP or ST. |
| **Security Attribute** | Characteristics of subjects, users, objects, information, and/or resources that are used for the enforcement of the TSP. |

| | |
|---|---|
| **Security Functions** | A part or parts of the TOE that have to be relied upon for enforcing a closely related subset of the rules from the TSP. |

### 1.3.5 Acronyms

| ACRONYM | DEFINITION |
|---|---|
| **CC** | **Common Criteria for Information Technology Security Evaluation** |
| **CEM** | **Common Evaluation Methodology** |
| **CM** | **Configuration Management** |
| **EAL** | **Evaluation Assurance Level** |
| **HTTP** | **Hyper Text Transfer Protocol** |
| **IT** | **Information Technology** |
| **J2EE** | **Java 2 Platform, Enterprise Edition** |
| **JSP** | **Java Server Pages** |
| **LDAP** | **Lightweight Directory Access Protocol** |
| **OS** | **Operating System** |
| **PP** | **Protection Profile** |
| **RDBMS** | **Relational Database Management System** |
| **SFP** | **Security Functional Policy** |
| **SFR** | **Security Functional Requirements** |
| **SOF** | **Strength of Function** |
| **SSL** | **Secure Sockets Layer** |
| **ST** | **Security Target** |
| **TOE** | **Target of Evaluation** |
| **TSF** | **TOE Security Function** |
| **TSP** | **TOE Security Policy** |

## 1.4 Security Target Organization

The Security Target contains the following additional sections:

- Section 2 – Target of Evaluation (TOE) Description: This section gives an overview of the TOE, describes the TOE in terms of its physical and logical boundaries, and states the scope of the TOE.
- Section 3 – Security Environment: This section details the expectations of the environment, the threats that are countered by Client and Server Management and its environment and the organizational policy that the Client and Server Management product must fulfill.
- Section 4 – Security Objectives: This section details the security objectives of Client and Server Management and its environment.
- Section 5 – IT Security Requirements: This section presents the security functional requirements (SFR) for Client and Server Management and IT Environment that supports the TOE, and details the requirements for EAL3.
- Section 6 – TOE Summary Specification: This section describes the security functions represented in the Client and Server Management product that satisfy the security requirements.
- Section 7 – Protection Profile Claims: This section identifies whether or not there is a Protection Profile in which conformance is being claimed.
- Section 8 – Rationale: This section closes the ST with the justifications of the security objectives, requirements and TOE summary specifications as to their consistency, completeness and suitability.

# 2. TOE Description

Marimba Client Management from BMC Software and Marimba Server Management from BMC Software are a family of software change and configuration management products produced by BMC Software, Inc., Marimba Product Line, 1030 W. Maude Avenue, Sunnyvale, CA, 94085, herein called simply Client and Server Management. The Server Management software is designed for use with groups of servers, while the Client Management software is designed for use with groups of desktop machines. For the evaluated configuration, all these products must be installed and configured.

The Client and Server Management products allow IT administrators to perform enterprise level change management for both packaged applications and file content, to distribute operating system patches, provision operating systems on 'bare-metal' systems (computer systems that have no operating system installed), track software usage characteristics and gather asset and inventory information for both software and hardware systems deployed in an enterprise. These product functions can be performed on both server and desktop systems, and on a variety of operating systems.

Marimba Client Management from BMC Software and Marimba Server Management from BMC Software operate on the Java Runtime Environment (JRE) version 1.3.1.10 or later.

## 2.1 Product Type

The Marimba Client and Server Management product family is a collection of Java / J2EE based applications that operate in a distributed environment. The product has a centralized administration server, implemented by Common Management Services (CMS), a centralized data server (Transmitter), a series of web-based applications (for example Policy Manager), a series of standalone Java applications (for example, Application packager), and a series of client-side agents (such as the Scanner Service).

The products provide functionality for both the distribution and configuration of packaged applications and content, and also the collection of hardware and software inventory information from within the enterprise and across the internet. The products use industry standard protocols such as HTTP(S) to implement this functionality.

## 2.2 Product Description

The TOE consists of a hierarchy of software packages, starting at the top level as:

- Marimba Patch Management by BMC Software 6.5
- Marimba Content Management by BMC Software with Content Replicator 6.5
- Marimba Desktop/Mobile Application Management by BMC Software with Application Packager 6.5
- Marimba Server Application Management by BMC Software with Application Packager 6.5
- Marimba Desktop OS Management by BMC Software 6.0.3
- Marimba Server OS Management for Unix and Linux by BMC Software 6.0.3

These software packages represent a range of software product 'solutions' produced by BMC Software. In addition, each one of these software packages includes a core set of server and administrative modules packaged as:

- Marimba Control Center by BMC Software 6.0.3 SP2, with SSL enabled, and Publisher and Channel Copier versions 4.6.2, Logging Service 5.0.1 and Policy Service 5.1

BMC Software markets two overriding product families, Marimba Client Management from BMC Software and Marimba Server Management from BMC Software, for managing groups of desktops and servers respectively. All the above listed software packages are included in both the client and server product families, with the exception of Marimba Content Management from BMC Software, which is for use in server environments only.

Each software package is comprised of a series of component external interfaces called 'channels'. A channel represents the smallest component of the BMC products, and BMC's Marimba products are implemented and delivered as a series of these channels. The major channels associated with each software package are described below, together with the purpose of each software package. A complete mapping of software packages to Marimba channels is included in section 2.4.

- **Marimba Patch Management by BMC Software 6.5**

Marimba Patch Management by BMC Software enables the management and deployment of security and functional patches on servers, desktops and laptops across the enterprise. By automating the critical patch management functions such as patch collection, preparation, testing, staging, deployment and auditing, the product helps administrators save time, improve response times and reduce attack related risks.

The major Marimba channel that implements the Patch Management package is called the Patch Manager. Patch Manager is a Marimba channel that enables reporting, packaging, and publishing of patches. This application also enables you to view and edit patch metadata. For example, you can edit metadata (patch details) to specify that certain patches or other software must be installed on a platform before the current patch can be installed.

- **Marimba Server Content Management by BMC Software with Content Replicator 6.5**

Marimba Server Content Management by BMC Software lets IT departments centrally control and monitor the distribution and activation of frequently changing content across servers located in remote offices and data centers. From a simple Web-based console, administrators can specify which files will be published and in what format, from anywhere they may be. Using simple point and click operations, they can package large amounts of content and deploy it to servers around the world in an automated fashion. This significantly reduces the amount of time administrators spend managing code and content updates.

The major channel that implements the Content Distribution package is the Content Replicator. The Content Replicator channel is used to distribute content, data files, such as database images and HTML pages, rather than packaged executable software applications to endpoint servers. Executable applications are packaged using the Application Packager component of the Application Management package.

- **Marimba Desktop/Mobile Application Management by BMC Software with Application Packager 6.5**
- **Marimba Server Application Management by BMC Software with Application Packager 6.5**

Marimba Application Management solutions from BMC software provide the capability to package a wide variety of software package types, in preparation for distribution to client and server endpoints via the Marimba Control Center. The product reduces the number of application packages that administrators must create and manage by allowing administrators to create a single package for application installation, update, self healing and removal.

The major Marimba channel that implements the Application Management package is the Application Packager channel. The Application Packager channel allows you to package a software application (and subsequent updates) into the Marimba package format. You can then distribute this package to target endpoints, such as desktops or servers, using the Marimba Control Center.

- **Marimba Desktop OS Management by BMC Software 6.0.3**
- **Marimba Server OS Management for Unix and Linux by BMC Software 6.0.3**

The Marimba OS Management solutions from BMC Software enable IT organizations to define an OS build, from the bare-metal installation of the OS to the required applications. Importantly, Marimba OS Management enforces standardization of OS builds, which helps IT organizations become more effective and secure

The major Marimba channels that implement Marimba OS Management are the Policy Service, Policy Manager and Infrastructure Administrator, which are also components of Control Center. Additionally, the BootManage Administrator channel is used for Windows OS management.

- **Marimba Control Center by BMC Software**

All the Marimba software packages listed in this section include the Control Center package. Marimba Control Center by BMC Software provides a base platform infrastructure on which all other Marimba applications run on. Control Center also implements the HTTP(S) protocols that enable BMC's Marimba products to efficiently and securely transfer and distribute content and software packages. The major components of Control Center are described below:

- The Tuner component serves two purposes: Firstly, the Tuner provides a Java execution environment for running all other Marimba products. The main service the Tuner provides, in addition to a Java Virtual Machine, is updating capabilities for all applications running in the Tuner environment. These applications can be both Marimba products, and third party packaged applications such as Microsoft Office XP. Due to the updating services provided, the Tuner is present on managed client endpoints in addition to the server and administrator computers.
- The Transmitter operates as a data server, providing the content for updating and distributing software packages. Since the Transmitter is a Marimba Java based channel, it also runs in the Tuner environment. This can sometimes be confusing, since the Tuner operates both as the client component, and as the Java environment for the Server components.

- The Common Management Services (CMS) is a part of the Marimba Control Center that provides a Servlet/JSP-based application server, on top of which Marimba applications are run. CMS also provides centralized services, such as database and LDAP connection pooling and user role support.
- Policy Manager allows an administrator to assign software application packages to groups of client and server endpoints based on membership in an LDAP directory-based user or machine group. Policy Manager distributes applications and content using a client-initiated request action.
- Deployment Manager allows an administrator to assign software application packages and content to groups of server endpoints based on membership in a machine group hierarchy that is internally implemented and stored in the product. Deployment Manager distributes applications and content using a server-initiated push action.

The following table summarizes the administrative interfaces that are used to manage security functions:

| Interface | Management Function |
|---|---|
| Infrastructure Administration (includes Transmitter Administrator, Tuner Administrator and Proxy Administrator) | Runs on Common Management Services (CMS) and is often referred to as the *console*. Allows the administrator to administer various Marimba infrastructure components. |
| Certificate Manager | Management of SSL certificates. |
| Channel Copier | Enables copying and publishing (*sending to the Marimba transmitter)* software packages prior to distribution using the Marimba products. |
| Publisher | Enables publishing (*sending to the Marimba transmitter)* software packages prior to distribution using the Marimba products. |
| Report Center | Runs on Common Management Services (CMS) and allows and administrators to view and report on audit logs, and also hardware and software inventory data. |
| Deployment Manager | Manages the distribution of software packages to server endpoints using a GUI. |
| Deployment Manager Command Line | Manages the distribution of software packages to server endpoints using a command line interface (CLI). Note: For Deployment Manager only, the command line functions are implemented using a separate Marimba channel. For all other command line functions, the functionality is integrated into the same channel that provides the GUI. |
| Policy Manager | Manages the policy based distribution of software packages to client and server endpoints using a GUI and command line interface (CLI). |
| Patch Manager | Manages the distribution of OS level patches to clients and servers. |
| Application Packager | Used to package and publish software packages prior to distribution using the Marimba products. |

**Table 1: Identification of Administrative Interfaces**

## 2.3 Product Security Features

The TOE implements the following features:

- Events within the Transmitter and Patch Manager are logged, including security events. Audit records include the name of the user associated with the event, a description of the event, and the date and time of the event. Authorized administrators may view audit logs using the Report Center component.
- Access to the channels, deployment manager folders, deployments, task groups, server groups, server keychains, policy manager targets, and patch groups is controlled by the identity of the user and/or group membership and the access control attributes associated with the named objects. Channel based access control is implemented by the Transmitter component. Deployment manager folders, deployments, task groups, server groups, and server keychains access control is implemented by the Deployment Manager. Policy manager target access control is implemented by the Policy Manager, and patch group access control is implemented by the Patch Manager.
- Marimba Client and Server Management from BMC Software provides user identification mechanisms for all administrative applications. The user is authenticated against an external user database such as LDAP for all components except the Deployment Manager. The LDAP server is external to the TOE. In the Deployment Manager component of Server Management, the administrative user is authenticated against a user database, which is part of the TOE.
- Marimba Control Center from BMC Software uses the secure sockets layer (SSL) protocol to ensure that information passed between the Transmitter and Tuner, and also information passed between web browsers and the Common Management Services (CMS) and Deployment Manager components is encrypted. Additionally, communication between the TOE, and the external LDAP server and the external Red Hat Satellite Server takes place over an encrypted channel. SSL communication can be individually enabled or disabled for each TSF that supports encryption. For the evaluated configurations, SSL functionality is enabled for all TSFs that support encryption. There are no TSF interfaces that do not support SSL encrypted communication.

## 2.4 Security Environment TOE Boundary

The TOE includes both physical and logical boundaries.

### 2.4.1 Physical Boundaries

The TOE is a suite of software applications operating on server and client machines, and the Java Runtime Environment (JRE) version 1.3.1.10 or later. The physical boundary for each component of the TOE is the environment that each component requires for effective operation. The operating system, external Relational Database Management System (RDBMS), Light-weight Directory Access Protocol (LDAP), and hardware are included by assumption and are not part of the TOE. The following is a description of the components that comprise the TOE.

For the purposes of describing the physical boundaries, the product will be described in terms of the component Marimba channels rather than the parent software packages. A mapping of software packages to Marimba channels is also included in this section. It should be noted that the Marimba product is packaged as a series of Marimba channels, and that this same channel format is also used by the product to package and distribute 3rd party software products to clients and servers. Some examples of 3rd party packages are application types such as Windows, Java, Visual Basic etc., and content, such as a series of HTML files for a website.

The Client and Server Management software packages consist of many Marimba channels, however the following list highlights only the *major* channels, and have been selected because they implement the majority of the security functions. These include the Tuner, the Transmitter, the Infrastructure Administrator, Patch Manager, Report Center,

Policy Manager, Deployment Manager, Content Replicator and Marimba Proxy. Other Marimba channels are included in the TOE, but do not warrant a complete description here as they typically either do not provide any security functions, or provide supporting functionality to the major channels. For example, the Schema Manager channel is used to install the underlying database schema on the RDBMS in the IT environment. The channel is accessed via CMS, but provides no security functions, is used only during installation of the product and hence is considered minor. The Scanner Service channel is used to gather hardware and software inventory data from managed endpoint computers, however the channel provides no security functions and has no user interface, and hence is considered minor.

The major Marimba channels are described below:

- The Tuner component is the application with which users subscribe to channels that have been published on the Transmitter component. The Tuner downloads the channel files or the updates to the channel files, to the user's workstation. In addition, the Tuner provides a Java execution environment for running all other Marimba products, including the Transmitter. For that reason the Tuner is present on all computers running the Marimba software, i.e. managed endpoints, servers, and administrative computers. Figure 1, shows the Tuner running on all five computer systems, and providing application change management to the computer designated as the Managed Endpoint Computer.
- The Transmitter component is a server that delivers channels to its clients' Tuners. The Transmitter itself runs in the Tuner environment, and operates as the server in the Marimba product distributed environment. The Transmitter is running on the computer designated as "Server Components" in Figure 1, and is providing application change management services to the "Managed Endpoint Computer".
- Infrastructure Administrator (contained in the CMS) provides the administrator with the ability to install, configure, and manage the components of the TOE. Infrastructure Administrator is shown in Figure 1 running on the computer designated as the "Administrator's Workstation".
- Policy Manager (contained in the CMS) is the application for assigning channels and otherwise managing the subscription capabilities.
- Report Center (contained in the CMS) provides the interface for scheduling data collection and otherwise administering the configuration and discovery process, as well as searching the collected data for specific information and reporting the results. Since this tool is used by administrators, it is shown in Figure 1, running on the computer designated as the "Administrator's Workstation".
- Patch Manager (contained in CMS) provides an interface that enables an administrator to perform administrative tasks on the Patch Management product, including managing patch groups, configuring the patch repository update schedule and configuring the Patch Service channel running on client and server endpoints
- The Deployment Manager and Deployment Manager Command Line components of Server Management provide centralized control and monitoring of content distribution. Since these tools are used by administrators, they are shown in Figure 1, running on the computer designated as the "Administrator's Workstation". The associated component running on managed endpoint computers is called the Deployment Service channel.
- The Content Replicator component (Content Distribution module) performs the tasks of installing data and content on managed server endpoints, and rolling back installations. Deployment Manager is used to run Content Replicator remotely. Since the Content Replicator component is used to provide content to server endpoints, it is shown running on the computer designated as the "Managed Endpoint Computer" in Figure 1.
- The Marimba Proxy is an architectural element that acts as an intermediary between clients and servers. In most cases, endpoint tuners act as the Proxy's clients, and transmitters act as the Proxy's servers. The Marimba Proxy is an HTTP-based proxy that caches Marimba channel content only.
- The Publisher, Channel Copier and Application Packager are used to publish data, for example software packages, to the Transmitter prior to distribution using the Marimba infrastructure.

The TOE is comprised of the following product packages, each of which contain the external interfaces (channels) indicated in the following table:

| Software Package Name | Channel Name (interface) |
|---|---|
| • Marimba Control Center by BMC Software | Channel Manager |
| | Infrastructure Administration |
| | Console Window |
| | Help Manager |
| | Infrastructure Service |
| | Schema Manager |
| | Certificate Manager |
| | Channel Copier |
| | Publisher |
| | Transmitter |
| | Proxy |
| | Subnet Repeater Policy |
| | Logging Service |
| | Scanner Service[1] |
| | Report Center |
| | Common Management Services |
| | Deployment Manager |
| | Deployment Service |
| | Deployment Manager Command Line |
| | Policy Manager[2] |
| | Policy Service[2] |
| | Subscription Reporter |
| | Tuner (not a channel, but a component of Control Center) |
| | |
| • Marimba Patch Management by BMC Software | Patch Manager |
| | Patch Service |
| | Solaris Patch Source |
| | Windows Patch Source |
| | Red Hat Enterprise Linux Patch Source |
| | Schema Manager |
| | Policy Manager[2] |
| | |
| • Marimba Server Content Management by BMC Software | Content Replicator |
| | |
| • Marimba Desktop/Mobile Application Management by BMC Software<br>• Marimba Server Application Management by BMC Software | Application Packager |
| | |
| • Marimba Desktop OS Management for Windows by BMC Software | Policy Service[2] |
| | Policy Manager[2] |
| • Marimba Server OS Management for Unix and Linux by | Infrastructure Administrator |

| Software Package Name | Channel Name (interface) |
|---|---|
| BMC Software | BootManage Administrator (Windows) |

**Table 2: Mapping of Software Packages to Components (Channels)**

1. The Scanner Service channel is also referred to as *Inventory Service* in various sections of the BMC Marimba guidance documentation.

2. The Policy Manager and Policy Service are sometime referred to as *Subscription Policy Manager* and *Subscription Policy Service* in various sections of the BMC Marimba guidance documentation.

The following diagram depicts the Client and Server Management component boundaries. This diagram shows the evaluated configuration of the TOE distributed amongst five computers. The RDBMS, LDAP server and Patch Vendor are not part of the TOE, but are part of the environment required to operate the products. The computer hardware, and associated components such as networking equipment are also not part of the TOE, but are part of the environment required to operate the products.

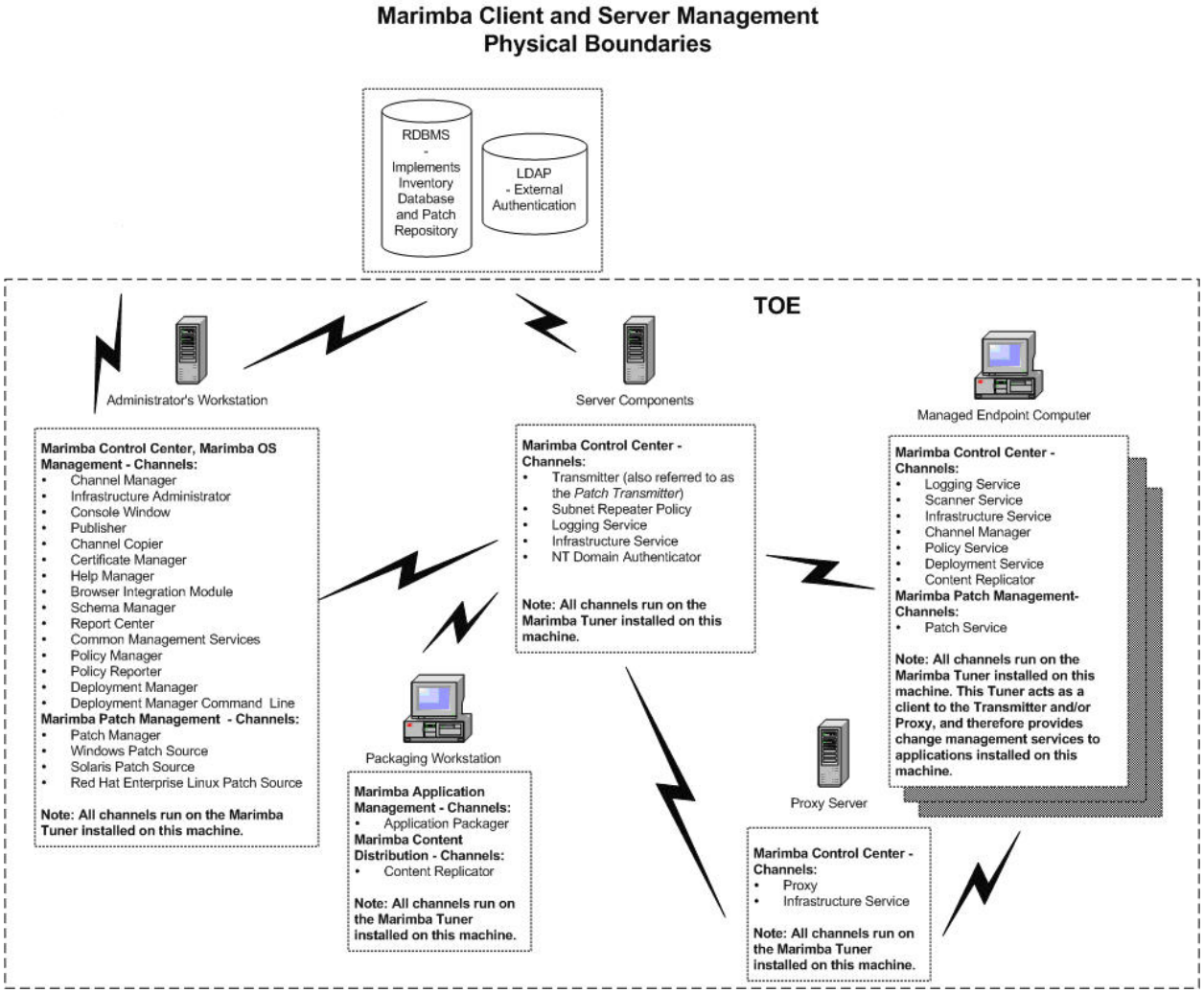


**Figure 1: Client and Server Management TOE Physical Boundaries**

## 2.4.2 Logical Boundaries

The logical boundaries of the TOE include the functions of the TOE interfaces. These functions include cryptographic support, audit, user data protection, identification and authentication, TOE access, trusted path/channel and the management of the security configurations.

### 2.4.2.1 Security Audit

Marimba Client and Server Management audits the actions that occur on the Transmitter and Patch Manager. The log files contain information about events such as starting the Transmitter and Patch Manager, modifying access control attributes associated with channels and patch groups, as well as any problems associated with those events.

#### 2.4.2.2 Cryptographic Support

The TSFs in Marimba Control Center utilize encrypted network protocols to provide secure communication paths between components of the TOE, and secure communication channels between the TOE and external components in the IT environment. The protocols used are HTTP and LDAPS which in turn utilizes the Secure Socket Layer (SSL) protocol. The TOE implements the cryptographic support, including all encryption and key management functions, using technology from RSA Security, Inc. The TOE cryptographic functions use the RC4 encryption algorithm and RSA key exchange.

#### 2.4.2.3 User Data Protection

Marimba Client and Server Management provides access control to the various channels and other named objects, controlled by the combination of user and group identification and the access control attributes associated with the various named objects.

#### 2.4.2.4 Identification and Authentication

Marimba Client and Server Management requires users to be identified and authenticated before they can access the TOE and the TOE security-relevant data.

#### 2.4.2.5 Security Management

The TOE provides a number of interfaces to manage the configuration and implementation of the policy enforced by the TOE. Security management includes managing the following items: access control of channels and configuring termination of inactive sessions.

#### 2.4.2.6 TOE Access

The CMS component monitors an established session for activity and if the session is inactive for the specified time period, CMS will terminate the session.

#### 2.4.2.7 Trusted Path/Channel

The TSFs in Marimba Control Center use the secure sockets layer (SSL) protocol, and the associated HTTPS protocol to enforce a trusted path for all communication between the Transmitter and client and server endpoint computers, and between the administrator's web browser and the Common Management Services (CMS) and Deployment Manager components.

The Infrastructure Administrator component uses the SSL protocol to provide a trusted path between the Infrastructure Administration channel, and the Tuner's, Transmitters and Proxies to which it is providing administration functionality. The Transmitter Administration functionality of the Infrastructure Administration component uses the SSL protocol to provide a trusted path when copying data (channels) between two Transmitters.

The Publisher, Channel Copier and Application Packager components use the SSL protocol to provide a trusted path between the components and the Transmitter, thus encrypting all published data (for example software packages being uploaded to the Transmitter prior to distribution)

Additionally, the TSFs enforce a trusted channel between Control Center and the LDAP server external to the TOE, and between the Red Hat Enterprise Linux Patch Source channel and the external Red Hat Satellite Server.

# 3. Security Environment

The TOE security environment describes the security aspects of the environment in which the TOE is intended to be used and the manner in which it is expected to be employed.

The statement of the TOE security environment defines the following:

- Threats that the product is designed to counter

- Organizational security policies with which the product is designed to comply
- Assumptions made on the operational environment and the method of use intended for the product

The TOE, Marimba Client and Server Management, has been developed for an operating environment with a medium level of risk to identified assets. The assurance requirements of EAL 3 and the minimum strength of function of SOF-basic were chosen to be consistent with that level of risk.

## 3.1 Threats to Security

| | |
|---|---|
| T.ACCESS | An unauthorized user may gain access to the TOE and exploit system privileges to gain access to TOE security functions and data. |
| T.AUDIT_CORRUPT | Unauthorized users may tamper with audit data by gaining unauthorized access to the audit trail. |
| T. DISCLOSURE_OF_ CERTIFICATES | A server certificate, which is used to enable encrypted communications, is copied or otherwise accessed by an administrator and then installed on a rogue server with the intent to allow a malicious server to impersonate a known secure server. |
| T.DISCLOSURE_OF_ COMMUNICATION | The contents of network communication between components of the TOE may be read by unauthorized personnel. Such an attack could occur using a commercially available network analyzer or 'packet sniffer' tool with the intent to view sensitive information such as passwords, which could then be used for further malicious attacks. |
| T.DISCLOSURE_OF_ PRIVATE_KEYS | A private key is disclosed or modified by a process executing in a network node that has no valid reason for viewing or modifying the key. SSL uses public key cryptography to encrypt network communications. The resulting encrypted communication is then decrypted using a private key. For SSL to be secure, the private key must never be disclosed to any malicious person trying to decrypt network communications. Such an attack could occur using a commercially available network analyzer or 'packet sniffer' tool. |
| T. PRIVILEGE | An authorized user of the TOE may gain access to a channel or other named object without having permission. |

## 3.2 Organizational Security Policies

An organizational security policy is a set of rules, practices, and procedures imposed by an organization to address its security needs. This section identifies the organizational security policies applicable to Client and Server Management.

| | |
|---|---|
| P.ACCOUNTABILITY | All users of the system shall be held accountable for their security relevant actions within the system. |
| P.MANAGE | The system must provide authorized administrators with utilities to effectively manage the security functions of the TOE. |

## 3.3 Secure Usage Assumptions

This section describes the security aspects of the environment in which the TOE will be utilized. This includes information about the physical, personnel, and system aspects of the environment.

### 3.3.1 Physical Assumptions

| | |
|---|---|
| A.CONNECT | Any network resources used for communication between TOE components will be adequately protected from unauthorized access. |
| A.PROTECT | The components of TOE software critical to security policy enforcement must be located within controlled access facilities that will be protected from unauthorized physical access and modification. |

### 3.3.2 Personnel Assumptions

| | |
|---|---|
| A.MANAGE | There will be one or more competent individuals assigned to manage the TOE and the security of the information it contains. |
| A.NOEVIL | The administrative personnel are not careless, willfully negligent, or hostile and will follow and abide by the instructions provided by the administrative guidance. |
| A.PLATFORM_REQUIREMENTS | The administrative personal have followed the instructions provided in the administrative guidance that recommend minimum hardware and software requirements on which the TOE operates. The administrative personal have configured the computer system on which the TOE is operating based on instructions in the administrative guidance. |

### 3.3.3 System Assumptions

| | |
|---|---|
| A.OPERATE_CORRECTLY | The computer platforms and operating systems in the environment are operating in a generally defect-free manner and follow either the manufacturers specifications, or accepted industry standards such as the secure sockets layer (SSL) protocol. |
| A.IDENT | The operating environment will provide a method of identification and authentication. |
| A.INSTALL | Those responsible for the TOE must ensure that the TOE is delivered, installed, managed, and operated in a manner that maintains the TOE security objectives. |
| A.SYSPROTECT | The operating environment will provide protection to the TOE and its related data. |
| A.TIME | The operating environment will provide reliable system time. |

# 4. Security Objectives

This section defines the security objectives of the Client and Server Management and the supporting environment. Security objectives, categorized as either IT or non-IT security objectives, reflect the stated intent to counter identified threats and/or comply with any identified organizational security policies. All of the identified threats and organizational policies are addressed under one of the categories below.

## 4.1 IT Security Objectives for the TOE

The following security objectives are intended to be satisfied by the TOE.

| | |
|---|---|
| O.AUTHORIZATION | The TSF must ensure that only authorized users gain access to the TOE and its resources. |
| O.CLIENT_TO_SERVER_ CONFIDENTIALITY | Communication data cannot be read by a user other than at the communication end-points, client and server. |
| O.SERVER_TO_LDAP_ CONFIDENTIALITY | Communication data between the TOE and the external LDAP server cannot be read by unauthorized users. |
| O.PATCH_VENDOR_ CONFIDENTIALITY | Communication data between the TOE and the Red Hat Satellite Server cannot be read by unauthorized users. |
| O.ENCRYPTION | The content of all communication to and from TSFs is encrypted using SSL. |
| O.OBJ_ACCESS | The TSF must limit access to named objects maintained by the TOE to users with authorization and appropriate privileges. The TSF must allow authorized users to specify which users may access their objects and the actions performed on the objects. |
| O.AUDIT | The TOE must record security-relevant events, associate these events with users, dates and times, and make audit information available to authorized administrators, prohibiting access to regular users. |
| O.MANAGE | The TOE must allow administrators to effectively manage the TOE and its security functions, and must ensure that only authorized administrators are able to access such functionality. |
| O.VERIFY_CERTIFICATES | The TOE must verify that when SSL encryption certificates are imported into the Certificate Manager they follow the correct certificate data format, and are rejected if they have been tampered with. |

## 4.2 IT Security Objectives for the Environment

The following security objectives for the IT environment of the TOE must be satisfied in order for the TOE to fulfill its own security objectives.

| | |
|---|---|
| OE.AUTH_ACCESS | The TOE operating environment must ensure that only authorized users gain access to the TOE. |
| OE.SSL_IMP | All web browsers in the operating environments shall provide an implementation of the SSL and HTTPS protocols whereby each session is managed in its own protocol domain. |
| OE.LDAP_SSL_IMP | The LDAP server which is external to the TOE shall provide an implementation of the SSL and LDAPS protocols whereby each session is managed in its own protocol domain. |
| OE.PATCH_VENDOR_ SSL_IMP | The Red Hat Satellite Server which is external to the TOE shall provide an implementation of the SSL protocol whereby each session is managed in its own protocol domain. |
| OE.SEP | The TOE operating environment shall provide mechanisms to isolate the TOE Security Functions (TSF) and assure that TSF components cannot be tampered with or bypassed. |
| OE.TIME_SOURCE | The IT environment for the TOE must provide a reliable time source for the TOE to generate accurate timestamps for audit records. |

## 4.3 Security Objectives of the Non-IT Environment

The following security objectives are intended to be satisfied by the environment of the TOE.

| | |
|---|---|
| OE.PLATFORM_SUPPORT | The TOE environment must provide reliable platform functions, including hardware and software that meet or exceed the minimum platform requirements, and hardware and software that has been configured correctly. |
| OE.INSTALL | Those responsible for the TOE must ensure that the TOE is delivered, installed, managed, and operated in a manner that maintains the TOE security objectives. |
| OE.PERSON | Authorized Administrators of the TOE shall be properly trained in the configuration and usage of the TOE and will follow the guidance provided. These users are not careless, negligent, or hostile. |
| OE.PHYCAL | Those responsible for the TOE must ensure that the parts of the TOE critical to security policy are protected from physical attack that might compromise the TOE security objectives. |

# 5. IT Security Requirements

This section of the ST details the security functional requirements (SFR) for the TOE and the IT Environment that will support the TOE. The SFR were drawn from the CC Part 2.

CC defined operations for assignment, iteration, selection, and refinement were used to tailor the requirements to the level of detail necessary to meet the stated security objectives. Refer to Conventions section for the format definition.

## 5.1 TOE Security Functional Requirements

| Security Functional Class | Security Functional Components |
|---|---|
| Security audit (FAU) | Audit data generation (FAU_GEN.1) |
| | User identity association (FAU_GEN.2) |
| | Audit review (FAU_SAR.1) |
| | Selective audit (FAU_SEL.1) |
| Cryptographic support (FCS) | Cryptographic Operation (FCS_COP.1(a)) |
| | Cryptographic Operation (FCS_COP.1(b)) |
| User data protection (FDP) | Subset access control (Deployment manager named objects) (FDP_ACC.1(a)) |
| | Subset access control (Transmitter folders, channels) (FDP_ACC.1(b)) |
| | Subset access control (Policy manager targets) (FDP_ACC.1(c)) |
| | Subset access control (Patch manager patch groups) (FDP_ACC.1(d)) |
| | Security attribute based access control (Deployment manager named objects) (FDP_ACF.1(a)) |
| | Security attribute based access control (Transmitter folders, channels) (FDP_ACF.1(b)) |
| | Security attribute based access control (Policy manager targets) (FDP_ACF.1(c)) |

| Security Functional Class | Security Functional Components |
|---|---|
| | Security attribute based access control (Patch manager patch groups) (FDP_ACF.1(d)) |
| Identification and authentication (FIA) | User attribute definition (FIA_ATD.1) |
| | Timing of authentication (FIA_UAU.1) |
| | User identification before any action (FIA_UID.2) |
| Security management (FMT) | Management of security attributes (Deployment manager named objects) (FMT_MSA.1(a)) |
| | Management of security attributes (Transmitter folders, channels) (FMT_MSA.1(b)) |
| | Management of security attributes (Policy manager targets) (FMT_MSA.1(c)) |
| | Management of security attributes (Patch manager patch groups) (FMT_MSA.1(d)) |
| | Secure Security Attributes (Handling of encryption certificates) (FMT_MSA.2) |
| | Static attribute initialization (Deployment Manager named objects) (FMT_MSA.3(a)) |
| | Static attribute initialization (Transmitter folder, channels) (FMT_MSA.3(b)) |
| | Static attribute initialization (Policy manager targets) (FMT_MSA.3(c)) |
| | Static attribute initialization (Patch manager patch groups) (FMT_MSA.3(d)) |
| | Management of TSF data (Audit) (FMT_MTD.1(a)) |
| | Management of TSF data (Access Control) (FMT_MTD.1(b) and FMT_MTD.1(c)) |
| | Specification of management functions (FMT_SMF.1) |
| | Security roles (FMT_SMR.1) |
| TOE access (FTA) | TSF-initiated termination (FTA_SSL.3) |
| Trusted Path/Channel (FTP) | Inter-TSF channel (FTP_ITC.1) |
| | Trusted Path (FTP_TRP.1) |

**Table 3: Security Functional Components**

## 5.1.1 Security Audit (FAU)

### 5.1.1.1 Audit data generation (FAU_GEN.1)

5.1.1.1.1 FAU_GEN.1.1

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

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

b) All auditable events for the [not specified] level of audit; and

c) [**Auditable events in Table 4 below**].

| Component | Auditable Event |
|---|---|
| FMT_MSA.1(b) | Use of the functions listed in this requirement |
| FMT_MSA.1(d) | Operations performed on the named objects listed in this requirement |

**Table 4: Auditable Events**

5.1.1.1.2 FAU_GEN.1.2

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

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

b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, [**object identity (channel),** and **severity**].

Note: Success and failure is not explicitly stated by a message such as "Success" or "Fail", but is deterministically implied by the event type description.

#### 5.1.1.2 User identity association (FAU_GEN.2)

5.1.1.2.1 FAU_GEN.2.1

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

#### 5.1.1.3 Audit review (FAU_SAR.1)

5.1.1.3.1 FAU_SAR.1.1

The TSF shall provide [**Primary Administrator, Administrator, Operator**] with the capability to read [**all audit trail data**] from the audit records.

5.1.1.3.2 FAU_SAR.1.2

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

#### 5.1.1.4 Selective audit (FAU_SEL.1)

5.1.1.4.1 FAU_SEL.1.1

The TSF shall be able to include or exclude auditable events **as instructed by a Primary Administrator, Administrator, or Operator** from the set of audited events, based on the following attributes:

a) [event type]
b) [**severity and event type ranges**].

### 5.1.2 Cryptographic support (FCS)

#### 5.1.2.1 Cryptographic operation (FCS_COP.1(a))

5.1.2.1.1 FCS_COP.1.1 (a)

The TSF shall perform [**encryption and decryption of network communications using the SSL protocol**] in accordance with a specified cryptographic algorithm [**RC4**] and cryptographic key sizes [**128**] that meet the following: [**RC4 algorithm as implemented by RSA**].

#### 5.1.2.2 Cryptographic operation (FCS_COP.1(b))

5.1.2.2.1 FCS_COP.1.1 (b)

The TSF shall perform **[key exchange for network communications using the SSL protocol]** in accordance with a specified cryptographic algorithm **[RSA]** and cryptographic key sizes **[512, 1024]** that meet the following: **[RSA key-exchange algorithm as implemented by RSA]**.

## 5.1.3 User Data Protection (FDP)

### 5.1.3.1 Subset access control (FDP_ACC.1(a))

5.1.3.1.1 FDP_ACC.1.1(a)

The TSF shall enforce the [**Deployment manager access control and create permissions SFP**] on [**subject: users and process acting on behalf of the user; objects: named objects, Deployment Manager Folders, deployments, task groups, server groups, server keychains; operations: read, write, execute, owner, and create, except the read operation in conjunction with the Deployment Manager Command Line Interface. The read operation refers to command line operations that do not modify data, and start with the prefix 'get'].**

### 5.1.3.2 Subset access control (FDP_ACC.1(b))

5.1.3.2.1 FDP_ACC.1.1(b)

The TSF shall enforce the [**Transmitter access control SFP**] on [**subject: users and process acting on behalf of the user; objects: named objects, Transmitter folders, channels; operations: read, write, and delete].**

### 5.1.3.3 Subset access control (FDP_ACC.1(c))

5.1.3.3.1 FDP_ACC.1.1(c)

The TSF shall enforce the [**Policy manager access control SFP**] on [**subject: users and process acting on behalf of the user; objects: named objects, policy manager targets; operations: ACL read, ACL write, policy read, policy write].**

### 5.1.3.4 Subset access control (FDP_ACC.1(d)

5.1.3.4.1 FDP_ACC.1.1(d)

The TSF shall enforce the [**Patch manager access control SFP**] on [**subject: users and process acting on behalf of the user; objects: named objects, patch manager patch groups; operations: modify, publish, delete].**

### 5.1.3.5 Security attribute based access control (FDP_ACF.1(a))

5.1.3.5.1 FDP_ACF.1.1(a)

The TSF shall enforce the [**Deployment manager access control and create permissions SFP**] to objects based on [

- **Deployment Manager access control for Deployment Manager Folders, Deployments, Server Groups, Server Keychains, Task Group named objects. Each of these named objects maintains an ACL containing the following attributes:**
  - **A user or group name**
  - **Four Boolean permission bits representing Read (r), Write (w), Execute (x) and Owner (o), associated with each user or group**
- **Deployment Manager create permissions for Deployment Manager Folders, Deployments, Server Groups, Server Keychains, Task Group named objects. Each user and group managed by the Deployment Manager has a series of attributes with the following values associated with it:**
  - **Five create attributes, one each for: Deployment Manager Folders, Deployments, Server Groups, Server Keychains, Task Groups**
  - **Each of these attributes can have three possible values: None, Create, Revoke].**

### 5.1.3.6 Security attribute based access control (FDP_ACF.1(b))

5.1.3.6.1 FDP_ACF.1.1(b)

The TSF shall enforce the [**Transmitter access control SFP**] to objects based on [

- **Transmitter access control for read, write and delete operations on Channel and Transmitter Folder named objects. Access is controlled by an ACL containing a single attribute, that can have the following values:**
    - **All users that have been authenticated by an external source**
    - **A specific user authenticated by an external source**
    - **A group, of which only authenticated members have access].**

#### 5.1.3.7 Security attribute based access control (FDP_ACF.1(c))

5.1.3.7.1 FDP_ACF.1.1(c)

The TSF shall enforce the [**Policy manager access control SFP**] to objects based on [

- **Policy Manager access control for Policy Manager Targets. Each user or group of users (stored and authenticated by an external source) has an ACL associated with it, containing the following attributes:**
    - **One or more Policy Manager Targets**
    - **Four Boolean permission bits associated with each Policy Manager Target for the following permissions: ACL read, ACL write, policy read, policy write].**

#### 5.1.3.8 Security attribute based access control (FDP_ACF.1(d))

5.1.3.8.1 FDP_ACF.1.1(d)

The TSF shall enforce the [**Patch manager access control SFP**] to objects based on [

- **Patch Manager access control for Patch Group named objects. Each Patch Group has an ACL associated with it that contains the following attributes:**
    - **One or more users or user groups (stored and authenticated by an external source)**
    - **Three Boolean permission bits associated with each user or user group for the following Patch Group permissions: modify, publish, delete].**

5.1.3.8.2 FDP_ACF.1.2

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

- **Named object access is allowed if at least one of the following conditions is true:**
    - **an ACL entry explicitly grants access to a user**
    - **an ACL entry explicitly grants access to a group of which the subject is a member**
    - **the subject is the object owner (folders, keychains, task groups, server groups, deployments, and patch manager patch groups)**].

5.1.3.8.3 FDP_ACF.1.3

The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: [**no additional rules**].

5.1.3.8.4 FDP_ACF.1.4

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

## 5.1.4 Identification and Authentication (FIA)

### 5.1.4.1 User attribute definition (FIA_ATD.1)

5.1.4.1.1 FIA_ATD.1.1

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

- **user identity**
- **group membership**
- **home (default) folder**
- **password (authentication data)**.
- **create permissions (for folders, keychains, task groups, server groups, and deployments)].**

### 5.1.4.2 Timing of authentication (FIA_UAU.1)[1]

5.1.4.2.1 FIA_UAU.1.1

The TSF shall allow [**Common Management Services operations to be performed after having been authenticated by an LDAP server in the IT environment**] on behalf of the user to be performed before the user is authenticated.

5.1.4.2.2 FIA_UAU.1.2

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

### 5.1.4.3 User identification before any action (FIA_UID.2)

5.1.4.3.1 FIA_UID.2.1

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

## 5.1.5 Security management (FMT)

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

5.1.5.1.1 FMT_MSA.1.1(a)

The TSF shall enforce the [**Deployment manager access control and create permissions SFP**] to restrict the ability to [change_default, modify,] the security attributes [**access control attributes associated with deployment manager folders, deployments, task groups, server groups, server keychains, and create permissions associated with users**] to [**Deployment Manager Administrator and Object Owners**].

[1] FIA_UAU.1.1 describes the fact that for CMS components, the user is authenticated against an external LDAP database. The LDAP server is external to the TOE.

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

5.1.5.2.1 FMT_MSA.1.1(b)

The TSF shall enforce the [**Transmitter access control SFP**] to restrict the ability to [change_default, modify, delete] the security attributes [**access control attributes associated with channels**] to [**Primary Administrator, Administrator**].

#### 5.1.5.3 Management of security attributes (FMT_MSA.1(c))

5.1.5.3.1 FMT_MSA.1.1(c)

The TSF shall enforce the [**Policy manager access control SFP**] to restrict the ability to [change default, modify, delete] the security attributes [**access control attributes associated with policy manager targets**] to [**Primary Administrator, users with ACL write permissions**].

#### 5.1.5.4 Management of security attributes (FMT_MSA.1(d))

5.1.5.4.1 FMT_MSA.1.1(d)

The TSF shall enforce the [**Patch manager access control SFP**] to restrict the ability to [modify, delete] the security attributes [**access control attributes associated with patch manager patch groups**] to [**Primary Administrator, Administrators and Object Owners (Administrator who created the patch group)**].

#### 5.1.5.5 Secure Security Attributes (FMT_MSA.2)

5.1.5.5.1 FMT_MSA.2.1

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

[NOTE: This SFR is present to cover the handling of encryption certificates only]

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

5.1.5.6.1 FMT_MSA.3.1(a)

The TSF shall enforce the [**Deployment manager access control and create permissions SFP**] to provide [restrictive] default values for security attributes that are used to enforce the SFP.

5.1.5.6.2 FMT_MSA.3.2(a)

The TSF shall allow the **[Deployment Manager Administrator or Object Owner (in Deployment Manager)]** to specify alternative initial values to override the default values when an object or information is created.

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

5.1.5.7.1 FMT_MSA.3.1(b)

The TSF shall enforce the [**Transmitter access control SFP]** to provide [permissive] default values for security attributes that are used to enforce the SFP.

5.1.5.7.2 FMT_MSA.3.2(b)

The TSF shall allow the **[Primary Administrator, or Administrator (in CMS)]** to specify alternative initial values to override the default values when an object or information is created.

#### 5.1.5.8 Static attribute initialization (FMT_MSA.3(c))

5.1.5.8.1 FMT_MSA.3.1(c)

The TSF shall enforce the [**Policy manager access control SFP]** to provide [restrictive] default values for security attributes that are used to enforce the SFP.

5.1.5.8.2 FMT_MSA.3.2(c)

The TSF shall allow the [**Primary Administrator or users with ACL write permissions (in Policy Manager)]** to specify alternative initial values to override the default values when an object or information is created.

#### 5.1.5.9 Static attribute initialization (FMT_MSA.3(d))

5.1.5.9.1 FMT_MSA.3.1(d)

The TSF shall enforce the [**Patch manager access control SFP]** to provide [restrictive] default values for security attributes that are used to enforce the SFP.

5.1.5.9.2 FMT_MSA.3.2(d)

The TSF shall allow the [**Primary Administrators or Object Owners (in Patch Manager)]** to specify alternative initial values to override the default values when an object or information is created.

#### 5.1.5.10 Management of TSF data (FMT_MTD.1(a))

5.1.5.10.1 FMT_MTD.1.1(a)

The TSF shall restrict the ability to [query] the [**audit data]** to the **[the Primary Administrator, Administrator, Operator**].

#### 5.1.5.11 Management of TSF data (FMT_MTD.1(b))

5.1.5.11.1 FMT_MTD.1.1(b)

The TSF shall restrict the ability to [modify, delete, [**initialize**]] **the** [Regular **User's security attributes**] to [**the Deployment Manager Administrator** ]

#### 5.1.5.12 Management of TSF data (FMT_MTD.1(c)))

5.1.5.12.1 FMT_MTD.1.1(c)

The TSF shall restrict the ability to [modify. delete, [**initialize**]] the [**owner's user security attributes**] to [**Regular Users who can only modify their own security attributes**].

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

5.1.5.13.1 FMT_SMF.1.1

The TSF shall be capable of performing the following security management functions: [

- **Management of access control to named objects**
- **The installation of encryption certificates**
- **Configure idle user timeout**].

#### 5.1.5.14 Security roles (FMT_SMR.1)

5.1.5.14.1 FMT_SMR.1.1

The TSF shall maintain the roles [

- **Deployment Manager - Deployment Manager Administrator, Regular User , Object Owners**
- **CMS (Infrastructure Administration, Report Center, Policy Manager, Patch Manager) - Primary Administrator, Administrator, Policy Administrator, Operator, users with ACL Write Permissions (also referred to as *ACL Administrator*), Object Owners**].

5.1.5.14.2 FMT_SMR.1.2

The TSF shall be able to associate users with roles.

### 5.1.6 TOE access (FTA)

#### 5.1.6.1 TSF-initiated termination (FTA_SSL.3)

5.1.6.1.1 FTA_SSL.3.1

The TSF shall terminate an interactive session after a [**a Deployment Manager Administrator or Primary Administrator specified time**].

### 5.1.7 Trusted Path/Channels

#### 5.1.7.1 Inter-TSF trusted channel (FTP_ITC.1)

5.1.7.1.1 FTP_ITC.1.1

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

5.1.7.1.2 FTP_ITC.1.2

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

5.1.7.1.3 FTP_ITC.1.3

The TSF shall initiate communication via the trusted channel for [**accessing user and user group entries, computer and computer group entries, other application data associated with Policy Manager from an LDAP server which is external to the TOE, and accessing external patch vendor meta-data and patches via the Red Hat Enterprise Patch Source channel**].

#### 5.1.7.2 Trusted Path (FTP_TRP.1)

5.1.7.2.1 FTP_TRP.1.1

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

5.1.7.2.2 FTP_TRP.1.2

The TSF shall permit [remote users] to initiate communication via the trusted path.

5.1.7.2.3 FTP_TRP.1.3

The TSF shall require the use of the trusted path for [[no selection] **the delivery of channels from the Transmitter to client and server endpoints, communications between the Infrastructure Administration channel and Tuners, Transmitters and Proxies, publish data sent to the Transmitter from the Publisher, Channel Copier and Application Packager, data copy operations performed by the Infrastructure Administration channel, administrative operations using CMS and Deployment Manager, logging information sent to the Deployment manager, and commands sent to the Deployment Service**].

## 5.2 Security Functional Requirements for the IT Environment

| Security Functional Class | Security Functional Components |
|---|---|
| Protection of the TSF (FPT) | Confidentially of exported TSF data (FPT_ITC.1) |
| Identification and Authentication (FIA) | User attribute definition (FIA_ATD.1) |
| | User authentication before any action (FIA_UAU.2) |
| | User identification before any action (FIA_UID.2) |
| Protection of the TSF (FPT) | TSF domain separation (FPT_SEP.1) |
| | Reliable time stamp (FPT_STM.1) |

**Table 5: Security Functional Components for the Environment**

### 5.2.1 User Data Protection (FDP)

#### 5.2.1.1 Inter-TSF confidentiality during transmission (FPT_ITC.1)

5.2.1.1.1 FPT_ITC.1.1

The ~~TSF~~ **IT environment** shall protect all TSF data transmitted from the TSF to a remote trusted IT product from unauthorized disclosure **and modification** during transmission.

### 5.2.2 Identification and Authentication (FIA)

#### 5.2.2.1 User attribute definition (FIA_ATD.1)

5.2.2.1.1 FIA_ATD.1.1

The ~~TSF~~ **IT environment** shall maintain the following list of security attributes belonging to individual users:

- [**user identity**
- **group membership**
- **password (authentication data)**].

#### 5.2.2.2 User authentication before any action (FIA_UAU.2)

5.2.2.2.1 FIA_UAU.2.1

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

#### 5.2.2.3 User identification before any action (FIA_UID.2)

5.2.2.3.1 FIA_UID.2.1

The ~~TSF~~ **IT environment** shall require each user to identify itself before allowing any other TSF-mediated actions on behalf of that user.

### 5.2.3 Protection of the TSF (FPT)

#### 5.2.3.1 TSF domain separation (FPT_SEP.1)

5.2.3.1.1 FPT_SEP.1.1

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

5.2.3.1.2 FPT_SEP.1.2

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

#### 5.2.3.2 Reliable time stamps (FPT_STM.1)

5.2.3.2.1 FPT_STM.1.1

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

## 5.3 TOE Security Assurance Requirements

The security assurance requirements for the TOE are the Evaluation Assurance Level (EAL) 3 components as specified in Part 3 of the Common Criteria. No operations are applied to the assurance components

| Assurance Class | Assurance Components |
|---|---|
| Configuration Management (ACM) | Authorization controls (ACM_CAP.3) |
| | TOE CM coverage (ACM_SCP.1) |
| Delivery and Operations (ADO) | Delivery procedures (ADO_DEL.1) |
| | Installation, generation, and start-up procedures (ADO_IGS.1) |
| Development (ADV) | Informal functional specification (ADV_FSP.1) |
| | Security enforcing high-level design (ADV_HLD.2) |
| | Informal correspondence demonstration (ADV_RCR.1) |
| Guidance Documents (AGD) | Administrator guidance (AGD_ADM.1) |
| | User guidance (AGD_USR.1) |
| Life Cycle Support (ALC) | Identification of security measures (ALC_DVS.1) |
| Tests (ATE) | Analysis of coverage (ATE_COV.2) |
| | Testing: high-level design (ATE_DPT.1) |
| | Functional testing (ATE_FUN.1) |
| | Independent testing – sample (ATE_IND.2) |
| Vulnerability Assessment (AVA) | Examination of guidance (AVA_MSU.1) |
| | Strength of TOE security function evaluation (AVA_SOF.1) |
| | Developer vulnerability analysis (AVA_VLA.1) |

**Table 6: EAL3 Assurance Components**

## 5.3.1 Configuration Management (ACM)

### 5.3.1.1 Authorization Controls (ACM_CAP.3)

5.3.1.1.1 ACM_CAP.3.1D

The developer shall provide a reference for the TOE.

5.3.1.1.2 ACM_CAP.3.2D

The developer shall use a CM system.

5.3.1.1.3 ACM_CAP.3.3D

The developer shall provide CM documentation.

5.3.1.1.4 ACM_CAP.3.1C

The reference for the TOE shall be unique to each version of the TOE.

5.3.1.1.5 ACM_CAP.3.2C

The TOE shall be labeled with its reference.

5.3.1.1.6 ACM_CAP.3.3C

The CM documentation shall include a configuration list and a CM plan.

5.3.1.1.7 ACM_CAP.3.4C

The configuration list shall uniquely identify all configuration items that comprise the TOE.

5.3.1.1.8 ACM_CAP.3.5C

The configuration list shall describe the configuration items that comprise the TOE.

5.3.1.1.9 ACM_CAP.3.6C

The CM documentation shall describe the method used to uniquely identify the configuration items.

5.3.1.1.10 ACM_CAP.3.7C

The CM system shall uniquely identify all configuration items.

5.3.1.1.11 ACM_CAP.3.8C

The CM plan shall describe how the CM system is used.

5.3.1.1.12 ACM_CAP.3.9C

The evidence shall demonstrate that the CM system is operating in accordance with the CM plan.

5.3.1.1.13 ACM_CAP.3.10C

The CM documentation shall provide evidence that all configuration items have been and are being effectively maintained under the CM system.

5.3.1.1.14 ACM_CAP.3.11C

The CM system shall provide measures such that only authorized changes are made to the configuration items.

5.3.1.1.15 ACM_CAP.3.1E

The evaluator shall confirm that the information provided meets all the requirements for content and presentation of evidence

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

5.3.1.2.1 ACM_SCP.1.1D

The developer shall provide a list of configuration items for the TOE.

5.3.1.2.2 ACM_SCP.1.1E

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

### 5.3.2 Delivery and Operation (ADO)

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

5.3.2.1.1 ADO_DEL.1.1D

The developer shall document procedures for delivery of the TOE or parts of it to the user.

5.3.2.1.2 ADO_DEL.1.2D

The developer shall use the delivery procedures.

5.3.2.1.3 ADO_DEL.1.1C

The delivery documentation shall describe all procedures that are necessary to maintain security when distributing versions of the TOE to a user's site.

5.3.2.1.4 ADO_DEL.1.1E

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence

#### 5.3.2.2 Installation, generation, and start-up procedures (ADO_IGS.1)

5.3.2.2.1 ADO_IGS.1.1D

The developer shall document procedures necessary for the secure installation, generation, and start-up of the TOE.

5.3.2.2.2 ADO_IGS.1.1C

The installation, generation and start-up documentation shall describe all the steps necessary for secure installation, generation, and start-up of the TOE.

5.3.2.2.3 ADO_IGS.1.1E

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

5.3.2.2.4 ADO_IGS.1.2E

The evaluator shall determine that the installation, generation, and start-up procedures result in a secure configuration.

## 5.3.3 Development (ADV)

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

5.3.3.1.1 ADV_FSP.1.1D

The developer shall provide a functional specification.

5.3.3.1.2 ADV_FSP.1.1C

The functional specification shall describe the TSF and its external interfaces using an informal style.

5.3.3.1.3 ADV_FSP.1.2C

The functional specification shall be internally consistent.

5.3.3.1.4 ADV_FSP.1.3C

The functional specification shall describe the purpose and method of use of all external TSF interfaces, providing details of effects, exceptions and error messages, as appropriate.

5.3.3.1.5 ADV_FSP.1.4C

The functional specification shall completely represent the TSF.

5.3.3.1.6 ADV_FSP.1.1E

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

5.3.3.1.7 ADV_FSP.1.2E

The evaluator shall determine that the functional specification is an accurate and complete instantiation of the TOE security functional requirements.

### 5.3.3.2 Security enforcing high-level design (ADV_HLD.2)

5.3.3.2.1 ADV_HLD.2.1D

The developer shall provide the high-level design of the TSF.

5.3.3.2.2 ADV_HLD.2.1C

The presentation of the high-level design shall be informal.

5.3.3.2.3 ADV_HLD.2.2C

The high-level design shall be internally consistent.

5.3.3.2.4 ADV_HLD.2.3C

The high-level design shall describe the structure of the TSF in terms of subsystems.

5.3.3.2.5 ADV_HLD.2.4C

The high-level design shall describe the security functionality provided by each subsystem of the TSF.

5.3.3.2.6 ADV_HLD.2.5C

The high-level design shall identify any underlying hardware, firmware, and/or software required by the TSF with a presentation of the functions provided by the supporting protection mechanisms implemented in that hardware, firmware, or software.

5.3.3.2.7 ADV_HLD.2.6C

The high-level design shall identify all interfaces to the subsystems of the TSF.

5.3.3.2.8 ADV_HLD.2.7C

The high-level design shall identify which of the interfaces to the subsystems of the TSF are externally visible.

5.3.3.2.9 ADV_HLD.2.8C

The high-level design shall describe the purpose and method of use of all interfaces to the subsystems of the TSF, providing details of effects, exceptions and error messages, as appropriate.

5.3.3.2.10 ADV_HLD.2.9C

The high-level design shall describe the separation of the TOE into TSP-enforcing and other subsystems.

5.3.3.2.11 ADV_HLD.2.1E

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

5.3.3.2.12 ADV_HLD.2.2E

The evaluator shall determine that the high-level design is an accurate and complete instantiation of the TOE security functional requirements.

#### 5.3.3.3 Informal correspondence demonstration (ADV_RCR.1)

5.3.3.3.1 ADV_RCR.1.1D

The developer shall provide an analysis of correspondence between all adjacent pairs of TSF representations that are provided.

5.3.3.3.2 ADV_RCR.1.1C

For each adjacent pair of provided TSF representations, the analysis shall demonstrate that all relevant security functionality of the more abstract TSF representation is correctly and completely refined in the less abstract TSF representation.

5.3.3.3.3 ADV_RCR.1.1E

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

### 5.3.4 Guidance Documents (AGD)

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

5.3.4.1.1 AGD_ADM.1.1D

The developer shall provide administrator guidance addressed to system administrative personnel.

5.3.4.1.2 AGD_ADM.1.1C

The administrator guidance shall describe the administrative functions and interfaces available to the administrator of the TOE.

5.3.4.1.3 AGD_ADM.1.2C

The administrator guidance shall describe how to administer the TOE in a secure manner.

5.3.4.1.4 AGD_ADM.1.3C

The administrator guidance shall contain warnings about functions and privileges that should be controlled in a secure processing environment.

5.3.4.1.5 AGD_ADM.1.4C

The administrator guidance shall describe all assumptions regarding user behavior that are relevant to secure operation of the TOE.

5.3.4.1.6 AGD_ADM.1.5C

The administrator guidance shall describe all security parameters under the control of the administrator, indicating secure values as appropriate.

5.3.4.1.7 AGD_ADM.1.6C

The administrator guidance shall describe each type of security-relevant event relative to the administrative functions that need to be performed, including changing the security characteristics of entities under the control of the TSF.

5.3.4.1.8 AGD_ADM.1.7C

The administrator guidance shall be consistent with all other documentation supplied for evaluation.

5.3.4.1.9 AGD_ADM.1.8C

The administrator guidance shall describe all security requirements for the IT environment that are relevant to the administrator.

5.3.4.1.10 AGD_ADM.1.1E

The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence

### 5.3.4.2 User Guidance (AGD_USR.1)

5.3.4.2.1 AGD_USR.1.1D

The developer shall provide user guidance.

5.3.4.2.2 AGD_USR.1.1C

The user guidance shall describe the functions and interfaces available to the non-administrative users of the TOE.

5.3.4.2.3 AGD_USR.1.2C

The user guidance shall describe the use of user-accessible security functions provided by the TOE.

5.3.4.2.4 AGD_USR.1.3C

The user guidance shall contain warnings about user-accessible functions and privileges that should be controlled in a secure processing environment.

5.3.4.2.5 AGD_USR.1.4C

The user guidance shall clearly present all user responsibilities necessary for secure operation of the TOE, including those related to assumptions regarding user behavior found in the statement of TOE security environment.

5.3.4.2.6 AGD_USR.1.5C

The user guidance shall be consistent with all other documentation supplied for evaluation.

5.3.4.2.7 AGD_USR.1.6C

The user guidance shall describe all security requirements for the IT environment that are relevant to the user.

5.3.4.2.8 AGD_USR.1.1E

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

## 5.3.5 Life Cycle Support (ALC)

### 5.3.5.1 Identification of security measures (ALC_DVS.1)

5.3.5.1.1 ALC_DVS.1.1D

The developer shall produce development security documentation.

5.3.5.1.2 ALC_DVS.1.1C

The development security documentation shall describe all the physical, procedural, personnel, and other security measures that are necessary to protect the confidentiality and integrity of the TOE design and implementation in its development environment.

5.3.5.1.3 ALC_DVS.1.2C

The development security documentation shall provide evidence that these security measures are followed during the development and maintenance of the TOE.

5.3.5.1.4 ALC_DVS.1.1E

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

5.3.5.1.5 ALC_DVS.1.2E

The evaluator shall confirm that the security measures are being applied.

## 5.3.6 Security Testing (ATE)

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

5.3.6.1.1 ATE_COV.2.1D

The developer shall provide an analysis of the test coverage.

5.3.6.1.2 ATE_COV.2.1C

The analysis of the test coverage shall demonstrate the correspondence between the tests identified in the test documentation and the TSF as described in the functional specification.

5.3.6.1.3 ATE_COV.2.2C

The analysis of the test coverage shall demonstrate that the correspondence between the TSF as described in the functional specification and the tests identified in the test documentation is complete.

5.3.6.1.4 ATE_COV.2.1E

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

### 5.3.6.2 Testing: high-level design (ATE_DPT.1)

5.3.6.2.1 ATE_DPT.1.1D

The developer shall provide the analysis of the depth of testing.

5.3.6.2.2 ATE_DPT.1.1C

The depth analysis shall demonstrate that the tests identified in the test documentation are sufficient to demonstrate that the TSF operates in accordance with its high-level design.

5.3.6.2.3 ATE_DPT.1.1E

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

### 5.3.6.3 Functional testing (ATE_FUN.1)

5.3.6.3.1 ATE_FUN.1.1D

The developer shall test the TSF and document the results.

5.3.6.3.2 ATE_FUN.1.2D

The developer shall provide test documentation.

5.3.6.3.3 ATE_FUN.1.1C

The test documentation shall consist of test plans, test procedure descriptions, expected test results and actual test results.

5.3.6.3.4 ATE_FUN.1.2C

The test plans shall identify the security functions to be tested and describe the goal of the tests to be performed.

5.3.6.3.5 ATE_FUN.1.3C

The test procedure descriptions shall identify the tests to be performed and describe the scenarios for testing each security function. These scenarios shall include any ordering dependencies on the results of other tests.

5.3.6.3.6 ATE_FUN.1.4C

The expected test results shall show the anticipated outputs from a successful execution of the tests.

5.3.6.3.7 ATE_FUN.1.5C

The test results from the developer execution of the tests shall demonstrate that each tested security function behaved as specified.

5.3.6.3.8 ATE_FUN.1.1E

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

#### 5.3.6.4 Independent testing – sample (ATE_IND.2)

5.3.6.4.1 ATE_IND.2.1D

The developer shall provide the TOE for testing.

5.3.6.4.2 ATE_IND.2.1C

The TOE shall be suitable for testing.

5.3.6.4.3 ATE_IND.2.2C

The developer shall provide an equivalent set of resources to those that were used in the developer's functional testing of the TSF.

5.3.6.4.4 ATE_IND.2.1E

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

5.3.6.4.5 ATE_IND.2.2E

The evaluator shall test a subset of the TSF as appropriate to confirm that the TOE operates as specified.

5.3.6.4.6 ATE_IND.2.3E

The evaluator shall execute a sample of tests in the test documentation to verify the developer test results.

### 5.3.7 Vulnerability Assessment (VLA)

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

5.3.7.1.1 AVA_MSU.1.1D

The developer shall provide guidance documentation.

5.3.7.1.2 AVA_MSU.1.1C

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

5.3.7.1.3 AVA_MSU.1.2C

The guidance documentation shall be complete, clear, consistent and reasonable.

5.3.7.1.4 AVA_MSU.1.3C

The guidance documentation shall list all assumptions about the intended environment.

5.3.7.1.5 AVA_MSU.1.4C

The guidance documentation shall list all requirements for external security measures (including external procedural, physical and personnel controls).

5.3.7.1.6 AVA_MSU.1.1E

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

5.3.7.1.7 AVA_MSU.1.2E

The evaluator shall repeat all configuration and installation procedures to confirm that the TOE can be configured and used securely using only the supplied guidance documentation.

5.3.7.1.8 AVA_MSU.1.3E

The evaluator shall determine that the use of the guidance documentation allows all insecure states to be detected.

### 5.3.7.2 Strength of TOE security function evaluation (AVA_SOF.1)

5.3.7.2.1 AVA_SOF.1.1D

The developer shall perform a strength of TOE security function analysis for each mechanism identified in the ST as having a strength of TOE security function claim.

5.3.7.2.2 AVA_SOF.1.1C

For each mechanism with strength of TOE security function claim the strength of TOE security function analysis shall show that it meets or exceeds the minimum strength level defined in the PP/ST.

5.3.7.2.3 AVA_SOF.1.2C

For each mechanism with a specific strength of TOE security function claim the strength of TOE security function analysis shall show that it meets or exceeds the specific strength of function metric defined in the PP/ST.

5.3.7.2.4 AVA_SOF.1.1E

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

5.3.7.2.5 AVA_SOF.1.2E

The evaluator shall confirm that the strength claims are correct.

### 5.3.7.3 Developer vulnerability analysis (AVA_VLA.1)

5.3.7.3.1 AVA_VLA.1.1D

The developer shall perform a vulnerability analysis.

5.3.7.3.2 AVA_VLA.1.2D

The developer shall provide vulnerability analysis documentation.

5.3.7.3.3 AVA_VLA.1.1C

The vulnerability analysis documentation shall describe the analysis of the TOE deliverables performed to search for obvious ways in which a user can violate the TSP.

5.3.7.3.4 AVA_VLA.1.2C

The vulnerability analysis documentation shall describe the disposition of obvious vulnerabilities.

5.3.7.3.5 AVA_VLA.1.3C

The vulnerability analysis documentation shall show, for all identified vulnerabilities, that the vulnerability cannot be exploited in the intended environment for the TOE.

5.3.7.3.6 AVA_VLA.1.1E

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

5.3.7.3.7 AVA_VLA.1.2E

The evaluator shall conduct penetration testing, building on the developer vulnerability analysis, to ensure obvious vulnerabilities have been addressed.

# 6. TOE Summary Specification

This chapter describes the security functions and associated assurance measures.

## 6.1 TOE Security Functions

Each of the security function descriptions is organized by the security requirements corresponding to the security function. Hence, each function is described by describing how it specifically satisfies each of its related requirements. This serves to both describe the security functions and rationalize that the security functions are suitable to satisfy the necessary requirements.

The TOE consists of a family of products named Marimba Client and Server Management from BMC Software. This family of products consists of the following software packages:

- Marimba Control Center by BMC Software
- Marimba Patch Management by BMC Software
- Marimba Content Management by BMC Software
- Marimba Desktop/Mobile Application Management by BMC Software
- Marimba Server Application Management by BMC Software
- Marimba Desktop OS Management by BMC Software
- Marimba Server OS Management for Unix and Linux by BMC Software

Each software package contains one or more components called ‘channels’, and a particular channel can be present in one or more packages. A mapping of software packages to channels is included in section 2.4.1, Physical Boundaries. TOE security functions are described in terms of channels.

### 6.1.1 Cryptographic Support

Marimba Client and Server Management from BMC Software provides the option to use encryption and decryption technology to enforce the information flow policy between the Transmitter, client and server endpoints, the administrator console, LDAP directory servers (external to the TOE) and the Red Hat Satellite Server (external to the TOE).

All communication, including control information and the transfer of software applications and patches between client and server endpoint computers and the Transmitter is encrypted, and subsequently decrypted to ensure

privacy. The encryption and decryption uses the Secure Socket Layer (SSL) protocol over HTTP, commonly referred to as HTTPS.

All communication between an administrator's web browser (such as Microsoft's Internet Explorer) and CMS and the Deployment Manager is encrypted, and subsequently decrypted to ensure privacy. The encryption and decryption uses the Secure Socket Layer (SSL) protocol over HTTP, commonly referred to as HTTPS.

All communication between the Publisher, Channel Copier and Application packager is encrypted, and subsequently decrypted to ensure the privacy of publishing data, such as when a software package is published to the Transmitter prior to distribution to endpoint computers. The encryption and decryption uses the Secure Sockets Layer (SSL) protocol.

All communication between the Infrastructure Administration channel's Transmitter, Tuner and Proxy Administrator interface and the Transmitters and Tuners for which the Infrastructure Administration channel is administering and copying data between is encrypted, and subsequently decrypted to ensure privacy. The encryption and decryption uses the Secure Sockets Layer (SSL) protocol.

Logging information sent by the Deployment Service component to the Deployment Manager is encrypted. This data is used by the Deployment Manager to report on the installation status of software applications deployed to server endpoints. Additionally, commands sent from the Deployment Manager to the Deployment Service are encrypted, and subsequently decrypted. The encryption and decryption uses the Secure Sockets Layer (SSL) protocol.

Communication between the Transmitter and CMS, and the external LDAP directory, used to source user ID, groups, passwords and other application specific data is encrypted, and subsequently decrypted to ensure privacy. The encryption and decryption uses the Secure Socket Layer (SSL) protocol over LDAP, commonly referred to as LDAPS.

The Red Hat Enterprise Linux Patch Source channel, used to source Linux operating system patches, and associated meta-data uses an encrypted channel to ensure privacy. The encryption and decryption uses the Secure Socket Layer (SSL) protocol.

All encrypted communication uses the RC4 encryption algorithm with a 128 bit key size, and the RSA algorithm during key exchange. The RSA key-exchange algorithm uses a 1024 bit key size by default, and also supports a 512 bit key. The RC4 and RSA algorithms, and the SSL protocol, are implemented using the BSAFE SSL-C 2.5.1 library technology from RSA Security, Inc. The use of client side (endpoint) certificates is not part of the evaluated configuration.

Each connection between these components is distinct from other connections. The encryption ensures that communication data is not modified or deleted by unauthorized users.

The Cryptographic Support security function satisfies the following security requirement:

- FCS_COP.1(a) (encryption/decryption)
- FCS_COP.1(b) (key exchange)

### 6.1.2 Security Audit

All events, as identified in **Table 4: Auditable Events**, are logged by the Transmitter and Patch Manager. Auditable security events include when the Transmitter is stopped and started. The audit records are stored both local to the Transmitter and Patch Manager, on the file system of the computer running these components, and also in a central log repository; the external Relational Database Management System (RDBMS). The Logging Service channel acts as a collection agent, collecting the audit records and sending them to the repository. The centralized location allows the Primary Administrator, Administrator, or Operator access to review them using the Report Center interface, which is part of the Marimba Control Center package. This audit data is presented in such a manner that the Primary Administrator, Administrator, or Operator can read and interpret the content of the information; hence the information is presented in a manner suitable for human interpretation. Each audit record records the user that performed the action, as well as the action (event), the date/time the action was performed, as well as the outcome of

either success or failure, the channel (object identity), and severity level. Success and failure is not explicitly stated by a message such as “Success” or “Fail”, but is deterministically implied by the event type description.

The Logging Service channel is configured using the Report Center interface. Report Center includes a log-filtering feature in which the Primary Administrator, Administrator, or Operator can specify which audit log messages they want collected by the centralized logging component, and what the minimum severity level should be (for example, MAJOR or CRITICAL messages only). Once the Primary Administrator, Administrator, or Operator has set specific logging filters based on severity and event type ranges, only messages with specified ID codes and severity levels are sent to the central log repository. Report Center is used to review the audit log entries present in the log repository.

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

- FAU_GEN.1
- FAU_GEN.2
- FAU_SAR.1
- FAU_SEL.1

### 6.1.3 User Data Protection

The TSF mediates access between subjects and the named objects. Subjects consist of users and processes acting on behalf of users. The following table lists the named objects under the control of the access control policy for the TOE.

| Named Object | Description |
|---|---|
| Deployment Manager Folder | Serves as an organizer of the Deployment, Task Group, Server Group and Server Keychain objects |
| Deployment | A set of jobs that distribute and manage content and applications on sets of servers. Each deployment job maps one task to at least one server group. |
| Task group | An object that contains the instructions for what you want to accomplish on your endpoints. For example, you could create a task group called Web Content, which contains instructions for managing the content (data files) on your website. Within this task group, you might create three related tasks:<br>•A task called Stage, which would download but not activate the content<br>•A task called Install, which would activate the content only if the Stage task was successful on all the servers<br>•A task called Rollback, which would revert the content to a previous version if necessary<br>Within each task, you would then create the specific commands to carry out that task. |
| Server group | A set of target servers on which you want to execute commands. |
| Server keychain | A server keychain specifies the user names and passwords (also called |

| Named Object | Description |
|---|---|
| | server credentials) that you want to use for accessing the servers. |
| Channel | An application or content that is published to a Transmitter. A channel can be:<br>•An application of any type (Windows, Java, Visual Basic, and so on) or a Java applet<br>•One or more content files, containing HTML or any data<br>•A Marimba software package component |
| Transmitter Folder | Serves as an organizer of the Channel objects |
| Policy Manager Target | An object used by the policy manager interface to assign a group of applications to. Typically the policy manager target object will correspond to a user or machine group in a directory system. A typical use-case would be "assign sales force automation applications to the sales group in active directory". |
| Patch Group | Security and functional patches, and associated patch information, packaged into a Marimba Channel. |

**Table 7: Named Objects**

The access control policy is the mechanism by which access to the named objects is controlled based solely on the identity of the user and/or group membership, create permissions associated with that user or group and the security attributes associated with the named object.

In the Deployment Manager module, the implementation of the access policy is accomplished by the association of permission bits, which are specific to the named object. Permissions restrict the access that a user (subject) has to a particular object (server group, server keychain, task group, deployment, or folder), and consequently restrict the operations (such as, read (r), write (w), execute (x), delete (d) and the ability to change permissions by virtue of being designated an object's owner (o)). Additionally, global create permissions, known as *privileges* can be associated with users and groups of users to explicitly grant or revoke create permissions for named objects.

In the Transmitter module, the implementation of the access policy is implemented using an ACL, which contains a single attribute indicating the user or user group that can access a channel.

In the Policy Manager module, the implementation of the access control policy is implemented using an ACL associated with a user or user group which contains a list of policy manager targets, and associated permission bits for the user or user group. The permission bits represent the following access control permissions: ACL read, ACL write, policy read, policy write.

In the Patch Manager module, the implementation of the access control policy is implemented using an ACL associated with a Patch Group which contains a list of users or user groups that have been granted permissions to the Patch Groups, and for each user or user group, three attributes representing the following Patch Group permissions: modify, publish, delete.

Users, or groups of users can gain access to the objects as long as they have the appropriate permissions. Access checks are performed on every reference to the named object.

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

- FDP_ACC.1(a)
- FDP_ACC.1(b)

- FDP_ACC.1(c)
- FDP_ACC.1(d)
- FDP_ACF.1(a)
- FDP_ACF.1(b)
- FDP_ACF.1(c)
- FDP_ACF.1(d)

### 6.1.4 Identification and Authentication

Users accessing Client and Server Management using CMS are identified and authenticated via the LDAP server in the environment and the identity is used by the TOE. However, when users access the TOE via the Deployment Manager component of Server Management, the TOE is performing identification and authentication. Before any security management functions can be performed on Deployment Manager, users must be successfully identified and authenticated. The identification and authentication is performed via the GUI Deployment Manager tool and via the Deployment Manager Command Line interface. The users are presented with a dialog box and once the logon dialog is displayed, the user enters their user ID and password. With the Deployment Manager Command Line interface, users submit their user IDs and passwords as command line parameters to be validated by the Deployment Manger. All users, regardless of where identification and authentication takes place, must be successfully identified and authenticated before allowing any other TSF-mediated actions on behalf of the user.

The user attributes for Deployment Manager are stored in the local user database, which is stored on the machine where Deployment Manager is installed. The following attributes are maintained for each user entry stored in the TOE:

- User ID – user name to uniquely identify a user.
- Password – the password must be at least 9 characters long.
- Home (default) folder – when the user account is created, a home or default folder must be specified.
- Group membership - all users are associated with groups to assist in defining roles as well as access permissions. By adding a user to a group, the user will have all the permissions assigned to that group.
- Create permissions, or privileges – when a user account or a group is created, you specify whether that user or group can create each type of named object (server group, server keychain, task group, deployment, or deployment manager folder).

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

- FIA_ATD.1
- FIA_UAU.1
- FIA_UID.2

### 6.1.5 Security Management

The TSF provides the ability to manage the security functions of the TOE. The functions include the following:

- Management of access control to named objects
- Installation of encryption certificates
- Configure idle user timeout

All management functions can be performed via either the GUI, command line interfaces, or both, depending on the specific operation. Management of access control to named objects can be performed by the GUI, but not all named object access control functions are available via the command line. Both installation of encryption certificates and

configuring idle user timeouts can only be performed using the GUI. Either interface requires successful identification and authentication of the authorized administrator before any security management functions can be performed.

All security functions are controlled through the assignment of roles (except Certificate Manager), which are determined through user and group membership. By default, for Deployment Manager, the owner of a named object has read, write, execute permission on the new objects and all other users have read permission. Folder owners can set default permissions that will be applied to all objects created within the folder. To grant access, the Deployment Manager Administrator associates users or user groups to the named objects. Only those users and/or user groups requiring access to named objects are granted access. By default, permissions for Deployment Manager Folders, deployments, task groups, server groups, and server keychains named objects are restrictive.

For Channels, and Transmitter Folders the Primary Administrator and Administrator, associate users or user groups to the named objects. Only those users and/or user groups requiring access to named objects are granted access. By default, permissions for channels and Transmitter folders are permissive.

For Policy Manager Targets the Primary Administrator and users with ACL write permissions associate users or user groups to the named objects. Only those users and/or user groups requiring access to named objects are granted access. By default, permissions for policy manager targets are restrictive.

For Patch Management Patch Groups the Primary Administrator, Administrator, and object owners can associate users or user groups to the named objects. Only those users and/or user groups requiring access to named objects are granted access. By default, permissions for patch groups are restrictive.

The following roles are available, together with an indication of which product they apply to:

- Primary Administrator (this role is available in all interfaces except Deployment Manager) - Primary Administrators have access to all product features available in the components (except Deployment Manager). Some of the security functions available to the Primary Administrator are the ability to assign the roles “Primary Administrator”, “Administrator” and “Operator” to other users, the ability to configure the authentication source (LDAP server external to the TOE) used by the Common Management Services (CMS) component of the TOE, the ability to modify access control settings on the Transmitter, the ability to configure all configuration settings in the Patch Manager and the ability to target and report on all target objects in Policy Manager and Report Center. Additionally, Primary Administrators can assign ACL write permissions to other users in Policy Manager.
- Administrator (this role is available in all interfaces except Deployment Manager) - Administrators can log in to the TOE and have access to most product features available, except those reserved for Primary Administrators. In Report Center, for example, the configuration page (used to configure various Report Center functions) is not available to Administrators, only Primary Administrators. In addition, Administrators cannot modify the system settings, so they cannot perform tasks such as giving login access to new users or configuring directory server settings.
- Operator (this role is available in all interfaces except Deployment Manager) - Operators can log in to the applications and perform certain tasks, but they cannot make changes or save any changes in the applications. For the most part, they have read-only access to the applications. The operator role has no access to the Patch Manager and Policy Manager components.
- Deployment Manager Administrator (this role is only available in the Deployment Manager interface) - Can modify the Deployment Manager’s system settings, and can grant new users and groups, access to the Deployment Manager, including specifying a default folder for them. In addition, the Deployment Manager can modify users’ security attributes. The Deployment Manager Administrator has the ability to modify, change default settings, and delete the security control attributes associated with Deployment Manager named objects.
- Regular Users (this role is only available in the Deployment Manager interface) – Non-administrative users can perform operations only on objects that they have sufficient permissions to do so. Each object in the Deployment Manager can have the following combination of permissions: read, write, execute and owner Non-administrative users can also change their own passwords.

Security functions that utilize encryption require the installation of a valid encryption certificate in a format, and containing values appropriate for the encryption algorithm. Additionally, the administrator can enable and disable encryption functions for the Transmitter, CMS, and Deployment Manager.

The Security Management functions are designed to satisfy the following security functional requirements:

- FMT_MSA.1(a)
- FMT_MSA.1(b)
- FMT_MSA.1(c)
- FMT_MSA.1(d)
- FMT_MSA.2
- FMT_MSA.3(a)
- FMT_MSA.3(b)
- FMT_MSA.3(c)
- FMT_MSA.3(d)
- FMT_MTD.1(a)
- FMT_MTD.1(b)
- FMT_MTD.1(c)
- FMT_SMF.1
- FMT_SMR.1

### 6.1.6 TOE Access

The TSF provides the ability for the Deployment Manager Administrator in the Deployment Manager component of server management, and the Primary Administrator for all other Client and Server Management components to specify a user session timeout value. The TSF monitors an established session for activity and if the session is inactive for the specified time period, the TSF will terminate the session. On the Deployment Manager, the default time is 15 minutes, on all other Client and Desktop Management components, the default time is 60 minutes. To reestablish a session, the user must re-enter their user ID and password and be successfully identified and authenticated.

The TOE Access function is designed to satisfy the following security functional requirement:

- FTS_SSL.3

### 6.1.7 Trusted Path/Channel

All successful client and server requests to the Transmitter, and all requests from web browsers to CMS and Deployment Manager initiate a trusted path. In each case, the client authenticates the server, and then encryption and decryption is used for all communications between these components. Each connection between these components is distinct from other connections. The trusted path ensures that communication data is not modified or deleted by unauthorized users, and also ensures that the Transmitter, CMS and Deployment Manager are valid hosts and have not been impersonated.

All communication between the Publisher, Channel Copier and Application packager is performed over a trusted path that uses encryption and decryption between these components. Each connection between these components is distinct from other connections. The trusted path ensures that communication data is not modified or deleted by unauthorized users.

All communication between the Infrastructure Administration channel's Transmitter, Tuner and Proxy Administrator interface and the Transmitters and Tuners for which the Infrastructure Administration channel is administering and copying data between is performed over a trusted path that uses encryption and decryption. Each connection between these components is distinct from other connections. The trusted path ensures that communication data is not modified or deleted by unauthorized users.

Logging information sent by the Deployment Service component to the Deployment Manager uses a trusted path for communication. This data is used by the Deployment Manager to report on the installation status of software applications deployed to server endpoints. Additionally, commands sent from the Deployment Manager to the Deployment Service also use a trusted path. These trusted paths use encryption and decryption to ensure that communication data is not modified or deleted by unauthorized users.

All connections to the external LDAP server, which is part of the IT environment, from both the Transmitter and CMS use a trusted channel. The trusted channels use encryption and decryption between these components. Each connection is distinct from other connections. The trusted channels ensure that communication data is not modified or deleted by unauthorized users, and also ensure that the LDAP server is a valid host and has not been impersonated.

Communication between the Red Hat Enterprise Patch Source channel and the external Red Hat Satellite Server providing patch meta-data and patches uses a trusted channel. The trusted channel uses encryption and decryption to ensure that no users can read the contents of these communication channels other than authorized users of the TOE

The Trusted Path/Channel security functions satisfies the following security requirements:

- FTP_TRP.1
- FTP_ITC.1

## 6.2 TOE Security Assurance Measures

The following assurance measures are applied to satisfy the Common Criteria EAL3 assurance requirements.

- Process Assurance;
- Delivery and Guidance;
- Design Documentation;
- Tests; and
- Vulnerability Assessment.

### 6.2.1 Process Assurance

#### 6.2.1.1 Configuration Management

The CM documentation describes the processes and procedure that are followed and automated tools that are utilized in the tracking and monitoring the changes to the CM items and the generation of the TOE. The configuration management measures applied by BMC Software ensure that configuration items are uniquely identified. BMC Software ensures changes to the implementation representation are controlled and that TOE associated configuration item modifications are properly controlled. BMC Software performs configuration management on the TOE implementation representation, design, tests, vulnerability analysis, delivery, installation, user and administrator guidance, lifecycle, and the CM documentation. These activities are documented in:

- Configuration Management Guide, Marimba Client and Server Management from BMC Software, Version 6.0.3 (070530)

The Configuration Management assurance measure satisfies the following Assurance requirements:

- ACM_CAP.3

- ACM_SCP.1

#### 6.2.1.2 Life Cycle Support

BMC Software ensures the adequacy of the procedures used during the development and maintenance of the TOE. BMC Software includes security controls on the development environment that are adequate to provide the confidentiality and integrity of the TOE design and implementation that is necessary to ensure the secure operation of the TOE. BMC Software achieves this through the use of documented procedures. These procedures are documented in:

- Configuration Management Guide, Marimba Client and Server Management from BMC Software, Version 6.0.3 (070530)

The Process Assurance measures satisfy the following assurance requirements:

- ALC_DVS.1

### 6.2.2 Delivery and Operation

BMC Software provides documentation that explains how the TOE is delivered, the carriers utilized, and the procedures that are followed to maintain security when distributed to the user's site. BMC Software's installation procedures describe the steps used for the secure installation, generation, and start-up of the TOE along with configuration settings to secure the TOE privileges and functions. These procedures are documented in:

- Sales Orders and Delivery Guide, Marimba Client and Server Management from BMC Software, Release 6.0.3 (060117)
- Marimba Documentation Addendum [1] for the BMC Marimba Product Line NIAP Certification version 6.0.3 (070302)

The Delivery and Guidance assurance measure satisfies the following Assurance requirements:

- ADO_DEL.1
- ADO_IGS.1

### 6.2.3 Design Documentation

BMC Software provides design documentation that identifies and describes the external interfaces and the decomposition of the TOE into subsystems. The design documentation consists of the following documents:

- ADV_FSP.1: The Functional Specification, Marimba Client and Server Management from BMC Software, Version 6.0.3 (060613) describes all the external interfaces of the TSF. The description includes the purpose and method of use of the interface, applicable parameters, effects, error messages, and exceptions as appropriate.
- ADV_HLD.2: The High Level Design, Marimba Client and Server Management from BMC Software, Release 6.0.3 (060331) decomposes the TOE into TSP-enforcing and other subsystems. Each subsystem will describe the purpose and method of use of all interfaces to the subsystems of the TSF. The description includes the purpose and method of use of the interface, applicable parameters, effects, error messages, and exceptions as appropriate.
- ADV_RCR.1: The way that this correspondence is evident within the design documentation is:
    - ST-TSS to FSP: The Marimba Correspondence Matrix identifies the interfaces that provide the security functions as described in the ST.

[1] The Addendum specifies the Common Criteria specific evaluation settings.

- FSP to HLD: The Marimba Correspondence Matrix identifies the interfaces of the subsystems that provide the security functions as described in the FSP.

The Design assurance measure satisfies the following Assurance requirements

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

### 6.2.4 Guidance Documentation

BMC Software provides administrator guidance on how to utilize the TOE security functions, the interfaces available to the administrator, and warnings to authorized administrators about actions that can compromise the security of the TOE. The procedures, included in the administrator guidance, describe the steps necessary to operate Client and Server Management in accordance with the evaluated configuration, detailing how to establish and maintain the secure configuration, and assumptions about the environment.

The user guidance describes the procedures to use the TOE security-related functions and the interfaces that are available to the non-administrative users.

The administrator and user guidance is documented in:

- *Release Notes - Version 6.0.3SP2 (051031)*
- *Release Notes - Controlled Availability Version 6.5 (051031)*
- *Content Replicator Release Notes Version 6.5 (051031)*
- *Application Packager Administrator's Guide - version 6.5 (050513)*
- *Patch Releases (Application Packager) Last Update: 10/28/05*
- *Release Notes Release date: 05/13/05 Last Updated: 05/13/05**

  *(*NOTE: Application Packager release notes)*
- *Certificate Manager Help - version 4.6.1 (020628)*
- *Channel Copier Help - version 4.6.1 (030610) **

  *(*NOTE: version 4.6.1 of this document applies to version 4.6.2 of the product)*
- *Deployment Guide - version 6.0.3, Patch Management 2.0.1 (051031)*
- *Infrastructure Administrator's Guide - version 6.0.3 (050331)*
- *Patch and Maintenance Releases (Infrastructure & Console 6.0.3.x) Last Update: 09/15/05*
- *Introduction to Marimba Products - September 2004 (040916)*
- *Marimba Reference - version May 2005 (050513)*
- *Marimba Documentation Addendum for the BMC Marimba Product Line NIAP Certification version 6.0.3 (060213)*
- *Release Notes Release date: 03/31/05 Last update: 03/31/05**

  *(*NOTE: OS Management release notes)*
- *Patch Management Administrator's Guide – version 6.5 (051031)*
- *Patch Management Deployment and Upgrade Guide – version 6.5 (051031)*
- *Release Notes Patch Management Version 6.5 (051031)*
- *Planning Guide - version 6.0 (031219)*

- *Policy Management Administrator's Guide - version 6.0.3 (050331)*
- *Patch and Maintenance Releases (Policy Management 6.0.3) Last Update: 06/28/05*
- *Publisher Help - version 4.6.1 (020628**

  *(*NOTE: version 4.6.1 of this document applies to version 4.6.2 of the product))*
- *Report Center Administrator's Guide - version 6.0.3 (050331)*
- *Patch and Maintenance Releases (Report Center, Inventory, Schema Management, and Software Usage 6.0.3) Last Update: 08/19/05*
- *Server Management Administrator's Guide - version 6.0.3 (050331)*
- *Server Management Advanced Topics Guide-version 6.0.3 (041207)*
- *Server Management Guide to the Command-Line Interface - version 6.0.3 (041207)*
- *Patch and Maintenance Releases (Server Management 6.0.3) Last Update: 08/03/05*
- *System Requirements for Marimba Products Version 6.0.3 Patch Management 2.0.1 (050414)*

The Guidance assurance measure satisfies the following Assurance requirements

- AGD_ADM.1
- AGD_USR.1

### 6.2.5 Test Documentation

BMC Software provides test documentation that describes how each of the TOE security functions is tested, as well as the actual results of applying the tests. The test documentation consists of the following document:

- Marimba Client and Server Management from BMC Software Test Plan and Test Cases, version 0.8 (May 17, 2007)

The Tests assurance measure satisfies the following assurance requirements:

- ATE_COV.2: The Test Cases descriptions describe the test cases for each of the security-relevant interfaces of the TOE. The descriptions indicate which tests are used to satisfy the test cases identified for each interface.
- ATE_DPT.1: The Test Cases descriptions include more detailed test case descriptions that demonstrate that the test are sufficient to demonstrate that the TSF operates in accordance with the high-level design and that all of the corresponding interfaces are appropriately exercised
- ATE_FUN.1: The Test Plan describes the security functions to be tested, how to successfully test all of them, the expected results, and the actual test results after exercising all of the tests.
- ATE_IND.2: The TOE and test documentation will be available for independent testing.

### 6.2.6 Vulnerability Assessment

#### 6.2.6.1 Evaluation of Misuse

The Evaluation Team's misuse analysis will demonstrate that the administrative and user guidance completely addresses managing the TOE in a secure configuration.

#### 6.2.6.2 Strength of TOE Security Functions and Vulnerability Analysis

BMC Software performs a SOF analysis of the authentication mechanism. The strength of TOE security function analysis demonstrates that the SOF claims made in the ST for all probabilistic or permutation mechanisms are correct. The TOE security function analysis is provided in the Section 8.7 of this ST.

BMC Software's vulnerability assessment provides the status of each identified vulnerability and demonstrates that each one cannot be exploited in the intended environment and that Client and Server Management is resistant to obvious penetration attacks.

The vulnerability analysis is documented in:

- Marimba Client and Server Management from BMC Software Vulnerability Analysis, Release 6.0.3, 5/30/07

The Vulnerability Assessment assurance measure satisfies the following assurance requirements:

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

# 7. Protection Profile Claims

There are no Protection Profile conformance claims for the TOE.

# 8. Rationale

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

- Security Objectives;
- Security Functional Requirements;
- Security Assurance Requirements;
- Explicitly Stated Requirements;
- Security Requirements Dependencies Rationale;
- TOE Summary Specification;
- Internal Consistency and Support Rationale; and
- Strength of Function (SOF) Rationale.

## 8.1 Security Objectives Rationale

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

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

### 8.1.1 Security Objective for the TOE Rationale

**Table 8: Mapping of TOE Security Objectives to Threats or Organizational Security Policies** provides a mapping of TOE security objectives to those threats that the security objectives that the TOE is designed to counter and organizational security policies that the TOE must enforce.

| TOE Security Objectives | Threats and Organizational Policies |
|---|---|
| O.AUTHORIZATION | T.ACCESS<br>T.AUDIT_CORRUPT |
| O.CLIENT_TO_SERVER_<br>CONFIDENTIALITY | T.DISCLOSURE_OF_<br>COMMUNICATION |
| O.SERVER_TO_LDAP_<br>CONFIDENTIALITY | T.DISCLOSURE_OF_<br>COMMUNICATION |
| O.PATCH_VENDOR_<br>CONFIDENTIALITY | T.DISCLOSURE_OF_<br>COMMUNICATION |
| O.ENCRYPTION | T.DISCLOSURE_OF_<br>COMMUNICATION<br>T.DISCLOSURE_OF_<br>PRIVATE_KEYS |
| O.OBJ_ACCESS | T.PRIVILEGE |
| O.AUDIT | P.ACCOUNTABILITY<br>T.AUDIT_CORRUPT |
| O.MANAGE | P.MANAGE<br>T.AUDIT_CORRUPT<br>T. DISCLOSURE_OF_CERTIFICATES |
| O.VERIFY_CERTIFICATES | T.DISCLOSURE_OF_<br>COMMUNICATION |

**Table 8: Mapping of TOE Security Objectives to Threats or Organizational Security Policies**

The following objectives will address the threats and organizational policies listed in the ST.

**O.AUTHORIZATION** - This objective counters the threats T.ACCESS and T.AUDIT_CORRUPT by requiring each user be identified and authenticated before any access to the TOE and its protected resources is granted.

**O.CLIENT_TO_SERVER_CONFIDENTIALITY** – This objective counters the threat T.DISCLOSURE_OF_COMMUNICATION by ensuring that all intra-TSF communication uses encryption, and hence cannot be read by unauthorized users.

**O.SERVER_TO_LDAP_CONFIDENTIALITY** – This objective counters the threat T.DISCLOSURE_OF_COMMUNICATION by ensuring that communication between the TSF and an external LDAP server is encrypted, and hence cannot be read by unauthorized users.

**O.PATCH_VENDOR_CONFIDENTIALITY** – This objective counters the threat T.DISCLOSURE_OF_COMMUNICATION by ensuring that communication between the TSF and external patch vendors is encrypted, and hence cannot be read by unauthorized users.

**O.ENCRYPTION** – This objective counters the threat T.DISCLOSURE_OF_COMMUNICATION and by ensuring that both intra and inter TSF communication is encrypted using the SSL network protocol. This objective also counters the threat T.DISCLOSURE_OF_PRIVATE_KEYS by ensuring that the private keys associated with encryption cannot be intercepted by unauthorized network nodes through the use of the SSL protocol which has features designed to counter these threats. SSL uses public key cryptography, whereby data is encrypted using a public key and then decrypted using a private key. During the initial network communication handshake, the SSL protocol exchanges only the public keys. Since the private keys, which are used to decrypt the data, are never exchanged, the SSL protocol is inherently secure from network related attacks designed to recover private keys.

**O.OBJ_ACCESS** - This objective counters the threat T. PRIVILEGE by ensuring access to named objects is explicitly granted, preventing an unauthorized user from gaining access to the named object.

**O.AUDIT –** This objective implements the security policy P.ACCOUNTABILITY, ensuring that all relevant TOE security actions are recorded. This objective also counters the threats T.AUDIT_CORRUPT by restricting access to all audit records to only authorized administrators.

**O.MANAGE** – This objective implements the security policy P.MANAGE, by ensuring that only authorized administrators can use the provided utilities for managing the security functions of the TOE and its resources. This objective also counters the threat T.AUDIT_CORRUPT by restricting access to all audit records to only authorized administrators, and T. DISCLOSURE_OF_CERTIFICATES by restricting access to the Certificate Manager interface that is used to install and manage SSL certificates.

O.VERIFY_CERTIFICATES – This objective counters the threat T.DISCLOSURE_OF_COMMUNICATION by ensuring that SSL encryption certificates follow the correct certificate format, and hence have not been tampered with.

## 8.1.2 Security Objectives for Environment Rationale

### 8.1.2.1 Security Objectives for the IT Environment Rationale

**Table 9: Security objectives for the IT environment mapped to assumptions** identifies security objectives for the IT environment in which the TOE is designed to operate and provides a mapping to assumptions that are made about that environment.

| TOE Security Objectives for the IT Environment | Threats and Assumptions |
|---|---|
| OE.AUTH_ACCESS | A.IDENT<br>A.SYSPROTECT<br>T.ACCESS |
| OE.SSL_IMP | A.OPERATE_CORRECTLY<br>T.DISCLOSURE_OF_COMMUNICATION<br>T.DISCLOSURE_OF_PRIVATE_KEYS |
| OE.LDAP_SSL_IMP | A.OPERATE_CORRECTLY<br>T.DISCLOSURE_OF_COMMUNICATION<br>T.DISCLOSURE_OF_PRIVATE_KEYS |
| OE.PATCH_VENDOR_SSL_IMP | A.OPERATE_CORRECTLY<br>T.DISCLOSURE_OF_COMMUNICATION<br>T.DISCLOSURE_OF_PRIVATE_KEYS |
| OE.SEP | A.CONNECT<br>A.SYSPROTECT<br>T.ACCESS |
| OE.TIME_SOURCE | A.TIME |

**Table 9: Security objectives for the IT environment mapped to assumptions**

**OE.AUTH_ACCESS** - This objective ensures that only authorized users have access to the TOE, thus countering T.ACCESS and assuring that A.IDENT and A.SYSPROTECT are addressed.

**OE.SSL_IMP** – This objective ensures that the web browser (such as Microsoft® Internet Explorer) correctly implements the SSL protocol to provide encrypted communication with the TSFs, assuring A.OPERATE_CORRECTLY is addressed. Additionally, the use of the SSL protocol counters T.DISCLOSURE_OF_COMMUNICATION and T.DISCLOSURE_OF_PRIVATE_KEYS through the use of encryption, and because this protocol has features that specifically address these threats.

**OE.LDAP_SSL_IMP** – This objective ensures that the LDAP server which is external to the TOE (such as Microsoft® Active Directory) correctly implements the SSL protocol to provide encrypted communication with the TSFs, assuring A.OPERATE_CORRECTLY is addressed. Additionally, the use of the SSL protocol counters T.DISCLOSURE_OF_COMMUNICATION and T.DISCLOSURE_OF_PRIVATE_KEYS through the use of encryption, and because this protocol has features that specifically address these threats.

**OE.PATCH_VENDOR_SSL_IMP** – This objective ensures that the Red Hat Satellite Server which is external to the TOE correctly implement the SSL protocol to provide encrypted communication with the TSFs, assuring A.OPERATE_CORRECTLY is addressed. Additionally, the use of the SSL protocol counters T.DISCLOSURE_OF_COMMUNICATION and T.DISCLOSURE_OF_PRIVATE_KEYS through the use of encryption, and because this protocol has features that specifically address these threats.

**OE.SEP** - This objective provides the support needed by the TOE to counter threats T.ACCESS by ensuring that the TOE cannot be tampered with or bypassed and assuring A.CONNECT and A.SYSPROTECT is addressed.

**OE.TIME_SOURCE -** The IT environment must provide a reliable time source for the TOE to provide an accurate timestamp for all audit records, thus assuring A.TIME is addressed.

#### 8.1.2.2 Security Objectives for the Non-IT Environment Rationale

**Table 10: Security objectives for the non-IT environment mapped to assumptions** identifies security objectives for the non-IT environment in which the TOE is designed to operate and provides a mapping to assumptions that are made about that environment.

| TOE Security Objectives for the Non-IT Environment | Assumptions |
|---|---|
| OE.PLATFORM_SUPPORT | A.PLATFORM_REQUIREMENTS<br>A.OPERATE_CORRECTLY |
| OE.INSTALL | A.INSTALL<br>A.PROTECT |
| OE.PERSON | A.NOEVIL<br>A.MANAGE<br>A.PLATFORM_REQUIREMENTS |
| OE.PHYCAL | A.CONNECT<br>A.PROTECT |

**Table 10: Security objectives for the non-IT environment mapped to assumptions**

**OE.PLATFORM_SUPPORT** - This objective ensures that the TOE is operating on the hardware, operating system, and associated software that would ensure the TOE operates correctly and has sufficient resources to execute the security functions correctly. This objective addresses A.PLATFORM_REQUIREMENTS and A.OPERATE_CORRECTLY

**OE.INSTALL -** Ensuring proper installation, management, and operation of the TOE to protect both itself and its resources addresses the assumption A.INSTALL and A.PROTECT.

**OE.PERSON -** This objective ensures that the TOE is operated in a secure manner by competent, trained personnel, which addresses A.NOEVIL assumption. This objective ensures that there are TOE administrators and they are properly trained and competent which addresses the A.MANAGE assumption, and also that a trained administrator has followed the platform requirements for the TOE to satisfy the A.PLATFORM_REQUIREMENTS assumption.

**OE.PHYCAL -** This objective ensures that the TOE is operated in an environment that will protect it from unauthorized access and physical threats and attacks that can disturb and corrupt the information generated. This objective addresses A.CONNECT and A.PROTECT.

## 8.2 Security Requirements Rationale

This section provides evidence supporting the internal consistency and completeness of the components (requirements) in the Security Target.

The purpose for the environmental objectives is to provide protection for the TOE that cannot be addressed through IT measures. The defined objectives provide for physical protection of the TOE, proper management of the TOE, and interoperability requirements on the TOE. Together with the IT security objectives, these environmental objectives provide a complete description of the responsibilities of TOE in meeting security needs.

### 8.2.1 Security Functional Requirements Rationale

**Table 11: SFRs mapped to Security Objectives** provides the correspondence mapping between security objectives for the TOE and the security functional requirements that satisfy them.

| SECURITY FUNCTIONAL REQUIREMENT | O.AUTHORIZATION | O.CLIENT_TO_SERVER_CONFIDENTIALITY_ | O.SERVER_TO_LDAP_ CONFIDENTIALITY | O.PATCH_VENDOR_CONFIDENTIALITY | O.ENCRYPTION | O.OBJ_ACCESS | O.AUDIT | O.MANAGE | O.VERIFY_CERTIFICATES | OE.AUTH_ACCESS | OE.SSL_IMP | OE.LDAP_SSL_IMP | OE.PATCH_VENDOR_SSL_IMP | OE.SEP | OE.TIME_SOURCE |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| FAU_GEN.1 | | | | | | | X | | | | | | | | |
| FAU_GEN.2 | | | | | | | X | | | | | | | | |
| FAU_SAR.1 | | | | | | | X | | | | | | | | |
| FAU_SEL.1 | | | | | | | X | | | | | | | | |
| FCS_COP.1(a-b) | | X | X | X | X | | | | | | | | | | |
| FDP_ACC.1(a) | | | | | | X | | | | | | | | | |
| FDP_ACC.1(b) | | | | | | X | | | | | | | | | |
| FDP_ACC.1(c) | | | | | | X | | | | | | | | | |
| FDP_ACC.1(d) | | | | | | X | | | | | | | | | |
| FDP_ACF.1(a) | | | | | | X | | | | | | | | | |
| FDP_ACF.1(b) | | | | | | X | | | | | | | | | |
| FDP_ACF.1(c) | | | | | | X | | | | | | | | | |
| FDP_ACF.1(d) | | | | | | X | | | | | | | | | |
| FPT_ITC.1 | | | | | | | | | | | X | X | X | | |
| FIA_ATD.1 | X | | | | | X | | | | X | | | | | |
| FIA_UAU.1/2 | X | | | | | | | | | X | | | | | |
| FIA_UID.2 | X | | | | | | | | | X | | | | | |
| FMT_MSA.1(a) | | | | | | X | | X | | | | | | | |
| FMT_MSA.1(b) | | | | | | X | | X | | | | | | | |
| FMT_MSA.1(c) | | | | | | X | | X | | | | | | | |

| SECURITY FUNCTIONAL REQUIREMENT | O.AUTHORIZATION | O.CLIENT_TO_SERVER_CONFIDENTIALITY_ | O.SERVER_TO_LDAP_ CONFIDENTIALITY | O.PATCH_VENDOR_CONFIDENTIALITY | O.ENCRYPTION | O.OBJ_ACCESS | O.AUDIT | O.MANAGE | O.VERIFY_CERTIFICATES | OE.AUTH_ACCESS | OE.SSL_IMP | OE.LDAP_SSL_IMP | OE.PATCH_VENDOR_SSL_IMP | OE.SEP | OE.TIME_SOURCE |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| FMT_MSA.1(d) | | | | | | X | | X | | | | | | | |
| FMT_MSA.2 | | | | | | | | | X | | | | | | |
| FMT_MSA.3(a) | | | | | | X | | X | | | | | | | |
| FMT_MSA.3(b) | | | | | | X | | X | | | | | | | |
| FMT_MSA.3(c) | | | | | | X | | X | | | | | | | |
| FMT_MSA.3(d) | | | | | | X | | X | | | | | | | |
| FMT_MTD.1(a) | | | | | | | | X | | | | | | | |
| FMT_MTD.1(b) | | | | | | X | | X | | | | | | | |
| FMT_MTD.1(c) | | | | | | | | X | | | | | | | |
| FMT_SMF.1 | | | | | | | | X | | | | | | | |
| FMT_SMR.1 | | | | | | | | X | | | | | | | |
| FPT_SEP.1 | | | | | | | | | | | | | | X | |
| FPT_STM.1 | | | | | | | | | | | | | | | X |
| FTA_SSL.3 | | | | | | | | X | | | | | | | |
| FTP_ITC.1 | | | X | X | X | | | | | | | | | | |
| FTP_TRP.1 | | X | | | X | | | | | | | | | | |

**Table 11: SFRs mapped to Security Objectives**

**O.AUTHORIZATION**

FIA_UAU.1 and FIA_UID.2 require a user be successfully be identified and authenticated before any access to the TOE and TOE-protected resources is allowed.

FIA_ATD.1 defines the unique attributes that are associated with individual users.

**O.CLIENT_TO_SERVER_CONFIDENTIALITY**

FTP_TRP.1 provides a trusted path between Transmitters and client and server endpoint computes, and between CMS, the Deployment Manager and the administrator's web browser. The trusted path ensures that no users can read the contents of these communication channels other than authorized users of the TOE. All communication is encrypted and complies with the RC4 encryption standards provided by FCS_COP.1(a) and RSA key exchange per FCS_COP.1(b).

**O.SERVER_TO_LDAP_CONFIDENTIALITY**

FTP_ITC.1 provides a trusted channel between Transmitters and CMS, and LDAP servers that are external to the TOE. The trusted channel ensures that no users can read the contents of these communication channels other than

authorized users of the TOE. All communication is encrypted and complies with the RC4 encryption standards provided by FCS_COP.1(a) and RSA key exchange per FCS_COP.1(b)..

**O.PATCH_VENDOR_CONFIDENTIALITY**

FTP_ITC.1 provides a trusted channel between the Red Hat Enterprise Linux Patch Source channel and the external Red Hat Satellite Server providing patch vendor meta-data and patches. The trusted channel ensures that no users can read the contents of these communication channels other than authorized users of the TOE. All communication is encrypted and complies with the RC4 encryption standards provided by FCS_COP.1(a) and RSA key exchange per FCS_COP.1(b)..

**O.ENCRYPTION**

A fundamental security function provided by the TSF is the use of encryption and key exchanges. An information flow security policy is maintained over all communication through the use of full session encryption that is provided through the exchange of cryptographic keys (FCS_COP.1(b)). Through the use of PKI technology on all session communication, the confidentiality of data exchanged between the sender and receiver is assured. Session communication through SSL ensures the sender and receiver must present encryption keys as the security attribute of the session information and no information may be imported into the TSF without a private session key (FTP_ITC.1 and FTP_TRP.1).

**O.OBJ_ACCESS**

FDP_ACC.1(a), FDP_ACC.1(b), FDP_ACC.1(c), FDP_ACC.1(d), and FDP_ACF.1(a), FDP_ACF.1(b), FDP_ACF.1(c), FDP_ACF.1(d) define the Access Control Policy, the subjects and objects that the policy covers, the security attributes that access to objects is based on, and the rules of access between subjects and objects. The Access Control Policy allows for the control of access to resources based on the user identity and group membership.

FIA_ATD.1 defines the security attributes that are associated to the user and used by the SFP.

FMT_MSA.1(a), FMT_MSA.1(b), FMT_MSA.1(c), FMT_MSA(d), FMT_MSA.3(a), FMT_MSA.3(b), FMT_MSA.3(c), FMT_MSA.3(d), FMT_MTD.1(b) and FMT_MTD.1(c) restrict the ability to change_default, query, modify, create, and delete security attributes to authorized users and ensures that restrictive default values are defined for the security attributes used to enforce the SFP.

**O.AUDIT**

FAU_GEN.1 and FAU_GEN.2 define the TOE events that will be audited, along with the details that will be recorded along with the event.

FAU_SAR.1 restricts access to the audit trail to authorized administrators and provides them a method for viewing the data according to various criteria. FAU_SEL.1 provides the capability for the authorized administrator to include or exclude records based on the channel as well as the severity and error code ranges.

**O.MANAGE**

FMT_MSA.1(a) restricts the ability to manage the associated security attributes (associated with Deployment Manager folders, deployments, task groups, server groups, and server keychains) of the Access Control SFP to the Deployment Manager Administrator.

FMT_MSA.1(b) restricts the ability to manage the associated security attributes (access control attributes associated with channels) of the Access Control SFP to the Primary Administrators and Administrators.

FMT_MSA.1(c) restricts the ability to manage the associated security attributes (access control attributes associated with policy manager targets) of the Access Control SFP to Primary Administrator and users with ACL write permissions.

FMT_MSA.1(d) restricts the ability to manage the associated security attributes (access control attributes associated with patch manager patch groups) of the Access Control SFP to the Primary Administrator and object owners.

FMT_MSA.3(a) enforces the restrictive default values for security attributes associated with a Deployment Manager named objects.

FMT_MSA.3(b) enforces the restrictive default values for security attributes associated with a Transmitter folder and channels.

FMT_MSA.3(c) enforces the restrictive default values for security attributes associated with a Policy Manager targets.

FMT_MSA.3(d) enforces the restrictive default values for security attributes associated with a Patch Manager patch groups.

FMT_MTD.1(a) provides the ability for the Primary Administrator, Administrator, and Operator to view the audit data

FMT_MTD.1(b) provides the ability for the Deployment Manager Administrator to manage the security attributes of other users.

FMT_MTD.1(c) provides the ability for Regular Users to modify their own security attributes.

FMT_SMR.1 requires the TOE to provide the ability to set roles for security relevant authority; Primary Administrator, Administrator, Operator, Deployment Manager Administrator, Transmitter Administrator, and Regular User.

FMT_SMF.1 requires that the TOE provide the ability to manage the security functions of TOE. Those functions include management of Access Control SFP and configuring idle user timeout.

FTA_SSL.3 provides the Deployment Manager Administrator and the Primary Administrator the ability to configure an idle user timeout; will automatically log out the current user after a specified period of time.

**O.VERIFY_CERTIFICATES**

FMT_MSA.2 ensures that imported SSL encryption certificates follow the correct certificate format, and are rejected if they do not.

**OE.AUTH_ACCESS**

FIA_ATD.1 defines the unique attributes that are associated with individual users.

FIA_UAU.2 and FIA_UID.2 require a user be identified and authenticated before any access to the TOE is allowed.

**OE.SSL_IMP**

The FPT_ITC.1 requirement ensures that there will be an implementation of SSL to support TOE operations. This requirement ensures the objective is adequately addressed.

**OE.LDAP_SSL_IMP**

The FPT_ITC.1 requirement ensures that there will be an implementation of SSL to support TOE operations. This requirement ensures the objective is adequately addressed.

**OE.PATCH_VENDOR_SSL_IMP**

The FPT_ITC.1 requirement ensures that there will be an implementation of SSL to support TOE operations. This requirement ensures the objective is adequately addressed.

**OE.SEP**

FPT_SEP.1 ensures the TOE maintains a separate execution domain to protect from external tampering.

**OE.TIME_SOURCE**

FPT_STM.1 ensures that an accurate time source will be available to the TOE.

## 8.3 Security Assurance Requirements Rationale

This ST contains the assurance requirements from the CC EAL3 assurance package and is based on good commercial development practices. This ST has been developed for a generalized environment with a medium level of risk to the assets. The security environment in which the TOE operates assumes physical protection. Client and Server Management provides a level of protection that is appropriate for IT environments that require secure automated change management, such as the distribution of application updates and patches throughout an enterprise. As such, it is believed that EAL3 provides an appropriate level of assurance in the security functions offered by the TOE.

## 8.4 Security Requirements Dependencies Rationale

The table below maps the TOE security functional requirements to the corresponding requirements they are dependent on. The dependencies of the TOE security functional requirements are, for the most part, met through the functionality of the TOE and/or by the security functionality of the IT environment.

| Functional Component | Dependency | Included |
|---|---|---|
| FAU_GEN.1 | FPT_STM.1 | Included (environment) |
| FAU_GEN.2 | FAU_GEN.1 | Included |
| | FIA_UID.2 | Included |
| FAU_SAR.1 | FAU_GEN.1 | Included |
| FAU_SEL.1 | FAU_GEN.1 | Included |
| | FMT_MTD.1 | Included (FMT_MTD.1(a)) |
| FCS_COP.1(a-b) | FCS_CKM.1, FCS_CKM.4 | No – See rationale below. |
| | FMT_MSA.2 | Included |
| FDP_ACC.1(a) | FDP_ACF.1(a) | Included |
| FDP_ACC.1(b) | FDP_ACF.1(b) | Included |
| FDP_ACC.1(c) | FDP_ACF.1(c) | Included |
| FDP_ACC.1(d) | FDP_ACF.1(d) | Included |
| FDP_ACF.1(a) | FDP_ACC.1(a) | Included |
| | FMT_MSA.3(a) | Included |
| FDP_ACF.1(b) | FDP_ACC.1(b) | Included |
| | FMT_MSA.3(b) | Included |
| FDP_ACF.1(c) | FDP_ACC.1(c) | Included |
| | FMT_MSA.3(c) | Included |
| FDP_ACF.1(d) | FDP_ACC.1(d) | Included |
| | FMT_MSA.3(d) | Included |
| FPT_ITC.1 | None | |
| FIA_ATD.1 | None | |
| FIA_UAU.1/2 | FIA_UID.1 | Met by FIA_UID.2 |
| FIA_UID.2 | None | |
| FMT_MSA.1(a, b, c, & d) | FDP_ACC.1 or FDP_IFC.1 | Included |

| Functional Component | Dependency | Included |
|---|---|---|
| | FMT_SMF.1 | Included |
| | FMT_SMR.1 | Included |
| FMT_MSA.2 | ADV_SPM.1 | No – See note below. |
| | FDP_ACC.1(a, b, c & d) | Included |
| | FMT_MSA.1 | Included |
| | FMT_SMR.1 | Included |
| FMT_MSA.3(a) | FMT_MSA.1 | Included |
| | FMT_SMR.1 | Included |
| FMT_MSA.3(b) | FMT_MSA.1 | Included |
| | FMT_SMR.1 | Included |
| FMT_MSA.3(c) | FMT_MSA.1 | Included |
| | FMT_SMR.1 | Included |
| FMT_MSA.3(d) | FMT_MSA.1 | Included |
| | FMT_SMR.1 | Included |
| FMT_MTD.1 (a, b & c) | FMT_SMR.1 | Included |
| | FMT_SMF.1 | Included |
| FMT_SMF.1 | None | |
| FMT_SMR.1 | FIA_UID.2 | Included |
| FPT_SEP.1 | None | |
| FPT_STM.1 | None | |
| FTA_SSL.3 | None | |
| FTP_ITC.1 | None | |
| FTP_TRP.1 | None | |

**Table 12: Requirement Dependency Rationale**

- FCS_CKM.1, FCS_CKM.4 – These requirements address cryptographic key creation and destruction. The keys associated with the RC4 and RSA algorithms are created and destroyed automatically upon connection and disconnection. Therefore, these requirements are not necessary to satisfy the FCS_COP.1(a-b) requirements.
- FMT_MSA.2 is present to cover the handling of encryption certificates. In this context the dependency on ADV_SPM.1 is not required, as 'secure values' are defined as those appropriate for the encryption algorithm.

## 8.5 TOE Summary Specification Rationale

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

This Section in conjunction with TOE Summary Specification section provides evidence that the security functions are suitable to meet the TOE security requirements. The collection of security functions work together to provide all of the security requirements. The security functions described in the TOE Summary Specification are all necessary for the required security functionality in the TSF.

**Table 13: Security Functions vs. Requirements Mapping** demonstrates the relationship between security requirements and security functions.

| | Cryptographic Operation | Audit | User Data Protection | Identification & Authentication | Security Management | TOE Access | Trusted Path/Channel |
|---|---|---|---|---|---|---|---|
| FAU_GEN.1 | | X | | | | | |
| FAU_GEN.2 | | X | | | | | |
| FAU_SAR.1 | | X | | | | | |
| FAU_SEL.1 | | X | | | | | |
| FCP_COP.1(a-b) | X | | | | | | |
| FDP_ACC.1(a) | | | X | | | | |
| FDP_ACC.1(b) | | | X | | | | |
| FDP_ACC.1(c) | | | X | | | | |
| FDP_ACC.1(d) | | | X | | | | |
| FDP_ACF.1(a) | | | X | | | | |
| FDP_ACF.1(b) | | | X | | | | |
| FDP_ACF.1(c) | | | X | | | | |
| FDP_ACF.1(d) | | | X | | | | |
| FIA_ATD.1 | | | | X | | | |
| FIA_UAU.1 | | | | X | | | |
| FIA_UID.2 | | | | X | | | |
| FMT_MSA.1(a) | | | | | X | | |
| FMT_MSA.1(b) | | | | | X | | |
| FMT_MSA.1(c) | | | | | X | | |
| FMT_MSA.1(d) | | | | | X | | |
| FMT_MSA.2 | | | | | X | | |
| FMT_MSA.3(a) | | | | | X | | |
| FMT_MSA.3(b) | | | | | X | | |
| FMT_MSA.3(c) | | | | | X | | |
| FMT_MSA.3(d) | | | | | X | | |
| FMT_MTD.1(a) | | | | | X | | |
| FMT_MTD.1(b) | | | | | X | | |
| FMT_MTD.1(c) | | | | | X | | |
| FMT_SMF.1 | | | | | X | | |
| FMT_SMR.1 | | | | | X | | |
| FTA_SSL.3 | | | | | | X | |
| FTP_ITC.1 | | | | | | | X |
| FTP_TRP.1 | | | | | | | X |

**Table 13: Security Functions vs. Requirements Mapping**

**FAU_GEN.1 Audit Data Generation**

The Transmitter and Patch Manager interface generate audit log messages which are subsequently collected and stored by the Centralized Logging channel.

**FAU_GEN.2 User Identity Association**

Both the Transmitter and Patch Manager audit records identify the user performing the associated actions being logged.

**FAU_SAR.1 Audit Review**

Report Center allows the user to query, filter, and sort audit logs collected using the Centralized Logging channel.

**FAU_SEL.1 Selective Audit**

Every Marimba channel has a logging mechanism that generates a channel-specific log file on each computer endpoint. The Report Center's interface enables an administrator to specify exactly which log messages should be collected from these endpoints.

**FCS_COP.1(a-b) Cryptographic Operation**

All encrypted communication utilizes the RC4 cryptographic algorithm using a 128 bit key size. Similarly, keys are exchanged using a 512 or 1024-bit RSA key exchange. This encryption, decryption, and key exchange functions are implementation is used by the secure sockets layer (SSL) network protocol, and is implemented by a RSA Security, Inc. BSAFE library linked into the TOE.

**FDP_ACC.1(a) Subset Access Control**

The Deployment Manager interface restricts access to users for the following names objects: Deployment Manager Folders, Deployments, Server Groups, Server Keychains and Task Groups.

**FDP_ACC.1(b) Subset Access Control**

The Transmitter restricts access to users for the following operations: Publish access control ('write' operations to the Transmitter), un-publish operations ('delete' operation on the Transmitter) and channel access (read operations on the Transmitter).

**FDP_ACC.1(c) Subset Access Control**

The Policy Manager interface restricts access to users for Policy Manager Target named objects.

**FDP_ACC.1(d) Subset Access Control**

The Patch Manager interface restricts access to users for Patch Group named objects.

**FDP_ACF.1(a) Security Attribute Based Access Control**

The Deployment Manager interface uses the following attributes to implement access control: An ACL containing a user or group name and four permission bits for read(r), write(w), execute(e) and owner(o). Additionally, each user and group object maintained by the Deployment Manager has the following create permission attributes associated with it: Five attributes, one for each Deployment manager named object, each containing three possible values, None, Create or Revoke.

**FDP_ACF.1(b) Security Attribute Based Access Control**

The Transmitter implements access control for a Transmitter Folder or Channel using an ACL associated with each named object that has a single attribute that can have the following values: a user authenticated using an external source, a specific user authenticated using an external source or a user group.

**FDP_ACF.1(c) Security Attribute Based Access Control**

The Policy Manager interface implements access control using an ACL associated with each user or group (specified in the external LDAP server), containing the following attributes: one or more Policy Manager Targets, four permission bits for ACL read, ACL Write, policy read and policy write.

**FDP_ACF.1(d) Security Attribute Based Access Control**

The Patch Manager interface implements access control using an ACL associated with each Patch Group containing the following attributes: one or more users or user groups (specified in the external LDAP server), three permission bits for modify, publish and delete.

**FIA_ATD.1 User Attribute Definition**

The Deployment Manager interface maintains the following attributes for each user managed by the interface: user identity, group membership, home folder, password, create permissions (also described in FDP_ACF.1(a) )

**FIA_UID.2 User Identification Before any Action/FIA_UAU.1 Timing of authentication**

Before any security management functions can be performed, the user must identify themselves and be authenticated by using the Deployment Manager interface GUI, or by external authentication using an LDAP directory service.

**FMT_MSA.1(a) Management of Security Attributes**

The Deployment Manager interface *Folder Contents* screen is used to manage the attributes for deployment manager named object access control

**FMT_MSA.1(b) Management of Security Attributes**

The Transmitter Administrator interface (a component of the Infrastructure Administration channel) is used to manage the security attributes for Transmitter Folder and Channel access control. User and user group create privileges are managed using the Deployment Manager interface *Edit User* and *Edit Group* screens.

**FMT_MSA.1(c) Management of Security Attributes**

The Policy Manager interface is used to manager the security attributes for Policy Manager Targets using either the *Targets View* or *User and Groups View* screens. Additionally, the Policy Manager access control function can be disabled or enabled using the command line interface. For the evaluated configuration, Policy Manager access control is always enabled.

**FMT_MSA.1(d) Management of Security Attributes**

The Patch Manager user interface allows permissions to be granted to users and user groups when a new patch group is created, and also by modifying an existing patch group.

**FMT_MSA.2 Secure Security Attributes**

The Certificate Manager enables an administrator to perform the following management tasks for SSL and root encryption certificates: requesting certificates from a certificate vendor, installing SSL certificates, viewing SSL certificate information, deleting SSL certificates, importing and exporting SSL certificates. The Certificate Manager ensures that imported certificates follow the correct certificate format, and are rejected if they do not.

**FMT_MSA.3(a) Static Attribute Initialization**

By default, the deployment manager does not grant any permissions to users trying to access Deployment Manager named objects. The deployment manager administrator or object owner uses the Deployment Manager interface to grant permissions to other users.

**FMT_MSA.3(b) Static Attribute Initialization**

By default, the Transmitter Administrator provides full access to users for Transmitter Folders and Channels. Primary Administrators must explicitly set permissions for these named objects to enable more restrictive access.

**FMT_MSA.3(c) Static Attribute Initialization**

By default, the Policy Manager interface does not grant any permissions to users trying to access Policy Manager Targets. Primary Administrators and ACL administrators use the Policy Manager interface to grant permissions for Policy Manager Targets.

**FMT_MSA.3(d) Static Attribute Initialization**

By default, patch group permissions to modify, publish and delete are not granted to any users or groups, and must be explicitly granted when a patch group is created. All users and groups can read the contents of a patch group.

**FMT_MTD.1(a) Management of TSF Data**

The Report Center interface provides authorized users with the ability to query audit log data that has been gathered from other TOE components using the Centralized Logging channel.

**FMT_MTD.1(b) Management of TSF Data**

Deployment Manager Administrators use the Deployment Manager Interface to initialize, modify and delete user accounts.

**FMT_MTD.1(c) Management of TSF Data**

The Deployment Manager interface only allows Deployment Manager Administrators to modify, initialize and delete user account attributes, however users are allowed to modify their own account attributes even if they are not assigned the Deployment Manager Administrator role.

**FMT_SMF.1 Specification of Management Functions**

The Client and Server Management products provide access control management functions for the various named objects associated with each product components, as described in FMT_MSA(a), FMT_MSA(b), FMT_MSA(c) and FMT_MSA(d). Additionally, the Certificate Manager interface provides management functions for encryption (SSL) certificates, and both CMS and the Deployment Manager allow user session timeout values to be specified, which log out users after a specified period of no activity.

**FMT_SMR.1 Security Roles**

The Deployment Manager interface provides the following user roles: Deployment Manager Administrator, Regular User, and Object Owner.

The Infrastructure Administration, Report Center, Policy Manager, Patch Manager interfaces (in CMS) provide the following user roles: Primary Administrator, Administrator, Policy Administrator, Operator, users with ACL Write Permissions (also referred to as *ACL Administrator*), and Object Owner.

**FTA_SSL.3 TSF-Initiated Termination**

Deployment Manager and Common Management Services (CMS) (and hence the Infrastructure Administration, Report Center, Policy Manager, Patch Manager interfaces) automatically log out users after a specified period of time.

**FTP_ITC.1 Inter-TSF Channel**

All communication between Transmitters and CMS and external LDAP servers uses an encrypted channel to ensure the privacy and integrity of the date is maintained. Additionally, communication between the Red Hat Enterprise Patch Source channel and the external Red Hat Satellite Server providing patch vendor meta-data and patches uses an encrypted channel to ensure the privacy and integrity of the data is maintained. All encrypted communication uses the Secure Sockets Layer (SSL) protocol.

**FTP_TRP.1 Trusted Path**

Marimba Client and Server Management uses a series of trusted path communication channels to encrypt data between various parts of the TOE to prevent modification and disclosure of user data. All encrypted communication uses the Secure Sockets Layer (SSL) protocol.

The following trusted communication paths are provided: the delivery of channels from the Transmitter to client and server endpoints, communications between the Infrastructure Administration channel and Tuners, Transmitters and Proxies, publish data sent to the Transmitter from the Publisher, Channel Copier and Application Packager, data copy operations performed by the Infrastructure Administration channel, administrative operations using CMS and Deployment Manager, logging information sent to the Deployment manager, and commands sent to the Deployment Service.

## 8.6 Internal Consistency and Support Rationale

The selected functional requirements for the TOE and IT Environment are internally consistent. All the operations performed are in accordance with the CC. The ST does not include any instances of a requirement that conflicts

with or contradicts another requirement. In instances where multiple requirements apply to the same functions, the requirements and their operations do not cause a conflict between each other.
The selected requirements are mutually supportive by supporting the dependencies as demonstrated in **Table 12: Requirement Dependency Rationale**. The rationale of the suitability of the requirements to meet the objectives; the inclusion of architectural requirements FPT_SEP.1, to protect the TOE; the inclusion of audit requirements to detect unauthorized actions and/or events, the inclusion of user data protection for access control, and the inclusion of security management requirements to provide a means to properly configure and manage the other security requirements.

## 8.7 Strength of Function (SOF) Rationale

The TOE minimum strength of function of SOF-basic was chosen to be consistent with the TOE environment. The SOF-claim is associated with the authentication mechanism described in Identification and Authentication, which supports FIA_UAU.1.2.

The list of relevant security functions and security functional requirements includes:

- Identification and Authentication Security Function
    - FIA_UAU.1.2 – Timing of authenticaiton, security functional requirement

The password is the only probabilistic or permutational function on which the strength of the authentication mechanism depends.

The system places the following restrictions on the passwords selected by the user:

- The password must be more than eight characters; a minimum of nine characters

Furthermore, the user is told to not use consecutive sequences, or easily guessable passwords

The password space is calculated as follows:

Patterns of human usage are important considerations that can influence the approach to searching a password space, and thus affect SOF. Assuming the worst case scenario and the user chooses a number comprising only nine characters, the number of password permutations is:

52 alpha characters (upper and lower)
10 digits
+ 16 special characters ( !, @, #, $, %, ^, &, *, (, ), +, =, <, >, :, ;)
78 possible values

78^9 = (78*78*78*78*78*78*78*78*78) = **106,868,920,913,284,608**

A proficient or expert person could create a program to guess passwords approximately 20 times every second without using any specialized equipment. This assumption is based on timing a logon attempt on a 3.4 GHz Pentium. The average total time to guess the correct password can be estimated by:

(53,434,460,456,642,304 passwords) * (.2 seconds / password guess) * (1 hour / 3600 seconds) * (1 day / 24 hours) * (1 year / 365 days) = 338879125 years
In accordance with annex B.3 in the CEM, the elapse time of attack over which the user would try to guess the password is too great to make the TOE vulnerable and thus results in a basic strength of function (SOF-basic) rating.