# d'COMPASS Security Target

## Common Criteria: EAL2

**Document version:** 1.0

**Document date:** 1-OCT-2014

# Document management

## Document identification

| | |
|---|---|
| **Document title** | d'COMPASS Security Target |
| **Document version** | 1.0 |
| **Document date** | 1-OCT-2014 |
| **Author** | Muzamir Mohamad |
| **Release Authority** | Han Yung Chai |

## Document history

| Version | Date | Description |
|---|---|---|
| 0.1 | 13-JUNE-2014 | Released for internal review |
| 0.2 | 27-JUNE-2014 | Released to evaluators |
| 0.3 | 7-JULY-2014 | Updated to address comments from evaluators |
| 0.4 | 1-AUG-2014 | Minor changes on TOE version number |
| 0.5 | 13-AUG-2014 | Updated to address EOR-ASE v1.0 |
| 0.6 | 25-AUG-2014 | Updated to address re-released EOR-ASE v1.1 |
| 0.7 | 25-SEP-2014 | Updated to address re-released EOR-AGD v1.0 |
| 1.0 | 1-OCT-2014 | Changes to minimum system requirements and final release |

# Table of Contents

# 1 Security Target introduction (ASE_INT.1)

## 1.1 ST reference

| | |
|---|---|
| **ST Title** | d'COMPASS Security Target |
| **ST Version** | 1.0 |
| **ST Date** | 1-OCT-2014 |

## 1.2 TOE reference

| | |
|---|---|
| **TOE Title** | d'COMPASS |
| **TOE Version** | 2.0.0 |

## 1.3 Document organization

This document is organized into the following major sections:

- Section 1 provides the introductory material for the ST as well as the TOE description (ASE_INT.1).
- Section 2 provides the conformance claims for the evaluation (ASE_CCL.1).
- Section 3 provides the definition of the security problem that the TOE has been designed to address (ASE_SPD.1).
- Section 4 defines the security objectives for the TOE and the environment (ASE_OBJ.2).
- Section 5 contains the security functional and assurance requirements derived from the Common Criteria, Part 2 and 3 respectively, which are in turn satisfied by the TOE and the development lifecycle (ASE_REQ.2).
- Section 6 provides a summary of the TOE specification, identifying the IT security functions provided by the TOE (ASE_TSS.1).

## 1.4 Defined terms

The following table defines all subjects, objects, operations, security attributes, external entities and other key terms that are used within the statements of security functional and assurance requirements. It also describes the acronym used in this documentation.

| Term | Description |
|---|---|
| Authentication Data | It is information used to verify the claimed identity of a user. |
| ACL | Access control lists |
| Java EE | Java Platform Enterprise Edition |
| RDBMS | Relational database management system |
| SHA-256 | SHA stands for Secure Hash Algorithm. SHA-256 is a set of cryptographic functions that falls under SHA-2 family designed by the U.S. National Security Agency (NSA) and published in 2001 by the NIST as a U.S. Federal Information Processing Standard (FIPS). |
| System Admin | The system admin is a pre-set user within the TOE that is created during TOE installation. All functions assigned to System admin are add new user profile and roles, edit/change and delete existing user profile and role setting, cancelling the form changes, closing the form, change password, and security setting |
| TSF data | Data created by and for the TOE, which might affect the operation of the TOE. |
| TSC | TOE Scope of Control, the set of interactions that can occur with or within a TOE and are subject to the rules of the TSP |
| TSP | TOE Security Policy, a set of rules that regulate how assets are managed, protected and distributed. |
| Unauthorized users | Unauthorized users can mean a legitimate user with access rights to certain web resource, an external entity that has no rights to any protected resource or data. |
| Users | It means any entity (human user or external IT entity) outside the TOE that interacts with the TOE. In this case, there are users of the TOE access the TOE through a web browser. |
| User data | Data created by and for the user, which does not affect the operation of the TSF. |

## 1.5 TOE overview

### 1.5.1 TOE usage and major security functions

The Target of Evaluation (TOE) is d'COMPASS version 2.0.0. The TOE is an online Treasury Management System that caters for managing and operating treasury, financial asset investment and financial risks. It provides a complete solution from back to the front office fund management operation. The TOE is a web application that is developed entirely from Java-based technologies. It is installed into a Java EE-compliant application server and is accessible via a web browser. Below are the primary features of the TOE:

- Single point of entry for every trade
- Centralised risk and compliance controls
- Centralised information in a global data repository
- Flexibility to incorporate new processes and external data flows
- Wide range user-defined intelligent enquiries and reports
- Ease of connection with external applications and tools

The following table highlights the range of security functions and features implemented by the TOE.

| Security function | Description |
|---|---|
| Access control | The TOE manages access control within each organisation based on user IDs, user roles and access control lists. Each ACL maps users and roles to the operations that they are permitted to perform on the object. |
| Identification and authentication | The TOE requires that each user is successfully identified (user IDs) and authenticated (password) before any interaction with protected resources is permitted. |
| Security Management | The TOE provides functions that allow management of the TOE and its security functions. The TOE restricts access to the management functions based on the role of the user. |
| Secure Communication | The TOE is able to protect the user data from disclosure and modification using SSL as a secure communication between users' browser and the TOE. |

### 1.5.2 TOE Type

The TOE is a web application designed to be used as a Treasury Management Systems for a web-based application environment.  The TOE provides security functionality such as access control, identification and authentication security management and secure communication.

### 1.5.3 Supporting Hardware, software and/or firmware

The underlying hardware and software that is used to support the TOE are:

| Minimum System Requirements | |
|---|---|
| **Application Server** | |
| Operating Systems | Microsoft Windows Server 2008 |
| Processor | Intel Core i3 |
| Memory (RAM) | 4 GB |
| Application | Wildfly 8.0 |
| Database (RDBMS) | Microsoft SQL Server 2008 |
| Supporting software | Java Runtime Environment 8.0 |
| **End-user** | |
| Web Browser | • Internet Explorer 11<br>• Firefox 30<br>• Chrome 35 |

**Note:** The TOE is tested on Microsoft Windows Server 2012 and Microsoft SQL Server 2012.

## 1.6 TOE description

### 1.6.1 Physical scope of the TOE

The TOE is web application (Java-based) for Treasury Management System hosted on a server. A typical installation of the TOE can be found in Figure 1 below, which identifies the various components of the TOE architecture.

**Formatte**

**Deleted:**

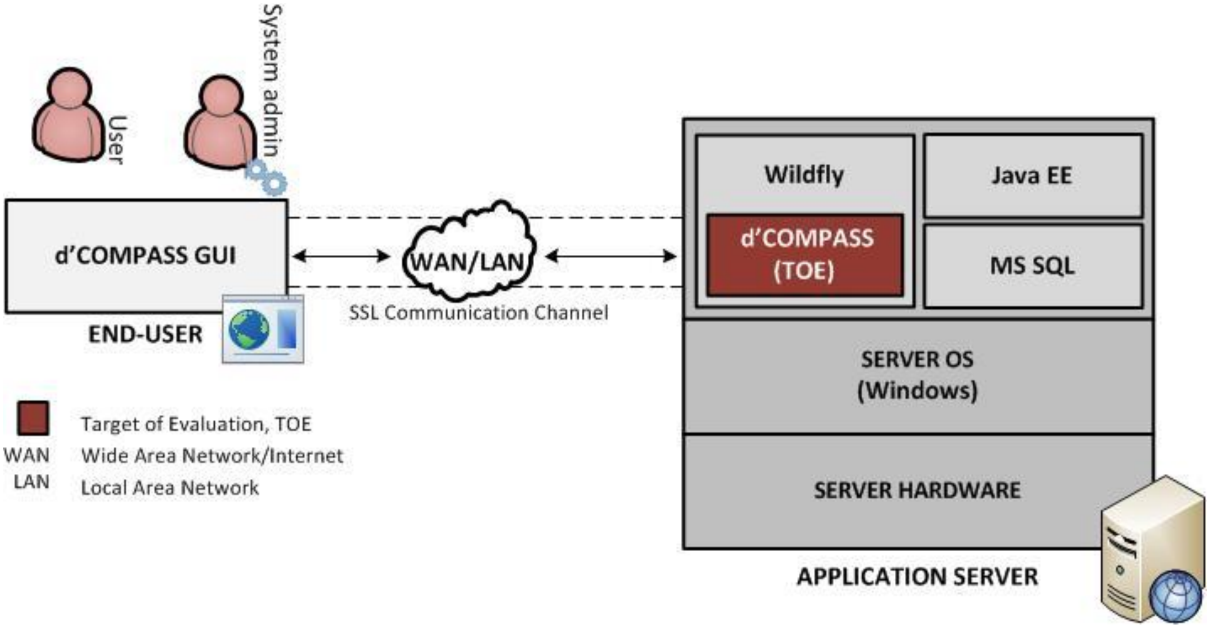

Figure 1 – TOE

### 1.6.2 Logical scope of the TOE

The logical boundary of the TOE is summarized below.

a) **Access Control.** The access control function permits a user to access a protected resource only if the user ID or user role has permission to perform the requested action on the resource. Access rules are stored in Access Control Lists associated with each object in the TSC.

b) **Identification & Authentication.** All users are required to be identified and authenticated before any information flows are permitted. The TOE checks the credentials presented by the user at the login page against the authentication information stored in the database.

c) **Security Management.** The TOE contains various management functions to ensure efficient and secure management of the TOE. The TOE maintains role-based access control mechanisms to ensure that functions are restricted to those who have the privilege to access them. The system admin has the ability to create users roles, who have privileged access to specific functions.The functions above are restricted based on this role.

d) **Secure communications.** The TOE provides a secure SSL channel between the end-user and the TOE.

# 2 Conformance Claim (ASE_CCL.1)

The ST and TOE are conformant to version 3.1 (REV 4) of the Common Criteria for Information Technology Security Evaluation.

The following conformance claims are made for the TOE and ST:

- **Part 2 conformant.** Conformant with Common Criteria for Information Technology Security Evaluation Part 2: Security functional requirements, version 3.1 (REV 4), September 2012
- **Part 3 conformant, EAL2.** Conformant with Common Criteria for Information Technology Security Evaluation Part 3: Security assurance requirements, version 3.1 (REV 4). Evaluation is EAL2, September 2012.

# 3 Security problem definition (ASE_SPD.1)

## 3.1 Overview

This section describes the nature of the security problem that the TOE is designed to address. The security problem is described through:

a) a series of ***threats*** that the TOE has been designed to mitigate,

b) specific ***assumptions*** about the security aspects of the environment (both IT related and non-IT related elements) in which the TOE will operate, and

c) any relevant ***organisational security policies*** statements that are made in terms of rules or guidelines that must be followed by the TOE and/or the operational environment.

## 3.2 Threats

| Identifier | Threat statement |
|---|---|
| T.EAVESDROP | An unauthorized person may eavesdrop the communication between Client-Side and Server-side. |
| T.MANAGEMENT | An unauthorized user modifies management data that they are not authorised to access resulting in a loss of integrity of the data that the TOE uses to enforce the security functions. |
| T.PASSWORD_DATA | An unauthorized user gains access to the passwords in the database and use them to authenticate to the TOE resulting in a loss of confidentiality or integrity of user and management data. |
| T.UNAUTHORISED_ACCESS | A user may gain unauthorized access to the TOE and residing data. |

## 3.3 Organisational security policies

No organisational security policies have been defined regarding the use of the TOE.

## 3.4 Assumptions

| Identifier | Assumption statement |
|---|---|
| A.ADMIN | It is assumed that the person who manages the TOE is not hostile and is competent. |

| Identifier | Assumption statement |
|---|---|
| A.ENVIRONMENT | The TOE environment will provide appropriate authentication and authorisation controls for all users in the underlying environment (including the database, network, operating system and Application Server). |
| A.PASSWORD | It is assumed that users will keep their passwords secret and not write them down or disclose them to any other system or user. It is also assumed that the user password is between a minimum of 8 and a maximum of 18 alphanumeric characters. |
| A.PATCH | It is assumed that the underlying operating system, application server and database are patched and hardened to protect against known vulnerabilities and security configuration issues. |
| A.PHYSICAL | It is assumed that the servers hosting the application and database servers are in a secure operating facility with restricted physical access and non-shared hardware. |
| A.SSL_CONFIG | It is assumed that the web application has valid SSL certificates installed (not revoked or expired), and are sourced from a trusted entity. |

# 4 Security objectives (ASE_OBJ.2)

## 4.1 Overview

The security objectives are a concise statement of the intended response to the security problem defined in Section 3. They are security objectives for the TOE to address and additional objectives that provide specific direction for the intended environment in which the TOE is to operate.

## 4.2 Security objectives for the TOE

| Identifier | Objective statements |
|---|---|
| O.ACCESS | The TOE must ensure that only authorised users are able to access protected resources or functions. |
| O.COMM | The TOE must ensure that user data traversing across the network to the application server is protected from disclosure and loss of integrity. |
| O.MANAGE | The TOE must allow System Admin to effectively manage the TOE, while ensuring that appropriate controls are maintained over those functions. |
| O.PASSWORD_DATA | The TOE must ensure that passwords stored in the database are not in plaintext. |
| O.USER | The TOE must ensure that all users are identified and authenticated before accessing protected resources or functions. |

## 4.3 Security objectives for the environment

| Identifier | Objective statements |
|---|---|
| OE.ADMIN | The owners of the TOE must ensure that the System Admin who manages the TOE is not hostile and is competent. |
| OE.AUTHDATA | The users of the TOE must not disclose their password that protects the TSF data. |
| OE.ENVIRONMENT | Those responsible for the TOE must ensure that there are appropriate authentication and authorisation controls for all users in the underlying environment (including the database, network, operating system and application Server). |
| OE.PHYSICAL | Those responsible for the TOE must ensure that the servers hosting the application and database servers are in a secure operating facility with restricted physical access and non-shared hardware. |

| | |
|---|---|
| OE.PATCH | Those responsible for the TOE must ensure that the underlying operating system, application server and database are patched and hardened to protect against known vulnerabilities and security configuration issues. |
| OE.SSL_CONFIG | Those responsible for the TOE must ensure that the application server has valid SSL certificates installed (not revoked or expired), and are sourced from a trusted entity. |

## 4.4 Security objectives rationale

This section provides the summary that all security objectives are traced back to aspects of the addressed assumptions and threats.

| THREATS/ ASSUMPTIONS<br>OBJECTIVES | T.EAVESDROP | T.MANAGEMENT | T.PASSWORD_DATA | T.UNAUTHORISED_ACCESS | A.ADMIN | A.ENVIRONMENT | A.PASSWORD | A.PATCH | A.PHYSICAL | A.SSL_CONFIG |
|---|---|---|---|---|---|---|---|---|---|---|
| O.ACCESS | | ✓ | | ✓ | | | | | | |
| O.COMM | ✓ | | | | | | | | | |
| O.MANAGE | | ✓ | | | | | | | | |
| O.USER | | ✓ | | ✓ | | | | | | |
| O.PASSWORD_DATA | | | ✓ | | | | | | | |
| OE. ADMIN | | | | | ✓ | | | | | |
| OE.AUTHDATA | | | | | | | ✓ | | | |
| OE. ENVIRONMENT | | | | | | ✓ | | | | |
| OE. PATCH | | | | | | | | ✓ | | |
| OE. PHYSICAL | | | | | | | | | ✓ | |
| OE.SSL_CONFIG | | | | | | | | | | ✓ |

### 4.4.1 TOE security objectives rationale

The following table demonstrates that all security objectives for the TOE trace back to the threats and OSPs in the security problem definition.

| Threats/OSPs | Objectives | Rationale |
|---|---|---|
| T.EAVESDROP | O.COMM | The objective ensures that all user data from the user to the web application will be secured using SSL protecting the user data from unauthorized disclosure and loss of integrity. |
| T.MANAGEMENT | O.USER | The objective ensures that the TOE identifies and authenticates all users before they access a protected resources or functions. |
| | O.MANAGE | This objective ensures that the TOE provides the tools necessary for the authorized system admin to manage the security-related functions and that those tools are usable only by users with appropriate authorizations. |
| | O.ACCESS | The objective ensures that the TOE restricts access to the TOE objects to the authorized users |
| T.PASSWORD_DATA | O.PASSWORD_DATA | The objective ensures that all passwords stored in the database are hashed using SHA-256 before written to the database. No one can see the password in plaintext and will not be able to use the password to authenticate to the TOE. |
| T.UNAUTHORISED_ACCESS | O.ACCESS | The objective ensures that the TOE restricts access to the TOE objects to the authorized users. |
| | O.USER | The objective ensures that the TOE identifies and authenticates all users before they access a protected resources or functions. |

### 4.4.2 Environment security objectives rationale

The following table demonstrates that all security objectives for the operational environment all trace back to assumptions or OSPs in the security problem definition.

| Assumptions | Objective | Rationale |
|---|---|---|
| A.ADMIN | OE.ADMIN | This objective ensures that those responsible for the TOE are competent and trustworthy individuals, capable of managing the TOE and the security of the information it contains. |

| Assumptions | Objective | Rationale |
|---|---|---|
| A.ENVIRONMENT | OE.ENVIRONMENT | This objective ensures that those responsible for the TOE ensure that appropriate authentication and authorisation controls for all users in the underlying environment (including the database, network, operating system and application Server). |
| A.PASSWORD | OE.AUTHDATA | This objective ensures that those responsible for the TOE ensure that the user will know the password but not disclose it to anyone else. |
| A.PHYSICAL | OE.PHYSICAL | This objective ensures that those responsible for the TOE ensure that the servers that host the application and database are hosted in a secure operating facility with restricted physical access with non-shared hardware. |
| A.PATCH | OE.PATCH | This objective ensures that those responsible for the TOE ensure that the underlying operating system, application server and database and are patched and hardened to protect against known vulnerabilities and security configuration issues. |
| A.SSL_CONFIG | OE.SSL_CONFIG | This objective ensures that those responsible for the TOE ensure that the web application has SSL certificates installed and are valid (not revoked or expired), are sourced from a trusted entity. |

# 5 Security requirements (ASE_REQ.2)

## 5.1 Overview

This section defines the security requirements satisfied by the TOE. Each requirement has been extracted from version 3.1 (REV 4) of the Common Criteria, part 2 providing functional requirements and part 3 providing assurance requirements.

Part 2 of the Common Criteria defines an approved set of operations that may be applied to security functional requirements. Following are the approved operations and the document conventions used within this ST to depict their application:

- **Assignment.** The assignment operation provides the ability to specify an identified parameter within a requirement. Assignments are depicted using bolded text and are surrounded by square brackets as follows [**assignment**].
- **Selection.** The selection operation allows the specification of one or more items from a list. Selections are depicted using bold italics text and are surrounded by square brackets as follows [***selection***].
- **Refinement.** The refinement operation allows the addition of extra detail to a requirement. Refinements are indicated using bolded text, for **additions**, and strike-through, for ~~deletions~~.
- **Iteration.** The iteration operation allows a component to be used more than once with varying operations. Iterations are depicted by placing a letter at the end of the component identifier as follows FDP_IFF.1a and FDP_IFF.1b.

## 5.2 Security functional requirements

### 5.2.1 Overview

The security functional requirements are expressed using the notation stated in Section 5.1 above and are itemised in the table below.

| Identifier | Title |
|---|---|
| FCS_COP.1 | Cryptographic operation |
| FDP_ACC.1 | Subset access control |
| FDP_ACF.1 | Security attribute based access control |
| FIA_UAU.2 | User authentication before any action |

| Identifier | Title |
|---|---|
| FIA_UID.2 | User identification before any action |
| FMT_MSA.1 | Management of security attributes |
| FMT_MTD.1a | Management of TSF data (Configuration) |
| FMT_MTD.1b | Management of TSF data (Password) |
| FMT_SMF.1 | Specification of Management Functions |
| FMT_SMR.1 | Security roles |
| FTP_TRP.1 | Trusted path |

### 5.2.2 FCS_COP.1 Cryptographic Operation

| | |
|---|---|
| Hierarchical to: | No other components. |
| FCS_COP.1.1 | The TSF shall perform [**secure hashing**] in accordance with a specified cryptographic algorithm [**SHA-256**] and cryptographic key sizes [**none**] that meet the following: [**None**]. |
| Dependencies: | [FDP_ITC.1 Import of user data without security attributes, or<br>FDP_ITC.2 Import of user data with security attributes, or<br>FCS_CKM.1 Cryptographic key generation]<br>FCS_CKM.4 Cryptographic key destruction |
| Notes: | This cryptographic operation does not use key. The password of the users are hashed and compared with the values stored in the authentication data database. |

### 5.2.3 FDP_ACC.1 Subset access control

| | |
|---|---|
| Hierarchical to: | No other components. |
| FDP_ACC.1.1 | The TSF shall enforce the [**Access Control SFP**] on [<br>**Subjects:**<br>**a) HTTP request on behalf of users**<br>**Objects:**<br>**a) Protected resources (methods and HTML pages)**<br>**Operations:**<br>**a) Methods execution**<br>**b) Serving of HTML pages**] |

| | |
|---|---|
| Dependencies: | FDP_ACF.1 – Security attribute based access control |
| Notes: | None. |

### 5.2.4 FDP_ACF.1 Security attribute based access control

| | |
|---|---|
| Hierarchical to: | No other components. |
| FDP_ACF.1.1 | The TSF shall enforce the [**Access Control SFP**] to objects based on the following: [<br>**Subject attribute:**<br>**a) ID of the user**<br>**b) corresponding user role**<br>**Object attributes:**<br>**a) Access Control List**] |
| FDP_ACF.1.2 | The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [<br>**a) The operation is allowed, if:**<br>**b) The Access Control List for an object permits the user ID to access that object; OR**<br>**c) The Access Control List for an object permits the User Role to access that Object**]. |
| FDP_ACF.1.3 | The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [**the System admin role can access all records and functions**]. |
| FDP_ACF.1.4 | The TSF shall explicitly deny access of subjects to objects based on the following additional rules: [**None**]. |
| Dependencies: | FDP_ACC.1 Subset access control |
| Notes: | None. |

### 5.2.5 FIA_UAU.2 User authentication before any action

| | |
|---|---|
| Hierarchical to: | FIA_UAU.1 Timing of authentication |
| 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. |
| Dependencies: | FIA_UID.1 Timing of identification |
| Notes: | None. |

### 5.2.6 FIA_UID.2 User identification before any action

| | |
|---|---|
| Hierarchical to: | FIA_UID.1 Timing of identification |
| FIA_UID.2.1 | The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user. |
| Dependencies: | No dependencies. |
| Notes: | None. |

### 5.2.7 FMT_MSA.1 Management of security attributes

| | |
|---|---|
| Hierarchical to: | No other components. |
| FMT_MSA.1.1 | The TSF shall enforce the [**Access Control SFP**] to restrict the ability to [***write or delete***] the security attributes [**that assign user Ids to roles to only the users that are mapped**] to [**the system admin role**]. |
| Dependencies: | [FDP_ACC.1 Subset access control, or<br>FDP_IFC.1 Subset information flow control]<br>FMT_SMR.1 Security roles<br>FMT_SMF.1 Specification of Management Functions |
| Notes: | None. |

### 5.2.8 FMT_MTD.1a Management of TSF data (Configuration)

| | |
|---|---|
| Hierarchical to: | No other components |
| FMT_MTD.1a.1 | The TSF shall restrict the ability to [***add, delete, edit*** [**Create**]] the [**Access Control Lists, assign users to roles, User ID**] to [**System admin**]. |
| Dependencies: | FMT_SMR.1 Security roles<br>FMT_SMF.1 Specification of Management Functions |
| Notes: | None. |

### 5.2.9 FMT_MTD.1b Management of TSF data (Password)

| | |
|---|---|
| Hierarchical to: | No other components |
| FMT_MTD.1b.1 | The TSF shall restrict the ability to [***modify***] the [**User Password**] to [**user defined (all users) and System Admin (all users)**]. |
| Dependencies: | FMT_SMR.1 Security roles |

| | |
|---|---|
| | FMT_SMF.1 Specification of Management Functions |
| Notes: | None. |

### 5.2.10 FMT_SMF.1 Specification of Management Functions

<table>
<tr><td>Hierarchical to:</td><td>No other components.</td></tr>
<tr><td>FMT_SMF.1.1</td><td>The TSF shall be capable of performing the following management functions: [
<table>
<tr><th>Roles</th><th>Functions</th></tr>
<tr><td>User<br>(user defined role)</td><td>All functions assigned to user:
<ul>
<li>creating deals</li>
<li>Approving deals</li>
<li>Setup of business related configuration</li>
<li>Perform Transaction settlement</li>
<li>View, edit, delete operation functions (Setup, Transaction, Processes, Settlement and approval)</li>
<li>Change password,</li>
<li>Cancelling form changes,</li>
<li>Closing the form</li>
</ul>
<b>Note:</b> The user is created by System admin with a specific roles and permissions based on defined role/permission (roles are categorized into Product Concept, Functions, Business Entities and Market Types).</td></tr>
<tr><td>System admin<br>(Pre-set role)</td><td>All functions assigned to System admin and:
<ul>
<li>Add new user profile and roles,</li>
<li>Edit/Change and Delete existing User Profile and Role Setting,</li>
<li>Cancelling the form changes,</li>
<li>Closing the form,</li>
<li>Change password,</li>
<li>Setup of system related configuration, and</li>
<li>Perform system maintenance</li>
</ul>
<b>Note:</b> The system admin is a pre-set user within the TOE that is created during TOE installation. For higher security features or environments, the system admin should be given the appropriate authentication and authorization controls in the underlying environment. (Including the Operating System, database, and Application Server).</td></tr>
</table>
</td></tr>
</table>

| | |
|---|---|
| | ] |
| Dependencies: | No dependencies. |
| Notes: | None. |

### 5.2.11 FMT_SMR.1 Security Roles

| | |
|---|---|
| Hierarchical to: | No other components. |
| FMT_SMR.1.1 | The TSF shall maintain the roles [**users and System admin**]. |
| FMT_SMR.1.2 | The TSF shall be able to associate users with roles. |
| Dependencies: | FIA_UID.1 Timing of identification |
| Notes: | System admin (setup of system related configurations, perform system maintenance and mainly use functions under **System Admin** menu) |

### 5.2.12 FTP_TRP.1 Trusted path

| | |
|---|---|
| Hierarchical to: | No other components. |
| FTP_TRP.1.1 | The TSF shall provide a communication path between itself and [***remote***] users that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from [***modification, disclosure***]]. |
| FTP_TRP.1.2 | The TSF shall permit [***remote users***] to initiate communication via the trusted path. |
| FTP_TRP.1.3 | The TSF shall require the use of the trusted path for [***initial authentication***]]. |
| Dependencies: | No dependencies |
| Notes: | None. |

## 5.3 TOE Security assurance requirements

EAL2 requires evidence relating to the design information and test results, but does not demand more effort on the part of the developer than is consistent with good commercial practice.

EAL2 provides assurance by a full security target and an analysis of the SFRs in that ST, using a functional and interface specification, guidance documentation and a basic description on the architecture of the TOE, to understand the security behaviours.

The analysis is supported by independent testing of the TSF, evidence of developer testing based on the functional specification, selective independent confirmation of the developer test results, and a vulnerability analysis (based upon the functional specification, TOE design, security architecture description and guidance evidence provided) demonstrating resistance to attackers with basic attack potential.

EAL2 also provides assurance through use of a configuration management system and evidence of secure delivery procedures.

| Assurance class | Assurance components |
|---|---|
| ADV: Development | ADV_ARC.1 Security architecture description |
| | ADV_FSP.2 Security-enforcing functional specification |
| | ADV_TDS.1 Basic design |
| AGD: Guidance documents | AGD_OPE.1 Operational user guidance |
| | AGD_PRE.1 Preparative procedures |
| ALC: Life cycle support | ALC_CMS.2 Parts of the TOE CM coverage |
| | ALC_CMC.2 Use of a CM system |
| | ALC_DEL.1 Delivery procedures |
| ASE: Security Target evaluation | ASE_CCL.1 Conformance claims |
| | ASE_ECD.1 Extended components definition |
| | ASE_INT.1 ST Introduction |
| | ASE_OBJ.2 Security objectives |
| | ASE_REQ.2 Derived security requirements |
| | ASE_SPD.1 Security Problem Definition |
| | ASE_TSS.1 TOE summary specification |
| ATE: Tests | ATE_IND.2 Independent testing - sample |
| | ATE_FUN.1 Functional testing |
| | ATE_COV.1 Evidence of coverage |
| AVA: Vulnerability assessment | AVA_VAN.2 Vulnerability analysis |

## 5.4 Security requirements rationale

### 5.4.1 Dependency rationale

The table below demonstrates the mutual supportiveness of the SFRs for the TOE by demonstrating how the SFR dependencies are fulfilled by the TOE and by justifying those dependencies that are not fulfilled.

The SARs relevant to the TOE constitute an evaluation assurance level of EAL2, as defined in the Common Criteria and include no extensions or augmentations. Therefore, as a complete evaluation assurance level, they are a mutually supportive set and require no further justification.

| SFR | Dependency | Inclusion |
|---|---|---|
| FDP_ACC.1 | FDP_ACF.1 Security attribute based access control | FDP_ACF.1 |
| FDP_ACF.1 | FDP_ACC.1 Subset access control | FDP_ACC.1 |
| FIA_UAU.2 | FIA_UID.1 Timing of identification | FIA_UID.2 |
| FIA_UID.2 | No dependencies | N/A |
| FMT_SMF.1 | No dependencies | N/A |
| FMT_MSA.1 | [FDP_ACC.1 Subset access control, or<br>FDP_IFC.1 Subset information flow control]<br>FMT_SMR.1 Security roles<br>FMT_SMF.1 Specification of Management Functions | FDP_ACC.1<br>FDP_ACF.1<br>FMT_SMF.1<br>FMT_SMR.1 |
| FMT_SMR.1 | FIA_UID.1 Timing of identification | FIA_UID.2 |
| FCS_COP.1 | [FDP_ITC.1 Import of user data without security attributes, or<br>FDP_ITC.2 Import of user data with security attributes, or<br>FCS_CKM.1 Cryptographic key generation]<br>FCS_CKM.4 Cryptographic key destruction | SHA-256 is a hashing algorithm and is a one-way function. Therefore it does not use any key for hashing and there is no FCS_CKM.1 and FCS_CKM.4 involved for the function. Therefore the dependencies are not applicable. |
| FMT_MTD.1a | FMT_SMR.1 Security roles<br>FMT_SMF.1 Specification of Management Functions | FMT_SMF.1<br>FMT_SMR.1 |

| SFR | Dependency | Inclusion |
|---|---|---|
| FMT_MTD.1b | FMT_SMR.1 Security roles<br>FMT_SMF.1 Specification of Management Functions | FMT_SMF.1<br>FMT_SMR.1 |
| FTP_TRP.1 | No dependencies | N/A |

### 5.4.2 Mapping of SFRs to security objectives for the TOE

| Security objective | Mapped SFRs | Rationale |
|---|---|---|
| O.ACCESS | FDP_ACC.1 | The requirement helps meet the objective by identifying the objects and users subjected to the access control policy. |
| | FDP_ACF.1 | The requirement meets the objective by ensuring the TOE only allows access to objects based on the defined access control policy. |
| O.USER | FIA_UID.2 | The requirement helps meet the objective by identifying the users before any TSF mediated actions |
| | FIA_UAU.2 | The requirement helps meet the objective by authenticating the users before any TSF mediated actions. |
| | FMT_SMR.1 | The TOE maintains System Admin and manages multiple user roles. |
| O.PASSWORD_DATA | FCS_COP.1 | The requirement helps to meet the objective by hashing all the passwords using SHA-256 before they are written to the database. |
| O.MANAGE | FMT_MSA.1 | The TOE allows the System Admin to determine who has access to the folder and the folder’s contents, and what actions the user can perform. |
| | FMT_MTD.1a | This requirements helps meet the objective by allowing only the System Admin roles to create, delete, modify access control lists, and mapping users to roles and user accounts to the respective organisation database. |
| | FMT_MTD.1b | This requirement helps meet the objective by allowing users of all roles to change their passwords. |
| | FMT_SMF.1 | The TOE allows the mapping of user to roles, creation of users, deletion of users, changing of passwords and management of ACLs. |
| | FMT_SMR.1 | The TOE maintains System Admin and manages multiple user roles. |

| Security objective | Mapped SFRs | Rationale |
|---|---|---|
| O.COMM | FTP_TRP.1 | This requirement helps meet the objective by establishing a SSL secure channel from the user's browser to the TOE, thus protecting the user data from disclosure and modification. |

### 5.4.3 Explanation for selecting the SARs

The assurance package for the evaluation of the TOE is Evaluation Assurance Level 2 (EAL2).

The TOE is intended to provide a number of capabilities, which are designed to support organisations to deploy and manage treasury management systems. EAL2 is sufficient to demonstrate that the TOE is resistant to attackers with a Basic attack potential.

# 6 TOE summary specification (ASE_TSS.1)

## 6.1 Overview

This section provides the TOE summary specification, a high-level description of how the TOE actually implements the claimed security functional requirements.

The TOE security functions include the following:

- **Access Control**
- **Identification and Authentication**
- **Security Management**
- **Secure communications**

## 6.2 Access Control

The TOE enforces an access control policy on protected resources. After a user is identified and authenticated to the TOE, the TOE will check all HTTP requests from the user to the protected resource. The TOE will permit a user to access a protected resource only if a user ID or role has permission to perform the requested action on the resource (**FDP_ACC.1, FDP_ACF.1**). The TOE maintains access control lists (ACL) for each object within an organisation. Each ACL maps users and roles to the operations that they are permitted to perform on the object.

There 2 users roles maintained by the TOE. They are users and systems admin (**FMT_SMR.1**). Each type of user will have different access rights to a protected resource. All users will have a unique user ID.

## 6.3 Identification and Authentication

When a user issues a request to the TOE to access a protected resource (methods or HTML pages), the TOE requires that the user (Users and System Admins) identify and authenticate themselves before performing any TSF mediated action (**FIA_UID.2, FIA_UAU.2**). The TOE compares the credentials by checking the information presented by the user at the login page against the authentication information stored in the database.

All user presented passwords are hashed before being used to authenticate to the TOE, or when users change their passwords (**FMT_MTD.1b**) to be written to the database. This is all done by the TOE (**FCS_COP.1**).

## 6.4 Security Management

The TOE contains various management functions to ensure efficient and secure management of the TOE (**FMT_SMF.1**):

System admin role can modify the access control list and mapping of users to roles (**FMT_MSA.1**). TOE provides a suite of management functions to system admin and users. These functions allow for the configuration of the TOE to suit the organization in which it is deployed. Additionally, management roles may perform the following tasks;

- assign user Ids to roles
- add, delete and edit user Ids and roles,
- delete, edit and view operation functions;
- cancelling of form changes,
- closing the form,
- changing of password, and
- security Setting

System admin may assign and adjust the functions available to users; users may assign and adjust the functions based on organization’s requirement(s) (**FMT_SMR.1 and FMT_MTD.1a**).

## 6.5 Secure communications

When a user accesses the TOE on their browser, by typing in the website address, the TOE will initiate a SSL secure channel establishment with the user’s browser (**FTP_TRP.1**). The TOE implements the Secure Sockets Layer (SSL v2/v3) and Transport Layer Security (TLS v1) protocols.