WebSphere Portal
EAL2
Security Target

IBM Global Services CLEF
IBM UK Ltd
Meudon House
Meudon Avenue
Farnborough
Hampshire GU14 7NB

Telephone No: (01252) 558472

Date: 18th August 2004
Issue: 2.8
Reference: LFF/WP/EAL2/ST/28

This Page Intentionally Left Blank.

Table of Contents

Glossary and Terminology

API	Application Programmable Interface
Authorised User	A user who may, in accordance with the TSP, perform an operation.
CC	Common Criteria
EAL	Evaluation Assurance Level
Entity	Subject
ID	IDentity
IT	Information Technology
ITSEC	IT Security Evaluation Criteria
J2EE	Java 2 Enterprise Edition
LDAP	Lightweight Directory Access Protocol
OS	Operating System
PAC	Portal Access Control
Portlet	A portlet is a small portal application, usually depicted as a small box in a web page.
Principal	An entity within the portal that can be authorized, i.e. user or group ID, a subject
Resource	An entity within the portal controlled by PAC (e.g. page, portlet) i.e. an object
SF	Security Function. A part or parts of the TOE that have been relied upon for enforcing a closely related subset of rules from the TSP.
SFR	Security Functional Requirement
SLES	SuSE Linux Enterprise Edition
SOF	Strength Of Function
ST	Security Target
TOE	Target Of Evaluation. An IT product or system and its associated administrator and user guidance documentation that is the subject of the evaluation.
TSF	TOE Security Function. A set consisting of all hardware, software and firmware of the TOE that must be relied upon for the correct enforcement of the TSP.
TSP	TOE Security Policy. A set of rules that regulate how assets are managed, protected and distributed within the TOE.
WAR	Web Application aRchive
WAS	WebSphere Application Server

Web Modules	Web modules are portlet WAR files that are installed on WAS.
WMM	WebSphere Member Manager
WP	WebSphere Portal
WPCP	WebSphere Portal Content Publishing
WPS	WebSphere Portal Server

1 Introduction

Security Target (ST) Title: WebSphere Portal EAL2 Security Target

Version: 2.8

Version Date: 18th August 2004

TOE identification: WebSphere Portal version 5.0.2

Common Criteria Identification: Common Criteria for Information Technology Security Evaluation, CCIMB-99-031, Version 2.1, August 1999.

Evaluated Assurance Level: EAL2

1.1 Target of Evaluation Overview

WebSphere Portal (WP) is a Java 2 Enterprise Edition (J2EE) application executed in the run-time environment provided by WebSphere Application Server (WAS) that provides users a consistent view of portal applications and allows users to define specific sets of applications which are presented in a single context. WP allows authorized users to establish protected portal resources like pages and portlets. As an example, authorized users (a team) can develop, share and store information of the types listed above for projects. This then allows for fast access to, and transfer of information between members of the team working on the same project.

The Access Control administration can be performed using corresponding portlets within the running portal or via the *XmlAccess* scripting interface.

WebSphere Portal 5.0.2 (also known as WebSphere Portal Server (WPS)) is provided within a set of products. There are four sets available, which are:

- WebSphere Portal Enable;
- WebSphere Portal Extend;
- WebSphere Portal Express; and
- WebSphere Portal Express Plus.

Note that the version 5.0.2 applies to both the product sets and WebSphere Portal.

1.1.1 WebSphere Portal Enable

WebSphere Portal Enable contains the following programs:

- WebSphere Application Server (WAS) 5.0;
- IBM HTTP Server;
- WebSphere Portal (WP);
- Collaboration APIs;
- WebSphere Portal Toolkit;

- WebSphere Translation Server;
- WebSphere Studio Site Developer;
- WebSphere Portal Content Publishing (WPCP);
- WebSphere Portal Document Manager;
- IBM Directory Server;
- DB2 UDB;
- WebSphere Member Manager (WMM).

1.1.2 WebSphere Portal Extend

WebSphere Portal Extend contains all those programs included within WebSphere Portal Enable, but has the following in addition:

- Sametime;
- QuickPlace;
- Domino;
- Collaboration Centre;
- Extended Search;
- Tivoli Web Site Analyser.

1.1.3 WebSphere Portal Express

WebSphere Portal Express contains the following products:

- WebSphere Application Server (WAS) 5.0;
- IBM HTTP Server;
- WebSphere Portal (WP);
- Collaboration APIs;
- WebSphere Portal Toolkit;
- WebSphere Studio Site Developer;
- WebSphere Portal Document Manager;
- IBM Directory Server;
- WebSphere Member Manager (WMM).

1.1.4 WebSphere Portal Express Plus

WebSphere Portal Express Plus contains all those programs included within WebSphere Portal Express, but has the following in addition:

- Sametime;
- QuickPlace;
- Domino;
- Collaboration Centre.

Each of these sets are for multi-platforms. Only three of the above products are security relevant: WAS, WMM and DB2. These are used by WebSphere Portal for user identification and attribute storage. Note: that there is no specific requirement for DB2 to be used as the database. All comerically available databses are compatible.

No further discussion is provided on the other, non-security relevant products.

WP contains the following components:

- Aggregation. This is used for generating the content returned to the client e.g. the objects to display on the browser;
- Deployment. This is used for installing new portlets on a running portal;
- Portal Access Control. This controls access to all protected portal resources;
- A number of Portlets are also included:
    - Manage Users and Groups;
    - Resource Permission;
    - User and Group permission;
    - Install Portlets;
    - Manage Portlet Applications;
    - Manage Portlets;
    - Manage Pages;
    - Manage Users and Groups;
    - URL Mapping.

The following Operating Systems (OS) are supported but outside the scope of this evaluation:

- AIX 5.1 and 5.2;
- RedHat Linux 8.0 and Advanced Server 2.1 for Intel;
- Solaris 8;
- SuSE 7.3 Linux for Intel;
- SuSE Linux Enterprise Edition (SLES) 7 & 8 for Intel;
- SuSE Linux Enterprise Edition (SLES) 7 for zSeries;
- Windows 2000 Server and Advanced Server;
- Windows 2003 Standard and Enterprise.

It is assumed that all hardware used within the operating environment is secured such that no potential vulnerabilities could be introduced that would circumvent the functionality described within this ST.

1.2 CC Conformance

This ST is [CC] *Part 2 extended with* FMT_MSA_E.3 *and Part 3 conformant*, with a claimed Evaluation Assurance Level of EAL2.

1.3 Strength Of Functions (SOF)

There is no SOF claim because the TOE does not identify any security functional requirements for which an explicit SOF is appropriate and does not identify any functions that are of a permutational or probabilistic nature. Therefore, a minimum SOF claim is not included for the TOE.

1.4 References

[CC] Common Criteria for Information Technology Security Evaluation, CCIMB-99-031, Version 2.1, August 1999.

1.5 Structure

The structure of this document is as defined by [CC] Part 1, Annex C:

- Section 2 is the TOE description;
- Section 3 provides a statement of the TOE security environment;
- Section 4 provides the statement of IT security objectives;
- Section 5 provides a statement of IT security requirements;
- Section 6 provides the TOE summary specification, which includes the detailed specification of the IT functions; and
- Section 7 provides the rationale for the security objectives, security requirements and TOE summary specification.

2 TOE Description

2.1 Introduction

WP allows authorized users to establish protected portal resources as defined in Section 2.2 of this document. As an example, authorized users (a team) can develop, share and store information of the types listed above for projects. This then allows for fast access to, and transfer of information between members of the team working on the same project.

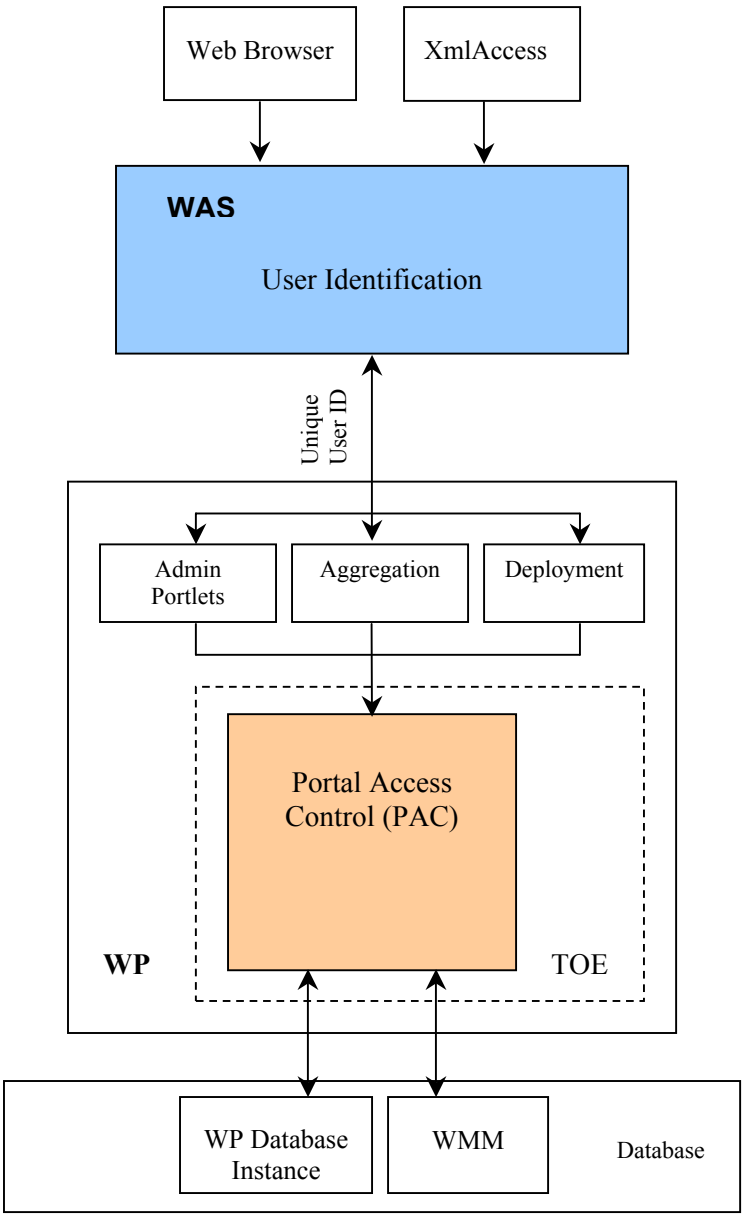

Figure 2.1: WP Dependencies and scope of evaluation.

WP provides access control to protected resources as identified within Section 2.2 of this document. Access control is performed by the Portal Access Control (PAC) component within WP. This is shown within Figure 2.1, which illustrates that PAC is the only component within the TOE.

When a user requests access to a resource from the web browser, WP relies upon WebSphere Application Server (WAS) to perform identification and management of users, WMM to provide the group membership and a database for the mapping of users to roles and the actions to resources. The request is passed onto PAC. Neither WAS or WMM are within the scope of evaluation and are therefore part of the TOE environment. WP also relies upon an OS and a database to operate however WP does not rely upon the either the OS or database to provide any security functionality.

Figure 2.1 shows the Target Of Evaluation (TOE) indicated by the dotted line, which only includes the PAC component. The main interfaces to PAC are:

- WAS;
- WMM;
- The scripting interface XmlAccess;
- The resources; and
- The administration Portlets;
- Aggregation; and
- Deployment.

Admin Portlets are the GUI Administration interface and XMLAccess is the scripting interface for PAC Administration. Admin Portlets describe the following portlets that are included within WP by default:

- Manage Users and Groups;
- Resource Permission; and
- User and Group permission.

The admin portlets in the following table are used to create the respective protected resources:

Resource	**Admininstration Portlet**
Web Module	Install Portlets
Portlet Application Definition	Manage Portlet Applications
Portlet	Manage Portlets
Page	Manage Pages
User Group	Manage Users and Groups
URL Mapping Context	URL Mapping

Aggregation is used for generating the content returned to the client e.g. the objects to display on the browser. Deployment is used for installing new portlets on a running

portal. WAS passes the request to access a resource to either the Aggregation or deployment components, depending on the action requested. These then request an access control decision from PAC. If PAC allows access to perform an operation on the resource then the aggregation and deployment components access that resource. Depending on the access control decision, then either an error message is returned to the browser/XmlAccess or the appropriate resource information.

The Aggregation, deployment and admin portlets are also referred to as PAC clients.

2.1.1 Portlets

Portlets are the heart of a portal. A portlet is a small portal application, usually depicted as a small box in a web page. Portlets are re-usable components that provide access to applications, web-based content and other resources. Portlets can be grouped together in a portlet application.

A Portlet is a complete application, following a standard model view controller design. Portlets run inside the portlet container of WP, similar to a servlet running on an application server.

2.2 Portal Access Control (PAC)

PAC is the single access control decision point within WP. It controls access to all sensitive portal resources. Protected resources are resources that can be accessed by a restricted set of users only. In order to be granted access to a protected resource in a specific way, the user needs a corresponding permission on this resource, e.g. a specific portal page can only be viewed by a specific user, if the user has the permission to perform the action 'View' on that page. The following types of resources are protected within the portal:

- Web Modules: Web modules are portlet archives that are installed on WAS. Web modules can contain multiple portlet applications. If a new Web module is installed, it is automatically a child of the Web Modules virtual resource;
- Portlet Application Definitions: Portlet applications provide a logical grouping of individual portlets. If a new Web module is installed, the portlet applications contained within that Web module are automatically child resources of the Portlet Applications virtual resource. Portlets contained within a portlet application appear as child nodes of that portlet application. A two-layer hierarchy consisting of portlet applications and the corresponding portlets exists beneath the Portlet Applications virtual resource;
- Portlets (Portlet Definitions): A portlet is an installed portlet having it's own portlet configuration. E.g. a Mail portlet can be configured to a specific mail server
- Content Nodes (Pages): Pages (also known as *content nodes*) contain the content that determines the portal navigation hierarchy. A portal page is basically the frame that contains a specific set of individual portlets arranged in a specific layout. If a new top-level page is created, it is automatically a child resource of the *Content Nodes* virtual resource. If a

new page is created beneath an existing page, the new page is automatically child of the existing page;

- User Groups: Users can be grouped into user groups (database records). User groups can be nested. Access privileges are propagated with user groups membership. If a new user group is created, it will appear as a corresponding child resource underneath the virtual resource *User Groups*.
- URL Mapping contexts: URL mapping contexts are user-defined definitions of URL spaces that map to portal content. If a new top-level URL mapping context is created, it is automatically a child resource of the *URL Mapping Contexts* virtual resource. If a new URL mapping context is created beneath an existing context, the new context is automatically a child the existing context. URL mapping contexts inherit access control configuration from their parent context unless role blocks are used;

Users (database records) are implicitly protected resources, which means that access to specific user profile data can only be obtained via corresponding privileges on a user group that contains the given user as a member i.e. implicitly protected resources are those resources that are not linked into the protected resource hierarchy. Implicitly protected resources behave in the same was as normal protected resources. The Users virtual resource protects sensitive operations that deal with user management. For example, in order to add a user to a user group you must have the Security Administrator@Users role.

PAC directly supports access control configuration of hierarchical resource topologies through the concept of permission inheritance. This concept reduces the administration overhead for an administrator when controlling access to a large number of portal resources. Inherited permissions are automatically assembled into roles that can be assigned to individual users and user groups, granting them access to whole sets of logically related portal resources. Permission inheritance can be prevented using role blocks. Role blocks can be either inheritance or propagation blocks, which prevent the inheritance of permissions to a child resource, or propagation of the permissions from a resource respectively.

Each of these resources has a database entry which contains a list of the roles that are authorised access to the resource. The access permissions are dependant upon those assigned to the role.

In addition to protected resources, portal access control supports the notion of virtual resources that are used to group resources of a specific type and to configure access to abstract concepts within the portal e.g. the virtual node *portal* provides a means to give a user full control over the portal. Access Control on the virtual resources behave in the same way as non-virtual resources. The portal defines a set of fixed virtual resources, which are virtual resources that are created and initialised during portal installation.

Figure 2.2 shows the general layout of the resource topology that is protected by Portal Access Control, Figure 2.3 depicts an example sub-set of this topology that could exist in a real portal setup. Implicitly protected resources (light yellow boxes in Figure 2.2) are protected via their non-implicit parent resources. Thus, they do not need to show up in the

PAC administration user interfaces. Implicitly protected resources are those resources that are not linked into the protected resource hierarchy.

It is possible to configure WP to allow the access control functionality to be performed externally, however WP has no control over external applications within the environment and therefore this functionality is outside the scope of the evaluation.

2.2.1 Virtual Resources

The portal defines a set of fixed virtual resources that are created and initialised during portal installation. Virtual resources are resources that are used to group resources of a specific type and to configure access to abstract concepts within the portal and cannot be accessed directly by a user. Fixed virtual resources are virtual resources that are supplied as part of WP.

Virtual resources have two functions:

- They protect sensitive operations that affect the entire portal or specific concepts in the portal. For example, the XmlAccess virtual resource protects the ability to execute scripts via that XML configuration interface.
- They are parent resources for all resource instances. For example, the Web Modules virtual resource is the root node of all Web modules instances within the portal. Role assignments on the Web Modules virtual resource permit access to all Web modules in the portal.

Virtual resources still operate within the Access Control policy as they can be accessed to assign resources and configuration.

Figure 2.2. shows the general layout of the resource topology that is protected by PAC.

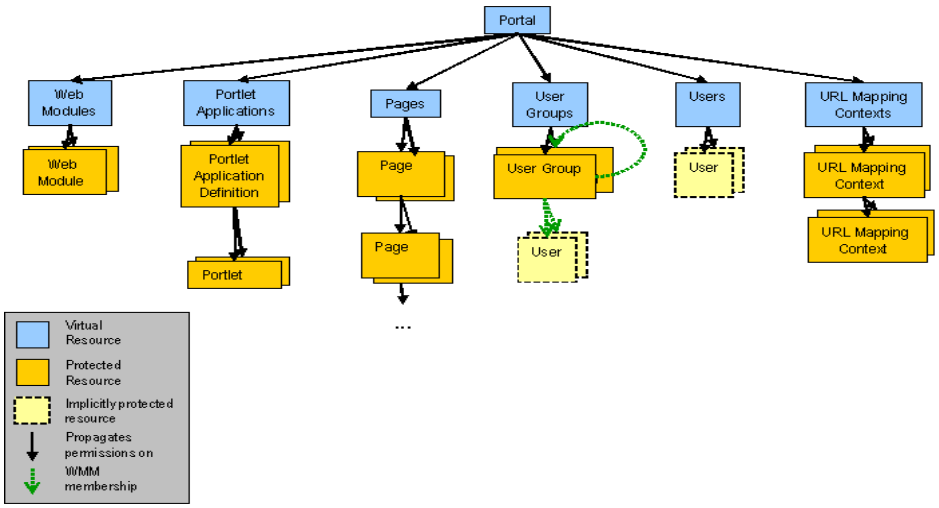

Figure 2.2: General Layout of Resource Topology

Figure 2.3. shows an example subset of the topology shown in Figure 2.2 that could exist in reality.

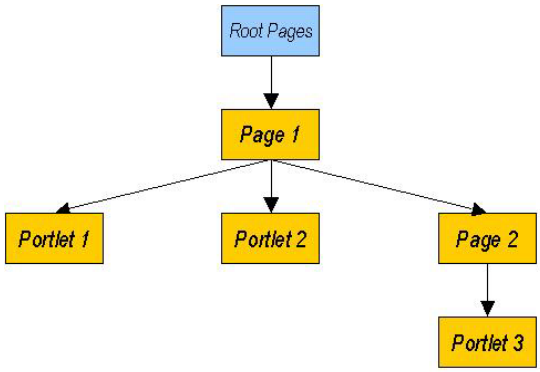

Figure 2.3: Example Sub-set of Resource Topology

WP also supports the virtual principals *Anonymous User* and *All-Authenticated-Users*. The *Anonymous User* can be used to grant permissions to users who have not been authenticated by the portal (i.e. WAS). The *All-Authenticated-Users* is the virtual group of users who have successfully authenticated (by WAS) to the portal[1]. These principals are called "virtual" since they do not map to entities within the users subsystem but are concepts within the access control component.

2.2.2 Actions

Actions model the different ways of accessing a specific resource. PAC supports the following actions:

- Grant Access On;
- Delegate To;
- Add Child;
- Add Private Child;
- Delete;
- Edit;
- Personalize;
- View.

2.2.3 Action Sets

An action set (also known as *role type*)is a named set of actions that provides a grouping of individual actions (e.g. *Editor* = {View, Edit}). The portal provides a set of predefined action sets each of which containing a set of actions that is typically needed to fulfil specific tasks within the portal (e.g. adjust and modify the layout of shared resources).

[1]Reference to Authentication is used purely as a description for these principals, and is not intended to imply that authentication functionality is included as part of this evaluation.

The default action sets within WP are:

- Admin;
- Security Admin;
- Delegator;
- Manager;
- Editor
- Privileged User; and
- User.

2.2.4 Roles

A Role is an action set that has been assigned to a resource. Roles are created within the portal by applying an action set to a specific resource within the resource topology. The resulting set of permissions is determined by combining the set of actions contained in the action set on the resource and all child resources (as long as no role blocks are encountered). For example, in Figure 2.3, the resulting permissions for Page1 would be the action set on Page 2 and Portlet 3. *Administrator@Portal* and *Security Administrator@Portal* are default roles created by WP.

Roles can be assigned to individual principals granting those principals the corresponding permissions. E.g. let there be a role called *User@SalesPage* containing the permissions (View, SalesPage) and (View, SalesPortletInstance). If this role is assigned to the user group SalesForce, all members of this group (including nested groups) are allowed the perform the action View on the Sales Page and the Sales Portlet, i.e. they are allowed to see the content of the Sales Page and use the Sales Portlet.

2.3 Data Storage

The PAC component environment includes WMM and a database for group membership and storage which are both outside the scope of this evaluation.

2.3.1 WebSphere Member Manager (WMM)

WMM provides the group membership(s) to WP. For the scope of this CC evaluation WMM is within the environment and responsible for the group membership of users. WMM relies upon a database for storage of the group memberships.

2.3.2 WebSphere Portal Database Instance

The PAC component relies upon a database in the environment to store the Group IDs, group memberships, mapping of users to roles and the actions to resources and the resources themselves.

WP has its own dedicated database instance that can be used or a third party database (e.g. DB2) can be used by the PAC component.

2.4 WebSphere Application Server (WAS)

The PAC component relies on WAS for the identification of principals (WAS is outside the scope of the evaluation). WAS provides the user's unique ID, which is retrieved from an authentication component (via a WAS Application Programmable Interface (API)). Once WP has received a user ID from WAS, then it consults the WMM, which provides the user's group membership(s).

3 TOE Security Environment

3.1 Introduction

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

The statement of TOE security environment therefore identifies the assumptions made on the operational environment and the intended method for the product, defines the threats that the product is designed to counter and the organisational security policies which the product is designed to comply.

3.2 Threats

The assumed security threats are listed below:

[T.ACCESS_RES] An authorised user of the TOE gains access to an object without the correct authority to access that object.

[T.APP] The applications that the TOE depends upon become compromised.

[T.NETWORK] Data transferred between workstations is disclosed to, or modified by unidentified users or processes, either directly or indirectly.

3.3 Organisational Security Policies (OSPs)

The TOE complies with the following OSP:

[P.ACCESS] The right to access a resource is determined on the basis of:

- User membership of a group(s);
- User or group(s) ID association with a role;
- Resource association with an Action set (and thus creation of a role); and
- Actions assigned to the action set;
- Permission inheritance given by the protected resource hierarchy and role blocks.

3.4 Assumptions

This section provides the minimum physical and procedural measures required to maintain security of WP.

3.4.1 Physical aspects

[A.APP] It is assumed that the applications that the TOE relies upon, have been configured in accordance with the manufacturer's installation guides and where applicable, in its evaluated configuration. It is securely configured such that the applications protect the TOE from any unauthorised users or processes.

[A.PROTECT] It is assumed that all hardware within the environment, including network and peripheral devices, have been approved for the transmittal of protected data. Such items are to be physically protected against threats to the confidentiality and integrity of the data.

3.4.2 Personnel Aspects

[A.ADMIN] It is assumed that there are one or more competent individuals that are assigned to manage the TOE and the security of the information it contains. Such personnel are assumed not to be careless, wilfully negligent or hostile.

4 Security Objectives

4.1 Security Objectives for the TOE

[O.ACCESS] The TOE must ensure that only those users with the correct authority are able to access a resource.

[O.MANAGE] The TOE must allow administrators to effectively manage the TOE and that this is only performed by authorised users.

4.2 Security Objectives for the TOE Environment

[O.ADMIN] Those responsible for the TOE environment are competent and trustworthy individuals, capable of managing the TOE environment and the security of the information it contains.

[O.CONFIG] Those responsible for the TOE environment must ensure that each user on the supporting applications have associated user IDs and where applicable have an associated Group ID.

[O.APP] Those responsible for the TOE environment must ensure that the supporting applications are installed and configured in accordance with the manufacturer's instructions, the evaluated configuration where applicable and is secure.

[O.PROTECT] Those responsible for the TOE environment must ensure that procedures and/or mechanisms exist to ensure that data transferred between workstations is secured from disclosure, interruption or tampering.

[O.RECOVER] Those responsible for the TOE environment must ensure that procedures and/or mechanisms are provided to ensure that after system failure or other discontinuity, recovery without a security compromise is obtained.

5 Security Requirements

This section specifies the Security Functional Requirements (SFRs) for the TOE and organises the SFRs by class. Within the text of each SFR, the selection and assignment operations (as defined within [CC]) are *italicised*.

Note: FMT_MSA_E.3 is an explicitly stated IT security requirement, and although based on [CC], have not been specified using CC Part 2 functional components.

The International Interpretations that have been applied for the Security Requirements are 058, 064, 065, and 103.

5.1 TOE Security Functional Requirements

The following table summarises the SFRs:

CLASS	FAMILY	COMPONENT	ELEMENT
FDP	FDP_ACC	FDP_ACC.2	FDP_ACC.2.1
			FDP_ACC.2.2
	FDP_ACF	FDP_ACF.1	FDP_ACF.1.1
			FDP_ACF.1.2
			FDP_ACF.1.3
			FDP_ACF.1.4
FMT	FMT_MSA	FMT_MSA.1	FMT_MSA.1.1
		FMT_MSA_E.3	FMT_MSA_E.3.1
	FMT_SMF	FMT_SMF.1	FMT_SMF.1.1
	FMT_SMR	FMT_SMR.1	FMT_SMR.1.1
			FMT_SMR.1.2

5.1.1 Access Control (FDP)

FDP_ACC.2.1 The TSF shall enforce the *Access Control SFP* on *users, groups and*

- *Web Modules;*
- *Portlet Application Definitions;*
- *Portlet;*
- *Content Nodes (Pages);*

- *User Groups;*
- *URL Mapping Contexts.*

and all operations among subjects and objects covered by the SFP.

FDP_ACC.2.2 The TSF shall ensure that all operations between any subject in the TSC and any object within the TSC are covered by an access control SFP.

FDP_ACF.1.1 The TSF shall enforce the *access control SFP* to objects based on the following:

Subject	*Security Attributes*
User	*User/Group IDs and role(s) association*
Object	*Security Attributes*
Web Modules;	*Action set association* *Inheritance block* *Propagation block*
Portlet Application Definitions;	
Portlet;	
Content Nodes (Pages);	
User Groups;	
URL Mapping contexts	

FDP_ACF.1.2 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed:

A user is granted access to a resource if:

- *the user ID attribute is associated to a role and that the action set within that role is associated with the resource or the resource's parent; or the user is associated with a group ID attribute that is associated with a role and that the action set within that role is associated with the resource or the resource's parent;*

and

- *the resource does not have an inheritance block on that role; or*
- *for users with inherited access to that resource, there is no propagation block on the resource's parent resource.*

The actions that the user is permitted to perform on the resource is defined by the role's action set.

FDP_ACF.1.3 The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: *no additional rules.*

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

5.1.2 Security Management (FMT)

FMT_MSA.1.1 The TSF shall enforce the A*ccess Control SFP* to restrict the ability to *create, view, and delete* the security attributes: *role and action set association, propagation blocks and inheritance blocks* to:

- *Users that are assigned the Administrator@Portal or Security Administrator@Portal; or:*

For role to user association for a role identified by Action Set (AS) and Resource (R) and a User Group (UG):

- *Users have been assigned:*
  - *Security Administrator@R and AS@R or Administrator@R role and*
  - *Delegator@UG, Security Administrator@UG, or Administrator@UG.*

For role blocks on a resource R and roles of type ActionSet :

- *Users have been assigned Security Administrator@R and AS@Resource or Administrator@R role.*

FMT_MSA_E.3.1 The TSF shall enforce the access control SFP to provide restrictive default values for security attributes that are used to enforce the SFP.

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

- *User and Group to action set association;*
- *Resource to action set association; and*
- *Inheritance and Propagation blocking.*

FMT_SMR.1.1 The TSF shall maintain the roles:

- *Administrator@Resource; and*
- *Security Administrator@Resource;*
- *Delegator@Resource;*
- *Manager@Resource;*
- *Editor@Resource;*
- *Privileged User@Resource; and*
- *User@Resource.*

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

5.2 Strength Of Function (SOF)

There is no strength of function claim because the TOE does not identify any security functional requirements for which an explicit Strength of Function (SOF) is appropriate and does not identify any functions that are of a permutational or probabilistic nature. Therefore, a minimum SOF claim is not included for the TOE.

5.3 TOE Security Assurance Requirements

The target evaluation assurance level for this product is EAL2. No augmented assurance requirements are defined.

5.4 Security Requirements for the IT Environment

This section specifies the Security Requirements for the IT environment.

5.4.1 Identification and Authentication (FIA)

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

- *User ID;*
- *Group ID.*

FIA_UID.1.1 The TSF shall allow *assumption of the anonymous user ID* on behalf of the user to be performed before the user is identified.

FIA_UID.1.2 The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user

6 TOE Summary Specification

6.1 IT Security Functions (SF)

6.1.1 Access Control

AC.1 The TSF shall ensure that users can only access the following resources based on their user/group IDs and association(s) with a role:

- Web Modules;
- Portlet Application Definitions;
- Portlets;
- Content Nodes (Pages);
- User Groups;
- URL Mapping contexts.

A web browser is used to access the resources. The user is identified within the environment by WAS and then the request forwarded to the PAC component. The User and group IDs are maintained within the environment. The associations with roles are maintained by the Admin@Portal, Security Admin@Portal, via the admin Portlets *User Group Permissions* and *Resource Permissions*. Access control is only granted to a resource if the user ID (or group IDs that the user ID is associated), are associated with corresponding roles that in turn is associated with the resource.

Roles are created by the association of action sets with resources. Access rights are stored in the WP database and are administered through the User Group Permissions and Resource Permissions portlets. Access to child resources can be blocked by a role block. This is either a propagation or inheritance block, which blocks the access control inheritance propagating to child resources or prevents access control inheritance from that resource respectively. The access control decision is returned to the appropriate PAC client as a boolean yes/no. If successfull then the PAC client performs the operation requested and the results returned to the user interface. If access is denied then the PAC client either does not display the resource or an error is given, dependent upon the operation requested.

6.1.1.1 Default Attributes

AC.2 The table below describes the initial access control settings.

User or Group	Role
The administrative user identified during the installation	Administrator@Portal
<*wpsadmins*>	Administrator@Portal
All Authenticated Users[1]	User@ the following portlet applications: • Edit page content and layout • Concrete Properties Web App • Welcome • Appearance Web Application • Set Permissions Portlets • Information Portlet Application • Page Properties • Organize Favourites • Page Customizer Privileged User@ the following pages: • My portal

Wpsadmins is a user group that is setup by default on installation of WebSphere Portal. A user has the actions associated with the User, Editor, and Privileged User role types on itself. There is no explicit role assignment for these actions. They are a part of the administration policy.

On creation of a resource, the TSF shall define default security attributes for access to that resource. Every time a protected resource is created within the portal, the user that created the resource becomes the owner of that resource. The owner of a resource is allowed to perform the following actions on the resource:

- Add Child (if the resource is a shared resource);
- Add Private Child (if the resource is a private resource);
- Delete;
- Edit (if the resource is a shared resource);
- Personalize (if the resource is a private resource);
- View.

In addition the resource inherits the permissions assigned to the parent resource, unless a propogation block is in place.

Private resources are those resources that can only be accessed by the owner of that resource. Therefore Add private child enables a child resource to be created that only allows access to the owner of that resource. Personalise is the same permission as edit, but for a private resource. Shared resources are resources that can be accessed by more than one user.

By default, the Action sets are as shown in AC.4.and no role blocks are set on creation of a resource.

6.1.1.2 Management of Access Control

AC.3 The Administrator@Portal and Security Administrator@Portal roles contain the (*Grant Access On, (the virtual resource) portal)* permission, which is not available to any other role. This permission allows the Administrator or Security Administrator to make arbitrary changes to the access control configuration of all resources that are internally managed by the portal. The Administrator and Security Administrator can view, create and delete roles, role assignments, and inheritance blocks.

The Access Control administration can be performed using corresponding portlets within the running portal or via the *XmlAccess* scripting interface. Running an XmlAccess script requires the actions (Grant Access On, (the virtual resource) portal) and (Grant Access On, (the virtual resource) XmlAccess).

WebSphere Portal supports delegated access control administration. An administrator is a user who is authorized to modify the access control configuration by changing role assignments and creating or deleting role blocks. Administrators can delegate specific subsets of their administrative privileges to other users or groups. These users or groups can in turn delegate subsets of their privileges to additional users and groups. The delegated administration policy determines how users are permitted to delegate their privileges.

The general policy for creating, viewing or deleting role assignments is as follows: A user *U* can view, create or delete a role assignment for a specific user or group *UG* to a role identified by Action Set *AS* and resource *R* in either of the following cases:

- All of the following criteria below are met:
  - *U* has the Security Administrator@*R* and *U* has the AS@*R* or Administrator@R role
  - *U* has the Delegator@*UG*, Security Administrator@*UG*, or Administrator@*UG* role.
- *U* has the Administrator@Portal or Security Administrator@Portal role

For example, in order to assign a group to the role type on a resource, you must have at least the Delegator@*Group* + Security_Administrator@*Resource* + RoleType@*Resource* roles.

The general policy for creating, viewing or deleting role blocks is as follows: A user *U* can view, create or delete a role block on a specific resource *R* and a Action Set *AS* in either of the following cases:

One of the following criteria is met:

- *U* has the Security Administrator@*R* and has the *AS*@*R* role
- *U has the* Administrator@*R* role
- *U* has the Security Administrator@Portal or Administrator@Portal role.

6.1.1.3 Actions

AC.4 Actions are provided as part of a set. The following actions are available:

- The *Grant Access On* action represents the activity of granting or revoking other principals access permissions to the access control configuration on a specific resource;
- The *Delegate To* action supports the activity of delegating a permission to a specific principal; For the complete set of actions necessary to allow a user to delegate a role assignment to a specific principal for a resource see the description of the Delegated Administrative Policy in AC.3;
- The *Add Child* action represents the creation of a new, shared resource underneath an existing resource;
- The *Add Private Child* action represents the creation of a new private resource underneath an existing resource that can be accessed by a single user only;
- The *Delete* action represents the deletion of a resource or the removal of a resource from its parent resource (e.g. when a resource is moved from one place in the topology to an other);
- The *Edit* action represents all modifications to a resource (e.g. changing the meta-information of a resource) that are visible not only to the owner of a resource;
- The *Personalize* action represents all modifications to a resource that are only visible to the owner of a resource (this may imply the creation of an implicitly derived resource);
- The *View* action represents the presentation of the content or meta-information of a resource.

Actions are part of an 'Action set'. Action sets are also called role types, since they characterize a specific type of roles that can be created from those action sets. The TSF shall maintain the following action sets to ensure secure operation of the TOE. These actions are set as default within WP and cannot be edited.

Action/Action Sets	Admin	Security Admin	Delegator	Manager	Editor	Privileged User	User
Grant Access On	✓	✓					
Delegate To	✓	✓	✓				
Add Child	✓			✓	✓		
Add Private Child	✓					✓	
Delete	✓			✓			
Edit	✓			✓	✓		
Personalize	✓					✓	
View	✓			✓	✓	✓	✓

6.2 Assurance Measures

Assurance measures will be adopted to address each of the EAL2 assurance requirements, as summarised in table B.1 within [CC] and the International Interpretations 003, 004, 016, 019, 027, 051 (Rev.1). The following table provides a summary:

Assurance Component	Description of how Requirement will be met
ACM_CAP.2	A description of the configuration management used by the developers will be provided together with a configuration list, which will identify the items that comprise the TOE. This document will uniquely reference the TOE stated within Section 1 of this ST. Confirmation that the TOE is labelled with the correct reference will be provided during testing.
ADO_DEL.1	The developers will provide the evaluators with the delivery procedures used to ensure that security is maintained when distributing versions of the TOE to the user's site.
ADO_IGS.1	Procedures for the secure installation, generation and start-up, will be provided.
ADV_FSP.1	An informal description of the TSF and its external interfaces, describing effects, exceptions and interfaces will be provided to the evaluators.
ADV_HLD.1	A high-level design will be provided, which informally describes the components of the TSF. The security of each of these

	components will be described. All hardware, software and firmware required by the TOE will be identified. A presentation of the functions provided by the supporting protection mechanisms implemented in these, will also be included. It will also identify the interfaces between the components and which of these are externally visible.
ADV_RCR.1	This correspondence information will be contained within the Functional Specification and high-level design. This will provide a correspondence analysis between the TOE summary specification, the functional specification and the high level design.
AGD_ADM.1	The WP operational documentation that described to the administrator how to operate the TOE in a secure manner will be provided. This will describe the administrative security functions and interfaces available to the administrator. All details of any warnings about functions and privileges and assumptions about user behaviour will be included. Secure parameters under the control of the administrator will be provided, indicating secure values where applicable.
AGD_USR.1	The WP operational documentation for normal users will be provided, which describes: • The security functions and interfaces available to non-administrators of the system; • The use of user accessible security functions; • User accessible functions and privileges; • Assumptions regards behaviour of the user; and • All security requirements for the IT environment.
ATE_COV.1	Coverage of the TSF by the developers functional testing to the functional specification will be provided to the evaluators as part of the testing documentation.
ATE_FUN.1	Testing documentation will be provided, which describes the functional tests performed by the developers. This document will include test plans, test procedures, expected and actual test results, It will also identify the security functions to be tested.
ATE_IND.2	Resources will be made available to the evaluators such that they are able to perform additional, independent testing.
AVA_SOF.1	There are no functions within the TOE that have a strength and therefore no Strength of Functions analysis will be produced.
AVA_VLA.1	A description and analysis of any potential vulnerability identified within the TOE will be performed. This will be documented together with an explanation of why the vulnerabilities cannot be exploited.

7 Rationale

This chapter presents the evidence used in the ST evaluation and supports the claims that the ST is a complete and cohesive set of requirements.

7.1 Correlation of Threats, Policies, Assumptions and Objectives

The following table provides a correspondence of the threats, policies, assumptions and objectives:

Objectives:	O.ACCESS	O.MANAGE	O.ADMIN	O.CONFIG	O.APP	O.PROTECT	O.RECOVER
T.ACCESS_RES	x	x				x	x
T.NETWORK					x	x	
T.APP			x	x	x	x	x
P.ACCESS	x	x	x	x	x		
A.APP			x	x	x		
A.PROTECT						x	
A.ADMIN			x				

7.2 Security Objectives Rationale

This section demonstrates that the security objectives stated in section 4 of this ST are traceable to all of the aspects identified in the TOE security environment and are suitable to cover them.

7.2.1 Threats

This section provides evidence demonstrating coverage of the threats by both the IT and non-IT security objectives.

[T.ACCESS_RES]

An authorised user of the TOE gains access to an object without the correct authority to access that object.

The objective O.ACCESS counters this directly by ensuring that only those users with the correct authority can access an object. This is supported by O.MANAGE, which ensures that privileged actions are performed effectively.

The following environmental objectives support O.ACCESS in countering the threat:

- O.PROTECT – ensures that no resources can be accessed via the cabling between the workstations on which the TOE is installed;
- O.RECOVER – ensures that following a system failure, the TOE is not operating in an insecure state whereby an unauthorised user can gain access to objects they are not authorised to access.

[T.NETWORK]

Data transferred between workstations is disclosed to, or modified by unidentified users or processes, either directly or indirectly.

Administrators must ensure that data transferred between workstations i.e. along network cabling, is suitably protected against physical or other (e.g. Sniffing) attacks that may result in the disclosure, modification or delay of information transmitted between workstations. Objective O.PROTECT ensures that this is achieved. O.APP ensures that the protocols used in the transmission of data have been correctly configured within the applications.

[T.APP]

The applications that the TOE depends upon, become compromised.

It is essential that the administrator manage the applications in a secure manner so that vulnerabilities do not exist, which may lead to compromise of the TOE. The objectives O.APP, O.CONFIG, O.PROTECT and O.RECOVER all ensure that the applications are managed in a secure manner. O.ADMIN further supports this threat by ensuring that the administrator is a competent individual who will apply the latest patch information within the environment and therefore ensuring that any vulnerabilities that may compromise the security of the applications that become known, will be countered.

7.2.2 Security Policy

This section provides evidence demonstrating coverage of the organisational security policy by both the IT and non-IT security objectives.

[P.ACCESS]

The right to access a resource is determined on the basis of:

- *User membership of a group(s);*
- *User or group(s) ID association with a role;*
- *Resource association with an Action set (and thus creation of a role); and*
- *Actions assigned to the action set;*
- *Permission inheritance given by the protected resource hierarchy and role blocks*

This policy is implemented through the objective O.ACCESS, which provides the means of controlling access to objects by users and processes. O.MANAGE supports this policy by the administrators ensuring that the policy is maintained.

O.ADMIN, O.CONFIG and O.APP further support this policy by ensuring that the applications are configured in a secure manner so that no vulnerability may exist that enables an unauthorised user to gain an authorised identity.

7.2.3 Assumptions

This section provides evidence demonstrating coverage of the assumptions by both the IT and non-IT security objectives.

[A.APP]

It is assumed that the applications that the TOE relies upon, have been configured in accordance with the manufacturer's installation guides and where applicable, in its evaluated configuration. It is securely configured such that the applications protect the TOE from any unauthorised users or processes.

O.APP is the primary environmental objective that satisfies the assumption. This ensures that the administrator installs and configures the supporting applications in accordance with:

- The manufacturers instructions; and
- Any evaluated configurations were applicable.

O.ADMIN and O.CONFIG support this by ensuring that the Administrator is a competent and trustworthy person and that the users have been set up appropriately.

[A.PROTECT]

It is assumed that all hardware within the environment, including network and peripheral devices, have been approved for the transmittal of protected data. Such items are to be physically protected against threats to the confidentiality and integrity of the data.

The environmental objective O.PROTECT ensures that the network cabling is suitably protected against threats of modification, tampering or interruption of the data transmitted via this medium.

[A.ADMIN]

It is assumed that there are one or more competent individuals that are assigned to manage the TOE and the security of the information it contains. Such personnel are assumed not to be careless, wilfully negligent or hostile.

O.ADMIN is the primary objective that meets this assumption, which ensures that the administrator is a competent and trustworthy person who is capable of managing the TOE in a secure manner.

7.3 Security Requirements Rationale

7.3.1 Security Functional Requirements Rationale

This section demonstrates that the functional components selected for the TOE provide complete coverage of the defined security objectives. The mapping of components to security objectives is illustrated in the table below.

Security Objective	Functional Component
O. ACCESS	Complete Access Control (FDP_ACC.2) Security Attribute Based Access Control (FDP_ACF.1) Management of Security Attributes (FMT_MSA.1) Static Attribute Initialisation (FMT_MSA_E.3)
O.MANAGE	Management of Security Attributes (FMT_MSA.1) Static Attribute Initialisation (FMT_MSA_E.3) Specification of Management Functions (FMT_SMF.1) Security Roles (FMT_SMR.1)

[O.ACCESS]

The TOE must ensure that only those users with the correct authority are able to access a resource.

The access control mechanism must have a defined scope of control [FDP_ACC.2] with defined rules [FDP_ACF.1]. Authorised users must be able to control who has access to the objects [FMT_MSA.1]. Protection of these objects must be continuous, starting from object creation [FMT_MSA_E.3]

[O.MANAGE]

The TOE must allow administrators to effectively manage the TOE and that this is only performed by authorised users.

The TSF must enable an authorised administrator to manage the TOE in accordance with the access control SFP [FMT_MSA.1]. On creation of resources default values will be used, which enables ease of management [FMT_MSA_E.3]. [FMT_SMF.1] specifies the management functions provided by the TOE. [FMT_SMR.1] defines roles in order that the TOE is managed effectively.

7.3.2 Security Environment Requirements Rationale

This section demonstrates that the functional components provided by the environment for the TOE, provide complete coverage of the defined security objectives. The mapping of requirements to security objectives is illustrated in the table below.

Requirement for Environment	Security Objective
User Attribute Definition (FIA_ATD.1)	O.CONFIG
User Identification (FIA_UID.1)	O.CONFIG

7.3.2.1 User Attribute Definition

[O.CONFIG] states that *Those responsible for the TOE environment must ensure that each user on the supporting applications have associated user IDs and where applicable have an associated Group ID.* This satisfies the requirement FIA_ATD.1 on the IT environment because the requirement ensures that the user and group IDs of users are maintained within the environment.

7.3.2.2 User Identification

[O.CONFIG] states that *Those responsible for the TOE environment must ensure that each user on the supporting applications have associated user IDs and where applicable have an associated Group ID.*. This satisfies the requirement on the IT environment FIA_UID.1 because the requirement ensures that each user shall be successfully identified and allows the assumption of the anonymous user ID. Therefore each user on

the system would be identified either by the unique ID supplied by WAS, or by the anonymous user ID.

7.3.3 Explicitly Stated Security Requirements Rationale

As stated within Section 5 of this ST, FMT_MSA_E.3 has been explicitly stated and was not specified using CC Part 2 functional components. The reason for this is because WP does not provide functionality to define alternate initial values that override the default values when an object has been created. This does not reduce security as the default values used are the most restricted that would enable normal operation of the TOE.

7.3.4 Security Assurance Requirements Rationale

This ST contains assurance requirements from the CC EAL2 assurance package.

The EAL chosen is based on the impact that the statements of the security environment and objectives within this ST have on the assurance level. The administrator shall be capable of managing the TOE such that the security is maintained (O.ADMIN) particularly within the applications that the TOE relies (O.APP), and that the physical environment protects the TOE from any potential vulnerability (O.PROTECT). This EAL level also provides a low to moderate level of independently assured security without demanding additional effort by the developers.

Given the level of assurance required to meet the TOE environment and the intent of EAL2, this assurance level was considered most applicable for the TOE described within this ST.

7.4 SFR Dependencies

The below table identifies all of the dependencies of the SFRs included in the ST. Only those SFRs that have a dependency, or are depended upon are shown in the table. The dependency is shown with a ‘x’.

	FDP_ACC.1	FDP_ACF.1	FIA_UID.1	FMT_MSA.1	FMT_MSA.3	FMT_SMF.1	FMT_SMR.1
FDP_ACC.2		x					
FDP_ACF.1	x				x		
FMT_MSA.1	x					x	x
FMT_MSA.3				x			x
FMT_SMR.1			x				

It should be noted that the dependency on FDP_ACC.1, has been satisfied by FDP_ACC.2 as this SFR is hierarchical.

As shown in [CC], all dependencies are satisfied by the TOE, with the exception of FIA_UID.1, which is met by the IT environment of the TOE.

The dependency upon FMT_MSA.3 has been provided by the explicitly stated requirement FMT_MSA_E.3. This satisfies the requirement as *FMT_MSA_E.3* provides restrictive default values. The difference between the reliance on FMT_MSA.3 and FMT_MSA_E.3, is that the TOE does not provide the ability for a user to modify those default values. This is inherently more secure as weaker configurations are not possible.

7.5 TOE Summary Specification Rationale

This section demonstrates that the TOE security functions and assurance measures are suitable to meet the TOE security requirements.

7.5.1 TSF correspondence to SFRs

This section demonstrates that the combination of the specified TSFs work together so that the SFRs are satisfied. The table below shows the TOE security functions, which together satisfy each SFR element.

SFR	SFs
FDP_ACC.2	AC.1
FDP_ACF.1	AC.1 and AC.4
FMT_MSA.1	AC.3
FMT_MSA_E.3	AC.2
FMT_SMF.1	AC.3
FMT_SMR.1	AC.2 and AC.3

FDP_ACC.2

This requirement defines the subjects and objects that are enforced by the Access control policy and that all operations between the subjects and objects are covered by the Access Control Policy.

AC.1 defines what entities are considered to be subjects and objects within the scope of control for the Access control policy. A description of the access control policy is also provided.

FDP_ACF.1

This requirement states the security attributes associated with each of the subjects and objects identified in the requirement FDP_ACC.2. and states the rules to determine if access is granted to a resource.

AC.1 describes the access control policy including details of the propagation and inheritance blocks. AC.4 describes the actions that are available within the action sets, and explains what actions are assigned to which action sets.

FMT_MSA.1

This requirement describes the policy that allows users to create, view and delete the *role and Action Set associations, propagation blocks and inheritance blocks* security attributes. AC.3 further describes this policy.

FMT_MSA_E.3

This requirement states that the TOE shall provide restrictive default values for the security attributes stated in requirement FDP_ACF.1.1. AC.2 provides the details of the default roles that are created on installation of the TOE, and the default security attributes on creation of a resource.

FMT_SMF.1

AC.3 describes the policy for users to manage the security attributes User and Group to action set association; Resource to action set association; and *Inheritance and propagation blocks.*

FMT_SMR.1

This requirement ensures that the roles Administrator@Portal; and Security Administrator@Portal are maintained by the TSF and that the TSF shall be able to associate users with roles. AC.2 confirms that these roles are created on installation of the TOE and further information on these roles are provided by AC.3