# AquaLogic Interaction Publisher 6.4 MP1 Patch 1 Security Target

Version 1.0
06/27/2008

**Prepared for:**
BEA Systems, Inc
475 Sansome Street
San Francisco, California 94111

**Prepared By:**
Science Applications International Corporation
**Common Criteria Testing Laboratory**
7125 Columbia Gateway Drive, Suite 300
Columbia, MD 21046

LIST OF TABLES

# 1. Security Target Introduction

This section identifies the Security Target (ST) and Target of Evaluation (TOE) identification, ST conventions, ST conformance claims, and the ST organization. The TOE is AquaLogic® Interaction Publisher 6.4 MP1 Patch 1 provided by BEA Systems, Inc. The TOE is a web-based software application that works with AquaLogic Interaction to provide the services required to deploy content-driven applications, such as a customer support knowledge base or sales support center, where users can create and manage Web content without HTML skills.

The Security Target contains the following additional sections:

- TOE Description (Section 2): This section gives an overview of the TOE, describes the TOE in terms of its physical and logical boundaries, and states the scope of the TOE.
- Security Environment (Section 3): This section details the expectations of the environment: describing the threats mitigated by the TOE and its environment, and the assumptions that the TOE and its environment must adhere to.
- Security Objectives (Section 4): This section details the security objectives of the TOE and its environment.
- IT Security Requirements (Section 5): The section presents the security functional requirements (SFR) for TOE and IT Environment that supports the TOE, and the assurance requirements for EAL 2 augmented with ALC_FLR.2.
- TOE Summary Specification (Section 6): The section describes the security functions represented in the TOE that satisfies the security requirements.
- Protection Profile Claims (Section 7): This section identifies the Protection Profile Claim made in the ST.
- Rationale (Section 8): This section closes the ST with the justifications of the security objectives, requirements and TOE summary specifications as to their consistency, completeness and suitability.

## 1.1 Security Target, TOE and CC Identification

**ST Title –** AquaLogic Interaction Publisher 6.4 MP1 Patch 1 Security Target

**ST Version** – Version 1.0

**ST Date** – 06/27/2008

**TOE Identification** – AquaLogic® Interaction Publisher 6.4 MP1 Patch 1

**TOE Developer** – BEA Systems, Inc

**Evaluation Sponsor** – BEA Systems, Inc

**CC Identification** – Common Criteria for Information Technology Security Evaluation, Version 2.3, August 2005.

## 1.2 Conformance Claims

This ST is conformant to the following CC specifications:

- Common Criteria (CC) for Information Technology (IT) Security Evaluation Part 2: Security functional requirements, Version 2.3, August 2005.
  - Part 2 Conformant
- Common Criteria (CC) for Information Technology Security Evaluation Part 3: Security assurance requirements, Version 2.3, August 2005.

- Part 3 Conformant
- Assurance Level: EAL 2 augmented with ALC_FLR.2

## 1.3 Conventions

This section specifies the formatting information used in the Security Target.

### 1.3.1 Conventions

The following conventions have been applied in this document:

- Security Functional Requirements – Part 2 of the CC defines the approved set of operations that may be applied to functional requirements: iteration, assignment, selection, and refinement.
    - Iteration: allows a component to be used more than once with varying operations. In the ST, iteration is indicated by a letter placed at the end of the component. For example FDP_ACC.1a and FDP_ACC.1b indicate that the ST includes two iterations of the FDP_ACC.1 requirement, a and b.
    - Assignment: allows the specification of an identified parameter. Assignments are indicated using bold and are surrounded by brackets (e.g., [**assignment**]).
    - Selection: allows the specification of one or more elements from a list. Selections are indicated using bold italics and are surrounded by brackets (e.g., [***selection***]).
    - Refinement: allows the addition of details. Refinements are indicated using bold, for additions, and strike-through, for deletions (e.g., "… **all** objects …" or "… ~~some~~ **big** things …").
- Other sections of the ST – Other sections of the ST use bolding to highlight text of special interest, such as captions.
- Explicitly stated requirements – Security Functional Requirements not defined in Part 2 of the CC are annotated with EX.

The terminology and acronyms are listed in the appendices.

# 2. TOE Description

The Target of Evaluation (TOE) is AquaLogic® Interaction Publisher 6.4 MP1 Patch 1 henceforth referred to as the TOE or Publisher. Publisher is part of the BEA AquaLogic® User Interaction (ALUI) suite of products and is designed to work with AquaLogic® Interaction 6.1 with AquaLogic® Interaction Development Kit 6.0[1], hereafter referred to as ALI.

Publisher is not a stand-alone product; rather it integrates directly with ALI and depends on ALI portal pages and security functions. Publisher functions as a remote server of ALI by providing the services required to deploy content-driven applications, such as a customer support knowledge base or sales support center, where users can create and manage Web content without HTML skills.

## 2.1 TOE Overview

Publisher is a web application that functions as a remote portlet server for AquaLogic Interaction (ALI). Publisher enables content creation, content publishing, and workflow management. TOE operation requires AquaLogic Interaction.

Publishing makes content available to end users as web pages. Publisher enables users to:

- Publish content to published content portlets, to the ALI Knowledge Directory, or to an external web site.
- Publish content immediately or schedule it to be published at a later date.
- Preview content before publishing it to confirm layout and appearance according to pre-defined presentation templates.
- Publish content to the ALI Knowledge Directory using a Publisher content crawler.
- Remove published content from the web server by setting it to *expire*. This removes it from the web server but keeps it in the Publisher directory. Users can set a published content item to expire immediately or schedule a future expiration.

Publisher's web publishing functions enable users without HTML skills to create and manage web content. Publisher supports the definition of structured content types; web browser form-based data entry; and publishing of content by combining data values with a text presentation template and copying the result to a file system or FTP server.

Content created and published with Publisher is made available to end-users through portlet integration with ALI. Web content can also be published to the ALI Knowledge Directory and to external web sites.

Publisher integration with ALI uses several ALI functions including the following:

- user and group management;
- document storage and management;
- content search;
- object security; and
- user identification and authentication.

Within the TOE, data is organized in a directory structure using folders. Publisher interfaces are used to add, edit, organize, preview and publish content. The core building block of Publisher content is the content item. Publisher provides the following features:

- Data Entry Templates. Content items can be created based on data entry templates. These templates define the properties available for creating a content item. When users create or edit a content item using a data entry template, these properties appear in the Content Item Editor as editable fields. Data entry templates are created and edited using the Data Entry Template Editor.

[1] Note: ALI 6.1 MPI was evaluated separate and is used by Publisher in ALI's evaluated configuration.

- Presentation Templates. These templates enable the definition of standardized pages with consistent branding. Data entry templates are always associated with a presentation template. When a content item is published to the web server or previewed on the preview site, the presentation template determines its appearance and format. Presentation templates are created and edited using the Presentation Template Editor. It takes some HTML skill to create and edit presentation templates.
- Published Content Portlets. Published content portlets present content through ALIthe portal and can enable users to submit and edit content. Publisher includes sample templates for three widely useful portlet types:
  - Announcement portlet,
  - News portlet, and
  - Community Directory portlet.

  Portlets can be created from these templates accepting all defaults, or the Configure Portlet Wizard can be used to edit the portlet objects included in the template to achieve the exact presentation and data formatting needed for specific uses.
- Published content portlet templates. Published content portlet templates enable a user without HTML skills, like a community manager, to create published content portlets using the Configure Portlet Wizard. Publisher provides six sample portlet templates that can be used as-is or modified to meet an organization's unique needs using the Configure Portlet Template Wizard. Users can also create their own portlet templates from scratch using this wizard.
- Content items. Content items are the base objects that are managed using Publisher; a content item can be a set of values entered via the *Content Item Editor* or an uploaded document or image file. Content Item Editor is the interface for creating and editing content items. In addition, users can use WebDAV (Web-based Distributed Authoring and Versioning) to map a Microsoft Windows network folder to the Publisher folder hierarchy and then use Microsoft Windows Explorer to: view file content items and the Publisher folder hierarchy; upload and download file content items between Publisher and the local filesystem; and open file content items for editing on the desktop. The user can also use the WebEdit feature for Microsoft Office files that have been uploaded into a content item.
- Publisher Explorer. Publisher Explorer is the central interface for viewing and managing Publisher objects, including content items, data entry templates, and presentation templates. Publisher Explorer enables users with administrative roles to set up and manage a folder structure to organize content items and published content portlets, and to assign security *roles* to users and groups by folder. Typically, the folders in Publisher Explorer reflect the organization and ownership of ALI communities, such that community managers and content owners are enabled to create and publish information within their scope of responsibility. Publisher Explorer allows as wide a spectrum of users as needed to view and search the Publisher folder hierarchy. Depending on a user's Publisher role, he can use Publisher Explorer to perform almost any Publisher function.
- Workflow. Publisher's workflow function enables an organization to manage the review, approval, and publishing of content using structured and repeatable processes. Users with the Configure Workflow activity right define workflows using the Workflow Editor. A workflow consists of an ordered list of workflow activities, each of them assigned to a user or group of users. Publisher provides portlets that enable tracking of personal workflow assignments and content items in workflow by folder. Workflow is similar to a checklist of reviews and approvals. The basic building block of workflow is the workflow definition—also called, simply, a workflow—which is a defined, ordered set of approval steps called workflow activities. Each workflow activity in a workflow is assigned to a user or group of users. When a content item is submitted to workflow, it is called a work item. When a work item is submitted or passed along to a workflow activity, the assigned users can do one of the following:
  - Approve the work item and send it on to the assignees for the next activity—and publish it if the item is publishable in that activity.
  - Reject it and send it back to the previous assignee.

  - Transfer or delegate the assignment to another user, if the workflow activity allows it. The Publisher workflow function automatically moves the work item through each activity and notifies each responsible party when it is his or her turn to review or take action on the item.

- Scheduled publishing and expiration. This feature enables users to schedule content publishing and automatically remove content from publishing targets and the ALI search index. Publishing places the formatted, web-ready file in a file location that can be accessed by ALI or other web server. Users with the appropriate permissions can publish a content item immediately (if it is in a publishable workflow activity or not subject to workflow), schedule a content item for future publication on a one-time or recurring basis, or set a published content item to *expire*, either immediately or at a future time.

## 2.2 TOE Architecture

Publisher consists of the following components:

- Publisher: A Java servlet-based web application providing the logic and the bulk of the user interface functionality for the creation and maintenance of content and for linear workflows that can be used to govern the approval and publishing of content. With one exception (a diagnostics page), all user interfaces are in the form of portlets that require ALI for display. Publisher includes administrative and content-related portlets and configuration wizards, including the Administer Publisher portlet. These portlets are added to and accessed from the ALI portal pages, however the functionality is provided and controlled by Publisher.

- Image Service Files: These required files provide the necessary images, styles, user interface controls, Java applets, and online help for Publisher. These files are integrated with the ALI's Image Service.

### 2.2.1 Physical Boundaries

The TOE is a web application functioning as a remote server to ALI within an ALUI deployment. Publisher is installed into an ALI deployment network and depends on components of the ALI installation as depicted in Figure 1. Publisher is installed on a Web Application Server. ALI and Publisher share ALI Databases while Publisher also manages its own databases (to store Publisher-specific data); each product having its own set of database tables as necessary.

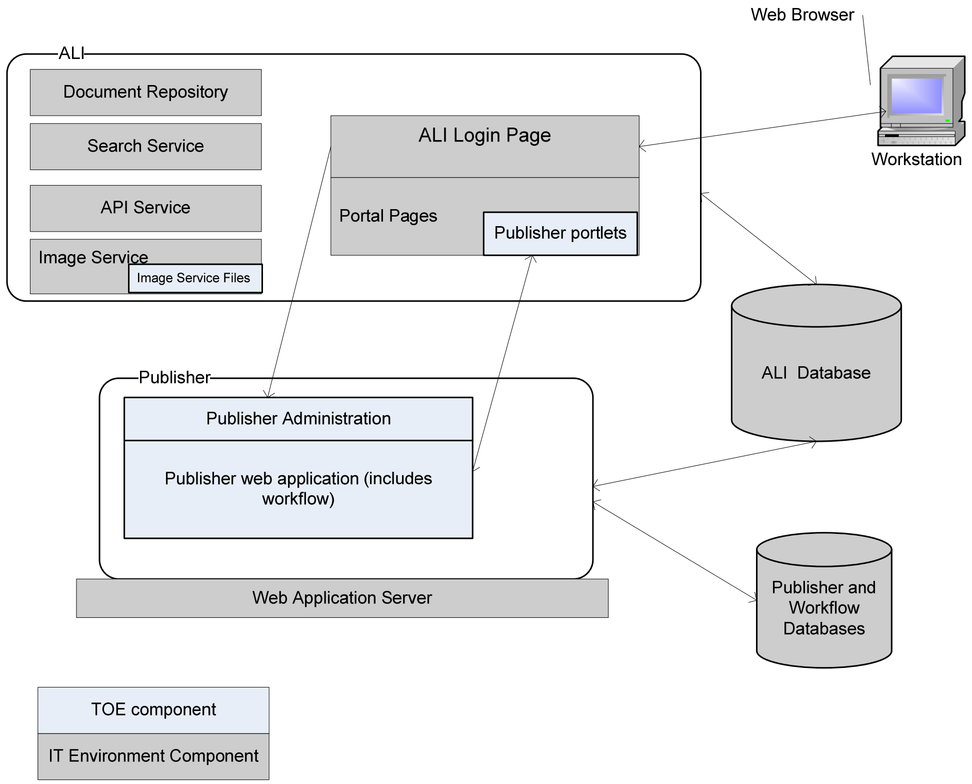

**Figure 1: Publisher and its IT Environment**

The following table lists the elements of the IT environment for the evaluated configurations.

**Table 2-1 IT Environment Components**

| IT Environment Components | Supported Versions/Descriptions |
|---|---|
| Operating systems | Microsoft Windows Server 2003 SP1<br>Solaris 10 (on SPARC)<br>Red Hat Enterprise Linux 4 Update 3 (x86)<br>*Operating systems host all of the components identified below.* |
| Web Application Server | JBoss Application Server, version 3.2.7 (bundled and installed with Publisher)<br>*Publisher is deployed on the JBoss application server, which is a separate product and not part of the TOE. The JBoss application server is not a product produced by BEA. JBoss is part of the IT environment for the TOE and is bundled and installed with Publisher as a convenience to the user.* |
| AquaLogic Interaction (ALI) | 6.1 MP1 |

| IT Environment Components | Supported Versions/Descriptions |
|---|---|
| | *ALI Includes the following components:*<br>• ***ALI Login Page****. This web page is part of the ALI user interface and is used by both administrators and non-administrative users. The* **Log In** *page enables users to log into their ALI portal.*<br>• ***API Service****. A component of ALI, the API Service enables remote client applications to call into ALI from other machines on the network. Publisher is one such remote client and makes calls to ALI to synchronize community users in ALI with the users of community projects that exist in Collaboration and with other projects that are accessible to ALI community members.*<br>• ***Document Repository****. Stores documents and files uploaded by ALUI components. This is where the Publisher folders are stored.*<br>• ***Image Service****. Serves images and other static content for ALI and its component ALUI products, such as Publisher.*<br>• ***ALI Portal Pages****. Web pages that are part of the ALI portal. These pages include the Login page, users' personal My Page, community pages, and the Knowledge Directory.*<br>• ***Search Service****. A component of ALI that returns indexed content stored in the ALI portal. The Search Service returns content that is indexed in the AquaLogic User Interaction system from the portal, Collaboration, and Publisher. Content that is indexed in the ALUI system includes documents, portlets, communities, and users as well as many other ALUI objects.* |
| Database Servers | Microsoft SQL Server 2005 (with SQL Server 2000 compatibility level)<br>Oracle 10g R2 (10.2.0.1 and above) in default or Oracle RAC configuration<br>*The ALI Database Server contains the ALI database tables including tables used to define Publisher instances. The ALI database stores portal objects, such as user and group configurations, document records, and administrative objects. The ALI database does not store the documents available through the portal.*<br>*The Publisher and Workflow Databases are used to store all Publisher-specific data.* |
| Web browsers | Internet Explorer 6.0 SP2<br>Mozilla Firefox 1.5<br>*Web browsers are used by clients (i.e.,, administrators and users) to access functions of the product.* |

### 2.2.2 Logical Boundaries

The logical boundary consists of the security functionality of TOE is summarized below.

#### 2.2.2.1 User data protection

The TOE defines a role-based access control policy to control the users that can access and act upon the TOE defined objects. This access control policy works in conjunction with the access control policy defined in ALI. The users of the TOE are defined, managed and maintained by ALI.

#### 2.2.2.2 Security management

The TOE provides the authorized administrator an interface to manage access control attributes, assignment of roles, and security-related functions as 'folder-level' security that supplements the security provided by ALI. The TOE depends upon ALI to define the administrators of the TOE.

#### 2.2.2.3 Protection of the TSF

The TOE enforces the access control policy to ensure that the security functions can not be bypassed. The TOE depends on its operating environment to store, protect, and ensure that the TOE functions are not tampered with or bypassed. The TOE leverages the security functions offered by ALI to ensure that users of the TOE are identified and authenticated before access to the TOE is granted. The TOE dependents upon ALI to define, maintain, and manage the administrator groups of the TOE, and the users, user groups, and community members that can be assigned to the roles in the TOE.

## 2.3 TOE Documentation

The TOE has a number of administrative, user and installation guides for the TOE. These documents and others are described in section 6.2.

# 3. Security Environment

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

- Assumptions made on the operational environment and the method of use intended for the TOE.
- Threat that the TOE and the IT environment mitigates.

## 3.1 Threats

T.ACCESS A user may gain unauthorized access to the TOE and the TOE's protected objects.

T.MANAGE A user may gain unauthorized access to the utilities available to manage the security-related functions of the TOE.

## 3.2 Assumptions

A.AUTH_USERS Only the users authorized to access the information within the TOE may access the TOE.

A.INSTALL Those responsible for the TOE must ensure that the TOE is delivered, installed, managed, and operated in a manner to maintain to the IT security objectives.

A.NOEVIL The administrative personnel are not careless, willfully negligent, or hostile and will follow and abide by the instructions provided by the administrative guidance.

A.PHYSICAL The TOE software critical to security policy enforcement will be protected from unauthorized physical modification.

A.OPE_ENV The TOE operating environment shall provide the mechanism to isolate the TOE components and resources, ensure the TOE cannot be tampered with or bypassed and the TOE data is protected from unauthorized deletions.

A.TRANSMIT The operating environment will protect the data transmitted to and from the TOE.

A.USER The authorized users will not be negligent or malicious and will follow the guidance provided.

# 4. Security Objectives

This section defines the security objectives of the TOE and its supporting environment. Security objectives, categorized as either IT security objectives or non-IT security objectives, reflect the stated intent to satisfy identified assumptions and mitigate the identified threats. All of the identified assumptions and threats are addressed under one of the categories below.

## 4.1 Security Objectives for the TOE

O.ACCESS The TSF shall restrict access of the TOE defined objects to specified users and users with appropriate privileges.  The TSF must allow authorized users to specify which users may access their objects and the actions performed on the objects.

O.MANAGE The TOE shall allow administrators to effectively manage the TOE and its security functions, and must ensure that only authorized users with the administrative privileges are able to access the functions.

## 4.2 Security Objectives for the IT Environment

OE.ACCESS The IT Environment shall restrict access of objects to identified users. The IT Environment must allow authorized administrator to specify which users may access the objects and the operations that may be performed.

OE.AUTH The operating environment shall ensure that all users have been identified and authenticated before access to the TOE is permitted.

OE.MANAGE The environment shall allow environment administrators to effectively manage the environment and its security functions.

OE.OPE_ENV The TOE operating environment shall provide the mechanism to isolate the TOE components and resources, ensure the TOE cannot be tampered with or bypassed and the TOE data is protected from unauthorized deletions.

## 4.3 Security Objectives for the Environment

OE.ADMIN The TOE administrators shall be competent, trustworthy, trained in the proper operation of the TOE and will follow the guidance provided.

OE.INSTALL Those responsible for the TOE must ensure that the TOE is delivered, installed, managed, and operated in a manner consistent with the IT security objectives.

OE.PHYSICAL The TOE software critical to security policy enforcement will be protected from unauthorized physical modification.

OE.TRANSMIT The operating environment shall protect the data transmitted by the TOE from disclosure.

OE.USER The authorized users will not be negligent or malicious and will follow the guidance provided.

# 5. IT Security Requirements

This section defines the security functional requirements satisfied by the TOE and security assurance requirements levying against the TOE in an evaluation. The security functional requirements are drawn from the CC Part 2.

## 5.1 TOE Security Functional Requirements

The following table identifies the SFRs satisfied by TOE.

**Table 5-1 TOE Security Functional Components**

| Requirement Class | Requirement Component |
|---|---|
| **FDP: User data protection** | FDP_ACC.1a: Subset access control |
| | FDP_ACF.1a: Security attribute based access control |
| **FMT: Security management** | FMT_MSA.1a: Management of security attributes |
| | FMT_MSA.3a: Static attribute initialization |
| | FMT_MTD.1a: Management of TSF data |
| | FMT_MTD.1b: Management of TSF data |
| | FMT_SMF.1a: Specification of Management Functions |
| | FMT_SMR.1a: Security roles |
| **FPT: Protection of the TSF** | FPT_RVM.1a: Non-bypassability of the TSP |

### 5.1.1 User data protection (FDP)

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

**FDP_ACC.1a.1** The TSF shall enforce the [**Publisher Access Control Policy**] on [

- **subjects: Users, Crawler Definers, and Crawlers;**
- **objects: folders;**
- **operations: the actions listed in table 6-1**].

**Application Note:** The primary distinction between Users, Crawler Definers and Crawlers is the interface used to access protected objects. The interfaces applicable to each type of subject are protected by the underlying ALI product. The User interface is intended for general purpose users of the TOE that will use the provided object management mechanisms to share information in controlled ways. The Crawler Definer interface is used to browse available document folders in order to select those that should be 'crawled' by a Crawler. The Crawler interface is for special purpose content crawlers that serve to catalog available information for subsequent use (via the other interface). In the context of these requirements, Crawlers and Crawler Definers are identified as security roles that are not available to Users as identified here.

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

**FDP_ACF.1a.1** The TSF shall enforce the [**Publisher Access Control Policy**] to objects based on the following: [

- **subject:**
  - **(non-Crawler and Crawler Definer) Users: identity and groups;**
  - **Crawlers: role;**
  - **Crawler Definer: identity, groups, and role;**
- **object:**

- **folders – folder security list**].

**FDP_ACF.1a.2** The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [

i. **Users are assigned to a role for a given folder if their user identity or an assigned group is assigned to that role in the folder's folder security list;**

ii. **Users can only perform operations as defined in table 6-1 on identified folders based on the highest access level assigned via their role(s) by the applicable folder security list:**

- **if the folder has an explicit folder security list and the user is not in the Publisher Administrator role, the explicit folder security list is used to determine the user's role(s); or**
- **if the folder doesn't have an explicit folder security list and the user is not in the Publisher Administrator role, the next folder security list defined higher in the folder hierarchy is used to determine the user's role(s);**

iii. **Crawler Definers are subject to rules i and ii above except that**

- **User is treated as Crawler Definer;**
- **the only applicable operation is to view folders; and**
- **at least the Reader access level is required in order to view each folder**].

**FDP_ACF.1a.3** The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: [

- **Crawlers can access the contents of any folder and**
- **Publisher Administrators can perform any functions on any folder**].

**FDP_ACF.1a.4** The TSF shall explicitly deny access of subjects to objects based on the [**no explicitly deny access rules**].

**Application Note:** Table 6-1 in section 6.1.1 outlines the available operations.

**Application Note:** Note that a User can potentially belong to more than one role. When that occurs, they will effectively have a superset of the access associated with the applicable roles for all applicable objects. Note also that the Publisher Administrator role is defined outside the context of specific folders and as such membership is determinable regardless of folder security lists.

## 5.1.2 Security management (FMT)

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

**FMT_MSA.1a.1** The TSF shall enforce the [**Publisher Access Control Policy**] to restrict the ability to [***modify***] the security attributes [**folder security list**] to [**Publisher and Folder Administrators**].

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

**FMT_MSA.3a.1** The TSF shall enforce the [**Publisher Access Control Policy**] to provide [***[inherited]***] default values for security attributes that are used to enforce the SFP.

**FMT_MSA.3a.2** The TSF shall allow the [**Publisher and Folder Administrators**] to specify alternative initial values to override the default values when an object or information is created.

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

**FMT_MTD.1a.1** The TSF shall restrict the ability to [***modify, copy, delete, [create], [enable], [disable]***] the [**workflow**] to [**Workflow Administrators**].

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

**FMT_MTD.1b.1** The TSF shall restrict the ability to [***assign***] the [**user and groups to workflow activities**] to [**Workflow Administrators**].

#### 5.1.2.5 Specification of Management Functions (FMT_SMF.1a)

**FMT_SMF.1a.1** The TSF shall be capable of performing the following security management functions: [**management of folder security list as specified in FMT_MSA.1a, management of the workflows as specified in FMT_MTD.1a and FMT_MTD.1b**].

#### 5.1.2.6 Security roles (FMT_SMR.1a)

**FMT_SMR.1a.1** The TSF shall maintain the roles [**Folder Administrator, Crawler Definers, and Crawlers**].

**FMT_SMR.1a.2** The TSF shall be able to associate users with roles.

**Application Note:** The Folder Administrator role is limited to those folders where the applicable user has been assigned that role.

### 5.1.3 Protection of the TSF (FPT)

#### 5.1.3.1 Non-bypassability of the TSP (FPT_RVM.1a)

**FPT_RVM.1a.1** The TSF shall ensure that TSP enforcement functions are invoked and succeed before each function within the TSC is allowed to proceed.

## 5.2 IT Environment Security Functional Requirements

The following table identifies the SFRs satisfied by the IT environment of TOE.

**Table 5-2 IT Environment Security Functional Components**

| Requirement Class | Requirement Component |
|---|---|
| **FDP: User data protection** | FDP_ACC.1b: Subset access control |
| | FDP_ACF.1b: Security attribute based access control |
| **FIA: Identification and authentication** | FIA_ATD.1: User attribute definition |
| | FIA_UAU.2: User authentication before any action |
| | FIA_UID.2: User identification before any action |
| **FMT: Security management** | FMT_MSA.1b: Management of security attributes |
| | FMT_MSA.3b: Static attribute initialization |
| | FMT_SMF.1b: Specification of Management Functions |
| | FMT_SMR.1b: Security roles |
| **FPT: Protection of the TSF** | FPT_RVM.1b: Non-bypassability of the TSP |
| | FPT_SEP.1: TSF domain separation |

### 5.2.1 User Data Protection

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

**FDP_ACC.1b.1** The ~~TSF~~ **IT Environment** shall enforce the [**ALI Access Control Policy**] on [

- **subjects: users;**
- **objects: Administrative folders, authentication sources, communities, community templates, content crawlers, content sources, content types, experience definitions, external operations, federated searches, filters, groups, invitations, jobs, pages, page**

**templates, portlets, portlet bundles, portlet templates, profile sources, properties, remote servers, snapshot queries, Web services;**

- **operations: view, modify, create, copy, move, delete**].

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

**FDP_ACF.1b.1** The ~~TSF~~ **IT Environment** shall enforce the [**ALI Access Control Policy**] to objects based on the following: [**subjects: Users: user name, group name (object creation activity rights are conferred via the group membership);**
**objects: ACL**].

**FDP_ACF.1b.2** The ~~TSF~~ **IT Environment** shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [

- **The move operation is granted if the ACL associated to the parent folder, destination folder, and the object grants the user name/group name permission.**
- **The create operation is granted if the user name via the user's group name has the appropriate object creation activity right.**
- **The copy operation is granted if user name via the user's group name has the appropriate object creation activity right and the ACL associated to the parent folder, destination folder, and the object grants the user name/group name permission.**
- **The other requested operations are granted if the ACL associated to the parent folder and the object grants the user name/group name permission**].

**FDP_ACF.1b.3** The ~~TSF~~ **IT Environment** shall explicitly authorize access of subjects to objects based on the following additional rules: [

- **Read access to authentication source, content types, filters, invitations, and properties is explicitly granted to all authorized users,**
- **The authorized administrator (built-in administrator or user in the Administrator group) is granted all access to the object**].

**FDP_ACF.1b.4** The ~~TSF~~ **IT Environment** shall explicitly deny access of subjects to objects based on the [**no explicitly deny access rules**].

### 5.2.2 Identification and authentication (FIA)

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

**FIA_ATD.1.1** The ~~TSF~~ **IT Environment** shall maintain the following list of security attributes belonging to individual users: [**user name, password, and groups**].

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

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

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

**FIA_UID.2.1** The ~~TSF~~ **IT Environment** shall require each user to identify itself before allowing any other TSF-mediated actions on behalf of that user.

### 5.2.3 Security management (FMT)

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

**FMT_MSA.1b.1** The ~~TSF~~ **IT Environment** shall enforce the [**ALI Access Control Policy**] to restrict the ability to [**assign to groups**] the security attributes [**activity rights**] to [**authorized administrator**].

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

**FMT_MSA.3b.1** The ~~TSF~~ **IT Environment** shall enforce the [**ALI Access Control Policy**] to provide [***[inherited]***] default values for security attributes that are used to enforce the SFP.

**FMT_MSA.3b.2** The ~~TSF~~ **IT Environment** shall allow the [**authorized administrator**] to specify alternative initial values to override the default values when an object or information is created.

#### 5.2.3.3 Specification of Management Functions (FMT_SMF.1b)

**FMT_SMF.1b.1** The ~~TSF~~ **IT Environment** shall be capable of performing the following security management functions: [**management of groups as specified in FMT_MSA.1b**].

#### 5.2.3.4 Security roles (FMT_SMR.1b)

**FMT_SMR.1b.1** The ~~TSF~~ **IT Environment** shall maintain the roles [**Publisher Administrator, Workflow Administrator, authorized administrator, and authorized user**].

**FMT_SMR.1b.2** The ~~TSF~~ **IT Environment** shall be able to associate users with roles.

### 5.2.4 Protection of the TSF (FPT)

#### 5.2.4.1 Non-bypassability of the TSP (FPT_RVM.1b)

**FPT_RVM.1b.1** The ~~TSF~~ **IT Environment** shall ensure that TSP enforcement functions are invoked and succeed before each function within the TSC is allowed to proceed.

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

**FPT_SEP.1.1** The ~~TSF~~ **IT Environment** shall maintain a security domain for its own execution that protects it from interference and tampering by untrusted subjects.

**FPT_SEP.1.2** The ~~TSF~~ **IT Environment** shall enforce separation between the security domains of subjects in the TSC.

## 5.3 TOE Security Assurance Requirements

The security assurance requirements for the TOE are the EAL 2 augmented with ALC_FLR.2 components as specified in Part 3 of the Common Criteria. No operations are applied to the assurance components.

**Table 5-3 EAL 2 augmented with ALC_FLR.2 Assurance Requirements**

| Requirement Class | Requirement Component |
|---|---|
| **ACM: Configuration management** | ACM_CAP.2: Configuration items |
| **ADO: Delivery and operation** | ADO_DEL.1: Delivery procedures |
| | ADO_IGS.1: Installation, generation, and start-up procedures |
| **ADV: Development** | ADV_FSP.1: Informal functional specification |
| | ADV_HLD.1: Descriptive high-level design |
| | ADV_RCR.1: Informal correspondence demonstration |
| **AGD: Guidance documents** | AGD_ADM.1: Administrator guidance |
| | AGD_USR.1: User guidance |
| **ALC: Life cycle support** | ALC_FLR.2: Flaw reporting procedures |
| **ATE: Tests** | ATE_COV.1: Evidence of coverage |
| | ATE_FUN.1: Functional testing |
| | ATE_IND.2: Independent testing - sample |

| | |
|---|---|
| **AVA: Vulnerability assessment** | AVA_SOF.1: Strength of TOE security function evaluation |
| | AVA_VLA.1: Developer vulnerability analysis |

## 5.3.1 Configuration management (ACM)

### 5.3.1.1 Configuration items (ACM_CAP.2)

**ACM_CAP.2.1d** The developer shall provide a reference for the TOE.

**ACM_CAP.2.2d** The developer shall use a CM system.

**ACM_CAP.2.3d** The developer shall provide CM documentation.

**ACM_CAP.2.1c** The reference for the TOE shall be unique to each version of the TOE.

**ACM_CAP.2.2c** The TOE shall be labelled with its reference.

**ACM_CAP.2.3c** The CM documentation shall include a configuration list.

**ACM_CAP.2.4c** The configuration list shall uniquely identify all configuration items that comprise the TOE.

**ACM_CAP.2.5c** The configuration list shall describe the configuration items that comprise the TOE.

**ACM_CAP.2.6c** The CM documentation shall describe the method used to uniquely identify the configuration items that comprise the TOE.

**ACM_CAP.2.7c** The CM system shall uniquely identify all configuration items that comprise the TOE.

**ACM_CAP.2.1e** The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

## 5.3.2 Delivery and operation (ADO)

### 5.3.2.1 Delivery procedures (ADO_DEL.1)

**ADO_DEL.1.1d** The developer shall document procedures for delivery of the TOE or parts of it to the user.

**ADO_DEL.1.2d** The developer shall use the delivery procedures.

**ADO_DEL.1.1c** The delivery documentation shall describe all procedures that are necessary to maintain security when distributing versions of the TOE to a user’s site.

**ADO_DEL.1.1e** The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

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

**ADO_IGS.1.1d** The developer shall document procedures necessary for the secure installation, generation, and start-up of the TOE.

**ADO_IGS.1.1c** The installation, generation and start-up documentation shall describe all the steps necessary for secure installation, generation and start-up of the TOE.

**ADO_IGS.1.1e** The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

**ADO_IGS.1.2e** The evaluator shall determine that the installation, generation, and start-up procedures result in a secure configuration.

## 5.3.3 Development (ADV)

### 5.3.3.1 Informal functional specification (ADV_FSP.1)

**ADV_FSP.1.1d** The developer shall provide a functional specification.

**ADV_FSP.1.1c** The functional specification shall describe the TSF and its external interfaces using an informal style.

**ADV_FSP.1.2c** The functional specification shall be internally consistent.

**ADV_FSP.1.3c** The functional specification shall describe the purpose and method of use of all external TSF interfaces, providing details of effects, exceptions and error messages, as appropriate.

**ADV_FSP.1.4c** The functional specification shall completely represent the TSF.

**ADV_FSP.1.1e** The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

**ADV_FSP.1.2e** The evaluator shall determine that the functional specification is an accurate and complete instantiation of the TOE security functional requirements.

### 5.3.3.2 Descriptive high-level design (ADV_HLD.1)

**ADV_HLD.1.1d** The developer shall provide the high-level design of the TSF.

**ADV_HLD.1.1c** The presentation of the high-level design shall be informal.

**ADV_HLD.1.2c** The high-level design shall be internally consistent.

**ADV_HLD.1.3c** The high-level design shall describe the structure of the TSF in terms of subsystems.

**ADV_HLD.1.4c** The high-level design shall describe the security functionality provided by each subsystem of the TSF.

**ADV_HLD.1.5c** The high-level design shall identify any underlying hardware, firmware, and/or software required by the TSF with a presentation of the functions provided by the supporting protection mechanisms implemented in that hardware, firmware, or software.

**ADV_HLD.1.6c** The high-level design shall identify all interfaces to the subsystems of the TSF.

**ADV_HLD.1.7c** The high-level design shall identify which of the interfaces to the subsystems of the TSF are externally visible.

**ADV_HLD.1.1e** The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

**ADV_HLD.1.2e** The evaluator shall determine that the high-level design is an accurate and complete instantiation of the TOE security functional requirements.

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

**ADV_RCR.1.1d** The developer shall provide an analysis of correspondence between all adjacent pairs of TSF representations that are provided.

**ADV_RCR.1.1c** For each adjacent pair of provided TSF representations, the analysis shall demonstrate that all relevant security functionality of the more abstract TSF representation is correctly and completely refined in the less abstract TSF representation.

**ADV_RCR.1.1e** The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

## 5.3.4 Guidance documents (AGD)

### 5.3.4.1 Administrator guidance (AGD_ADM.1)

**AGD_ADM.1.1d** The developer shall provide administrator guidance addressed to system administrative personnel.

**AGD_ADM.1.1c** The administrator guidance shall describe the administrative functions and interfaces available to the administrator of the TOE.

**AGD_ADM.1.2c** The administrator guidance shall describe how to administer the TOE in a secure manner.

**AGD_ADM.1.3c** The administrator guidance shall contain warnings about functions and privileges that should be controlled in a secure processing environment.

**AGD_ADM.1.4c** The administrator guidance shall describe all assumptions regarding user behaviour that are relevant to secure operation of the TOE.

**AGD_ADM.1.5c** The administrator guidance shall describe all security parameters under the control of the administrator, indicating secure values as appropriate.

**AGD_ADM.1.6c** The administrator guidance shall describe each type of security-relevant event relative to the administrative functions that need to be performed, including changing the security characteristics of entities under the control of the TSF.

**AGD_ADM.1.7c** The administrator guidance shall be consistent with all other documentation supplied for evaluation.

**AGD_ADM.1.8c** The administrator guidance shall describe all security requirements for the IT environment that are relevant to the administrator.

**AGD_ADM.1.1e** The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

### 5.3.4.2 User guidance (AGD_USR.1)

**AGD_USR.1.1d** The developer shall provide user guidance.

**AGD_USR.1.1c** The user guidance shall describe the functions and interfaces available to the non-administrative users of the TOE.

**AGD_USR.1.2c** The user guidance shall describe the use of user-accessible security functions provided by the TOE.

**AGD_USR.1.3c** The user guidance shall contain warnings about user-accessible functions and privileges that should be controlled in a secure processing environment.

**AGD_USR.1.4c** The user guidance shall clearly present all user responsibilities necessary for secure operation of the TOE, including those related to assumptions regarding user behaviour found in the statement of TOE security environment.

**AGD_USR.1.5c** The user guidance shall be consistent with all other documentation supplied for evaluation.

**AGD_USR.1.6c** The user guidance shall describe all security requirements for the IT environment that are relevant to the user.

**AGD_USR.1.1e** The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

## 5.3.5 Life cycle support (ALC)

### 5.3.5.1 Flaw reporting procedures (ALC_FLR.2)

**ALC_FLR.2.1d** The developer shall provide flaw remediation procedures addressed to TOE developers.

**ALC_FLR.2.2d** The developer shall establish a procedure for accepting and acting upon all reports of security flaws and requests for corrections to those flaws.

**ALC_FLR.2.3d** The developer shall provide flaw remediation guidance addressed to TOE users.

**ALC_FLR.2.1c** The flaw remediation procedures documentation shall describe the procedures used to track all reported security flaws in each release of the TOE.

**ALC_FLR.2.2c** The flaw remediation procedures shall require that a description of the nature and effect of each security flaw be provided, as well as the status of finding a correction to that flaw.

**ALC_FLR.2.3c** The flaw remediation procedures shall require that corrective actions be identified for each of the security flaws.

**ALC_FLR.2.4c** The flaw remediation procedures documentation shall describe the methods used to provide flaw information, corrections and guidance on corrective actions to TOE users.

**ALC_FLR.2.5c** The flaw remediation procedures documentation shall describe a means by which the developer receives from TOE users reports and enquiries of suspected security flaws in the TOE.

**ALC_FLR.2.6c** The procedures for processing reported security flaws shall ensure that any reported flaws are corrected and the correction issued to TOE users.

**ALC_FLR.2.7c** The procedures for processing reported security flaws shall provide safeguards that any corrections to these security flaws do not introduce any new flaws.

**ALC_FLR.2.8c** The flaw remediation guidance shall describe a means by which TOE users report to the developer any suspected security flaws in the TOE.

**ALC_FLR.2.1e** The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

## 5.3.6 Tests (ATE)

### 5.3.6.1 Evidence of coverage (ATE_COV.1)

**ATE_COV.1.1d** The developer shall provide evidence of the test coverage.

**ATE_COV.1.1c** The evidence of the test coverage shall show the correspondence between the tests identified in the test documentation and the TSF as described in the functional specification.

**ATE_COV.1.1e** The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

### 5.3.6.2 Functional testing (ATE_FUN.1)

**ATE_FUN.1.1d** The developer shall test the TSF and document the results.

**ATE_FUN.1.2d** The developer shall provide test documentation.

**ATE_FUN.1.1c** The test documentation shall consist of test plans, test procedure descriptions, expected test results and actual test results.

**ATE_FUN.1.2c** The test plans shall identify the security functions to be tested and describe the goal of the tests to be performed.

**ATE_FUN.1.3c** The test procedure descriptions shall identify the tests to be performed and describe the scenarios for testing each security function. These scenarios shall include any ordering dependencies on the results of other tests.

**ATE_FUN.1.4c** The expected test results shall show the anticipated outputs from a successful execution of the tests.

**ATE_FUN.1.5c** The test results from the developer execution of the tests shall demonstrate that each tested security function behaved as specified.

**ATE_FUN.1.1e** The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

#### 5.3.6.3 Independent testing - sample (ATE_IND.2)

**ATE_IND.2.1d** The developer shall provide the TOE for testing.

**ATE_IND.2.1c** The TOE shall be suitable for testing.

**ATE_IND.2.2c** The developer shall provide an equivalent set of resources to those that were used in the developer's functional testing of the TSF.

**ATE_IND.2.1e** The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

**ATE_IND.2.2e** The evaluator shall test a subset of the TSF as appropriate to confirm that the TOE operates as specified.

**ATE_IND.2.3e** The evaluator shall execute a sample of tests in the test documentation to verify the developer test results.

### 5.3.7 Vulnerability assessment (AVA)

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

**AVA_SOF.1.1d** The developer shall perform a strength of TOE security function analysis for each mechanism identified in the ST as having a strength of TOE security function claim.

**AVA_SOF.1.1c** For each mechanism with a strength of TOE security function claim the strength of TOE security function analysis shall show that it meets or exceeds the minimum strength level defined in the PP/ST.

**AVA_SOF.1.2c** For each mechanism with a specific strength of TOE security function claim the strength of TOE security function analysis shall show that it meets or exceeds the specific strength of function metric defined in the PP/ST.

**AVA_SOF.1.1e** The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

**AVA_SOF.1.2e** The evaluator shall confirm that the strength claims are correct.

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

**AVA_VLA.1.1d** The developer shall perform a vulnerability analysis.

**AVA_VLA.1.2d** The developer shall provide vulnerability analysis documentation.

**AVA_VLA.1.1c** The vulnerability analysis documentation shall describe the analysis of the TOE deliverables performed to search for obvious ways in which a user can violate the TSP.

**AVA_VLA.1.2c** The vulnerability analysis documentation shall describe the disposition of obvious vulnerabilities.

**AVA_VLA.1.3c** The vulnerability analysis documentation shall show, for all identified vulnerabilities, that the vulnerability cannot be exploited in the intended environment for the TOE.

**AVA_VLA.1.1e** The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

**AVA_VLA.1.2e** The evaluator shall conduct penetration testing, building on the developer vulnerability analysis, to ensure obvious vulnerabilities have been addressed.

# 6. TOE Summary Specification

This chapter describes the security functions and associated assurance measures.

## 6.1 TOE Security Functions

### 6.1.1 User data protection

Publisher enforces a role-based access control policy on Publisher folders where the users are assigned to roles in the folder security list. The access control policy determines if the user is able to perform the requested operation based on the role associated to the user or the user's groups for the folder.

The authorized administrator can modify the folder security by doing the following:

- Add users and groups to the Publisher folder
- Assign a specific Publisher role to each newly added user and group

The TOE provides two ways for the authorized administrator to assign roles to Publisher users:

- Explicitly assign roles to users and groups in each folder
- Associate roles to users and groups based on the ALI-defined object access level of ALI[2] communities, folders, portlets, and portlet templates.

The Publisher folder Content Security page is used to assign users to folders and to associate the users to specific Publisher roles.

When an ALI object ACL is used to associate users to roles, Publisher automatically adds all of the users and groups on the ALI object's ACL to the Publisher security list for the folder using the specified (using the Content Security page) Publisher role for each ALI access privilege. For example, users with Read access to an ALI community might be assigned the Reader role in Publisher. If a user is associated to the more than one role on the folder, the access control policy will enforce the highest role associated to the user. A user may be assigned different roles in different folders.

Table 6-1 outlines the operations a user can perform on the folders and it contents based on the role the user is assigned to in the folder. The roles are hierarchical and each higher level role is permitted all operations of the roles below it in the hierarchy.

**Table 6-1 Publisher Roles and Functions Matrix**

| Roles with Access | Function |
|---|---|
| **Administrator** | Create, edit, copy, and delete top-level folders and all functions of the other roles. |

[2] ALI is used in this section to refer to the AquaLogic Interaction 6.1 MP1 portal TOE which has been evaluated independently and is a component of the Publisher TOE environment in the context of this Security Target.

| Roles with Access | Function |
|---|---|
| **Folder Administrator** | Attach workflow to a folder in Publisher Explorer<br>Copy, rename, move, and delete top-level folders<br>Detach portlet from folder<br>Override workflow<br>Security: assign users, groups, and roles to folders<br>Undo checkout (by other user)<br>Includes functions permitted to the Producer, Editor, Contributor, Submitter, and Reader roles |
| **Producer** | Create, delete, copy, move, and rename lower-level folders (move requires permission on both the move from and move to folders)<br>Create new Data Entry Templates, Presentation Templates, and selection lists<br>Edit content items assigned to another user in workflow<br>Set publishing targets<br>Includes functions permitted to the Editor, Contributor, Submitter, and Reader roles |
| **Editor** | Publish to directory<br>View and edit publishing information in Content Item Editor<br>Publish content item or folder<br>Schedule publishing and expiration<br>Includes functions permitted to the Contributor, Submitter, and Reader roles |
| **Contributor** | View and restore versions in Content Item Editor<br>Includes functions permitted to the  Submitter and Reader roles |
| **Submitter** | Set user Quicklinks and preferences in Publisher Explorer<br>Preview content items<br>Check content items in and out<br>Create, edit, copy, delete, and rename content items<br>Use WebEdit when creating and editing content items<br>Use the Copy to function in Publisher Explorer (security applies to the folder copied to)<br>Use the Move to function in Publisher Explorer (security applies to the folder moved to<br>Create a new folder in a delivered Community Directory portlet<br>Access WebDAV and set up a Web folder for Publisher<br>Includes functions permitted to Reader role |

| Roles with Access | Function |
|---|---|
| **Reader** | Browse, search, and view published content portlets<br>Browse, search, and view intrinsic Publisher portlets<br>Be assigned to workflow activities<br>Approve or reject a work item |

By default, when a parent or first level folder is created in the Publisher, the Administrator role has full access, all other roles are denied access. Subfolders inherit the security of the parent folder by default. However, this can be overridden as explained in section 6.1.2, below.

**Content Crawlers**

‘Crawlers’ use a special purpose interface in order to “crawl” the applicable folders. This interface, like those used by normal users, is instantiated and controlled by the underlying ALI product. Any user that can access the crawler interface is in effect a crawler and can only access the limited crawler functions (collect descriptive information, properties and link/location) which do not include any of the operations identified above.

Note that there is a special purpose interface made available to support the definition of crawlers. Users of this interface (i.e., known as ‘Crawler Definers’ in this Security Target) are able only to select folders, to which they have at least the Reader access level, to be subsequently crawled by the resulting crawler.

The User data protection function is designed to satisfy the following security functional requirements:

- FDP_ACC.1a: The TOE defines an access control policy for the access to the TOE objects.
- FDP_ACF.1a: The TOE defines an access control policy based on the roles associated to users associated to objects.
- FMT_MSA.3a: The TOE defines a default folder security list to newly created parent or top level folders which allows full access to the Administrator role and denies access to all other roles. The default folder security list can be edited after creation. The TOE implements an inheritance policy where newly created child folders inherit the folder security list associated to the parent. Publisher and Folder Administrators can specify alternative initial values by specifying a folder security list for a portlet template or portlet when using the Configure Portlet Template and Configure Portlet wizards. Otherwise, the default folder security list cannot be altered by any user prior to association to a folder.

### 6.1.2 Security management

The TOE provides web-based administrative interfaces to manage the access control policy and workflow functions of the TOE. Some administrative interfaces are accessed via the ALI Administrative Portal. The Publisher folder-level security interfaces are accessed via the Content Security page of the Publisher Explorer interface.

Publisher has two types of authorized administrators to manage the folders, each with a different scope of permissible action on the TOE’s security and a third administrator to manage the workflow:

- Publisher Administrators - users and user groups assigned the Administrator Publisher activity right or users who are members of the ALI Administrators group.
- Folder Administrators – users and user groups assigned to the Folder Administrator on a folder. The Folder Administrator is only able to manage assigned folders.
- Workflow Administrators – users in groups assigned the Configure Workflow activity right. A workflow administrator is able to manage the workflow by accessing Workflow Administration interfaces to create workflow definitions. Note: there is not an explicit Publisher role called workflow administrator.

Publisher activity rights are assigned using ALI interfaces. The Administer Publisher activity right allows the management of Publisher functions with the exception of the management of workflow. The Configure Workflow activity right allows users to manage the workflow. Note that since Publisher and Workflow Administrators are based on Publisher activity rights, those roles are implemented outside the TOE, unlike Folder Administrators.

Publisher Administrators and Folder Administrators use the Content Security page to assign users, groups, and ALI object ACL's access levels to roles to the folders in Publisher. Once a Publisher Administrator has configured the folder security list for the top-level folders, the Folder Administrator can configure the folder security list for any folders for which they have that role, including top-level folders. By default, folder security list settings are inherited by the lower-level folders from the top-level folders, unless an authorized administrator overrides the security inheritance for the lower-level folder. The inherited Publisher Administrator and Folder Administrator role assignments for a folder cannot be overridden.

A Workflow Administrator performs the following tasks using the Workflow Editor interface:

- Define the workflow by defining the work item activities and their sequence.
- Enable the workflow. Only enabled workflows can be attached to Publisher folders.
- Define the work item activity properties as follows:
    - Specify the work item activity name and description
    - Assign or remove users to workflow activities**.** Users assigned to an activity in a workflow are called *assignees*. Assigning an individual or a group to an activity restricts access to work items in that activity as long as they remain in that activity. All activities in a workflow must have assignees in order for that workflow to be available for attachment to a Publisher folder. Although users in the Editor role and above can edit a content item at any time, Contributors and Submitters cannot edit a content item unless they are assigned to the current activity as individuals or members of a group.
    - Enable users in the Editor role or higher to publish a content item as part of the activity.
    - Enable users to delegate an activity to another user. Delegation enables users to reassign an activity to another user, while retaining the ability to reclaim and handle the activity themselves.
    - Enable users to transfer an activity to another user. Transferring, unlike delegation, enables a user to reassign a workflow activity to another user without retaining the ability to handle the activity themselves.

Note that Assignees must also have the proper security access for any folders to which a workflow is attached.

In addition to normal users and their roles as described above, a special purpose crawler interface is available for the purpose of crawling folders to catalog and index the contents. Users of this interface have access only to the limited crawler functions and are not specifically or individually identified, unlike other users. As such, '**Crawlers**' is treated as a distinct role for the purpose of differentiating those users in the context of this Security Target. Similarly, a special purpose interface is available for the purpose of defining crawlers. Users of this interface are identified as '**Crawler Definers'** in the requirements of this Security Target, and the interface facilitates only the selection of folders to be crawled by the defined crawler.

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

- FMT_MSA.1a: The TOE restricts the Content Security page to the authorized administrator. The page is used to modify the folder security list. The page allows the authorized administrator to assign user, groups, ALI's object ACL to the pre-defined roles of the folder security list.
- FMT_MTD.1a, FMT_MTD.1b: The TOE restricts the interface to manage the workflow and its items to the Workflow Administrator. The Workflow editor interface allows the Workflow administrator to create,

delete, modify, copy, enable or disable a workflow, attached the workflow to a folder, and assign users and groups to the workflow and its items.

- FMT_SMF.1a: The TOE provides the web-based interfaces that allow the authorized administrator to manage the access control policy and the workflow.
- FMT_SMR.1a: The TOE implements a Folder Administrator security management role in addition to recognizing the Publisher Administrator and Workflow Administrator defined in its environment (i.e., ALI) by associating Publisher activity rights with groups managed by ALI. While there are other roles associated with various access as described in the previous section, those are not security management roles but rather representative of other types of TOE users. The TOE also realizes crawler and crawler definer roles by virtue of distinct interfaces provided exclusively to support the crawling and crawler definition functions.

### 6.1.3 Protection of the TSF

The TOE enforces the access control policy to restrict the access to the TOE objects and the operations a user may performed on the object. The TOE utilizes activity rights and roles to restrict what objects can be created and what administrative functions can be accessed by authorized administrative user.

Publisher relies on the ALI to identify and authenticate all users attempting to access Publisher. ALI defines, maintains, and manages the administrators and administrative groups of Publisher, the users, user groups, and ALI objects which can be assigned to roles in Publisher. ALI uses the Publisher activity rights to limit the access to the Publisher administrative interfaces. Publisher enforces the privileges allowed by the activity right.

The Protection of the TSF function is designed to satisfy the following security functional requirements:

- FPT_RVM.1a: The TOE enforces access control mechanisms to ensure that the TOE security functions are not bypassed.

## 6.2 TOE Security Assurance Measures

The following assurance measures are applied to satisfy the Common Criteria EAL 2 augmented with ALC_FLR.2 assurance requirements:

- Configuration management;
- Delivery and operation;
- Development;
- Guidance documents
- Life cycle support;
- Tests; and
- Vulnerability assessment.

### 6.2.1 Configuration management

The configuration management measures applied by BEA Systems ensure that configuration items are uniquely identified, and that documented procedures are used to control and track changes that are made to the TOE. BEA Systems performs configuration management on the TOE implementation representation, design, tests, user and administrator guidance, delivery and operations, life cycle support, vulnerability assessment and the CM documentation.

These activities are documented in:

- AquaLogic® User Interaction Configuration Management

The Configuration management assurance measure satisfies the following EAL 2 augmented with ALC_FLR.2 assurance requirements:

- ACM_CAP.2

### 6.2.2 Delivery and operation

BEA Systems provides delivery documentation and procedures to identify the TOE, secure the TOE during delivery, and provide necessary installation and generation instructions. The BEA Systems' delivery procedures describe all applicable procedures to be used to prevent in appropriate access to the TOE. BEA also provides documentation that describes the steps necessary to install the TOE in the evaluated configuration.

These activities are documented in:

- AquaLogic® User Interaction Delivery and Operation
- Release Notes for AquaLogic Interaction Publisher 6.4
- BEA AquaLogic Interaction Publisher Installation and Upgrade Guide, Version 6.4
- Installation Worksheet for AquaLogic Interaction Publisher 6.4
- Deployment Guide for BEA AquaLogic™ User Interaction – this information is divided into 5 separate guides as follows:
    - BEA AquaLogic User Interaction Deployment Overview
    - BEA AquaLogic User Interaction Deployment Planning
    - BEA AquaLogic User Interaction Customization Overview
    - BEA AquaLogic User Interaction Deployment Maintenance Guide
    - BEA AquaLogic User Interaction Networking and Authentication Guide

The Delivery and operation assurance measure satisfies the following EAL 2 augmented with ALC_FLR.2 assurance requirements:

- ADO_DEL.1
- ADO_IGS.1

### 6.2.3 Development

BEA Systems has documents describing all facets of the design of the TOE. These documents serve to describe the security functions of the TOE, its interfaces both external and between subsystems, the architecture of the TOE (in terms of subsystems), and correspondence between the available design abstractions (including the ST).

These activities are documented in:

- AquaLogic Interaction Publisher 6.4 (ADV_FSP)
- AquaLogic Interaction Publisher 6.4  (ADV_HLD)
- AquaLogic Interaction Publisher 6.4 Representation (ADV_RCR)

The Development assurance measure satisfies the following EAL 2 augmented with ALC_FLR.2 assurance requirements:

- ADV_FSP.1
- ADV_HLD.1

- ADV_RCR.1

### 6.2.4 Guidance documents

BEA Systems provides administrator and user guidance on how to utilize the TOE security functions and warnings to administrators and users about actions that can compromise the security of the TOE.

These activities are documented in:

- AquaLogic Interaction Publisher 6.4 Guidance Documents (AGD)
- BEA AquaLogic Interaction Publisher Administrator Guide, Version 6.4
- AquaLogic Interaction Publisher 6.4 Online Help

The Guidance documents assurance measure satisfies the following EAL 2 augmented with ALC_FLR.2 assurance requirements:

- AGD_ADM.1
- AGD_USR.1

### 6.2.5 Life cycle support

BEA Systems has procedures that define the process for accepting and acting upon user reports of security flaws. These procedures describe the acceptance criteria for security flaws, how all security flaws and the status of fixes for each security flaw is tracked, and how corrections and corrective measures are made available as applicable.

These activities are documented in:

- AquaLogic® User Interaction Flaw Remediation

The Life cycle support assurance measure satisfies the following EAL 2 augmented with ALC_FLR.2 assurance requirements:

- ALC_FLR.2

### 6.2.6 Tests

The test documents describe the overall test plan, testing procedures, the tests themselves, including expected and actual results. In addition, these documents describe how the functional specification has been appropriately tested.

These activities are documented in:

- AquaLogic® Interaction Publisher 6.4 Testing Documentation

The Tests assurance measure satisfies the following EAL 2 augmented with ALC_FLR.2 assurance requirements:

- ATE_COV.1
- ATE_FUN.1
- ATE_IND.2

### 6.2.7 Vulnerability assessment

BEA Systems performs regular vulnerability analyses of the entire TOE (including documentation) to identify weaknesses that can be exploited in the TOE.

These activities are documented in:

- AquaLogic® Interaction Publisher 6.4 Vulnerability Assessment

The Vulnerability assessment assurance measure satisfies the following EAL 2 augmented with ALC_FLR.2 assurance requirements:

- AVA_SOF.1
- AVA_VLA.1

# 7. Protection Profile Claims

The ST does not claim compliance to a Protection Profile.

# 8. Rationale

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

- Security Objectives;
- Security Functional Requirements;
- Security Assurance Requirements;
- Strength of Functions;
- Requirement Dependencies;
- TOE Summary Specification; and,
- PP Claims.

## 8.1 Security Objectives Rationale

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

### 8.1.1 Security Objectives Rationale for the TOE and Environment

This section provides evidence demonstrating the coverage of threats and usage assumptions by the security objectives.

**Table 8-1 Environment to Objective Correspondence**

| | T.ACCESS | T.MANAGE | A.AUTH_USERS | A.INSTALL | A.NOEVIL | A.PHYSICAL | A.OPE_ENV | A.TRANSMIT | A.USER |
|---|---|---|---|---|---|---|---|---|---|
| **O.ACCESS** | X | X | | | | | | | |
| **O.MANAGE** | | X | | | | | | | |
| **OE.ACCESS** | X | | | | | | | | |
| **OE.AUTH** | | | X | | | | | | |
| **OE.MANAGE** | | X | | | | | | | |
| **OE.OPE_ENV** | | | | | | | X | | |
| **OE.ADMIN** | | | | | X | | | | |
| **OE.INSTALL** | | | | X | | | | | |
| **OE.PHYSICAL** | | | | | | X | | | |
| **OE.TRANSMIT** | | | | | | | | X | |
| **OE.USER** | | | | | | | | | X |

#### 8.1.1.1 T.ACCESS

*A user may gain unauthorized access to the TOE and the TOE's protected objects.*

This Threat is addressed by ensuring that:

- O.ACCESS: The objective ensures that the TOE restricts access to the TOE objects to the authorized users.
- OE.ACCESS: This objective ensures that the IT environment facilitates the control of access to the underlying objects that make up the TOE and its objects.

#### 8.1.1.2 T.MANAGE

*A user may gain unauthorized access to the utilities available to manage the security-related functions of the TOE.*

This Threat is addressed by ensuring that:

- O.ACCESS: The objective ensures that the TOE protects its objects so that they can be accessed only by appropriate users as controlled by authorized users.
- O.MANAGE: This objective ensures that the TOE provides the tools necessary for the authorized administrator to manage the security-related functions and that those tools are usable only by users with appropriate authorizeations.
- OE.MANAGE: This objective ensures that the environment provides the tools necessary for the authorized administrator to manage the security functions in the environment including those necessary to control access to the TOE.

#### 8.1.1.3 A.AUTH_USERS

*Only those users who have been authorized to access the information within the TOE may access the TOE.*

This Assumption is satisfied by ensuring that:

- OE.AUTH: The objective ensures that users of the TOE are identified and authenticated before access to the TOE and its functions is allowed.

#### 8.1.1.4 A.INSTALL

*Those responsible for the TOE must ensure that the TOE is delivered, installed, managed, and operated in a manner to maintain to the IT security objectives.*

This Assumption is satisfied by ensuring that:

- OE.INSTALL: The objective ensures that the TOE will be installed, managed and operated in a manner that maintains the security objectives.

#### 8.1.1.5 A.NOEVIL

*The administrative personnel are not careless, willfully negligent, or hostile and will follow and abide by the instructions provided by the administrative guidance.*

This Assumption is satisfied by ensuring that:

- OE.ADMIN: This objective ensures that the TOE is managed and administered in a secure manner by competent and security aware personnel in accordance with the administrator documentation.

#### 8.1.1.6 A.PHYSICAL

*The TOE software critical to security policy enforcement will be protected from unauthorized physical modification.*

This Assumption is satisfied by ensuring that:

- OE.PHYSICAL: The objective ensures that the TOE is protected from physical attacks.

#### 8.1.1.7 A.OPE_ENV

*The TOE operating environment shall provide the mechanism to isolate the TOE components and resources, ensure the TOE cannot be tampered with or bypassed and the TOE data is protected from unauthorized deletions.*

This Assumption is satisfied by ensuring that:

- OE.OPE_ENV: The objective ensures that the TOE's operating environment protects the TOE and it associated data.

#### 8.1.1.8 A.TRANSMIT

*The operating environment will protect the data transmitted to and from the TOE.*

This Assumption is satisfied by ensuring that:

- OE.TRANSMIT: The objective ensures that the TSF data imported to and exported from the TOE is protected from disclosure.

#### 8.1.1.9 A.USER

*The authorized users will not be negligent or malicious and will follow the guidance provided.*

This Assumption is satisfied by ensuring that:

- OE.USER: The objective ensures that the user of the TOE will work co-operatively and will follow the provided guidance.

## 8.2 Security Requirements Rationale

This section provides evidence supporting the internal consistency and completeness of the components (requirements) in the Security Target. Note that Table 8-2 identifies the requirements that effectively satisfy the individual objectives.

### 8.2.1 Security Functional Requirements Rationale

All Security Functional Requirements (SFR) identified in this Security Target are fully addressed in this section and each SFR is mapped to the objective for which it is intended to satisfy.

**Table 8-2 Objective to Requirement Correspondence**

| | O.ACCESS | O.MANAGE | OE.ACCESS | OE.AUTH | OE.MANAGE | OE.OPE_ENV |
|---|---|---|---|---|---|---|
| **FDP_ACC.1a** | X | | | | | |
| **FDP_ACF.1a** | X | | | | | |
| **FMT_MSA.1a** | X | X | | | | |
| **FMT_MSA.3a** | X | | | | | |
| **FMT_MTD.1a** | | X | | | | |
| **FMT_MTD.1b** | | X | | | | |
| **FMT_SMF.1a** | | X | | | | |
| **FMT_SMR.1a** | | X | | | | |
| **FPT_RVM.1a** | X | X | | | | |
| **FDP_ACC.1b** | | | X | | | |
| **FDP_ACF.1b** | | | X | | | |
| **FIA_UAU.2** | | | | X | | |
| **FIA_UID.2** | | | | X | | |
| **FMT_MSA.1b** | | | | | X | |
| **FMT_MSA.3b** | | | X | | | |
| **FMT_SMF.1b** | | | | | X | |
| **FMT_SMR.1b** | | | | | X | |
| **FPT_RVM.1b** | | | | | | X |
| **FPT_SEP.1** | | | | | | X |

#### 8.2.1.1 O.ACCESS

*The TSF shall restrict access of the TOE defined objects to specified users and users with appropriate privileges. The TSF must allow authorized users to specify which users may access their objects and the actions performed on the objects.*

This TOE Security Objective is satisfied by ensuring that:

- FDP_ACC.1a: The requirement helps meets the objective by identifying the objects and users subjected to the access control policy.
- FDP_ACF.1a: The requirement meets this objective by ensuring the TOE only allows access to objects based on the defined access control policy.
- FMT_MSA.1a: The TOE allows the authorized users to determine who will have access to the folder and the folder’s contents and what actions the user can be perform.
- FMT_MSA.3a: The TOE enforces a restrictive access when a new object is created. The TOE has a default ACL which is assigned to all newly-created objects. This default ACL cannot be altered by any user.
- FPT_RVM.1a: The TOE enforces the access control policy to ensure the objects are accessed by authorized users and only permitted actions are allowed.

#### 8.2.1.2 O.MANAGE

*The TOE shall allow administrators to effectively manage the TOE and its security functions, and must ensure that only authorized users with the administrative privileges are able to access the functions.*

This TOE Security Objective is satisfied by ensuring that:

- FMT_MSA.1a: The TOE will restrict the ability to modify the folder security list to authorized administrators.
- FMT_MTD.1a, FMT_MTD.1b: The TOE restricts that ability to manage the workflow to the user with the Configure Workflow activity right.
- FMT_SMF.1a: The TOE will provide the interfaces to manage the access control policy.
- FMT_SMR.1a: The TOE has pre-defined roles which can be associated to the user.
- FPT_RVM.1a: The TOE limits access to the security functions to users with appropriate administrative activity rights.

#### 8.2.1.3 OE.ACCESS

*The IT Environment shall restrict access of objects to identified users. The IT Environment must allow authorized administrator to specify which users may access the objects and the operations that may be performed.*

This IT Environment Security Objective is satisfied by ensuring that:

- FDP_ACC.1b: The requirement helps meets the objective by identifying the user objects subject to the access control policy.
- FDP_ACF.1b: The requirement meets this objective by ensuring the IT environment only allows access to user objects based on the defined access control policy.
- FMT_MSA.3b: The IT environment enforces a restrictive access when a new object is created. Newly-created objects inherit the ACL from the parent folder.

#### 8.2.1.4 OE.AUTH

*The operating environment shall ensure that all users have been identified and authenticated before access to the TOE is permitted.*

This IT Environment Security Objective is satisfied by ensuring that:

- FIA_UAU.2: The environment ensures that users are authenticated before access to the TOE is permitted. New users are able to create a new account before access is granted.
- FIA_UID.2: The environment ensures that users are identified before access to the TOE is permitted. New users are able to create a new account before access is granted.

#### 8.2.1.5 OE.MANAGE

*The environment shall allow environment administrators to effectively manage the environment and its security functions.*

This IT Environment Security Objective is satisfied by ensuring that:

- FMT_MSA.1b: The environment will restrict the ability to assign the activity right to user's groups to environment administrators.
- FMT_SMF.1b: The environment will provide the interfaces to manage the environment.
- FMT_SMR.1b: This requirement ensures the environment defines an authorized administrator.

#### 8.2.1.6 OE.OPE_ENV

*The TOE operating environment shall provide the mechanism to isolate the TOE components and resources, ensure that the TOE cannot be tampered with or bypassed and the TOE data is protected from unauthorized deletions.*

This IT Environment Security Objective is satisfied by ensuring that:

- FPT_RVM.1b: The IT environment will enforce mechanisms to ensure that the TOE security functions can not be bypassed.
- FPT_SEP.1: The IT environment will protection TOE and ensure that it is not tampered with by untrusted subject.

## 8.3 Security Assurance Requirements Rationale

This ST contains the assurance requirements from the CC EAL 2 augmented with ALC_FLR.2 assurance package. The EAL chosen is based on the statement of the security environment (assumptions and threats) and the security objectives defined in this ST. The sufficiency of the EAL chosen is justified based on those aspects of the environment that have impact upon the assurance needed in the TOE. The users are conscientious, non-hostile and will follow the guidance (A.NOEVIL, A.USER, OE.ADMIN, OE.USER). The TOE server component is physically protected and properly and securely configured (A.INSTALL, OE.INSTALL). Given these aspects, a TOE based on good commercial development practices is sufficient. EAL 2 augmented with ALC_FLR.2 is an appropriate level of assurance for the TOE described in this ST.

## 8.4 Strength of Functions Rationale

The TOE does not implement any other functions that are of a permutational or probabilistic nature. Therefore, a minimum SOF claim is not made for the TOE.

## 8.5 Requirement Dependency Rationale

The ST satisfies all the requirement dependencies of the Common Criteria. Table 8-3 Security Requirement Dependencies lists each requirement from Sections 5.1 and 5.2 and indicates which requirements were included to satisfy the dependencies, if any.

**Table 8-3 Security Requirement Dependencies**

| ST Requirement | CC Dependencies | ST Dependencies |
|---|---|---|
| **FDP_ACC.1a** | FDP_ACF.1 | FDP_ACF.1a |
| **FDP_ACF.1a** | FDP_ACC.1 and FMT_MSA.3 | FDP_ACC.1a and FMT_MSA.3a |
| **FMT_MSA.1a** | FMT_SMR.1 and FMT_SMF.1 and (FDP_ACC.1 or FDP_IFC.1) | FMT_SMR.1a and FMT_SMF.1a and FDP_ACC.1a |
| **FMT_MSA.3a** | FMT_MSA.1 and FMT_SMR.1 | FMT_MSA.1a and FMT_SMR.1a |
| **FMT_MTD.1a** | FMT_SMR.1 and FMT_SMF.1 | FMT_SMR.1b and FMT_SMF.1a |
| **FMT_MTD.1b** | FMT_SMR.1 and FMT_SMF.1 | FMT_SMR.1b and FMT_SMF.1a |
| **FMT_SMF.1a** | none | none |
| **FMT_SMR.1a** | FIA_UID.1 | FIA_UID.2 |
| **FPT_RVM.1a** | none | none |
| **FDP_ACC.1b** | FDP_ACF.1 | FDP_ACF.1b |
| **FDP_ACF.1b** | FDP_ACC.1 and FMT_MSA.3 | FDP_ACC.1band FMT_MSA.3b |
| **FIA_ATD.1** | none | none |
| **FIA_UAU.2** | FIA_UID.1 | FIA_UID.2 |
| **FIA_UID.2** | none | none |
| **FMT_MSA.1b** | FMT_SMR.1 and FMT_SMF.1 and (FDP_ACC.1 or FDP_IFC.1) | FMT_SMR.1b and FMT_SMF.1b and FDP_ACC.1b |
| **FMT_MSA.3b** | FMT_MSA.1 and FMT_SMR.1 | FMT_MSA.1b and FMT_SMR.1b |
| **FMT_SMF.1b** | none | none |
| **FMT_SMR.1b** | FIA_UID.1 | FIA_UID.2 |
| **FPT_RVM.1b** | none | none |
| **FPT_SEP.1** | none | none |

## 8.6 Explicitly Stated Requirements Rationale

All requirements included in this ST are drawn from the CC Part 2 and Part 3. The Security Target does not define explicitly stated requirements.

## 8.7 TOE Summary Specification Rationale

Each subsection in Section 6, the TOE Summary Specification, describes a security function of the TOE. Each description is followed with rationale that indicates which requirements are satisfied by aspects of the corresponding security function. The set of security functions work together to satisfy all of the security functions and assurance requirements. Furthermore, all of the security functions are necessary in order for the TSF to provide the required security functionality.

This Section in conjunction with Section 6, the TOE Summary Specification, provides evidence that the security functions are suitable to meet the TOE security requirements. The collection of security functions work together to provide all of the security requirements. The security functions described in the TOE summary specification are all necessary for the required security functionality in the TSF. Table 8-4 Security Functions vs. Requirements Mapping demonstrates the relationship between security requirements and security functions.

**Table 8-4 Security Functions vs. Requirements Mapping**

| | **User data protection** | **Security management** | **Protection of the TSF** |
|---|---|---|---|
| **FDP_ACC.1a** | X | | |
| **FDP_ACF.1a** | X | | |
| **FMT_MSA.1a** | | X | |
| **FMT_MSA.3a** | X | | |
| **FMT_MTD.1a** | | X | |
| **FMT_MTD.1b** | | X | |
| **FMT_SMF.1a** | | X | |
| **FMT_SMR.1a** | | X | |
| **FPT_RVM.1a** | | | X |

## 8.8 PP Claims Rationale

See Section 7, Protection Profile Claims.

# Appendix A: Terminology

***Activity rights*** Activity rights are settings that confer system-wide privileges maintained in ALI. Activity rights are defined on a group basis. Users who do not have permission to perform a particular activity do not see the corresponding user interface elements in ALI. Activity rights are not assigned directly to individual users, rather activity rights are assigned to groups. Then users are also assigned to groups and the users acquire the associated group activity rights conferred by their specific group membership.

***Assignee*** User assigned to an activity in a workflow.

***Content item*** Publisher's primary publishable object. Content items can be objects created from Data Entry Templates, or they can be imported files or images.

***Content crawler*** An administrative object used to import content into ALI from external content repositories.

***Content*** Text and images published on a web page; including things like articles, customer profiles, employee reports, news stories, job postings.

***System*** The system is the TOE and its operating environment.

# Appendix B: Acronyms

| | |
|---|---|
| ACL | Access Control List |
| ALI | AquaLogic Interaction |
| ALUI | AquaLogic User Interaction |
| CC | Common Criteria for Information Technology Security Evaluation |
| CEM | Common Evaluation Methodology for Information Technology Security |
| CM | Configuration Management |
| EAL | Evaluation Assurance Level |
| FDP | User Data Protection CC Class |
| FIA | Identification and Authentication CC Class |
| FMT | Security Management CC Class |
| FSP | Functional Specification |
| HLD | High Level Design |
| IT | Information Technology |
| MOF | Management of Functions |
| MTD | Management of TSF Data |
| OSP | Organization Security Policy |
| PP | Protection Profile |
| SAR | Security Assurance Requirement |
| SFP | Security Function Policy |
| SFR | Security Functional Requirement |
| SM | Security Management |
| SMR | Security Management Roles |
| SOF | Strength of Function |
| ST | Security Target |
| TOE | Target of Evaluation |
| TSC | TSF Scope of Control |
| TSF | TOE Security Function |
| TSP | TOE Security Policy |
| UAU | User Authentication |
| UDP | User Data Protection |