Security Target for IBM Tivoli License Compliance Manager version 2.2, Fix Pack 1

Version 2.2, 18 December 2006

Table of Contents

1 Introduction

1.1 ST Identification

Title: Security Target for IBM Tivoli License Compliance Manager, version 2.2, Fix Pack 1

Assurance level: EAL 2 augmented by ALC_FLR.1

Keywords: Tivoli, License Manager, IBM Tivoli License Manager (ITLM), Software License Management, Software License Monitoring, LCM, IBM Tivoli License Compliance Manager (ITCLM), License Compliance Manager

Note: Shortly before general release of version 2.2 of the product, the product name was changed to from IBM Tivoli License Manager to IBM Tivoli License Compliance Manager. References to the product name in the product guidance, design documents, and most other evidence use the previous name for the product, IBM Tivoli License Manager.

1.2 ST Overview

This document defines the Security Target (ST) for the Common Criteria evaluation of **IBM Tivoli License Compliance Manager, version 2.2, Fix Pack 1** and **IBM Tivoli License Compliance Manager for IBM Software, version 2.2, Fix Pack 1**. This ST describes the target of evaluation (TOE), its IT environment, IT security requirements, and security functions. This ST has been developed in accordance with the Common Criteria for Information Technology Security Evaluation (CC) version 2.3 (August 2005).

1.2.1 Product naming convention

Unless there is a need to distinguish between the two forms of the product, references to “IBM Tivoli License Compliance Manager, version 2.2, Fix Pack 1“ throughout this Security Target and other product guidance, design documents, and other evidence apply to both forms of the product. See section 2.5.3.1 for more information about the two product forms.

1.3 Tivoli License Compliance Manager Overview

IBM Tivoli License Compliance Manager, version 2.2, Fix Pack 1 is a software only product that helps organizations manage software assets. Information about installed software and software use (software usage records) is collected by agents deployed on monitored computers, and stored in a central DB2 database that can be accessed by an authorized administrator through reports generated using a Web interface. Using Tivoli License Compliance Manager’s capabilities enables organizations to know exactly what software licenses they have, which licenses are being used, and which licenses they need.

1.4 CC Conformance Claim

This ST is Part 2 conformant and Part 3 conformant to the CC v2.3 (August 2005). The ST is conformant with the security functional requirements specified in CC Part 2 and with the security assurance requirements for Evaluation Assurance Level 2 (EAL 2) augmented with ALC_FLR.1 specified in CC Part 3, including the Common Criteria Interpretations Maintenance Board (CCIMB) final interpretations as of February 2006, the date of application.

This ST does not claim conformance to any Protection Profile (PP).

1.5 Strength of Function

The claimed Strength of Function (SOF) for this TOE is: **SOF-basic**.

1.6 Related Standards and Documents

[CC] ISO 15408 – Information Technology – Security Techniques – Evaluation Criteria for IT Security, version 2.3 (also known as the Common Criteria or CC).

[CCG] ISO/IEC PDTR 15446 – ISO-Guide for the Production of Protection Profiles and Security Targets, Draft 2004-01-04.

[CEM] Common Methodology for Information Security Evaluation, version 2.3 (CEM).

2 TOE Description

This chapter provides a general description of the IBM Tivoli License Compliance Manager, version 2.2, Fix Pack 1 product focusing on the product's security-related features and aspects, defines the target of evaluation (TOE) and the intended TOE environment, and introduces the security functions that we will demonstrate address the security requirements placed on the product.

2.1 Product Type

IBM Tivoli License Compliance Manager, version 2.2, Fix Pack 1 is a software only product that provides software inventory, use metering, and license allocation services. Information about installed software and software use (software usage records) is collected from monitored computers and stored in a central DB2 database. The software usage records can be accessed by an authorized administrator to produce reports for billing and tracking of software license use within a defined organization.

The product is available in two forms with identical security functionality, as described in section 2.5.3.1.

2.1.1 Product architecture

Tivoli License Compliance Manager is based on a three-tier architecture composed of multiple servers with associated databases, agents, and related components supporting the product's functionality. **Figure 1** shows a very simple deployment.

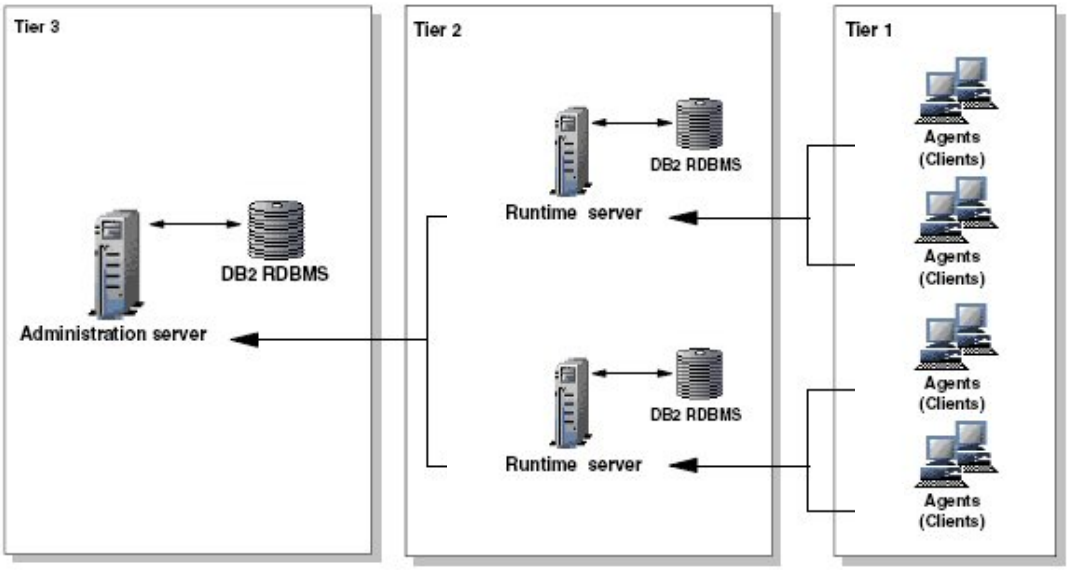

Figure 1: Tivoli License Compliance Manager, three-tier architecture

These are the main components of Tivoli License Compliance Manager, as illustrated in **Figure 1**:

– a single **Administration Server** through which various administrative, monitoring, and reporting capabilities are provided and which with its associated **DB2 database**, provides a repository for product, license agreement, license use, installed software, and organization information. (In the context of this evaluation, the DB2 database is part of the environment, not part of the TOE.)

- one or more **Runtime Servers**, which act as a proxy between agents and the Administration Server. Each Runtime Server has an associated **DB2 database**, whose primary functions are (1) to provide a repository of information required by the agents to support monitoring tasks (catalog of products to be monitored, schedule for performing installed software scans) and (2) to provide a temporary repository for data collected by the agents connected to the Runtime Server. This data is then transmitted to the Administration Server at regular, configurable intervals. Scalability of the monitored organization is addressed by adding more Runtime Servers as needed in order to balance performance requirements. (In the context of this evaluation, the DB2 database is part of the environment, not part of the TOE.)
- a **license management agent** deployed on each computer that is to be monitored. The agent performs an inventory of the software installed on the computer and monitors use of installed software products, and forwards this information (software usage records) to the Runtime Server, which in turn forwards the software usage records to the Administration Server.

As shown in **Figure 1**, the software usage records managed by the TOE flow from the agents to the Runtime Servers, then from the Runtime Servers to the Administration Server.

2.1.2 Intended use and environment

The primary purpose for a customer to use Tivoli License Compliance Manager is to monitor software license use within one or more defined organizations. Reliable software usage records are essential to satisfy billing requirements in some software pricing models and to plan for software purchase decisions. Ensuring the integrity of the software usage records collected from the monitored computers in the organization is the key security-related priority in Tivoli License Compliance Manager.

The hierarchical structure of Tivoli License Compliance Manager allows for flexible deployment. For example, the Administration Server and Runtime Server components can be deployed on the same machine, or Runtime Servers may be deployed within the same physical network as the Administration Server, or Runtime Servers may be deployed at remote locations. Agents may be deployed on machines physically within or outside the secure network. Administrators are trusted to make appropriate decisions to ensure secure communication if components are not deployed within a secure network.

Tivoli License Compliance Manager is an internally-deployed system, accessible only by a trusted and competent group of administrators in a controlled environment. There are no typical “end users” interacting with the product.

2.1.3 Administration server logging

The Administration Server performs logging of actions, which includes the logging of failed login attempts for both Runtime Servers and TOE users. The events are saved in the Message Log. The Message Log includes a log management system. When a log file reaches a maximum size defined by TLM, the file is closed and a new file created. After the number of log files reaches the maximum number of files defined by TLM, the oldest log file is overwritten with new events.

The logging function is not part of the evaluated configuration in this release.

2.2 Security Functionality in the TOE

2.2.1 Identification and authentication

2.2.1.1 Administrator authentication to the Administration Server

When the Super Administrator (see section 2.3.3) creates a new administrator (user) of the TOE, the administrator is assigned a user name and password. To access the Administration Server using the GUI (via a web browser), all administrators including the Super Administrator are required to provide their credentials (user name and password). The login information submitted via the web browser is collected by the WebSphere Application Server and passed to the Administration Server, which performs the user identification and authentication operations by comparing the credentials provided by the user to the user credentials stored in the Administration Server database.

2.2.1.2 Agent to Runtime Server authentication

In the evaluated configuration, an install-time parameter must be set to require agents to authenticate to the Runtime Servers using the authentication feature of SSL as provided by IBM Global Security Kit (GSKit), which is deployed with the license management agent software and is part of the TOE. Configuring SSL with client authentication for the agent to Runtime Server connection means also that the environment must provide for creation, distribution, and installation of an X.509v3 certificate to each agent (management of the certificate is outside the TOE itself). Agents then connect to a Runtime Server through the Runtime Web Server using the SSL protocol configured to use encryption and digests. (The Web Server is part of the TOE environment)

2.2.1.3 Runtime Server to Administration Server authentication

Depending on the nature of the deployment, IBM Tivoli License Compliance Manager, version 2.2, Fix Pack 1 can be configured at install time to require Runtime Server to Administration Server authentication. This model might be chosen, for example, if the Runtime Server is deployed remotely from the Administration Server. If, on the other hand, the Runtime Server and the Administration Server are installed on the same machine or installed in a well-protected Data Center, the system administrator or other individual choosing security levels is likely to conclude that such authentication is not necessary. Other deployments will fall somewhere between these two extremes, and administrators will decide whether implementing secure communication is appropriate.

If Runtime authentication is to be used, during installation of each Runtime Server, a Runtime Server name and communication password are created. The password is stored encrypted in the passwd.properties file on the Runtime Server. During registration of the Runtime Server to the Administration Server, these credentials are written to the Administration Server database. Each time it needs to initiate a transaction to the Administration Server, the Runtime Server is required to provide its credentials (Runtime Server name and password) to the Administration Server. The Administration Server identifies and authenticates the Runtime Server by comparing the credentials submitted by the Runtime Server to the Runtime Server credentials stored in the Administration Server database.

2.2.2 Password policy enforcement

In order to resist attempts to guess a password, the TOE enforces a global password policy for administrator (user) and Runtime Server communication passwords when

configured through the Admin Server GUI[1]. All administrators are required to comply with the policy regardless of their role and organization. The policy defines a minimum password length of 8 characters and password composition rules (passwords must contain a minimum of 2 non-alphabetic characters, passwords may consecutively repeat a maximum of 2 characters).

2.2.3 Session timeout

An Administration Server GUI session will terminate if the time interval *n* set for session inactivity timeout (the sessionTimeout parameter in the system.properties configuration file on the Administration Server) is exceeded.

2.3 Security Roles and Management

2.3.1 Association of software usage records to organizations

An important element in limiting access by individual administrators to software usage records that are relevant to organizations in which the administrator plays a role is associating each software usage record to an organization as the data is collected and then stored in the Administration Server database.

The Super Administrator (see section 2.3.3) creates logical organizations for which software licensing needs to be managed. Both the Runtime Server and the agent are configured to a specific organization, so that software usage data collected and passed by those components is identified with that organization. Each Runtime Server is associated with an organization at the time the Runtime Server is registered to the Administration Server. Each agent is associated with an organization and a Runtime Server when the agent is installed.

Software usage records are then collected from agents and sent to the Runtime Server, and then sent on the Administration Server database. In the Administration Server database, each software usage record is associated with its configured organization.

2.3.2 Access control

The Administration Server controls access to software usage records in the Administration Server database based on the organizations, roles, and privacy policies associated with an administrator (note that all “users” of the TOE are administrators)

Each administrator’s profile contains a list of organizations that the administrator belongs to, as well as the administrator’s configured role within each organization and the administrator’s privacy policy setting within each organization.

Each administrator is only allowed to access software usage records associated with the organizations listed in his or her profile. The access rights that an administrator is granted to the software usage records depend on the roles assigned to the administrator within the organization.

[1] As described in sections 2.3.5.1 and 2.3.5.2, the password must be changed on the Administration Server via the Administration Server GUI and also on the Runtime Server via the Runtime Server CLI, and the passwords must match exactly. The Administration Server GUI enforces the password rules. The Runtime Server CLI does not enforce the password rules, but because the passwords entered through both interfaces must match exactly in order for the Runtime Server to connect to the Administration Server, in effect, the rules enforced by the Administration Server GUI prevail.

The privacy policy associated with the administrator defines the level of detail about individual monitored computers in an organization that an administrator can view. If the privacy policy parameter is set to show computer information, then search criteria and report data related to individual agents and computers is displayed. If the parameter is set to not show computer information, the search criteria in reports is hidden to limit report data of defined agents and computers, and the data is queried on a division granularity basis only.

2.3.3 Security roles

The Super Administrator role is created at install time.

The Super Administrator can assign these roles to administrators, each of which entitles the individual assigned to the role with a specific level of access to data stored in the Administration Server database:

- Administrator
- Procurement Manager
- Software Resources Manager
- License Administrator
- System Resources Manager
- Procurement and Licensing Manager

See Table 4: Roles and Tasks for information about the specific access for each role.

The roles and the access rights for each role are hard coded into the product (i.e., not configurable). As previously mentioned, roles are specific to an organization.

2.3.4 Secure data transfer between components

2.3.4.1 Agent to Runtime Server communication

In the evaluated confirmation, communication between the agent and the Runtime Server must be configured to use SSL encryption and client authentication in order to protect the confidentiality and integrity of the data flow between them. The TOE uses the IBM Global Security Kit (GSKit) library for the implementation of the SSL protocol and its underlying cryptographic functions at the agent side.

2.3.4.2 Runtime Server to Administration Server communication

For Runtime Server to Administration Server communication, administrators will decide whether to enable Runtime Server to Administration Server security based on the nature of the deployment, as previously described in section 2.2.1.3.

If secure communication is selected, encryption is provided by the SSL protocol as a function of the environment (through the JSSE library of the IBM JDK in the WebSphere Application Server). In addition, mutual authentication is performed. The Administration Server authenticates itself to the Runtime Server using an SSL certificate. Authentication of the Runtime Server to the Administration Server is provided by the TOE using Runtime Server credentials (a unique name and password for each Runtime Server) known by both parties.

2.3.4.3 Web browser to Administration Server communication

In the evaluated configuration, communication between the Administration Server GUI (accessed through a Web browser) and the Administration Server must be configured to use the HTTPS protocol for all data transmissions. (This is a requirement on the TOE environment.)

2.3.4.4 Guaranteed data delivery

The data flow between an agent and Runtime Server and between a Runtime Server and Administration Server also provides an extra level of data protection beyond the communications protocol. Specifically, when a server receives software usage records from a client, the server sends a message back to the client after the server saves the software usage records in its database. This message informs the client that the data has been successfully persisted at its target. The client then can delete the transmitted software usage records. For example, when an agent sends software usage records to the Runtime Server, the Runtime Server signals the agent when it has saved the software usage records in its database. The agent then is free to delete the software usage records from its own internal storage.

2.3.5 Management of security functions

2.3.5.1 Administration Server GUI

The Administration Server GUI (via a web browser) provides the Super Administrator with the ability to manage the security functionality of the TOE. After successful identification and authentication (see section 2.2.1.1), the Super Administrator can perform the following management tasks:

- create and manage organizations
- create and manage administrators (users) including assigning roles, organizations, and privacy policies to administrators (users)
- change administrator (user) passwords and his or her own password
- change Runtime Server communication passwords (note that in addition, the password must be changed on the Runtime Server – see section 2.3.5.2)

Other administrator roles are able to perform a single security function management task through the Administration Server GUI:

- change their own passwords

2.3.5.2 Runtime Server CLI

The Runtime Server provides a command line interface (CLI) to manage the following security-relevant tasks:

- changing the Runtime Server to Administration Server communication password using the **rtpasswd** command (note that in addition, the password must be changed on the Administration Server through the Administration Server GUI - see section 2.3.5.1)
- changing the Runtime Server to Runtime Server Database password using the **dbpasswd** command
- changing the password used by the Runtime Server to open its truststore file (key.jks) using the **sslpasswd** command.

- re-encrypting all passwords in the password.properties file on the Runtime Server using the **kstoreupdate** command.

The command line can only be used by Windows users with administrator rights and UNIX user root from the UNIX Shell. Executing the **rtpasswd**, **dbpasswd**, or **sslpasswd** command requires entering the old password.

2.3.5.3 Administration Server CLI

The Administration Server provides a command line interface (CLI) to manage the following security-relevant tasks:

- changing the Administration Server to Administration Server Database password using the **dbpasswd** command
- re-encrypting all passwords in the password.properties file on the Administration Server using the **kstoreupdate** command

The command line can only be used by Windows users with administrator rights and UNIX user root from the UNIX Shell. Executing the **dbpasswd** command requires entering the old password.

2.4 Deployment and installation of agents

Multiple mechanisms are available to securely deploy and install the Tivoli License Compliance Manager agent software, including pre-requisite software and certificates and keys required for SSL encryption and mutual authentication. An administrator in the system will choose the secure deployment model that is best-suited to the environment in which the Tivoli License Compliance Manager will run, taking into account the strengths and limitations of each deployment method as documented in the administrator guidance.

Deployment and installation using the available mechanisms automatically configures the agent securely on each target computer. To further reduce the possibility of contamination of the software usage records gathered by the agents, the file permissions of the software usage records files on the agents are set to allow access only by a user with administrator (Windows) or root (UNIX/Linux) privileges.

2.5 TOE Definition

2.5.1 TOE boundary

The TOE is a subset of the IBM Tivoli License Compliance Manager, version 2.2, Fix Pack 1 package. **Figure 2** shows an installation of the product including the components previously described, as well as associated components. Components that are part of the TOE are marked in blue. All other components are considered to be part of the TOE environment.

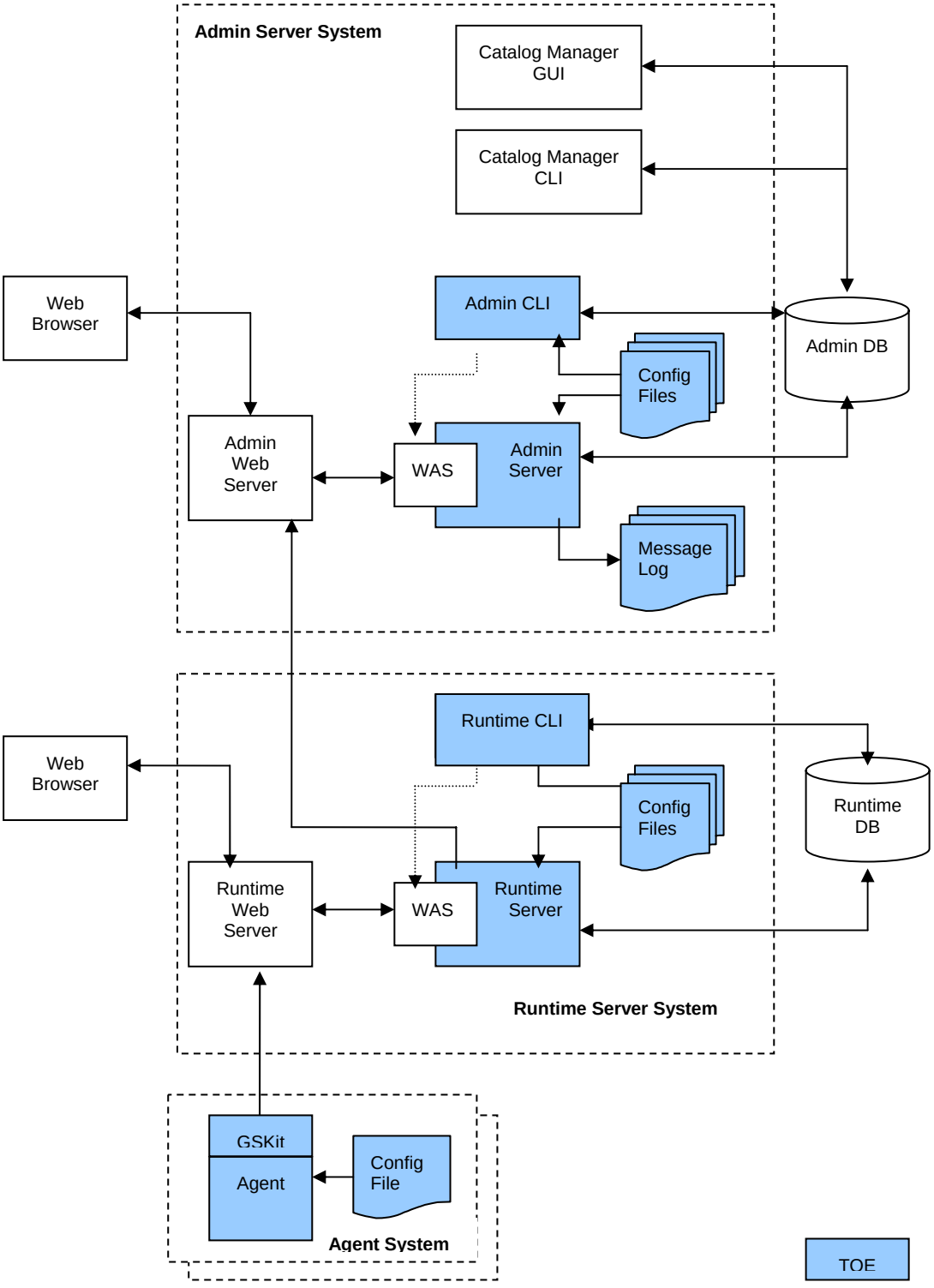

Figure 2: TOE Boundary

An actual deployment includes many ITLM agents and might also include multiple Runtime Servers. There is always exactly one Administration Server.

The agents communicate to the Runtime Server through the Runtime Web Server using SSL. The agent SSL implementation is from the IBM Global Security Kit (GSKit) library package, which is part of the TOE. The Runtime Web Server communicates to the WebSphere Application Server (WAS) using a protected connection. WAS passes the information to the Runtime Server. The Runtime Server stores the software usage record

from the agent into the Runtime Database. The Runtime Server is a Java application that runs in the context of WAS and uses the Java Runtime Environment (JRE).

The Runtime Server System also supports a web browser whose functionality is limited to being able to download a standard agent installation image. No authentication is required for downloading this image.

The Runtime Command-Line Interface (CLI) allows an OS system administrator to manage the passwords of the database and Runtime Server, and to start/stop the Runtime Server. The dotted line from the Runtime CLI to WAS indicates the start/stop function of the Runtime CLI. The Runtime CLI is protected by the operating system (OS) so that only an OS administrator can use this interface.

The Runtime Server communicates to the Administration Server through the Administration Web Server using SSL if configured to do so (see section 2.3.4.2). The Administration Web Server communicates to WAS using a protected connection. WAS passes the information to the Administration Server. The Administration Server stores the software usage record (which originates from the agent) from the Runtime Server into the Administration Database. The Administration Server is a Java application that runs in the context of WAS and uses the JRE.

The Administration Server supports a GUI interface (via a web browser) where the management functions of the TOE are performed. These management functions include creating and managing organizations, creating and managing users (including assigning users to organizations and roles within the organizations), changing administrator passwords, and changing Runtime Server communication passwords. The interface requires an administrator (user) of the TOE to log into the Administration Server before performing any tasks. The GUI uses SSL to protect the communications between the browser and the Administration Server.

After the user has logged into the Administration Server, a session tracking ID is generated to associate the HTTPS request that will follow with the session just started. The session ID is then passed back and forth in the transactions that follow so that each transaction can be associated with the user login instance.

Through the Administration Server's GUI interface, the Administration Server controls access to the software usage records contained in the database. Each user has a profile that specifies the organizations in which he or she is a member and the roles he or she has within each organization. Access to the software usage records in the database is dependent on the organizations and roles of the user.

The Administration Server System contains the Administration CLI. This CLI allows an OS system administrator to maintain (backup/restore) the ITLM configuration and perform other maintenance and problem determination tasks. The CLI can also start and stop the Administration Server. The dotted line from the Administration CLI to WAS indicates the start/stop function of the Administration CLI. The Administration CLI is protected by the operating system (OS) so that only an OS administrator can use this interface.

In the full commercial version of the TOE (IBM Tivoli License Compliance Manager, version 2.2, Fix Pack 1—see section 2.5.3.1), the Administration Server System contains a Catalog Manager GUI and CLI which are used to maintain the software catalog information in the database (note that use of the available Catalog Manager is optional and includes no security functionality). The catalog includes information about the products that can be monitored and the signatures that are used by the agent to detect the presence and use of products on monitored computers. The Catalog Manager is in the TOE environment.

The agent, Runtime Server, and Administration Server use configuration information contained in configuration files to securely configure themselves. The standard method for updating these files is through the use of a text editor.

2.5.2 TOE environment

The hardware and the operating system on which the servers and agents run are part of the TOE environment.

The TOE environment includes a number of components that are required in order for the product to work in its evaluated configuration.

The following required components are packaged with IBM Tivoli License Compliance Manager, version 2.2, Fix Pack 1:

- IBM WebSphere Application Server (WAS) version 6.0
- IBM WebSphere Application Server plug-in, version 6.0
- IBM HTTP Web server, version 6.0
- IBM DB2 Universal Database, Enterprise Edition, version 8.2

The following optional component is packaged with IBM Tivoli License Compliance Manager, version 2.2, Fix Pack 1[2]

- Catalog Manager, version 2.2. Catalog Manager also requires Java Runtime Environment (JRE), version 1.4 as a pre-requisite.

The following elements are assumed to exist in the TOE environment and are not packaged with IBM Tivoli License Compliance Manager, version 2.2, Fix Pack 1:

- a Web browser
- an application to read the PDF documentation files

In addition, the TOE environment is responsible for having a PKI certificate infrastructure to support the secure communication feature in the evaluated configuration. The PKI infrastructure must supply all PKI tasks (for example, generating certificates, reissuing expired certificates, and revoking certificates).

2.5.3 Evaluated configuration

Although the product supports several additional languages, the evaluation is based on the English version of IBM Tivoli License Compliance Manager, version 2.2, Fix Pack 1.

[2] (Note: Catalog Manager is not packaged with the IBM Tivoli License Compliance Manager for IBM Software form of the TOE – see section 2.5.3.1)

2.5.3.1 Product forms

The product is available in two forms:

- a full commercial form that enables enterprise-wide monitoring and management of both IBM and non-IBM software products, including software products defined by an authorized administrator. This form of the product is called IBM Tivoli License Compliance Manager. (Specifically, this form of the TOE is IBM Tivoli License Compliance Manager, version 2.2, Fix Pack 1.)
- a subset form that tracks installation and use of specific IBM software products to enable reporting requirements associated with the sub-capacity pricing model. This form is called IBM Tivoli License Compliance Manager for IBM Software. (Specifically, this form of the TOE is IBM Tivoli License Compliance Manager for IBM Software, version 2.2, Fix Pack 1.)

The security functions are the same for both forms of the product. All components that are present in both forms of the product are identical.

2.5.3.2 IBM GSKit

The agent requires and includes IBM Global Security Kit (GSKit), a library package that implements SSL. The following GSKit versions are supported:

Supported Agent Operating System	GSKit Version
AIX	7.0.3.15
Linux	7.0.3.15
Sun Solaris	7.0.3.17
Windows	7.0.3.20

2.5.3.3 Evaluated platforms

The IBM Tivoli License Compliance Manager, version 2.2, Fix Pack 1 servers, databases, and catalog server are supported on a range of operating system platforms. The evaluated configuration of IBM Tivoli License Compliance Manager, version 2.2, Fix Pack 1 is supported on the following operating system platforms.

Tivoli License Compliance Manager servers (Runtime Servers and Administration Server):

- Windows Server 2003 Standard or Enterprise Edition, Windows 2000 Advanced Server, Windows 2000 Server
- IBM AIX 5.3 and 5.2
- HP/UX 11i
- Red Hat Enterprise Linux 4.0 and 3.0
- SUSE LINUX Enterprise Server 9 and 8
- Sun Solaris 10 and 9

Tivoli License Compliance Manager agent:

- Windows Server 2003 Standard Edition
- IBM AIX 5.3

- Sun Solaris 9
- Red Hat Enterprise Linux 4.0

2.5.3.4 Acquiring the TOE

The general availability (GA) component of the TOE (IBM Tivoli License Compliance Manager, version 2.2 or IBM Tivoli License Compliance Manager for IBM Software, version 2.2) is purchased through established IBM product distribution channels. The GA product may be acquired either by secure electronic download or by requesting and installing from physical media (CD-ROM or DVD).

In the evaluated configuration, the fix pack component of the TOE (Fix Pack 1) must be acquired by requesting a CD-ROM through established IBM Tivoli support channels.

2.5.3.5 Installation requirements

- The TOE must be configured to use the database authentication mechanism.
- The TOE must be configured to use SSL with client authentication to protect data flow between the agents and the Runtime Server.
- The Runtime Server Agent Self Update feature must be disabled.

2.5.3.6 TOE environment components requirements:

- The web server must be set up to use the HTTPS protocol for communication with the Administration Server.
- Server certificates for SSL use must not be created using the self-signed certificate option. Certificates must be requested from a well-known or customer certificate authority (CA).
- The WebSphere Application Server global security settings must be set to require authentication before stopping or uninstalling an application server running within a cell.

3 TOE Security Environment

This section describes the general environment in which the TOE is expected to be used. The description is made in the form of assumptions about intended use, physical, and personnel aspects that are handled by the environment rather than by the TOE and that support the correct and secure operation of the TOE. This section also outlines threats, potential for attacks, and applicable policies that are enforced.

The underlying operating system on which the TOE is installed is expected to operate according to its specification and to have no security-critical side effects on the operation of the TOE. The operating system is not part of the TOE, but is considered to belong to the TOE environment; however, the functionality provided by the TOE relies on proper support from the operating system.

The underlying hardware on which the TOE is installed is expected to operate according to its specification and to have no security-critical side effects on the operation of the TOE. Hardware is not part of the TOE but is considered to belong to the TOE environment; however, the functionality provided by the TOE relies on proper hardware support.

In the evaluated configuration, the TOE requires the use of X.509v3 certificates for client authentication of the agent components using SSL. At least some aspects of certificate generation, signing, and management take place in the environment. It is assumed that the certificates have been generated in a secure and proper manner and are properly formed.

The administrators of the TOE are considered to be trustworthy. The TOE does not provide protection against a misbehaving administrator trying to configure the TOE to an insecure state. Administrators are well trained, thus reducing the risk of accidental mis-configuration leading to insecure states

3.1 Assumptions

This section describes assumptions about the environment in which the TOE will be used or is intended to be used. These assumptions include the following:

- Assumptions about the intended usage of the TOE.
- Assumptions about the environment of use of the TOE, including physical and personnel aspects.

3.1.1 Intended usage of the TOE

A.CRYPTO It is assumed that cryptographic keys and certificates used in authentication between components of the TOE are generated, managed, and stored in a secure way to ensure their confidentiality and integrity.

A.ENV_CONFIG It is assumed that the TOE environment is configured and well-managed in accordance with the administrator documentation to protect the TOE and its data, including when data is transferring from the Runtime Server to the Administration Server and when data is transferring between each server and its associated database.

A.TOE_CONFIG It is assumed that the TOE is configured and operated in accordance with the administrator documentation and that the agent software is installed using a secure deployment method for

the intended environment as documented in the administrator documentation.

A. DATA_INT It is assumed that administrators will ensure the integrity and confidentiality of data transferred between the Runtime Server and the Administration Server.

Application note: Administrators are trusted to make a sound decision about whether to configure the TOE to use SSL encryption and/or client authentication to ensure integrity and confidentiality. In some deployments, administrators might legitimately conclude that selecting this option is necessary (see section 2.2.1.3).

3.1.2 Environment of use of the TOE

3.1.2.1 Physical aspects

A.PHYSICAL It is assumed that all machines housing components of the TOE and components in the TOE environment on which the TOE relies are protected against unauthorized physical access and modification.

A.TIME It is assumed that a reliable time function is provided by the TOE environment to support the inactivity timeout function.

3.1.2.2 Personnel aspects

A.TOE_ADMIN It is assumed that TOE Administrators are competent and trustworthy to perform their tasks, and that organizational procedures and policies are sufficient to ensure that they are held accountable for their security-relevant actions.

A. ENV_ADMIN It is assumed that TOE Environment Administrators (e.g., individuals who have administrator privileges on the administration and runtime databases) are competent and trustworthy to perform their tasks.

A.COOP It is assumed that all non-administrator users in the environment are part of a well-managed and cooperative user community.

3.2 Threats

This section describes the threats to be countered by the TOE and its environment in terms of human threat agents and assets potentially subject to attacks.

3.2.1 Assets and threat agents

The **assets** to be protected by the TOE security functions are described below.

Asset	Description	Type of Data
Software Usage Records	Software installation and use data collected by the agents deployed on monitored computers, transmitted through the Runtime Server and the Administration Server, and stored in the Administration Server DB2 database.	User data

The **threat agents**, their attack potential, resources, and level of expertise are described below.

Threat Agent	Description
User	This threat agent is a legitimate human user in the TOE environment. Non-administrator TOE users have no authorized access at all to the TOE resources, but may attempt to access assets protected by the system. This threat agent is considered to have a low motivation to attack, limited resources, and limited opportunity, but might have a high level of expertise and competence.
Non-authorized administrator	This threat agent is a legitimate human administrator of the TOE with access to specific data or function of TOE, but without authorized access to other data or function functions of TOE. An administrator with limited access might attempt to access assets he or she is not authorized to access. This threat agent is considered to have a low motivation to attack and limited resources, but has a high level of expertise, competence, and opportunity.

It is assumed that both sets of potential attackers come from a well-managed user community in a non-hostile working environment. The TOE is not intended to be used in an environment in which protection against determined or sophisticated attacks is required.

3.2.2 Threats addressed by the TOE

These threats must be countered by security functions implemented by the TOE.

T.BYPASS A user or a non-authorized administrator might bypass TSP enforcement functions to access data or resources protected by the TOE by penetrating or manipulating portions of the TOE.

T.ACCESS An administrator might see software usage records for an organization in which he or she does not play a role.

T.DATA_INT A user or a non-authorized administrator might compromise the integrity or confidentiality of data being transferred from the agent to the Runtime Server.

T.DATA_PERSIST A legitimate user (because of user error), an attacker (maliciously), or a system error might cause loss of data by interfering with successful completion of the transfer of data from one TOE component to another component (successful completion means successful write to the target database).

3.2.3 Threats addressed by the TOE Environment

The threats described below must be countered by security mechanisms implemented by the TOE environment.

TE.PASS A user or a non-authorized administrator might bypass the TOE to access data or resources protected by the TOE by attacking the underlying operating system or database.

TE.SPOOF	A user or a non-authorized administrator might record or modify user data on an inter-TOE communication link or on a communication link between TOE and non-TOE components in order to obtain unauthorized access to user data or to manipulate user data to be recorded.

3.3 Organizational Security Policies

There are no organization security policies for the TOE.

4 Security Objectives

The security objectives provide a concise statement of the intended response to the security problem. It will describe which security needs will be addressed by the TOE and which will be addressed by the TOE environment, in the form of a statement of security objectives.

4.1 Security Objectives for the TOE

O.AUTHENTICATE The TOE must ensure that administrators are identified and authenticated before being granted access to usage records in the Administrator Server database.

Application Note: User data is stored as usage records in the Administration Server database, which is in the TOE environment.

O.AUTHORIZE The TOE must provide the ability to specify and manage access rights to administrative functions and objects managed by the TOE.

O.ACCESS The TOE must ensure that access by TOE administrators to software usage data in the Administration Server database is managed according to organization and role

O.DATA_INT The TOE must ensure that the integrity and confidentiality of user data transferred from the agent to the Runtime Server is ensured.

O.DATA_PERSIST The TOE must ensure that loss of user data as a result of transfer from one TOE component to another TOE component is prevented.

4.2 Security Objectives for the non-IT Environment

Some security needs are beyond the capability of the TOE to be adequately satisfied without support from the TOE operational environment. Non-IT requirements on the TOE environment are listed in this section.

OE.,CLIPROT The environment must protect the CLI environment.

OE.COOP Non-administrator users in the TOE environment must be well-managed and cooperative.

OE.CRYPTO Cryptographic keys and certificates used in authentication between components of the TOE must be generated, managed, and stored in a secure way to ensure their confidentiality and integrity.

OE.DATA_INT TOE administrators must ensure the integrity and confidentiality of data transferred between the Runtime Server and the Administration Server based on the nature of the deployment of these components.

OE.ENV_ADMIN TOE environment administrators must be competent and trustworthy to perform their tasks.

OE.ENV_CONFIG TOE environment administrators must ensure that components in the TOE environment are installed and configured in the evaluated configuration as defined in the administrator

	guidance.
OE.PHYSICAL	All machines housing the components of the TOE must be protected against unauthorized physical access and modification.
OE.SPOOF	The TOE environment must ensure that unauthorized users cannot record or modify user data on inter-TOE communication links.
OE.TOE_ADMIN	TOE administrators must be competent and trustworthy to perform their tasks.
OE.TOE_CONFIG	TOE administrators must ensure that the TOE is installed and configured in the evaluated configuration as defined in the administrator guidance.

4.3 Security Objectives for the IT Environment

IT requirements on the TOE environment are listed below.

OE.TIME	The TOE environment must provide a reliable time source to support the session timeout feature.

5 IT Security Requirements

This chapter contains the Security Functional Requirements (SFRs) and Security Assurance Requirements (SARs) that must be satisfied by the TOE. These requirements consist of functional components from Part 2 of the CC and Evaluation Assurance Level (EAL) 2 assurance components from Part 3 of the CC, augmented with ALC_FLR.1 for flaw remediation. In addition the Security Functional Requirements (SFRs) for the TOE IT environment are described.

5.1 TOE Security Functional Requirements

This section identifies and specifies the SFR components that the TOE is intended to meet for the purposes of this CC evaluation. All of these SFR components are chosen from Part 2 of the CC to directly or indirectly (i.e., via a functional component dependency) satisfy the security objectives for the TOE, summarized in Table 1.

The following table gives an overview of the functional components from the Common Criteria Part 2 that are relevant for this TOE.

Table 1: TOE Security Functional Requirements

Component	Component Name
FDP_ACC.1	Subset access control
FDP_ACF.1	Security attribute based control
FDP_ITT.1	Basic internal transfer protection
FIA_ATD.1	User attribute definition
FIA_SOS.1	Verification of secrets
FIA_UAU.2	User authentication before any action
FIA_UID.2	User identification before any action
FMT_MOF.1	Management of security functions behavior
FMT_MSA.1a	Management of security attributes (Super Administrator)
FMT_MSA.1b	Management of security attributes (user)
FMT_MSA.3	Static attribute initialization
FMT_SMF.1	Specification of management functions
FMT_SMR.1	Security management roles
FTA_SSL.3	TSF-initiated termination

Operations that are completed on the SFR components are indicated throughout this section by the use of bold text.

Iterations are indicated by adding an alpha identifier, like FMT_MSA.1a and FMT_MSA.1b.

Application notes have been added after some requirements are identified; the string "**Application note**" indicates such information.

5.1.1 Class FDP User data protection

5.1.1.1 FDP_ACC.1 Subset access control

FDP_ACC.1.1 The TSF shall enforce the **Management SFP** on **users as subjects and usage records as objects, among the following operations performed by the subject on the objects:**

- **Produce Reports**
- **Manage Batch Report**
- **Manage Licenses**
- **Assign Licenses**
- **Define Product Properties**
- **Schedule Software Scans**
- **Manage Resources**
- **Manage Complex Products**
- **Manage Infrastructure**
- **Manage Organizations**
- **Manage Access**
- **Define Custom Fields**
- **Export IBM Use**

5.1.1.2 FDP_ACF.1 Security attribute based access control

FDP_ACF.1.1 The TSF shall enforce the **Management SFP** to objects based on the following: **the roles and privacy policy of a user in an organization vs. the organization to which the software usage record belongs**.

FDP_ACF.1.2 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: **user can only perform the allowable tasks of the roles and privacy policy they have been assigned in the organization to which the software usage record belongs**.

FDP_ACF.1.3 The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: **the Super Administrator can act in any role within any organization**.

FDP_ACF.1.4 The TSF shall explicitly deny access of subjects to objects based on ~~the~~ **no additional rules**.

5.1.1.3 FDP_ITT.1 Basic internal transfer protection

FDP_ITT.1.1 The TSF shall enforce the **Management SFP** to prevent the **disclosure, modification, or loss of use** of user data when it is transmitted between physically-separated parts of the TOE.

Application note: Internal transfer protection to prevent **disclosure and modification** of user data (software usage records) is established from the agent to the Runtime Server using the GSKit communications software to authenticate and implement SSL encryption.

Application note: Internal transfer protection to prevent **loss of use** is provided by a guaranteed data delivery mechanism to persist data on the sending component until confirmation is received that the data has been successfully written to the database in the target component. This mechanism is used for both agent to Runtime Server, and Runtime Server to Administration Server software usage records transfer.

5.1.2 Class FIA Identification and authentication

5.1.2.1 FIA_ATD.1 User attribute definition

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

- **user name**
- **password**
- **organizations**
- **role within each organization**
- **privacy policy within each organization**

Application Note: An administrator (user) may have different roles and privacy policies in different organizations. Note that Runtime Servers also have passwords, but they are not considered users in this context.

5.1.2.2 FIA_SOS.1 Verification of secrets

FIA_SOS.1.1 The TSF shall provide a mechanism to verify that secrets meet **the password policy constraints, defined by the following attributes:**

- **minimum length 8 characters**
- **minimum 2 non-alphabetic characters**
- **maximum 2 consecutively repeated characters**

Application Note: Note that the password policy applies to passwords assigned to administrators (users) and to communication passwords assigned to Runtime Servers to identify and authenticate to the Administration Server.

The claimed SOF for the authentication mechanism is **SOF-basic**. This claim is based on the setting above for the verification of secrets when creating or changing a user password.

5.1.2.3 FIA_UAU.2 User authentication before any action

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

Application note: This applies to the authentication of administrators (users) to perform actions through the Administration GUI (via a web browser) and to the authentication of Runtime Servers to the Administration Server.

5.1.2.4 FIA_UID.2 User identification before any action

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.

Application note: This applies to the identification of administrators (users) to perform actions through the Administration GUI (via a web browser) and to identification of the Runtime Servers to the Administration Server.

5.1.3 Class FMT Security management

5.1.3.1 FMT_MOF.1 Management of security functions behavior

FMT_MOF.1.1 The TSF shall restrict the ability to **modify the behavior of** the function **authentication** to **the Super Administrator**.

Application note: This applies to the authentication of administrators (users) to perform actions through the Administration GUI (via a web browser) and to authentication of Runtime Servers to the Administration Server.

5.1.3.2 FMT_MSA.1a Management of security attributes (Super Administrator)

FMT_MSA.1.1 The TSF shall enforce the **Management SFP** to restrict the ability to **modify** the security attributes:

- **organizations assigned for each user**
- **roles assigned within each of organization for each user**
- **privacy policy assigned within each organization for each user**
- **password assigned for each user**

to **the Super Administrator**.

5.1.3.3 FMT_MSA.1b Management of security attributes (user)

FMT_MSA.1.1 The TSF shall enforce the **Management SFP** to restrict the ability to **modify** the security attributes:

- **own password**

to **each user**.

Application note: Each user (note that all users of the TOE are Administrators with specific roles as described in section 6.1.2.1) can change his or her own user password.

5.1.3.4 FMT_MSA.3 Static attribute initialization

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

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

Application note: Apart from the Super Administrator, there are no other users in a default installation of the TOE. New users created will have no organizational association and no roles assigned to any of these organizations.

5.1.3.5 FMT_SMF.1 Specification of management functions

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

- **organization management**
- **user management**

- **user password management**
- **Runtime Server user name management**
- **Runtime Server communication password management**
- **Runtime Server database password management**
- **Runtime Server truststore file (key.jks) password management**
- **Administration Server database password management**

Application note: In addition, the TOE provides another aspect of server password management to re-encrypt all passwords in the password.properties file on the Runtime Server and the Administration Server.

5.1.3.6 FMT_SMR.1 Security roles

FMT_SMR.1.1 The TSF shall maintain the roles:

- **Super Administrator**
- **Administrator**
- **Procurement Manager**
- **Software Resources Manager**
- **License Administrator**
- **System Resources Manager**
- **Procurement and Licensing Manager**

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

Application note: Each administrator (user) may be assigned to one or more organizations. Each of these organizations will allow the administrator to work as one or more roles. The set of roles depends on the organization; for example, the administrator may be Administrator for one organization, but License Administrator or System Resources Manager for another organization.

5.1.4 Class FTA TOE Access

5.1.4.1 FTA_SSL.3 TSF-initiated termination

FTA_SSL.3.1 The TSF shall terminate an inactive session after ~~a~~ **60 minutes**.

Application Note: Sixty minutes of inactivity is the default timeout period for user sessions via the Administration Server GUI (via a web browser).

5.2 TOE Security Assurance Requirements

The target assurance components for this TOE are those for EAL2 augmented with ALC_FLR.1, as specified in Part 3 of the CC. The following table provides an overview of the assurance components that form the assurance level for the TOE:

Table 2: TOE Security Assurance Requirements

Assurance class	Assurance components
Configuration management	ACM_CAP.2 CM capabilities
Delivery and operation	ADO_DEL.1 Delivery
	ADO_IGS.1 Installation, generation, and start-up
Development	ADV_FSP.1 Functional specification
	ADV_HLD.1 High-level design
	ADV_RCR.1 Representation correspondence
Guidance documents	AGD_ADM.1 Administrator guidance
	AGD_USR.1 User guidance
Life cycle support	ALC_FLR.1 Basic flaw remediation
Tests	ATE_COV.1 Coverage
	ATE_FUN.1 Functional tests
	ATE_IND.2 Independent testing
Vulnerability assessment	AVA_SOF.1 Strength of TOE security functions
	AVA_VLA.1 Vulnerability analysis

5.3 Security Requirements for the IT Environment

This section identifies and specifies the environmental SFR components that the environment of the TOE is intended to meet for the purposes of this CC evaluation. All of these SFR components are chosen to directly or indirectly (i.e., via a functional component dependency) satisfy the IT environmental security objectives for the TOE.

Table 3: Security Functional Requirements for the IT Environment

Component	Component Name
FPT_STM.1	Reliable time stamps

5.3.1 IT Security Requirements for the underlying Operating System

5.3.1.1 FPT_STM.1 Reliable time stamps

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

6 TOE Summary Specification

This chapter provides a description of the TOE security functions and TOE assurance measures that meet the TOE security requirements specified in Chapter 5.

6.1 TOE Security Functions

6.1.1 Identification and Authentication (SF.IA)

6.1.1.1 User to Administration Server identification and authentication (SF.IA.1)

Administrators interact with the TOE via a GUI (Administration Server only, via a web browser). When using the Administration Server GUI, users are authenticated by providing a login and password through a web browser.

The product architecture is such that the TOE is a WebSphere application. The method used to send the login credential is the standard HTTP form. The administrator login page is created by Tivoli License Compliance Manager and contains the keyword “login”. When the user enters login information and submits the page, the WebSphere Application Server parses the html, finds the “login” keyword, and hands the data to Tivoli License Compliance Manager as a Request object. Tivoli License Compliance Manager then performs identification and authentication on the data contained in the Request object and returns a Response object to the WebSphere Application Server, and from there back to the web browser.

Although the authentication mechanism is pluggable and may support different mechanisms (such as LDAP authentication), only the default custom authentication mechanism, with credentials stored in the Tivoli License Compliance Manager database is considered to be the evaluated configuration.

When the custom Tivoli License Compliance Manager database authentication mechanism is used, users’ passwords are not encrypted; they are digested (or hashed) using SHA-1 and a salt. The SHA-1 algorithm is designed to be irreversible, or at least prohibitively expensive to reverse. The salt defends from dictionary attacks.

6.1.1.2 Password policy enforcement (SF.IA.2)

The TOE enforces the following password length and composition requirements on administrator (user) passwords and on Runtime Server communication passwords:

- mimimumPasswordLength parameter - minimum password length is 8 characters
- maxNumberCharsRepeated parameter - maximum consecutively repeated characters is 2 characters
- minimumNumberNonAlphChars parameter - minimum non-alphabetic characters is 2 characters

Application note: The allowable character set for the Runtime Server communication password is: A-Z, a-z, 0-9, +, -, |, =, *. (This limitation is imposed by the Runtime Server CLI.) The allowable character set for user passwords is limited only by the installed character set.

6.1.1.3 Agent to Runtime Server identification and authentication (SF.IA.3)

In the evaluated configuration, agents are configured to use SSL encryption with client authentication when connecting to the Runtime Server in order to transfer software

usage records. The TOE uses the IBM Global Security Kit (GSKit) library for the implementation of the SSL protocol and its underlying cryptographic functions at the agent side.

6.1.1.4 Runtime Server to Administration Server identification and authentication (SF.IA.4)

If Tivoli License Compliance Manager is configured at install time (or changed later) to require Runtime Server to Administration Server authentication, the Runtime Server authenticates to the Administration Server using a user name and password. This authentication is initiated by the Runtime Server when the Runtime Server has information to pass on to the Administration Server without any human interference.

Passwords are selected at installation time and can be changed by the Super Administrator whenever decided.

If Tivoli License Manager is configured at install time (or changed later) to NOT require Runtime Server to Administration Server authentication, the Runtime Server identifies itself to the Administration Server by sending its user name only.

6.1.1.5 Inactivity timeout (SF.IA.5)

The Administration Server supports a per-session inactivity timeout for user login sessions. A session is identified by a session tracking ID composed of 48 bytes including three pseudorandom numbers based on seeds that ensure the session ID is unique across multiple logins by the same user. If a session has been idle for a configured time interval, the session is terminated. The time interval is set using the sessionTimeout parameter in the system.properties file on the Administration Server. By default, the time interval is set to 60 minutes.

6.1.2 Access Control (SF.ACCESS)

6.1.2.1 Administration Server GUI

Tivoli License Compliance Manager implements a role-based Access Manager, which controls access to specific operations using the concept of roles. The Access Manager gets roles information from the internally-configured database tables. An administrator's role is specific to an organization. An administrator can play different roles in different organizations. It is not possible to add roles or modify the definition of existing roles.

These are the defined user roles:

Super Administrator	The Super Administrator can run any task on the Administration Server, and is the only role that can manage organizations and user accounts. There is exactly one Super Administrator per installation. The Super Administrator, named tlmroot, is created during product installation. The Super Administrator cannot be deleted and its profile cannot be modified.
Administrator	An Administrator can run any task on the Administration Server except managing organizations and user accounts. It is the responsibility of the Administrator to define the topology of the licensing environment and to define the users. Users with the Administrator role do not perform any security-relevant functions except that they can manage their own passwords.
Procurement Manager	A Procurement Manager is responsible for procuring licenses. Procurement Managers own the information necessary for

	decision-making regarding software procurement within their organization. Users with this role can manage contracts, procure licenses, and work with reports. Users with the Procurement Manager role do not perform any security-relevant functions except that they can manage their own passwords.
Software Resources Manager	A Software Resources Manager is responsible for working with reports to define resource requirements. Users with the Software Resources Manager role do not perform any security-relevant functions except that they can manage their own passwords.
License Administrator	A License Administrator is responsible for defining product compliance and distributing licenses. License Administrators configure entitlements to match software contracts, determine the computers or groups of computers that must have access to licenses, and generate reports to assist with software acquisition analysis and planning. Users with the License Administrator role do not perform any security-relevant functions except that they can manage their own passwords.
System Resources Manager	A System Resources Manager is responsible for ensuring that the software and systems are running correctly. System Resources Managers are responsible for the deployment of software products. They ensure that the complex products that are installed can be accurately monitored, that the infrastructure and agents are deployed, and that the total solution is scalable, secure, and easy to maintain. The System Resources Manager can schedule software scans, manage and servers and agents, and define the mappings of complex products on monitored computers. Users with the System Resources Manager role do not perform any security-relevant functions except that they can manage their own passwords.
Procurement and Licensing Manager	The Procurement and Licensing Manager role is a combination of Procurement Manager and License Administrator, combining the responsibilities of both roles. Users with the Procurement and Licensing Manager role do not perform any security-relevant functions except that they can manage their own passwords.

According to their assigned roles in the organization, administrators are able to perform specific tasks through the Administration Server GUI. The Super Administrator can perform all tasks, including the management tasks of creating and managing organizations and users. The other roles perform a subset of tasks, as shown in the table below.

Table 4: Roles and Tasks

	Super Administrator	Administrator	Procurement Manager	Software Resources Manager	License Administrator	System Resources Manager	Procurement and Licensing Manager
Produce Reports	Yes	Yes	Yes	Yes	Yes	No	Yes
Manage Batch Report	Yes	Yes	Yes	Yes	Yes	No	Yes
Manage Procurement	Yes	Yes	Yes	No	No	No	Yes
Assign Licenses	Yes	Yes	No	No	Yes	No	Yes
Define Product Properties	Yes	Yes	No	No	Yes	No	Yes
Schedule Software Scans	Yes	Yes	No	No	No	Yes	No
Manage Resources	Yes	Yes	No	No	No	Yes	No
Manage Complex Products	Yes	Yes	No	No	No	Yes	No
Manage Infrastructure	Yes	Yes	No	No	No	Yes	No
Manage Organizations	Yes	No	No	No	No	No	No
Manage Access	Yes	No	No	No	No	No	No
Define Custom Fields	Yes	Yes	Yes	No	No	No	Yes
Export IBM Use	Yes	Yes	No	No	Yes	No	Yes

Note: the security-relevant management tasks of the TOE are associated with the Super Administrator (except that each user can manage his or her own password). These are described as part of SF.MGMT.

6.1.3 Data Protection During Transfer (SF.DATA)

6.1.3.1 Agent to Runtime Server secure channel (SF.DATA.1)

The TOE includes the Global Security Kit (GSKit) library package for the agents that implements SSL to provide encryption and authentication between agents and the Runtime Server.

6.1.3.2 Data availability protection (SF.DATA.2)

Interruptions of data flow between components can occur in the event of network failure, database problems, or system crash. In order to prevent loss of software usage data if one of these interruptions to normal behavior occurs, Tivoli License Compliance Manager uses a guaranteed data delivery mechanism to persist data on the sending component until confirmation is received that the data has been successfully written to the database in the target component.

The mechanism implemented is a synchronous system:

1. The client sends the data in the body of the HTTPS request.

2. Each packet is marked with an ID, and it is processed server side transactionally. The server processes the data, and only at the successful end of the transaction (when the data has been written to the database associated with the server) returns a positive acknowledgment to the client in the body of the HTTPS response.
3. If the client receives the positive acknowledgement, the client deletes the data; else, the client schedules a resend.

Using the persistence confirmation mechanism prevents data loss during transmission from one component of the TOE to another.

6.1.4 Management of Security Functions (SF.MGMT)

6.1.4.1 Management via the Administration Server GUI

Management of security functions is performed by the Super Administrator using the Administration Server GUI. While other users can use the Administration Server GUI to perform other tasks, the security management functions are exclusive to the Super Administrator, with the single exception that users are allowed to change their own passwords.

The Super Administrator can perform the administrative tasks defined here using the TOE (that is, through the Administration Server GUI).

6.1.4.1.1 Create and manage organizations (SF.MGMT.1)

The Super Administrator can add or remove organizations from the TOE.

6.1.4.1.2 Create and manage users (SF.MGMT.2)

The Super Administrator can add, remove, or change the profiles of other users, including changing a user's organizations, roles, privacy policies, or passwords.

6.1.4.1.3 Change own user passwords (SF.MGMT.3)

Each user can change his or her own password. New passwords must comply with the password policy enforced by the TOE.

6.1.4.1.4 Change Runtime Server communication password (SF.MGMT.4)

The Super Administrator can change the Runtime Server to Administration Server communication password. New passwords must comply with the password policy enforced by the TOE.

Note that if the Runtime Server communication password is changed on the Administration Server, the password must also be changed on the Runtime Server using the Runtime Server CLI (see section 6.1.4.2.1).

6.1.4.2 Management via the Runtime Server CLI

The Runtime Server provides a command line interface (CLI) to manage the following security-relevant tasks. The command line can only be used by Windows users with administrator rights and UNIX user root from the UNIX Shell.

6.1.4.2.1 Change the Runtime Server communication password (SF.MGMT.5)

The Runtime Server to Administration Server communication password can be changed using the **rtpasswd** command. Executing the **rtpasswd** command requires entering the old password.

Note that if the Runtime Server communication password is changed on the Runtime Server, the password must also be changed on the Administration Server through the Administration Server GUI (see section 6.1.4.1.4).

6.1.4.2.2 Change the Runtime Server Database password (SF.MGMT.6)

The Runtime Server to Runtime Server Database password can be changed using the **dbpasswd** command. Executing the **dbpasswd** command requires entering the old password.

Note that if the Runtime Server to Runtime Server Database password is changed on the Runtime Server, the password for the tlmsrv user on the Runtime Server Database machine must also be changed to match.

6.1.4.2.3 Change the password used to open the truststore file (SF.MGMT.7)

The password used by the Runtime Server to open its truststore file (key.jks) can be changed using the **sslpasswd** command. Executing the **sslpasswd** command requires entering the old password.

Note that if the truststore file password is changed on the Runtime Server, the access password of the key.jks file must also be changed to match using the ikeyman application, which is bundled with the IBM HTTP Server.

6.1.4.2.4 Re-encrypt all passwords (SF.MGMT.8)

All passwords in the password.properties file on the Runtime Server can be re-encrypted using the AES algorithm by executing the **kstoreupdate** command. On the Runtime Server, this means that the database password, Runtime Server communication server password, and the trust store password are re-encrypted. New generated keystores are saved into the keys.jks file, all passwords are re-encrypted, and the accessing keys are replaced into the password.properties file. Keys temporarily stored in memory are refreshed.

Note that running **kstoreupdate** does not change any passwords; rather the command generates a new set of keys and re-encrypts all passwords stored in the password.properties file.

6.1.4.3 Management via the Administration Server CLI

The Administration Server provides a command line interface (CLI) to manage the following security-relevant tasks. The command line can only be used by Windows users with administrator rights and UNIX user root from the UNIX Shell.

6.1.4.3.1 Change the Administration Server Database password (SF.MGMT.9)

The Administration Server to Administration Server Database password can be changed using the **dbpasswd** command. Executing the **dbpasswd** command requires entering the old password.

Note that if the Administration Server to Administration Server Database password is changed on the Administration Server, the password for the tlmsrv user on the Administration Server Database machine must also be changed to match.

6.1.4.3.2 Re-encrypt all passwords (SF.MGMT.10)

The database password in the password.properties file on the Administration Server can be re-encrypted using the AES algorithm by executing the **kstoreupdate** command.

Note that running **kstoreupdate** does not change any passwords; rather the command generates a new set of keys and re-encrypts passwords stored in the password.properties file.

6.1.4.3.3 Define and manage runtime server security attributes (SF.MGMT.11)

If Tivoli License Manager is configured to NOT require Runtime Server to Administration Server authentication, the Runtime Server identifies itself to the Administration Server by sending its user name only. The Runtime Server user name is configured on the Runtime Server when the software is installed, and configured on the Administration Server when the Runtime Server is registered to the Administration Server. The Super Administrator, Administrator, or System Resources Manager registers the Runtime Server to the Administration Server.

If Tivoli License Manager is configured to require Runtime Server to Administration Server authentication, the Runtime Server identifies and authenticates itself to the Administration Server by sending its user name and password. The Runtime Server user name and password are configured on the Runtime Server when the software is installed, and configured on the Administration Server when the Runtime Server is registered to the Administration Server. The Super Administrator, Administrator, or System Resources Manager registers the Runtime Server to the Administration Server.

Once configured on the Runtime Server, the user name cannot be changed. If the user name is incorrectly configured on the Administration Server, it can be corrected.

The Runtime Server to Administration Server password can be changed. See sections 6.1.4.1.4. and 6.1.4.2.1.

6.2 Assurance Measures

The assurance requirements for the EAL 2 augmented assurance level are met by the following assurance measures. These assurance requirements provide, primarily via review of supplied evidence, independent confirmation that these actions have been competently performed. They also include the following independent, third-party analysis:

- Confirmation of effective configuration management
- Confirmation of product delivery and installation procedures
- Confirmation that the guidance documentation is adequate
- Verification of a sample of the vendor functional testing
- Verification of the developer's analysis for vulnerabilities and resistance against obvious penetration attacks
- Independent functional testing

To define the assurance measures claimed to satisfy the security assurance requirements specified in chapter 5.2, a mapping is provided between the security assurance requirements (SARs) and the assurance measures, which are intended to satisfy the assurance requirements. As shown in Table 5, the Assurance Measures are provided in the form of references to the relevant and appropriate document associated with each requirement.

Table 5: TOE Assurance Measures

SAR	Assurance Measure
ACM_CAP.2	M.CAP CMVC, Global Storage Architecture (GSA), Test Tracking Tool (TTT), and Lotus Notes Team Rooms are used to manage configuration items. Configuration lists are provided by the Developer. A Configuration Management Systems roadmap document is provided.

SAR	Assurance Measure
ADO_DEL.1	M.DEL Secure methods for the delivery of the TOE by secure electronic download or by shipping of physical media in sealed packages using traceable shipping mechanisms are provided and documented by the Developer.
ADO_IGS.1	M.IGS Installation, generation, and start-up guidance for the evaluated configuration of the TOE is provided by the Developer. General product installation instructions are provided in the Planning, Installation, and Configuration Guide. Instructions specific to installing and configuration the evaluated configuration are provided in the Common Criteria Guide.
ADV_FSP.1	M.FSP A Functional Specification identifying and describing all security interfaces is provided by the Developer. This is a top-level document which provides pointers to information contained in a number of other documents.
ADV_HLD.1	M.HLD A High-level Design identifying and describing all subsystems and the internal and externally visible interfaces is provided by the Developer. This is a top-level document which provides pointers to information contained in a number of other design documents.
ADV_RCR.1	M.RCR Correspondence mapping between the TOE Summary Specification and the Functional Specification, and between the Functional Specification and the High-level Design is provided by the Developer.
AGD_ADM.1	M.ADM Guidance for TOE administrators describing how to administer the TOE in a secure manner is provided by the Developer. The documentation set includes an Overview; Administration Guide; Planning, Installation and Configuration Guide; Commands Reference; Data Dictionary; Problem Determination Guide; Security Management Guide; and Release Notes. In addition, as previously noted, the documentation for the TOE includes a Common Criteria Guide created for the evaluation configuration. The documentation set for the product also includes a Catalog Management Guide; the Catalog Manager is not a part of the TOE.
AGD_USR.1	M.USR There are no non-administrator users of the TOE.
ALC_FLR.1	M.FLR Appropriate management of potential security flaws in the TOE is implemented and documented by the Developer.
ATE_COV.1	M.COV Test Coverage Analysis is provided by the Developer.

SAR	Assurance Measure
ATE_FUN.1	M.FUN Testing of the TOE Security Functions was performed by the Developer on the range of platforms as defined by the ST. Test results are documented such that they can be repeated.
ATE_IND.2	M.IND The Evaluation Facility performed independent testing of a subset of the TOE security functions to determine whether the TOE behaves as specified, and to gain confidence in the developer's test results by performing a sample of the developer's tests.
AVA_SOF.1	M.SOF A Strength of Function Analysis is provided for the password policy mechanism. The SOF for the password policy is SOF-basic.
AVA_VLA.1	M.VLA A Vulnerability Analysis to determine the existence and exploitability of flaws and weaknesses in the TOE in the intended environment is provided by the Developer.

7 PP Claims

This Security Target does not claim conformance with any Protection Profile.

8 Rationale

The rationale chapter demonstrates how the security objectives of the TOE are met and how objectives, threats, and security functions relate to each other. The rationale section identifies which security functions contribute to which objectives and identify which threats are countered by the individual security functions.

8.1 Security Objectives Rationale

8.1.1 Security Objectives Coverage

The following table demonstrates how each security objective for the TOE correlates to at least one threat.

Table 6: TOE security objectives mapped to threats

TOE Objective	Threats
O.AUTHORIZE	T.BYPASS
O.AUTHENTICATE	T.BYPASS
O.ACCESS	T.ACCESS
O.DATA_INT	T.DATA_INT
O.DATA_PERSIST	T.DATA_PERSIST

The following table demonstrates how each security objective for the TOE environment correlates to at least one threat or assumption.

Table 7: TOE IT environment security objectives mapped to threats or assumptions

TOE Environment Objective	Threats or Assumptions
OE.TIME	A.TIME

Table 8: TOE non-IT environment security objectives mapped to threats or assumptions

TOE Environment Objective	Threats or Assumptions
OE.CLIPROT	TE.PASS
OE.COOP	A.COOP
OE.CRYPTO	A.CRYPTO
OE.DATA_INT	A.DATA_INT
OE.ENV_ADMIN	A.ENV_ADMIN
OE.ENV_CONFIG	A.ENV_CONFIG
OE.PHYSICAL	TE.PASS, A.PHYSICAL
OE.SPOOF	TE.SPOOF

TOE Environment Objective	Threats or Assumptions
OE.TOE_ADMIN	A.TOE_ADMIN
OE.TOE_CONFIG	A.TOE_CONFIG

8.1.2 Security Objectives Sufficiency

The following rationale provides justification that the security objectives are suitable to counter each individual threat and that each security objective tracing back to a threat, when achieved, actually contributes to the removal, diminishing or mitigation of that threat:

Table 9: Sufficiency of objectives countering threats

Threat	Rationale for Objective
T.BYPASS	The threat that a user or a non-authorized administrator might circumvent TSP enforcement functions by penetrating or manipulating the TOE is removed by the O.AUTHENTICATE and O.AUTHORIZE objectives, which ensure that any user who wishes to access user data protected by the TOE must first authenticate to the TOE, and then be authorized by the TOE to allow access.
T.ACCESS	The threat that an administrator might see software usage records for an organization in which he or she does not play a role is removed by the O.ACCESS objective, which ensures that software usage data in the Administration Server database is managed according to organization and role.
T.DATA_INT	The threat that a user or a non-authorized administrator might compromise the integrity or confidentiality of data being transferred from the agent to the Runtime Server is removed by the O.DATA_INT objective, which ensures the integrity and confidentiality of user data transferred from the agent to the Runtime Server.
T.DATA_PERSIST	The threat that a legitimate user (because of user error or system error) or an attacker (maliciously) might cause loss of data by interfering with successful completion of the transfer of data from one TOE component to another component (successful completion means successful write to the target database) is removed by the objective O.DATA_PERSIST, which states that the TOE must prevent loss of user data as a result of transfer from one TOE component to another TOE component.
TE.PASS	The threat that a user or non-authorized user might bypass the TOE to access data or resources protected by the TOE by attacking the underlying operating system or database is removed by the objective OE.CLIPROT, with support from the OE.PHYSICAL objective.
TE.SPOOF	The threat that a user or non-authorized user may record or modify user data on an inter-TOE communication link is addressed by OE.SPOOF.

The following rationale provides justification that the security objectives for the environment are suitable to cover each individual assumption, that each security objective for the environment that traces back to an assumption about the environment of use of the TOE, when achieved, actually contributes to the environment achieving

consistency with the assumption, and that if all security objectives for the environment that trace back to an assumption are achieved, the intended usage is supported:

Table 10: Sufficiency of objectives countering assumptions

Assumption	Rationale for Objective
A.COOP	OE.COOP requires that non-administrator users in the environment are part of a well-managed and cooperative user community. In a well-managed, cooperative user community, non-administrator users (who have no legitimate reason to access or interfere with TOE resources or user data protected by the TOE) will not attempt such access or interference.
A.CRYPTO	OE.CRYPTO requires that cryptographic keys and certificates are generated, managed, and stored in a secure way to ensure their confidentiality and integrity. Keys and certificates are used for SSL encryption and mutual authentication between components.
A.ENV_ADMIN	OE.ENV_ADMIN requires that TOE Environment Administrators are competent and trustworthy to perform their tasks, This impacts the underlying hardware, operating system, and the DB2 databases in the TOE environment that are essential for the successful working of the TOE.
A.ENV_CONFIG	OE.ENV_CONFIG requires that the TOE environment will be installed and configured in the evaluated configuration as defined in the administrator guidance, so that security functions in the TOE environment on which the TOE relies are available and correctly configured
A.PHYSICAL	OE.PHYSICAL requires that all machines housing the components of the TOE are protected against unauthorized physical access and modification, so that the TOE cannot be attacked in this manner.
A.TOE_ADMIN	OE.TOE_ADMIN requires that TOE Administrators are competent and trustworthy to perform their tasks, which means that they can be trusted to perform their security-related tasks as directed by the TOE guidance.
A.TOE_CONFIG	OE.TOE_CONFIG requires that the TOE will be installed and configured in the evaluated configuration as defined in the administrator guidance, so that the security functions in the TOE are available and correctly configured.
A.DATA_INT	OE.DATA_INT requires TOE administrators to ensure the integrity and confidentiality of data transferred between the Runtime Server and the Administration Server based on the nature of the deployment of these components.
A.TIME	OE.TIME requires that the TOE environment must provide a reliable time source to the TOE.

There are no organizational security policies to analyze against objectives.

8.2 Security Requirements Rationale

This section provides the rationale for the selection of security requirements.

8.2.1 Security Requirements Coverage

The following table demonstrates how each TOE security functional requirement correlates to at least one security objective for the TOE.

Table 11: Mapping TOE SFRs to objectives

SFR	Objective
FDP_ACC.1	O.AUTHORIZE
FDP_ACF.1	O.AUTHORIZE; O.ACCESS
FDP_ITT.1	O.DATA_INT; O.DATA_PERSIST
FIA_ATD.1	O.AUTHORIZE
FIA_SOS.1	O.AUTHENTICATE
FIA_UAU.2	O.AUTHENTICATE
FIA_UID.2	O.AUTHENTICATE
FMT_MOF.1	O.AUTHORIZE
FMT_MSA.1a	O.AUTHORIZE
FMT_MSA.1b	O.AUTHENTICATE
FMT_MSA.3	O. AUTHORIZE
FMT_SMF.1	O.AUTHORIZE
FMT_SMR.1	O.AUTHORIZE
FTA_SSL.3	O.AUTHENTICATE

8.2.2 Functional Requirements Sufficiency

The following rationale provides justification for each security objective for the TOE, showing that the TOE security functional requirements are suitable to meet and achieve the security objectives.

Table 12: Mapping TOE objectives to SFRs for the TOE

Objective	SFRs
O.AUTHENTICATE	FIA_UID.2 requires that a user must be identified before performing any TSF mediated operations. FIA_UAU.2 requires that a user must be authenticated before performing any TSF mediated operations. FIA_SOS.1 requires that user passwords and Runtime Server communication passwords must comply with the password policy. FMT_MSA.1b enables each user to change his or her own password, which is the attribute he or she must provide in order to authenticate to the TOE. FTA__SSL.3 requires that after 60 minutes of inactivity, a user's connection to the TOE is terminated; user must re-authenticate in order to re-establish connection to the TOE.
O.AUTHORIZE	FDP_ACC.1 defines the operations that require access control by the TOE, specifically by the Management SFP. FDP_ACF.1 requires that access is based on users' roles within an organization. FDP_ATD.1 requires that the following attributes be assigned to users: username, password, organizations, roles. FMT_MOF.1 defines that only the Super Administrator can manage

	authentication. FMT_MSA.1a defines that only the Super Administrator can modify security attributes of other users of the TOE (with the exception that (FMT_MSA.1b) each user can change his or her own password, as noted in the O.AUTHENTICATE mapping above). FMT_MSA.3 requires restrictive defaults for user attributes and specifies that only the Super Administrator can change the defaults. FMT_SMF.1 defines that the security management functions are: organizational management, user management, user password management, and Runtime Server communication password management, Runtime Server database password management, Runtime Server truststore password management, Administration Server database password management. FMT_SMR.1 defines security roles.
O.ACCESS	FDP_ACF.1 requires that access to data is based on users' roles within an organization
O.DATA_INT	FDP_ITT.1 requires the TOE to provide a secure channel between the agents and the Runtime Server in order to protect the integrity and confidentiality of data during transfer.
O.DATA_PERSIST	FDP_ITT.1 requires the TOE to provide a mechanism to protect against loss of data during transfer between distributed components of the TOE.

The following rationale provides justification for each security objective for the IT environment, showing that the security functional requirements for the IT environment are suitable to meet and achieve the security objectives.

Table 13: Mapping TOE environment objectives to SFRs for the IT environment

Objective	SFR
OE.CLIPROT	no dependency to any SFR
OE.COOP	no dependency to any SFR
OE.CRYPTO	no dependency to any SFR
OE.DATA_INT	no dependency to any SFR.
OE.ENV_ADMIN	no dependency to any SFR
OE.ENV_CONFIG	no dependency to any SFR
OE.PHYSICAL	no dependency to any SFR
OE.SPOOF	no dependency to any SFR.
OE.TOE_ADMIN	no dependency to any SFR
OE.TOE_CONFIG	no dependency to any SFR
OE.TIME	FPT_STM.1 requires the IT environment to provide a reliable time source.

8.2.3 Security Requirements Dependency Analysis

Dependencies within the EAL 2 package selected for the security assurance requirements have been considered by the authors of CC Part 3 and are not analyzed here. The ALC_FLR.1 component that augments the EAL 2 package has no dependencies on other requirements.

The security functional requirements in this ST do not introduce dependencies on any security assurance requirement; neither do the security assurance requirements in this ST introduce dependencies on any security functional requirement.

The following table demonstrates the dependencies of SFRs modeled in CC Part 2 and how the SFRs for the TOE and the TOE environment resolve those dependencies. (Italics indicates an SFR on the TOE environment.)

Table 14: SFR Dependency Analysis

SFR	Dependencies	Resolved?
FDP_ACC.1	FDP_ACF.1	Yes
FDP_ACF.1	FDP_ACC.1, FMT_MSA.3	Yes
FDP_ITT.1	FDP_ACC.1	Yes
FIA_ATD.1	none	–
FIA_SOS.1	none	–
FIA_UAU.2	FIA_UID.1	Yes, by FIA_UID.2
FIA_UID.2	none	–
FMT_MOF.1	FMT_SMF.1 and FMT_SMR.1	Yes
FMT_MSA.1a	[FDP_ACC.1 or FDP_IFC.1] and FMT_SMR.1 and FMT_SMF.1	Yes, by FDP_ACC.1, FMT_SMR.1 and FMT_SMF.1
FMT_MSA.1b	[FDP_ACC.1 or FDP_IFC.1] and FMT_SMR.1 and FMT_SMF.1	Yes, by FDP_ACC.1, FMT_SMR.1 and FMT_SMF.1
FMT_MSA.3	FMT_MSA.1 and FMT_SMR.1	Yes
FMT_SMF.1	none	–
FMT_SMR.1	FIA_UID.1	Yes, by FIA_UID.2
FTA_SSL.3	none	–

Table 15: IT Environment SFR Dependency Analysis

SFR	Dependencies	Resolved?
FPT_STM.1	none	–

There are no unsatisfied dependencies.

8.2.3.1 Internal consistency and mutual support of SFRs

The dependency analysis demonstrates how supportive dependencies between SFRs are satisfied. This section further demonstrates that the SFRs are mutually supportive by highlighting and discussing the additional supportive dependencies that ensure that security policies are enforced.

FIA_UID.2 ensures that no security mediated functions can be initiated on behalf of a user until the user is uniquely identified to the TOE, while FIA_UAU.2 provides additional protection as it ensures that no security-mediated functions can be initiated on behalf of a user until the user has been authenticated to the TOE. FIA_SOS.1 provides further assurance related to identification and authentication by preventing brute force password guessing attacks against administrator account. Finally, FTA_SSL.3 terminates an

inactive user session with the TOE after a configured time interval (default 60 minutes), reducing the potential for an attacker to use the credentials of a logged on, absent user to perform actions he or she is not authorized to perform. FTA_SSL.3 needs a reliable time source, which is provided by the TOE environment (FPT_STM.1).

Identification and authentication is the first step in the process to perform actions. The second step is for the identified and authenticated user to be able to perform tasks associated with the TOE and the software usage records protected by the TOE.

FDP_ACC.1 specifies that performance of specific operations is managed by the Management SFP, whose rules are defined in FDP_ACF.1. The rules are based on user attributes that are defined according to the requirements listed in FDP_ATD.1. The Super Administrator is the key to managing users and their access to the TOE and the software usage records protected by the TOE. The management functions of the Super Administrator are defined in FMT_MOF.1 (the Super Administrator manages authentication) and FMT_MSA.1a (the Super Administrator can modify the security attributes of users). FMT_MSA.1b defines the single security management function that each user can perform; that is, each user can change his or her own password. In addition, FSM_MSA.3 defines that the Super Administrator sets restrictive default values for user attributes. Further, FMT_SMF.1 defines the security management functions that the Management SFP manages, while FMT_SMR.1 defines the security roles that govern which administrator can perform specific actions within an organization.

The reason to identify and authenticate users and to associate those users with specific roles in the organization is ultimately to protect software usage records. Another aspect to protecting software usage records is protecting the transfer of software usage records from one component of the TOE to another component. FDP_ITT.1 requires the TOE to provide an SSL channel between the agents and the Runtime Server. In addition, FDP_ITT.1 requires the TOE to provide a mechanism to persist data on the sending component until confirmation is received that the data has been successfully written to the database in the target component.

8.2.4 Appropriateness of Assurance Requirements

The TOE is intended to protect against attackers of limited resources. The assurance requirements EAL 2 augmented with ALC_FLR.1 brings sufficient assurance elements for the TOE, operating within its environment as described in this document. Furthermore, the EAL 2 assurance level augmented with ALC_FLR.1 is technically feasible and achievable based on the requirements on life-cycle support, development documents, secure delivery procedure, and configuration management. It is appropriate to satisfy users' expectations.

8.3 TOE Summary Specification Rationale

The TOE security functions work together to satisfy the security functional requirements. Below is a justification for each SFR, how the related security functions work together to meet the requirements and as well for the sum of security assurance requirements.

8.3.1 TOE Security Functions Rationale

The following table shows that the IT security functions specified in the TOE summary specification meet all the security functional requirements for the TOE and work together to satisfy the TOE security functional requirements.

Table 16: Mapping TOE SFRs to TSFs

TOE SFR	TOE Security Function
FDP_ACC.1	SF.ACCESS
FDP_ACF.1	SF.ACCESS
FDP_ITT.1	SF.DATA1; SF.DATA.2;SF.IA.3
FIA_ATD.1	SF.ACCESS
FIA_SOS.1	SF.IA.2
FIA_UAU.2	SF.IA.1; SF.IA.4
FIA_UID.2	SF.IA.1; SF.IA.4
FMT_MOF.1	SF.MGMT.2
FMT_MSA.1a	SF.MGMT.1; SF.MGMT.2; SF.MGMT.4
FMT_MSA.1b	SF.MGMT.3
FMT_MSA.3	SF.MGMT.1; SF.MGMT.2
FMT_SMF.1	SF.MGMT.1; SF.MGMT.2; SF.MGMT.3; SF.MGMT.4; SF.MGMT.5; SF.MGMT.6; SF.MGMT.7; SF.MGMT.8; SF.MGMT.9; SF.MGMT.10, SF.MGMT.11
FMT_SMR.1	SF.ACCESS
FTA_SSL.3	SF.IA.5

Table 16 shows that all the TOE security functional requirements are addressed by the TOE security functions. Below is described how the IT security functions of the TOE are suitable to meet the TOE security functional requirements.

FDP_ACC.1 The requirement to enforce the Management SFP on users as subjects and user data as objects among the named set of operations performed the objects is satisfied by the function SF.ACCESS, which describes that Tivoli License Compliance Manager controls access to user data based on roles.

FDP_ACF.1 The requirement to enforce access to user data based on the user's role in the organization is satisfied by the function SF.ACCESS, which specifies the actions allowed for users with specific roles in an organization.

FDP_ITT.1 The requirement for the TOE to provide a secure channel for communication and data transfer between the agents and the Runtime Server is satisfied by SF.DATA.1. The requirement for the TSF to acknowledge, when requested by another part of the TSF, the receipt of an unmodified TSF data transmission is satisfied by SF.DATA.2, which specifies a mechanism for one component to notify another component when data has been successfully persisted by the target component.

FIA_ATD.1 The requirement to associate a specific set of attributes with each user is satisfied by the function SF.ACCESS, which specifies that users' access to user data protected by the TOE is based on their role within the organization. Because the user cannot be granted access to anything without first authenticating, user name and password are also required attributes.

FIA_SOS.1 The requirement to comply with the user and Runtime Server communication password policy is satisfied by the function SF.IA.2, which identifies that compliance to the password policy is enforced by the TOE for all users.

FIA_UAU.2 The requirement to allow no TSF mediated operations on behalf of the user before the user is authenticated is satisfied by function SF.IA.1, which specifies that a human user must be authenticated before performing any action on the Administration Server and SF.IA.4, which specifies that if the security communication option is chosen, the Runtime Server must be authenticated to the Administration Server before performing any action.

FIA_UID.2 The requirement to allow no TSF mediated operations on behalf of the user before the user is identified is satisfied by function SF.IA.1, which specifies that a human user must be identified before performing any action on the Administration Server and SF.IA.4, which specifies that the Runtime Server must be identified to the Administration Server before performing any action.

FMT_MOF.1 The requirement for only the Super Administrator to be able to modify the function authentication is satisfied by SF.MGMT.2, which allows the Super Administrator to manage users.

FMT_MSA.1a The requirement for only the Super Administrator to be able to modify the security attributes of users is satisfied by SF.MGMT.1 and SF.MGMT.2, which enable the Super Administrator to change security attributes of users of the TOE; and SF.MGMT.4, which enables the Super Administrator to change the security attribute of the Runtime Server (with regard to its ability to authenticate to the Administration Server).

FMT_MSA.1b The requirement for each user to be able to modify a single security attribute - his or her own password - is satisfied by SF.MGMT.3.

FMT_MSA.3 The requirement for restrictive default values for security attributes of users and for only the Super Administrator to specify alternative values for security attributes of users is satisfied by SF.MGMT.1 and SF.MGMT.2, which covers managing organizations and users.

FMT_SMF.1 The requirement that names the security functions managed by the TOE is satisfied in full by SF.MGMT.1, SF.MGMT.2, SF.MGMT.3, SF.MGMT.4, SF.MGMT.5, SF.MGMT.6, SF.MGMT.7, SF.MGMT.8, SF.MGMT.9, SF.MGMT.10, and SF.MGMT.11.

FMT_SMR.1 The requirement to manage the security roles named in the requirement is satisfied by SF.ACCESS, which matches the list of security roles named.

FTA_SSL.3 The requirement to terminate a user session after 60 minutes of inactivity is satisfied by SF.IA.5.

8.3.2 Assurance Measures Rationale

This section shows that the identified assurance measures are appropriate to meet the assurance requirements by providing mapping between the identified assurance measures and the assurance requirements, as shown in Table 17.

Table 17: Mapping of Assurance Measures to Assurance Requirements

	ACM_CAP.2	ADO_DEL.1	ADO_IGS.1	ADV_FSP.1	ADV.HLD.1	ADV_RCR.1	AGD_ADM.1	AGD_USR.1	ALC_FLR.1	ATE_COV.1	ATE_FUN.1	ATE_IND.2	AVA_SOF.1	AVA_VLA.1
M.CAP	X													
M.DEL		X												
M.IGS			X											
M.FSP				X										
M.HLD					X									
M.RCR						X								
M.ADM							X							
M.USR								X						
M.FLR									X					
M.COV										X				
M.FUN											X			
M.IND												X		
M.SOF													X	
M.VLA													X	X

Table 4 provides justification relating the assurance measures to the assurance requirements.

8.3.3 Minimum Strength of Function Rationale

The TOE mechanisms will resist technical attacks by unauthorized users. The TOE mechanisms will also resist user errors, system errors, or non-malicious actions by authorized users. The environment also assumes that those individuals who have authorized physical access to the TOE are trusted to not behave maliciously.

Consequently, the TOE has a strength of function basic (**SOF-basic**), which indicates that overall, the probabilistic and permutational functions of the TOE provide adequate protection against casual breach of TOE security by attackers possessing a low attack potential is consistent with the security objectives of the TOE.

This claim applies to the password policy for passwords associated with user and Runtime Server identification and authentication as specified in FIA_SOS.1, using the described settings and satisfied by SF.IA.2.

8.4 PP Claims Rationale

No claims to any Protection Profile are made.