RSA SECURITY™

RSA ClearTrust 4.7

Administrator’s Guide

Last Revised: March 6, 2002 5:17 pm

Contact Information

See our Web sites for regional Customer Support telephone and fax numbers.

RSA Security, Inc.
www.rsasecurity.com

RSA Security Ireland Limited
www.rsasecurity.ie

Trademarks

License agreement

Note on encryption technologies

This product may contain encryption technology. Many countries prohibit or restrict the use, import, or export of encryption technologies, and current use, import, and export regulations should be followed when exporting this product.

Distribution

Limit distribution of this document to trusted personnel.

First Printing: March 2002

P/N 3502A0

Contents

Preface

This *Administrator’s Guide* provides a complete overview of the RSA ClearTrust® administration tool, the Entitlements Manager. This guide explains the RSA ClearTrust administration concepts and data model. As an administrator, you will use the RSA ClearTrust Entitlements Manager to manage users and groups, protect resources and assign access privileges.

The Entitlements Manager is a Web-based, Java Server Page (JSP) application that you can set up on any supported application server or servlet engine. Administrators can then access the Entitlements Manager from any computer via a Web browser. See the *RSA ClearTrust Installation and Configuration Guide* for installation instructions.

The Entitlements Manager also includes an online help program with step-by-step instructions for completing specific tasks. Use this guide together with Entitlements Manager online help to understand and perform the administration tasks of RSA ClearTrust.

About this Guide

This guide is intended to give a general understanding of the RSA ClearTrust Entitlements Manager administration tool — its data model, features, and administration concepts. The intended audience of this document is security administrators or help desk personnel who are responsible for managing user access to Web-based resources using RSA ClearTrust.

This guide contains the following chapters:

- Chapter 1, “Getting Started with RSA ClearTrust Administration”. This chapter contains all of the information you will need to know prior to administering users, resources and policy in RSA ClearTrust. This chapter familiarizes you with the administrative concepts and terminology of RSA ClearTrust, and helps you plan and prepare your initial Web access management deployment.
- Chapter 2, “Managing Delegated Administration”. This chapter explains RSA ClearTrust’s administrative model. The Entitlements Manager allows you to delegate administration responsibilities by creating logical *Administrative Groups*.
- Chapter 3, “Managing Users and Groups”. This chapter explains how to manage your user and group data in RSA ClearTrust.
- Chapter 4, “Managing Resources”. This chapter explains how to define and maintain the resources that you will protect with RSA ClearTrust.
- Chapter 5, “Managing Security Policy”. This chapter gives detailed information on the concepts and methods for protecting system resources with RSA ClearTrust.

This guide also contains the following appendices:

Appendix A, "Glossary of Terms". This appendix provides definitions to terminology used in the Entitlements Manager, and concepts related to RSA ClearTrust administration.

Note: The Entitlements Manager online help program, which is installed with the product, contains detailed step-by-step procedures for all of the administration tasks described in this guide.

Related Documentation

For more information about the RSA ClearTrust product, refer to the following guides in this 4.7 documentation set:

- **Overview Guide.** This guide provides a comprehensive overview of the system components, supported platforms, and features of RSA ClearTrust.
- **Installation and Configuration Guide.** This guide provides instructions for installing and configuring the RSA ClearTrust Servers, Data Adapters, Web Server Agents and the Entitlements Manager Web-based administration tool on your chosen operating system. This guide also contains detailed descriptions of the different configuration options, features and production environment considerations.
- **Developer's Guide.** This guide provides information about developing custom programs using the RSA ClearTrust application programming interfaces (APIs). Reference documentation for the RSA ClearTrust APIs can also be found in the `/api` directory of your RSA ClearTrust 4.7 CD.

Document Conventions

These document conventions are used consistently throughout the documentation.

Typographical Conventions

Convention	Meaning	Example
San-serif bold	User interface elements such as buttons, menus choices, window names, dialog boxes, field names and so on will appear in san-serif bold text.	Select **File** ▸ **Print**. Click **Save**.
SAN-SERIF BOLD UPPERCASE	Keyboard keys, including letters and numbers as well as Tab, CTRL, ALT, and so on, will appear in san-serif bold uppercase text.	Press **CTRL+ALT+DELETE**
Italics	New terms, emphasized words or book titles will appear in italics.	See the *Administrator’s Guide* for more information about using the Entitlements Manager.
UPPERCASE	Environment variables, SQL commands, logical operators, device names, acronyms, registry settings, system commands, and so on will appear in all uppercase letters.	SELECT object_name FROM user_objects Mount your CD-ROM drive.
`Courier`	Code examples, files, directories, class names, commands, parameters and on-screen computer output will appear in courier font.	Edit the `aserver.conf` file in the `\conf` directory.
`Courier bold`	Typed input, as opposed to on-screen computer output, will appear in bold courier font.	Enter the hostname of your Web server here: **`web1.rsa.com`**

`<`*`italics`*`>`	Italicized text contained within the less than (<) and greater than (>) symbols denotes information to be determined by the reader. Substitute the appropriate name, directory, or other specific information.	`print <`*`filename`*`>`
		`<`*`ct_home`*`>/cleartrust/conf`

`[]`	Text contained within square brackets denotes optional information.	`reject [-d] <`*`filename`*`>`
`|`	Text separated by the pipe symbol denotes an either/or relationship.	`true|false`
`$`	Bourne, Bourne Again or Korn shell prompt for UNIX commands	`$`

Convention	Meaning	Example
%	C shell prompt for UNIX commands	%
#	Super user or root prompt for UNIX commands	#

Comment Icons

Comment icons identify particular types of information, as the following table describes.

Icon	Alert Labels	Description
!	**Warning:** **Important:**	Identifies paragraphs that contain vital instructions, cautions or critical information.
🗎	**Note:** **Tip:**	Identifies paragraphs that contain notes, RSA recommendations or other helpful product information.

Getting Support and Service

SecurCare® Online	www.rsasecurity.com/support/securcare
General Technical Support Information	www.rsasecurity.com/support

1 Getting Started with RSA ClearTrust Administration

This chapter contains information to prepare you for RSA ClearTrust administration, including prerequisites, overview information and an administrator's task list. This chapter contains the following sections:

- "Before You Begin"
- "Understanding the RSA ClearTrust Data Model"
- "Getting Started in the Entitlements Manager UI"
- "Administrator's Task List"

Before You Begin

Before you begin using the RSA ClearTrust Entitlements Manager, you will need to understand the basic administrative concepts and terminology. Also, your RSA ClearTrust system must be installed and configured prior to using the Entitlements Manager, and your logical architecture should be planned out ahead of time. You may need to collaborate with your system administrators, directory administrators, or other IT support technicians, as many of these preconditions involve the technical setup and configuration of the RSA ClearTrust components.

Before you can begin RSA ClearTrust administration:

□ **Become familiar with the concepts and terminology.** See "Understanding the RSA ClearTrust Data Model" on page 3, as well as the "Glossary of Terms". Once you have reviewed the administration concepts in this guide, use the Entitlements Manager online help program for step-by-step procedure instructions.

□ **Plan your logical architecture for RSA ClearTrust.** This involves defining your users and groups, determining your delegated administration hierarchy, identifying all of the resources you want to protect, and deciding on your security policy. The information in the *RSA ClearTrust Overview Guide, Chapter 5* will help you plan your logical architecture for RSA ClearTrust.

□ **Install the RSA ClearTrust Servers.** In order to run the Entitlements Manager administration tool, you must have your RSA ClearTrust Servers installed and configured. *See the RSA ClearTrust Installation and Configuration Guide, Chapter 2* for more information.

- □ **Install and configure the RSA ClearTrust Agents.** In order to protect a Web server (or application server) in the Entitlements Manager, you must first install and configure the RSA ClearTrust Agent on each server you want to protect. The Web server name in the Agent's configuration file must match the Web server's name in the Entitlements Manager in order for the resources on that server to be protected. See the *RSA ClearTrust Installation and Configuration Guide, Chapter 5* for more information.

- □ **Set up your RSA ClearTrust Data Stores.** The Entitlements Manager interacts with data stored in your native data server environment, such as an LDAP directory. You must prepare your data server environment before you can use the Entitlements Manager. See the *RSA ClearTrust Installation and Configuration Guide, Chapter 3* for instructions on installing the LDAP Data Adapter, and mapping your existing user and group data for use by RSA ClearTrust.

- □ **Set up the Entitlements Manager Web application.** The Entitlements Manager is a Web-based JSP (Java Server Page) application, and must first be deployed on an application server. Once the Entitlements Manager application is installed, administrators can access the application from any machine using a Web browser. See the *RSA ClearTrust Installation and Configuration Guide, Chapter 4* for installation instructions.

- □ **Determine if you are using active or passive authorization mode.** When you configure your RSA ClearTrust Authorization Servers, you must choose your default authorization mode. The default authorization mode controls access to resources that are not explicitly protected in the Entitlements Manager tool. *Active* mode (which is the default setting) means that users can access any resource on a protected Web server, unless they are *explicitly denied* access in the Entitlements Manager. *Passive* mode means that users cannot access any resource on a protected Web server, unless they have been *explicitly granted* access in the Entitlements Manager. See Chapter 4, "Authorization Mode: Active and Passive Modes" for more information.

  The default authorization mode is controlled by a configuration parameter in your RSA ClearTrust's Authorization Servers' configuration files. See the *RSA ClearTrust Installation and Configuration Guide, Appendix A* for more information.

- □ **Determine if you will be using the Entitlements Manager to edit your User and Group data.** The Entitlements Manager has the capability to add, modify and delete user and group accounts. However, you should only use the Entitlements Manager to edit your user and group data if that data was created within RSA ClearTrust. If you originally saved your users and groups with a non-RSA ClearTrust tool, or if you are using multivalued LDAP attributes, RSA Security recommends that you continue to use your existing user administration tool to manage this data.

You can configure RSA ClearTrust to access your LDAP user and group data in read-only mode, thus protecting the integrity of your data. If your RSA ClearTrust Entitlements Server is configured as read-only, then you will not be permitted to modify or create user and group accounts from the Entitlements Manager. See the *RSA ClearTrust Overview Guide, Appendix B* and the *RSA ClearTrust Installation and Configuration Guide, Chapter 3* for more information.

Conversely, if you use the Entitlements Manager to access and modify your user and group data, make sure that you use *only* the Entitlements Manager. Manually modifying user and group data in other environments may break crucial relationships with administrative groups or other RSA ClearTrust data objects.

Understanding the RSA ClearTrust Data Model

This section introduces important terms and concepts of RSA ClearTrust administration. All of these terms and concepts are discussed in greater detail in the following chapters of this guide.

Understanding Users and Groups

People that can be given access to a protected item in RSA ClearTrust are referred to as *Users*. Users can be logically grouped together into *Groups*, and a group can contain users as well as other groups.

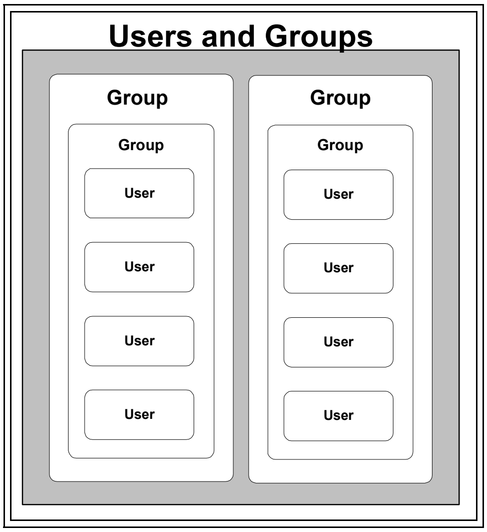


Figure 1.1 Example of the User and Group Hierarchy

Since RSA ClearTrust 4.7 uses your native data server as the backend data source, your user and group hierarchy for RSA ClearTrust resides in a directory schema in LDAP or relational tables in SQL. For this release of RSA ClearTrust, however, only LDAP directories are supported.

For more detail on user and group management, see Chapter 3, "Managing Users and Groups".

Understanding Resources

The term *Resources* refers to all of the items that are protected by RSA ClearTrust. A resource can be a Web server, a directory on your Web server, an application, or a particular file.

In the Entitlements Manager, you will first define the *Web servers* (or application servers) that you want to protect. Web servers contain files (*Uniform Resource Indentifiers* or URIs) and directories (URIs or *Server Trees*).

An *Application* is a collection of resources that are logically grouped together. An Application can consist of URIs and Application Functions. For example, you can create an Application called HR, which contains various resources such as benefit plans, the employee handbook, and so on. These application resources may reside in different directories or on different Web servers, but are logically grouped together in RSA ClearTrust.

An *Application Function* is a customized set of rules that specify how other resources can be accessed and manipulated. Application Functions are useful for access checking in situations that do not involve URI requests. For example, if you want to create policy for a non-Web Java application that has a sensitive method that you do not wish to make available to all users, you can protect it with an Application Function. Application Functions are developed using the RSA ClearTrust Administrative API, and then specified in the Entitlements Manager.

For more information on managing resources, see Chapter 4, “Managing Resources”.

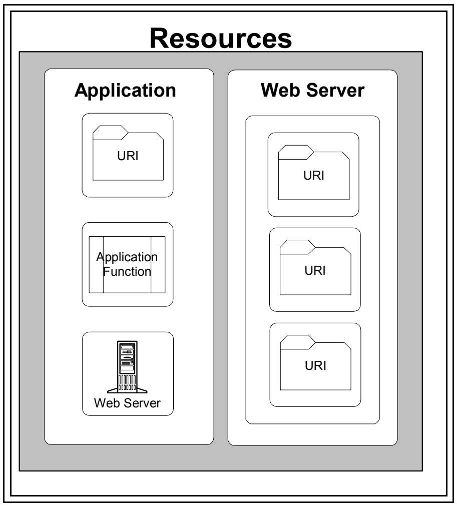


Figure 1.2 Example of the Resources Hierarchy

Understanding Policy

Policy is the link between users and resources. There are two types of policy in RSA ClearTrust—*SmartRules* and *Basic Entitlements*.

A Basic Entitlement allows you to either allow or deny access to a specific resource for a specific user or group of users. A Basic Entitlement can be applied to an *Application*, *Application Function* or to an individual *URI*.

SmartRules are dynamic access control policies that are created by Administrators. SmartRules are based on *User Properties* that you define. For example, you can create a SmartRule that only allows users that have an account balance of over $500.00 to see a certain page on your Web site. You would first have to create a User Property definition for Account Balance. Then you would define the SmartRule in the Entitlements Manager (Account Balance > $500.00). You can then apply the SmartRule to an Application, Application Function or to an individual URI. When the user tries to access the protected page, RSA ClearTrust makes the access control decision dynamically by determining if that user meets the criteria of the SmartRule at the moment of the request.

For more information on managing security policy, see Chapter 5, "Managing Security Policy".

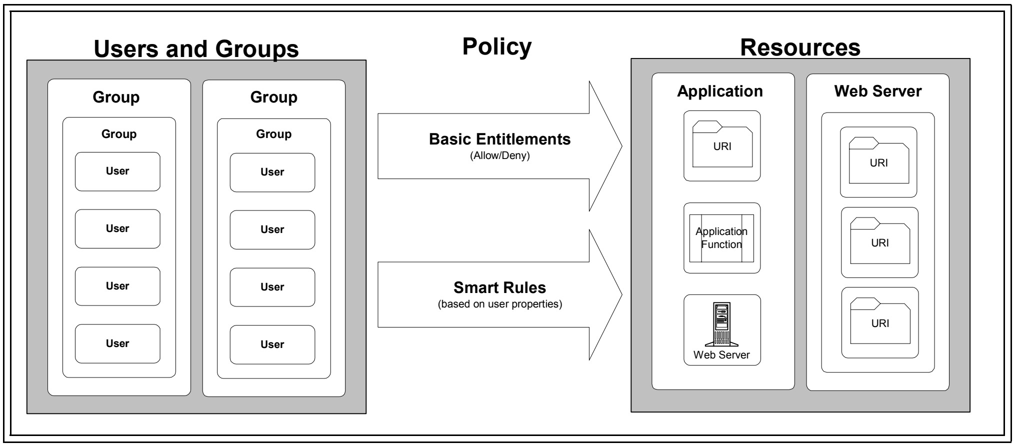


Figure 1.3 Example of the Policy Relationships

Understanding Delegated Administration

You have to create a separate *Administrative User* account in the Entitlements Manager for every user that you want to be an administrator in RSA ClearTrust. The administrative data for RSA ClearTrust is kept separate from your regular user data. An administrative user is assigned one or more *Administrative Roles*, and is also assigned to an *Administrative Group*.

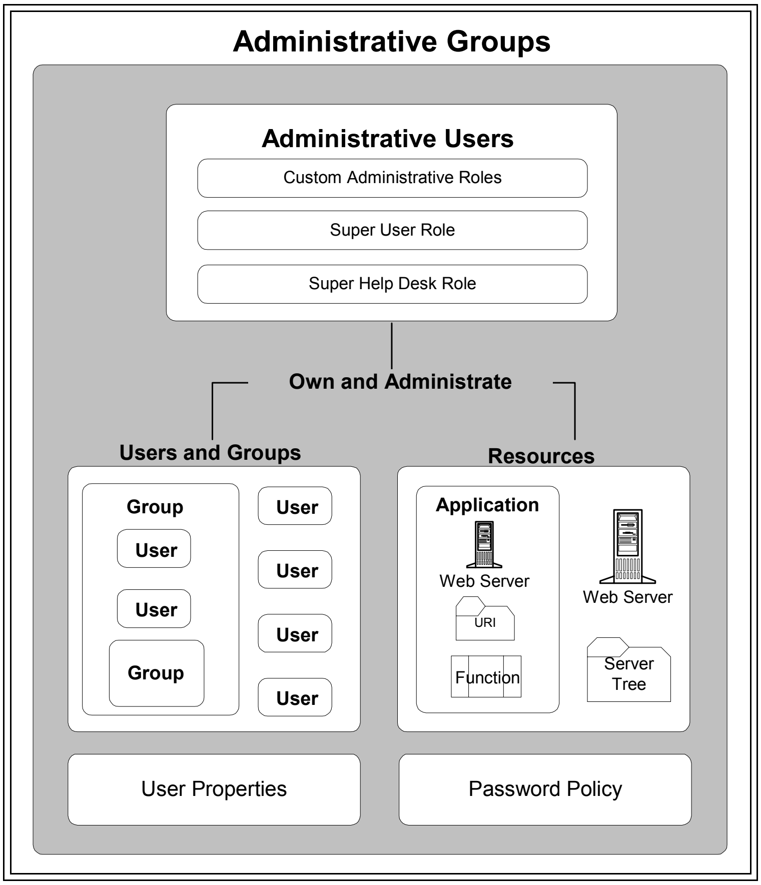


Figure 1.4 Example of the RSA ClearTrust Delegated Administration Model

Administrative Groups

An *Administrative Group* is a collection of objects (Users, Groups, Web servers, Server Trees, Applications, and User Properties) that are *owned* by the administrators assigned to that group. This allows you to delegate administration responsibilities by creating these logical administration groups and assigning ownership of these objects to the group's administrators. Each administrator in RSA ClearTrust belongs to an Administrative Group. When you first install the RSA ClearTrust Entitlements Manager, a default Administrative Group is created (called *Default Administrative Group*).

Administrative Groups are also referred to as *Virtual Business Units* (VBUs).

Public and Private Objects

Any object that can be owned by an Administrative Group (Users, Groups, Web servers, Server Trees, Applications, and User Properties) can be designated as either *Private* or *Public*. Private objects can only be viewed by the administrative users within the same Administrative Group. Public objects can be viewed by all administrative users, regardless of Administrative Group ownership.

Administrative Roles

Administrative users are also assigned one or more *Administrative Roles*, which determine the actions that they can perform within RSA ClearTrust. Administrative roles are usually named to reflect the real-world role of the administrator (for example, Help Desk or HR). The Administrative Role controls what actions the administrator can perform on the objects owned by their Administrative Group. When you first install the RSA ClearTrust Entitlements Manager, a default administrative role is created (called *Default Administrative Role*).

In addition to the administrative roles that you define, there are two additional roles that can be assigned to an administrative user—the *Super User* role and the *Super Help Desk* role.

Super Users

Super Users are administrators who have access to the highest level of administrative privileges. Super Users can view, create or modify any object in the RSA ClearTrust system, regardless of the Administrative Group that owns the object. They also have the permissions to manage the accounts of other administrators.

When you first install the RSA ClearTrust Entitlements Manager, an initial administrative user account is automatically created. This default administrative user account (named *admin*) has Super User privileges. This Super User account is the starting point for setting up your RSA ClearTrust administration model.

At a minimum, the RSA ClearTrust system can function with only one administrator, who is the Super User. However, delegated administration relieves the Super User of having to perform all the administrative tasks associated with controlling access to system resources. See Chapter 2, “Managing Delegated Administration” for more information.

Super Help Desk Users

The *Super Help Desk* role is a special kind of administrative user whose primary purpose is to reset passwords in RSA ClearTrust. A super help desk user is able to view all user accounts and User Properties and reset passwords for users, regardless of the Administrative Group that owns the user.

Password Policies

Each Administrative Group in RSA ClearTrust is assigned a password policy. Password policies establish certain rules regarding users’ passwords, such as a minimum length, or the maximum lifetime for a password. The password policy applies to all users within that Administrative Group. For more information on password policies, see “Password Policies” on page 19.

Getting Started in the Entitlements Manager UI

The RSA ClearTrust Entitlements Manager is a Web application that you access from a Web browser. This section explains some of the main functional areas in the Entitlements Manager user interface (UI), as well as instructions for logging on to the system for the first time.

Detailed instructions for performing specific tasks in the Entitlements Manager are available in chapters 2 through 5, and in the online help files, which are installed with the Entitlements Manager application.

Note: The Entitlements Manager has a five minute timeout. If you begin a lengthy process, such as updating maximum password lifetime for a large administrative group, the Entitlements Manager may time out and close before the process is finished. The process, however, will continue and finish on the servers.

Understanding the Menu Options

As shown in Figure 1.5, the Entitlements Manager has several menu options at the top of the screen. The menu options allow you to access different areas of functionality within the application, and complete specific tasks within each functional area.

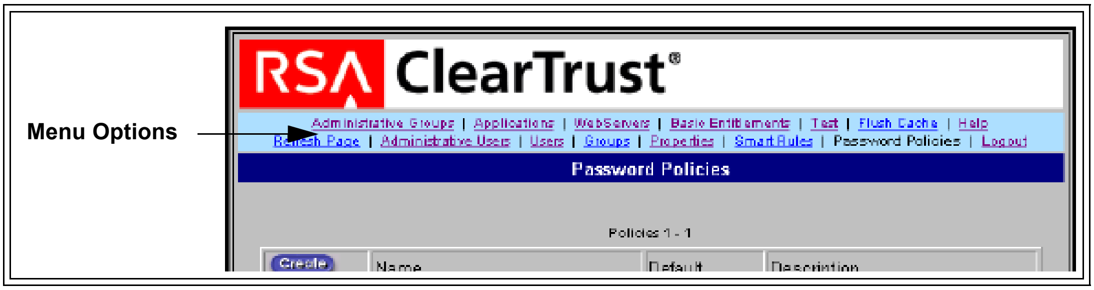

Menu Options
RSA ClearTrust®
Administrative Groups | Applications | WebServers | Basic Entitlements | Test | Flush Cache | Help
Refresh Page | Administrative Users | Users | Groups | Properties | SmartRules | Password Policies | Logout
Password Policies
Policies 1 - 1
Create
Name
Default
Description

Figure 1.5 RSA ClearTrust Entitlements Manager Window

The Entitlements Manager has the following menu options and commands (from left to right):

- **Administrative Groups.** This functional area allows you to define and manage Administrative Groups and Roles. Select this menu option to perform the following tasks:

  - Create a new Administrative Group definition or edit an existing one
  - Transfer ownership from one Administrative Group to another
  - Create an Administrative Role definition or edit an existing one
  - Assign Administrative Users to an Administrative Role

- **Applications.** This functional area allows you to define logical groupings of resources called *Applications* in RSA ClearTrust. Select this menu option to perform the following tasks:
  - Create a new Application definition or edit an existing one
  - Add an Application Function to an Application or edit an existing one
  - Add a URI to an Application or edit an existing one
  - Assign Administrative Group ownership for an Application or Application Function

- **Web Servers.** This functional area allows you to define Web Servers that are protected by RSA ClearTrust. Select this menu option to perform the following tasks:
  - Create a new Web Server definition or edit an existing one
  - Add a Server Tree (directory) definition to a Web Server or edit an existing one
  - View the URIs associated with a Web Server (URIs are created in the Applications area)
  - Assign Administrative Group ownership for a Web Server or a Server Tree

- **Basic Entitlements.** This functional area allows you to create Basic Entitlements (static access policy) between selected users (or groups of users) and resources. Select this menu option to perform the following tasks:
  - Select a User or a Group, then select Applications and create Basic Entitlements to the entire Application or to specific URIs and Application Functions within that Application.
  - Select a User or a Group, then select Web Servers and create Basic Entitlements to specific URIs on that Web Server.
  - Select a User or a Group, and view, edit or delete the existing Basic Entitlements for the selected User or Group.

- **Test.** This functional area allows you to test your security policy (both Basic Entitlements and SmartRules) by simulating a user trying to access a Web Server (or a particular URI on a Web Server). You can see if the user is granted or denied access to the resource by the RSA ClearTrust system, thus verifying that your access control policy is set up as expected.

- **Flush Cache.** This command clears all records that are currently cached on all of the RSA ClearTrust Authorization Servers. Incremental updates are automatically refreshed in the cache as you save your changes in the Entitlements Manager. Do not use this command unless you want to empty the cache for the entire RSA ClearTrust runtime system.
- **Help.** This command launches the Entitlements Manager online help program. The online help contains step-by-step procedures for performing administrative tasks in the Entitlements Manager application.
- **Refresh Page.** This command reloads the current page you are viewing. Use this command to update the page, and see the results of your most recent changes.
- **Administrative Users.** This functional area allows you to define and manage the Administrative Users for RSA ClearTrust. Only administrators with Super User privileges can create or modify other Administrative User's accounts. Select this menu option to perform the following tasks:
  - Search for an Administrative User
  - Create a new Administrative User account or edit an existing one
  - Assign Administrative Roles to an Administrative User
  - Assign an Administrative User to an Administrative Group
- **Users.** This functional area allows you to define and manage users for RSA ClearTrust. You should only use the Entitlements Manager to manage user accounts created within RSA ClearTrust. If your LDAP directory of users was created prior to installing RSA ClearTrust, RSA Security recommends that you continue to use your existing user administration tool to manage this data. Conversely, if you use the Entitlements Manager to access and modify your user and group data, make sure that you use *only* the Entitlements Manager. Select this menu option to perform the following tasks:
  - Search for a user accounts
  - View a user account

  If your user data is not configured as read-only, then you can also perform the following tasks:
  - Create a new user account or modify an existing user
  - Assign Administrative Group ownership for a user
  - Reset a user's password
  - Activate or deactivate a user account in RSA ClearTrust
  - Set the value of a User Property for a user

- **Groups.** This functional area allows you to define and manage groups for RSA ClearTrust. You should only use the Entitlements Manager to manage groups created within RSA ClearTrust. If your LDAP directory of users and groups was created prior to installing RSA ClearTrust, RSA Security recommends that you continue to use your existing user and group administration tool to manage this data. Conversely, if you use the Entitlements Manager to access and modify your user and group data, make sure that you use *only* the Entitlements Manager. Select this menu option to perform the following tasks:
  - View a list of all Groups
  - View an existing Group definition

  If your group data is not configured as read-only, then you can also perform the following tasks:
  - Create a new group definition or modify an existing group
  - Assign Administrative Group ownership for a group
  - Add users and other groups to an existing group
- **Properties.** This functional area allows you to define User Properties. User Properties are special attributes that are used for dynamic access control decisions (SmartRules). For RSA ClearTrust version 4.7, a User Property must map to an existing LDAP attribute in your user data store. Select this menu option to perform the following tasks:
  - View a list of the currently defined User Properties
  - Create a new User Property definition or modify an existing one
  - Assign Administrative Group ownership for a User Property
  - Specify if a User Property should be read-only (RSA ClearTrust will not update the property values in your LDAP directory data)
- **SmartRules.** This functional area allows you to create SmartRules (dynamic access policy) for selected resources. User access to those resources is then determined at runtime based on the business logic you define in the SmartRule. Before you can create a SmartRule, you must have created your User Property definitions. Select this menu option to perform the following tasks:
  - Select an Application, then create a SmartRule for that Application.
  - Select an Application, select an Application Function and then create a SmartRule for that Application Function.
  - Select a Web Server, select a URI and then create a SmartRule for that URI.
  - Select a resource (Application, Application Function or URI) and view or edit existing SmartRules for that resource.
- **Password Policies.** This functional area allows you to create the Password Policies that are associated with Administrative Groups.
- **Logout**. This command reloads ends your administration session. Select this command if you want to log out and then log back on as a different Administrative User.

Navigating in the Entitlements Manager Application

It is not possible to navigate in the Entitlements Manager application by using the **Forward** and **Back** buttons on your browser. If you need to exit a screen and go back to the previous screen, use the **Cancel** button in the lower right corner of the window.

To navigate to other pages in a lengthy list, click on the **Next** and **Previous** buttons.

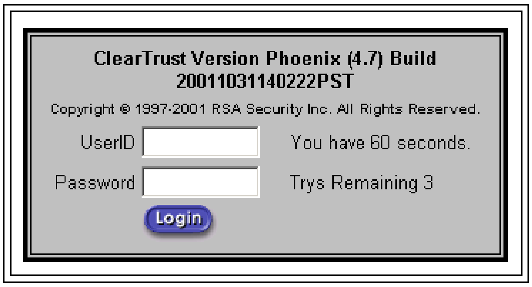
To select an object for certain processes, click the **Set** button to search for your selection. To select an item for use from a list of items, click the **Use** button next to the name of the item in the list.

Logging on to the Entitlements Manager

These instructions presume that you have installed and configured the Entitlements Manager; see the *Installation Guide, Chapter 4* for details.

To log on to the RSA ClearTrust Entitlements Manager

1. Access the URL of the Entitlements Manager from your browser as you would any other Web page (by typing the URL into the address field or by choosing the page from your bookmarks).
2. In a few seconds, the **Login** screen displays:



ClearTrust Version Phoenix (4.7) Build 20011031140222PST

Copyright © 1997-2001 RSA Security Inc. All Rights Reserved.

UserID

Password

You have 60 seconds.

Trys Remaining 3

Figure 1.6 Entitlements Manager Login Screen

3. Enter your **User ID** and **Password** and click **Login** to continue.
4. If this is the first time you are launching the Entitlements Manager application, log on using the default administrator account (with Super User privileges) and password:

   User ID: admin

   Password: *admin1234*
5. After you log on for the first time, select **Administrative Users** and edit the default Super User (**admin**) account. Change the user ID and password to something more secure.

Administrator's Task List

This task list presents the basic administrative tasks required to protect system resources with RSA ClearTrust. In the right column next to each task is a brief description of issues to consider before embarking on that task. Refer to chapters 2 through 5 of this guide for more information about these tasks.

Table 1.1 Administrator's Checklist

	Task	Considerations
Administrative Groups		
□	Create Administrative Groups. See "Administrative Groups" on page 16	Decide the organizing principle of your groups -- geographical divisions, business units, etc.
□	Create Administrative Roles. See "Administrative Roles" on page 21	Decide who will manage your administrative groups, and the range of each administrator's privileges.
□	Create Administrative Users. See "Administrative Users" on page 22	
Users and Groups		
□	Create User Properties. See "User Properties" on page 29	These are based on your plans to use SmartRules for certain security policies or your need to store additional user information. Each user property must be mapped to an LDAP attribute.
□	Create users (if you are not accessing LDAP user and group data in read-only mode). See "Users" on page 25	You can set User Properties as you create users; alternately, you can create or import users and set property values using the RSA ClearTrust Administrative API.
□	Create groups (if you are not accessing LDAP user and group data in read-only mode). See "Groups" on page 33	Your group structure will depend on your organization's structure as well as your plans to use Basic Entitlements to allow or deny group access to resources.
Resources		
□	Create web servers. See "Web Servers" on page 37	You must have on hand valid hostname and port numbers as well as the Web server Name matching the corresponding entry in the Web Server Agent configuration file. Also, you can assign ownership of the Web server to an Administrative Group.

Table 1.1 Administrator's Checklist

	Task	Considerations
□	Create Server Trees. See "Server Trees" on page 38	If you have planned to divide any single Web server's resources between Administrative Groups, define trees at this stage so that delegate administrators can create their own Applications, etc.
□	Create Applications. See "Applications" on page 39	The organization of these Applications will reflect your planning of logically related resources.
□	Create Application Resources. See "Application Resources" on page 39	Enter the specific URIs that you need to protect. Each of these will be entered under one of the applications you have created.
□	Create Application Functions. See "Application Functions" on page 43	To protect non-Web-based resources, collaborate with Developers to create Application Functions.
Security Policy		
□	Create SmartRules. See "SmartRules" on page 45	Create rules to meet your dynamic security policy needs.
□	Test SmartRules. See "Testing Security Policy" on page 53	Test SmartRules prior to implementation.
□	Create Basic Entitlements. See "Basic Entitlements" on page 49	Create rules to control access based on group or user identity.
□	Test Basic Entitlements. See "Testing Security Policy" on page 53	Test entitlements prior to implementation.

2 Managing Delegated Administration

Rather than require a single system administrator to control all the security policy for a protected system, RSA ClearTrust supports delegated administration. In delegated administration, the Super User can divide the system's resources and users into *Administrative Groups*, and then assign the administrative responsibility for these groups to other administrators. Each administrator has privileges specified in an *Administrative Role*, also created and assigned by the Super User.

Managing delegated administration involves these tasks:

- Grouping protected resources and users into administrative groups.
- Defining sets of privileges, or administrative roles, to define administrators' levels of control over the group's resources and users.
- Assigning administrative roles to selected administrative users, creating administrators of the groups.

When these tasks are completed for an administrative group, its administrators can manage the group's users and resources according to the privileges of their administrative roles, without any further involvement by the Super User. In this way, the Super User can delegate administrative privileges and responsibilities to other administrators.

This chapter discusses these key concepts of delegated administration in greater detail under the following headings:

- "Administrative Groups"
- "Administrative Roles"
- "Administrative Users"

Since password policies are associated with administrative groups, they are discussed in this chapter under "Password Policies" in the administrative groups section.

Administrative Groups

Administrative Groups are collections of resources, users, and other objects that have been grouped together for delegated administration. Administrative groups can be based on organizational structure, reflecting departmental divisions like Marketing, Sales, Shipping, and Engineering. They may be based on geography, with separate groups for the regions or administrative centers of the enterprise. In such cases, the group could own the department's or the region's users and groups, the Web servers that house its particular resources, and any special Web applications used by it.

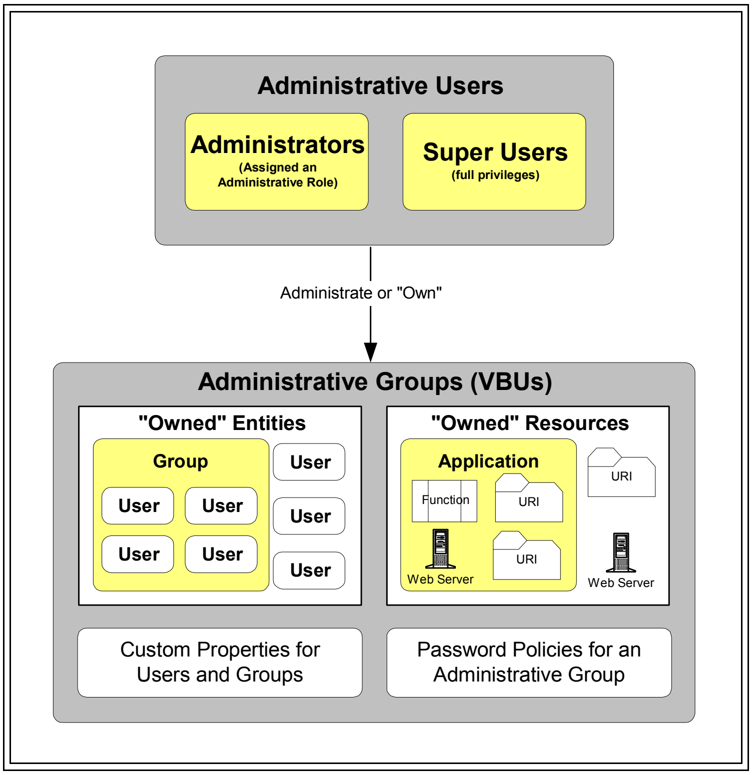


Figure 2.1 Administrative Groups and Administrative Users

Administrative groups can also include extranet partners, customers, and others external to the organization. In all cases, delegated administration should relieve the Super User of excessive responsibilities, and should distribute administrative tasks to the people most knowledgeable about a specific group of users and resources.

Administrative Group Ownership

Once the Super User creates an administrative group, all resources or users that are placed in the group or created within it are "owned" by the group. This means that only that particular group's administrators can modify or protect those resources. If the resources are designated as Private, then only administrators of the group can view them.

The following objects can be owned:

- **User** - A person's account on the system. A user's account information in the RSA ClearTrust system always includes Fixed Attributes (name, email, password, and so on). Administrators with the appropriate privilege can also define new User Properties (for example, Social Security Number or Job Title) that can be used to control access.
- **Group** - A group of users or member groups that are logically associated with each other. For example, you could create a group for each department in your organization (for example, Sales, HR, Finance, IT, and so on).
- **Application** - A set of programs and data that has been grouped together and named. Applications are a way of logically grouping resources together regardless of their physical location.
- **URI** - (Uniform Resource Identifier) A file or directory name on a Web server (usually the same as a URL without the domain name).
- **Server Tree** - Logical groupings of resources on the same Web server, Server Trees allow different administrative groups to own different directories on the server.
- **Web Server** - A logical name for your protected Web server or application server that you define in the Entitlements Manager.
- **User Property Definition** - Customized information to be associated with each user account. For more information about User Properties, see "User Properties" on page 29.

Only the Super User has the ability to give ownership, remove ownership, or transfer ownership of an administrative group's resources from one administrator to another. See the online help for detailed instructions on these procedures.

Note: When you delete an administrative group, ownership of all its resources is transferred to the Default Administrative Group. If you need to delete or reorganize an administrative group, you may wish to first transfer its resources to another group.

Public and Private Objects

Objects in the RSA ClearTrust system are typically designated as public - that is, they can be seen in the Entitlements Manager UI by all administrators, regardless of whether or not they are members of the administrative group that owns that object (though only members of the owning administrative group with the appropriate privileges can modify and delete public objects).

To control administrators' access to sensitive information, it is possible to designate an object as private. An object that is private only appears in the Entitlements Manager UI when the Manager is being used by administrators of the administrative group that owns the object.

The following objects can be designated as private by selecting the corresponding option in the Entitlements Manager when creating or editing the object:

- **User** (including all Fixed Attributes and User Properties associated with that user)
- **Group**
- **Application**
- **Web Server**
- **Server Tree**
- **User Property Definition**

In Figure Figure 2.2, "Public and Private Objects" on page 18 , administrators in the Extranet Admin Group cannot see any of the objects owned by the Intranet Admin Group (Web server 1, User 2 and User 1) because these objects have been made private. On the other hand, administrators in the Intranet Admin Group can see the Online Order Application and User 3 owned by the Extranet Admin Group because these objects are public.

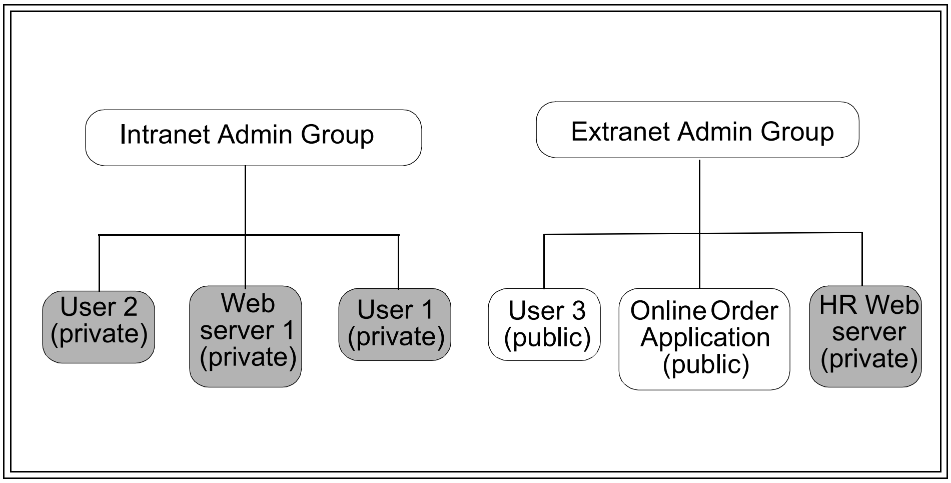


Figure 2.2 Public and Private Objects

By default, all objects are public. An object can be designated as private when it is being created, or modified to be private later.

Note: It is possible for a Super User to give a user access to resources which do not belong to that user's administrative group and which have been designated as Private. Those resources and their associated access permissions will not be visible to the user's administrative group's administrators. This functionality should be used with discretion when using the system as a Super User

Password Policies

RSA ClearTrust administrators can define and enforce *Password Policies* for each administrative group. These password policies are enforced in passwords for Basic authentication. See the *RSA ClearTrust Installation and Configuration Guide* for more detail on implementing Basic authentication.

If no password policy has been defined for an administrative group, then the default password policy will be applied. There must be a default password policy for the system at all times. The existing default password policy cannot be deleted, only modified.

When designing a password policy, consider user needs in balance with your security needs. Enforcing an excessively strict password policy (for example, one that requires overly long passwords or very frequent password changes) may cause users to compromise security -- most commonly by writing their passwords down.

Password policies are defined by their required length, restricted characters and words, lifetime, and expiration dates.

Important: Creating password policies or updating password lifetime places heavy demands on system resources. Be prepared for long processing time when you save password policies, especially for administrative groups owning large numbers of users.

Length

Passwords that are too short are vulnerable to brute force attacks, but passwords that are too long can be difficult to remember, and may cause problems in some programs. RSA ClearTrust allows administrators to specify a minimum and maximum required length for user passwords.

A password shorter than six characters is probably unacceptably weak, but a password longer than 32 characters could also be problematic. Because the algorithm used by RSA ClearTrust for encrypting passwords encrypts all passwords to be the same length, the length of the password is unimportant to the RSA ClearTrust system itself.

Non-Alphabetic Characters

Because the most common attacks used by crackers are dictionary attacks, adding a few non-alphabetic characters to a password can enhance a password's security greatly (for example, changing "password" to "password-327"). RSA ClearTrust password policies can be configured to require at least one non-alphabetic character.

Most common alphanumeric substitutions (the numeral 1 for the letter l, the numeral 3 for the letter E and the numeral 7 for the letter T, among others) have been integrated into password cracking tools.

Character Exclusion

If a password is going to be used in more than one environment (particularly if it is going to be used in a UNIX environment), it would be better if it did not contain certain non-alphabetic characters. Characters such as `&`, `*`, and `/` can have unpredictable effects when used in strings passed to some common UNIX commands. RSA ClearTrust allows you to reject potential passwords that include specified characters.

Dictionary Search

RSA ClearTrust will match potential new passwords against a list of words (in the file `words.txt`) and exclude passwords that are on the list. `words.txt` contains several thousand commonly-used words that will likely be included as part of any dictionary attacks on the system (for example, the word "password"). The RSA ClearTrust installation also includes empty.txt, to which you can add your own words to be excluded (for example, your company's name).

Password Lifetime

The longer a password exists, the more likely it is to be compromised. RSA ClearTrust includes a function to force users to change their passwords after a specified period of time has passed. When users' passwords expire, they will be locked out of any RSA ClearTrust-protected resources until they choose a new, valid password.

Important: Because each user record is retroactively updated, updating password lifetime places heavy demands on system resources. Be prepared for long processing time when you update password lifetime settings, especially for administrative groups owning large numbers of users.

Also, make sure that you enter an appropriate parameter or the Maximum Password Lifetime field -- either **d** (days), **h** (hours), **m** (minutes), or **s** (seconds). Failing to do this returns errors.

Password Expiration

A user password can be expired before the password lifetime is reached. Expiring the password forces the user to change his/her password the next time he or she accesses the system. This can be useful when creating new user accounts, as you can assign a default password, and then require the user to immediately choose their own password before they can access the system.

Unless you have implemented a dynamic solution such as the JSP example provided in the *RSA ClearTrust Developer's Guide*, users are presented with only a "Password Expired" page on password expiration.

You can expire passwords in these ways:

- Using the RSA ClearTrust Administrative API
- On creating or modifying a user in the Entitlements Manager, setting the user password expiration date to match the current date

- Selecting the **Expire Now** option in the user screen. See "User Information" on page 25 for more information on this setting.

As with all other Entitlements Manager password operations, your changes apply to passwords for basic authentication only -- not NT or SecurID authentication, which must be managed in the native NT or RSA Ace Server environment.

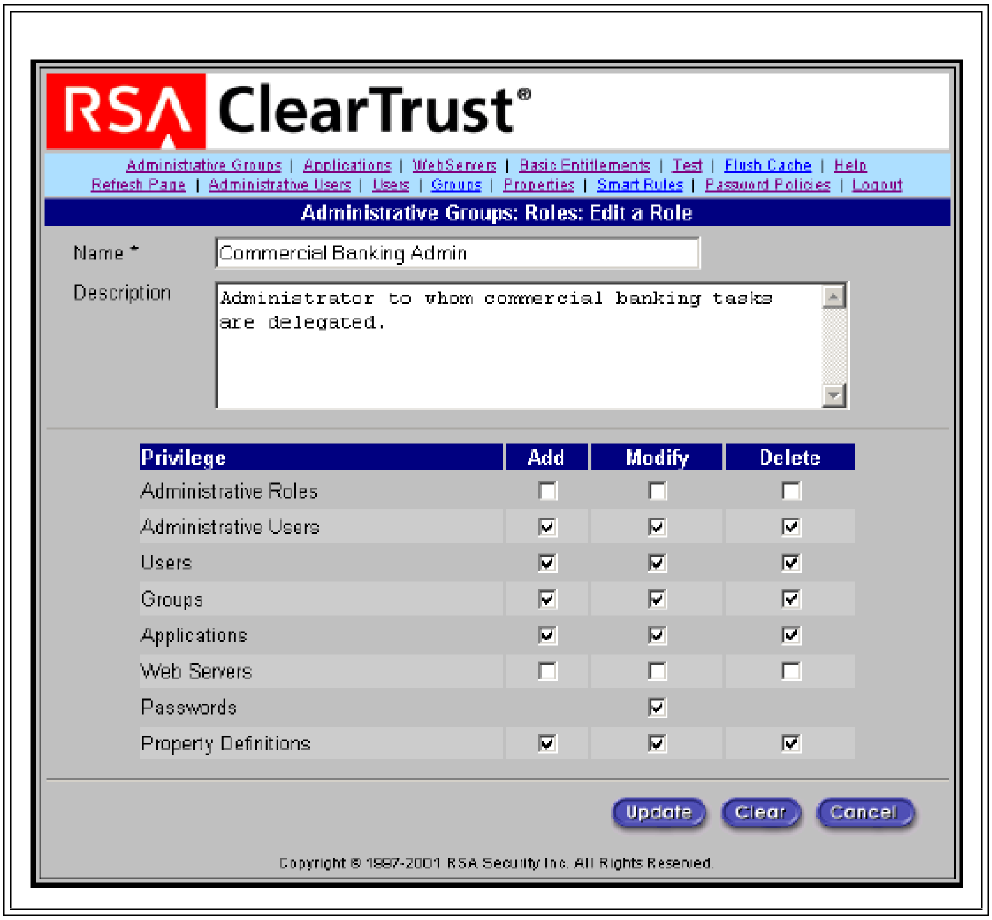
Administrative Roles

An Administrative Role is a collection of privileges that can be assigned to an administrator in an administrative group. For instance, a limited administrative role might include only the ability to add new applications, but not the ability to modify or delete them. A more extensive role might share all of the Super User's privileges except the ability to delete user property definitions.

You will name administrative roles when you create them, and then refer to the role names later when you apply them to particular administrative users. Roles typically reflect an administrator's function in the organization, such as Help Desk or Human Resources. Available administrative role privileges include:

- **Administrative Roles** - The ability to add, modify, or delete administrative roles.
- **Administrative Users** - The ability to add, modify and delete administrative users.
- **Users** - The ability to add users, modify a user's fixed attributes and user properties, and delete users.
- **Groups** -The ability to add groups, modify a group's properties, or delete groups.
- **Applications** - The ability to add, modify, or delete Web applications, including application resources (URIs).
- **Web Servers** - The ability to add, modify, or delete Web servers.
- **Passwords** - The ability to modify (but not create or delete) a user's password.
- **Property Definitions** - The ability to add, delete, or modify user property definitions.

Administrative roles are defined and updated in the **Create a Role** or **Edit a Role** pages of the Entitlements Manager, to be applied later when you create administrative users. In this example, the administrators of the Commercial Banking administrative group will have all privileges except the manipulation of administrative roles and web servers:


RSA ClearTrust®
Administrative Groups | Applications | WebServers | Basic Entitlements | Test | Flush Cache | Help
Refresh Page | Administrative Users | Users | Groups | Properties | Smart Rules | Password Policies | Logout
Administrative Groups: Roles: Edit a Role
Update
Clear
Cancel
Copyright © 1997-2001 RSA Security Inc. All Rights Reserved.

Name * Commercial Banking Admin

Description Administrator to whom commercial banking tasks are delegated.

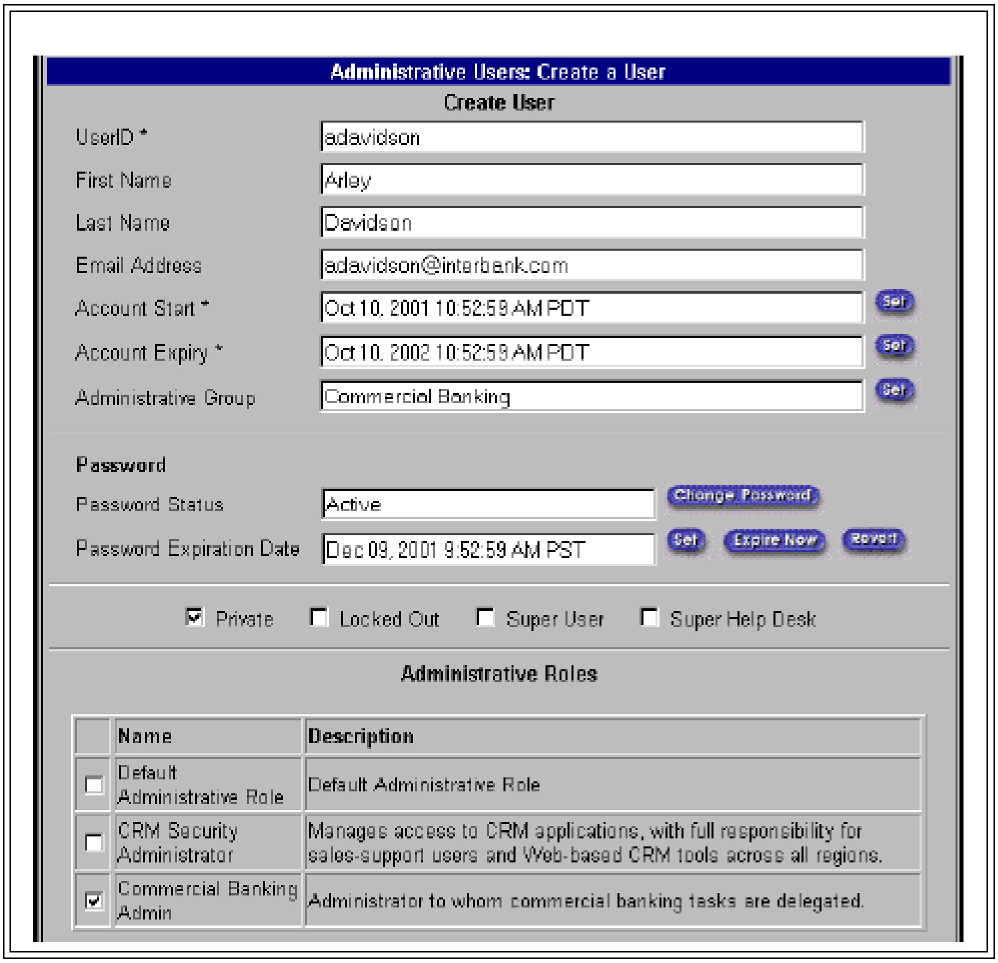
Privilege	Add	Modify	Delete
Administrative Roles	□	□	□
Administrative Users	☑	☑	☑
Users	☑	☑	☑
Groups	☑	☑	☑
Applications	☑	☑	☑
Web Servers	□	□	□
Passwords		☑	
Property Definitions	☑	☑	☑

Figure 2.3 Administrative Roles: Edit a Role Page

Administrative Users

After you have defined administrative groups and roles, you can associate them with each other in the process of creating of Administrative Users, the final step in delegating administrative privileges and responsibilities. As illustrated below, you must select an administrative group and role in the process of creating an administrative user; doing so makes this user an administrator of the group you selected, with the privileges defined in the role you selected.

In this example from the Entitlements Manager, the user "adavidson" is designated an administrator of the Commercial Banking administrative group, with that role's privileges:



Administrative Users: Create a User

Create User

UserID *	adavidson	
First Name	Arley	
Last Name	Davidson	
Email Address	adavidson@interbank.com	
Account Start *	Oct 10, 2001 10:52:59 AM PDT	Set
Account Expiry *	Oct 10, 2002 10:52:59 AM PDT	Set
Administrative Group	Commercial Banking	Set

Password

Password Status	Active	Change Password
Password Expiration Date	Dec 09, 2001 9:52:59 AM PST	Set Expire Now Revert

☑ Private ☐ Locked Out ☐ Super User ☐ Super Help Desk

Administrative Roles

	Name	Description
☐	Default Administrative Role	Default Administrative Role
☐	CRM Security Administrator	Manages access to CRM applications, with full responsibility for sales-support users and Web-based CRM tools across all regions.
☑	Commercial Banking Admin	Administrator to whom commercial banking tasks are delegated.

Figure 2.4 Administrative Users: Create a User Page

3 Managing Users and Groups

RSA ClearTrust controls access to your system resources based on user or group identity, or on customized user properties assigned to selected users. This chapter will describe concepts and procedures for managing RSA ClearTrust users and groups, including the creation and specification of user properties.

You should only use the Entitlements Manager to manage user accounts created within RSA ClearTrust. If your LDAP directory of users was created prior to installing RSA ClearTrust, RSA Security recommends that you continue to use your existing user administration tool to manage this data. Conversely, if you use the Entitlements Manager to access and modify your user and group data, make sure that you use *only* the Entitlements Manager. Manually modifying user and group data in other environments may break crucial relationships with administrative groups or other RSA ClearTrust data objects

Users

A *User* is an individual logon account in the RSA ClearTrust system. User accounts are usually associated with one person, although it is possible for more than one person to share an account (*guest*, for example). It is also possible for one person to have access to more than one account, and for user accounts to be associated with job descriptions instead of people.

Each user account must define a range of required information about the user. These fields must be assigned valid values, whether you manage your users in an LDAP browser, the Entitlements Manager, or another administrative tool. You can also define certain optional user information, as well as user properties. Proper management of all these types of information is discussed in this section.

Note: Administrative Users and system users are separate entities with important differences. Except where clearly indicated, the term "User" refers to typical system users. For more information on administrative users, see Chapter 2, "Managing Delegated Administration".

User Information

The section describes RSA ClearTrust user information. Some items are required, some are optional, and some will probably be used only rarely (for example, **Locked Out**).

Table 3.1 User Information descriptions

Field	Description	Notes
UserID	Login ID for the user	Required. Must **not** contain any of these prohibited characters: **`','`, `'+'`, `'"'`, `'\'`, `'<'`, `'>'`** or **`';'`**. The default LDAP attribute name for this field is **`cn`**.
First Name	User's first name	Required. The default LDAP attribute name for this field is **`givenname`**.
Last Name	User's last name	Required. The default LDAP attribute name for this field is **`sn`**.
Certificate DN	User's distinguished name for certificate authentication.	The DN of the client-side certificate for authentication must match this value.
Email Address	Email address for the user	Required. The default LDAP attribute name for this field is **`mail`**.
Account Start	Date and time the user account becomes active	Default is the host machine local system time when the user's information is first saved (thus, when the user is created). Time Zone is determined by local system time.
Account Expiry	Date and time the account will expire	The default is one year after Account Start.
Administrative Roles (for administrative users only)	An administrative user's collection of privileges to create, edit and delete items.	This table list is relevant only to administrative users, not system users.
Administrative Group	Administrative Group that owns the user account	The default is the Administrative Group of the Administrator that created the user.
Password	The user's unique password for Basic Authentication.	Required. This password is **not** used in systems configured for NT, SecurID or Certificate Authentication -- it is only used in Basic Authentication. The default LDAP attribute name for this field is **`userpassword`**.
Password status	Indicates whether the password is *active* or has *expired*	

Table 3.1 User Information descriptions

Field	Description	Notes
Password expiration date	The date the password will expire. (Use **Set** button to modify. It is also possible to force expiration of a password by checking the **Expire Now** check box. Among other uses, this can be used to force newly created users to change their passwords the first time they log on).	The default lifetime of a password is determined by the password policy associated with the Administrative Group that owns the user. Time Zone for the date is determined by local system time. **Note:** password expiration dates that you set with the Entitlements Manager apply only to passwords for RSA ClearTrust Basic Authentication. You cannot use the Entitlements Manager to expire NT or SecurID passwords.
Private	Identifies the user account as private; only the owning Administrative Group will be able to view this user and their associated account information.	The default is public. Private users can only be seen by an administrator in the same Administrative Group as the administrator who created the user.
Super User	Creates the user as a Super User. Super Users can perform any action on any object or resource.	This check box is only enabled if the administrator creating or modifying the user is also a Super User, and the user being created is an *Administrative User.* Assign with caution.
Super Help Desk	Allows an administrative user the ability to change or reset passwords across all existing administrative groups.	Only Super Users can enable this feature.
Locked Out	Immediately disables any permissions granted to the user, and blocks them from accessing protected resources.	Only Super Users can enable this feature.

Entering User Information

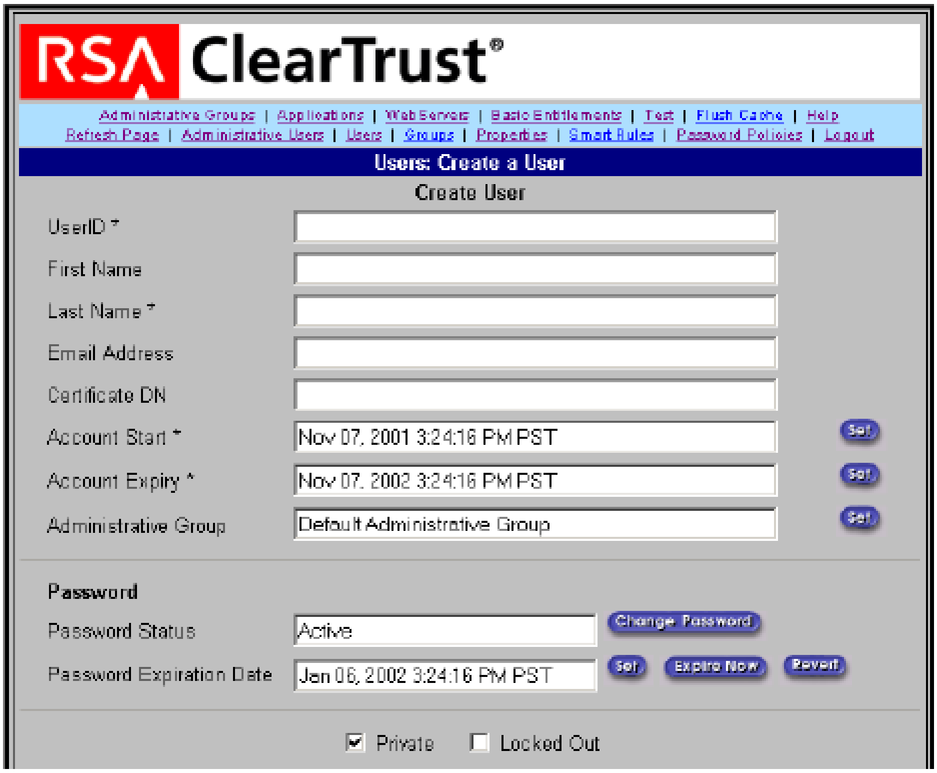
User information, both required and optional, can be specified when the user is being created, or can be modified later (except for *Login ID*, which cannot be modified).

Note: If you plan to control access to a resource by a user property, you should define it before creating any users, so that you can specify the value of that user property for each user as you create user accounts. For more information about controlling access via a user property, see Chapter 5, “Managing Security Policy”.

You can enter user information into the RSA ClearTrust system by these methods:

- **LDAP Administrative Tools.** If your organization stores user data in an LDAP database, you can use native administrative tools or an LDAP browser to enter and update users. Make sure you have covered all the relevant LDAP considerations detailed in Chapter 1, “Getting Started with RSA ClearTrust Administration”**.**
- **The Administrative API** When entering large numbers of users, you may prefer to automate the process with a custom program. Work with developers and refer to the *RSA ClearTrust Developer’s Guide*.
- **The Entitlements Manager** If you are using the RSA ClearTrust Entitlements Manager to create users, all information is entered in the Create a User page:

Important: When entering values in the UserID field, take care not to use any of these prohibited characters: `','`, `'+'`, `'"'`, `'\'`, `'<'`, `'>'` or `';'`**.** These characters are restricted due to limitations in acceptable LDAP DN syntax.


RSA ClearTrust®
Administrative Groups | Applications | Web Servers | Basic Entitlements | Test | Flush Cache | Help
Refresh Page | Administrative Users | Users | Groups | Properties | Smart Rules | Password Policies | Logout
Set
Change Password
Set
Expire Now
Revert

Users: Create a User

Create User

UserID *	
First Name	
Last Name *	
Email Address	
Certificate DN	
Account Start *	Nov 07, 2001 3:24:16 PM PST
Account Expiry *	Nov 07, 2002 3:24:16 PM PST
Administrative Group	Default Administrative Group

Password

Password Status	Active
Password Expiration Date	Jan 06, 2002 3:24:16 PM PST

☑ Private ☐ Locked Out

Figure 3.1 Create a User Page

Note: If you use the Entitlements Manager to modify existing users, you can configure the system to automatically update the LDAP object class for all optional user information fields. See details of the parameter `cleartrust.data.ldap.user.update_objectclass_on_modify` in the *Installation and Configuration Guide*.

User Properties

A user property is a custom data field that you define in order to store any type of information in user records. The chief purpose of user properties is to create evaluation criteria for SmartRules. See Chapter 5, “Managing Security Policy” for information about SmartRules.

User Properties can include any data that your organization stores or maintains for its users, including age, account status, department, date of hire, customer type, and so on. If you wanted to add a data field to every user indicating what region that user lives in, you might create a user property called `RegionCode`. Note, however, that user properties may not share conflicting names with the required user information fields listed on page 25.

In addition to their use in SmartRules, user properties that are marked as "exportable" can be read by any RSA ClearTrust Runtime API client program. In either case, user property values are used by RSA ClearTrust to make authorization decisions or user personalization decisions.

User Property Types

User properties must be one of five specific data types, which are described in the table below.

Table 3.2 User property types and examples

Type	Description	Example	Format/Allowed Values
BOOLEAN	True or False	Current depositor? External user? Customer?	1 = True 0 = False
STRING	A character string	The user's street address	Any string
INT	An integer	The user's zip code The user's level of security clearance	Minimum: -2147483648 Maximum: 2147483647
FLOAT	A floating point decimal value	The user's account balance The user's shoe size	Minimum: 1.40129846432481707e-45f Maximum: 3.40282346638528860e+38f
DATE	A date	The user's birthday The user's retirement date	yyyy-MM-dd HH:mm:ss.S

User Property Fields

User properties are defined in the Entitlements Manager by the following fields. Required information is noted.

Table 3.3 User Property Fields

Field	Description	Notes
Name	Name for the user property.	Must be identical to its corresponding LDAP attribute (though not case-sensitive). See "Mapping User Properties to LDAP Attributes" on page 31. **Required.**
Type	User property type: Boolean, String, Integer, Float, Date.	See preceding section for details and examples. **Required**.

Table 3.3 User Property Fields

Field	Description	Notes
Administrative Group	Administrative Group that owns the user property.	The default is the Administrative Group of the Administrator that created the property.
Description	Text description of the user property.	
Private	When checked, this field identifies the property as private; only the owning administrative group or an administrator with appropriate role status will be able to view this user property.	
Read Only	When checked, this field prevents users or API programs from modifying the values set for this property.	
Help Desk	When checked, this field allows Super Help Desk users to view the property, regardless of its designation as private.	
Exportable	When checked, this field allows the Runtime API to retrieve the values for this property for non-authorization uses, such as personalization.	Exportable properties may only be retrieved over secure connections; the runtime client must be configured for authenticated SSL.

Mapping User Properties to LDAP Attributes

If you are using the RSA ClearTrust LDAP Data Adapter, you must map user properties between the LDAP data store and RSA ClearTrust. You must make sure an identically named LDAP attribute exists in your LDAP schema before you can create a user property definition in the Entitlements Manager.

When you are building SmartRules based on user information already stored in your LDAP database, mapping is a simple matter of entering the exact attribute name when you create a user property definition in RSA ClearTrust. For new user properties, you must create matching entries in your LDAP database and in the Entitlements Manager.

For example, if you decided to create a policy limiting the access of users under age 18, you would first use your LDAP administration tools to create a user attribute titled *Age* in your LDAP schema. Then you would use the Entitlements Manager to create an integer-type user property named *Age* to identify the same value.

Important: User properties may not share conflicting names with the required user information fields listed on page 25.

Do not create any user properties with the following reserved names:

- `cn`
- `givenname`
- `sn`
- `mail`
- `userpassword`

Also avoid system-level attributes. If you are using the default user object class, the following attributes cannot be used as user properties:

- `dn`
- `uid`
- `o`
- `dc`

Failing to map to LDAP attributes in this manner will result in errors when you attempt to create user properties.

Creating and Setting User Properties

You can create user properties in the **Create a Property Definition** page of the Entitlements Manager, and then set values from the **Create a User** page. See the online help files for detailed, step-by step procedural guidance.

In cases where user properties must be applied to a large numbers of users, you may prefer to create properties in the Entitlements Manager, and then work with developers to set values automatically using the RSA ClearTrust Administrative API.

Note: If you are using the RSA ClearTrust LDAP Data Adapter, user properties must be mapped to LDAP attributes. You can create user properties only for attributes that have already been defined in your LDAP data store.

Setting and Updating User Property values

The values of user properties can be updated in these ways:

- Selecting the corresponding check boxes in the **Edit a User** or **Create a User** screen in the Entitlements Manager
- Using a custom application that changes the value of the user property via the RSA ClearTrust Administrative API. To use this method, work with your developers and refer to the *RSA ClearTrust Developer's Guide*.

Important: You cannot create multivalued user properties using the Entitlements Manager nor the Administrative API. If your need to create multivalued properties, you must use your existing, non-RSA ClearTrust user management tool (such as an LDAP browser) to save and edit users.

Administrative Users

With the LDAP Data Adapter, data for administrators and Super Users is stored in the RSA ClearTrust policy data store, separate from the data for system users. Therefore, even if an individual has a user account in the system, you must create another, separate account in the Entitlements Manager to designate that individual as an administrator.

Though similar in their information fields and options, the Entitlements Manager pages for managing **Administrative Users** are clearly distinguished from the pages for **Users**. See the online help files for detailed, step-by-step guidelines for creating and managing administrative users.

More information on the specific roles and privileges of administrative users is presented in Chapter 2, "Managing Delegated Administration".

Groups

Groups are groups of users and "children" or member groups. A group must have a unique name, and may also have an associated description. A user can belong to more than one group, and a group may contain multiple children groups. This "nesting" of groups allows for multiple layers of grouping:

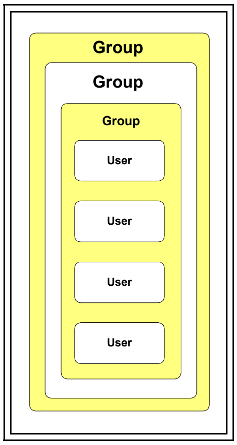


Figure 3.2 Nested Groups

There is no strict limit on the number of groups that may be nested in this manner. However, that there is a mounting performance penalty with each added level of group nesting.

Important: When entering group names in the Entitlements Manager **Name** field, take care not to use any of these prohibited characters: ',', '+', '"', '\', '<', '>' or ';'. These characters are restricted due to limitations in acceptable LDAP DN syntax.

How you organize your RSA ClearTrust users into groups will depend on the needs of your organization. A school, for example, might create a group called *Teachers*, with access to word processing software, database software, student records, assignments and test answers, a group called *Students*, with access only to software and assignments, and a group called *Administration*, with access to all of the above plus teacher’s salaries. Nested groups may be especially useful for defining logical subgroups, such as divisions within departments, or teams within divisions. In this example, the Sales, IT and Management teams are defined as children groups of the parent group

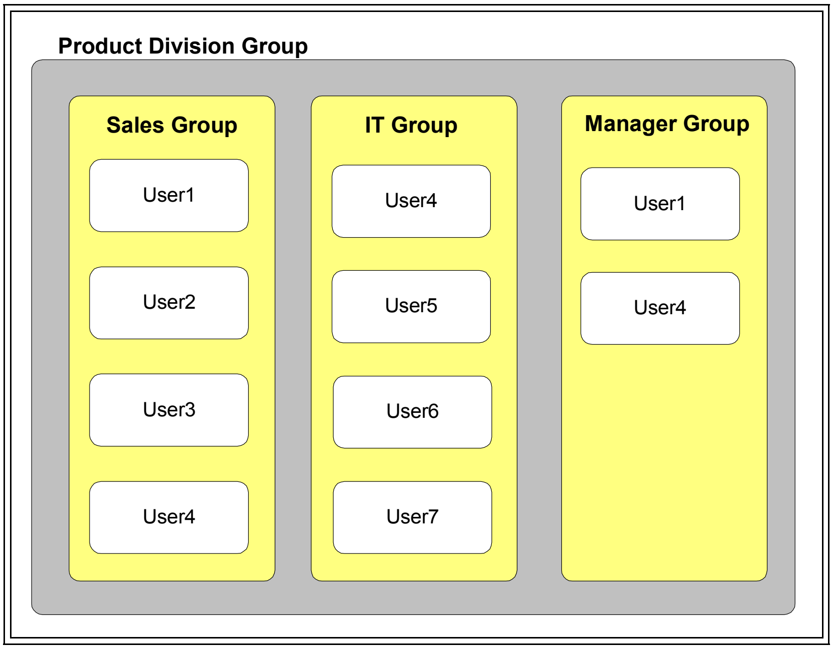


Figure 3.3 Example of Group Nesting

In an approach similar to traditional access control listing, you could create a basic entitlement allowing or denying the Sales group’s access to the IT group’s sensitive resources, and vice versa. Access to system resources can be controlled according to group membership as well as according to the individual user. To limit the access of Manger User1 to certain resources relevant only to Manager User2, you could create an individual basic entitlement that would take precedence over any entitlements allowing access by the Manager Group or its parent group. See Chapter 5, “Managing Security Policy” for more details on creating basic entitlements.

4 Managing Resources

Your primary concern as a RSA ClearTrust administrator is the protection of system resources. Before they can be protected, however, you must create each resource in RSA ClearTrust by entering its *URI* (Universal Resource Identifier) or other specific name. This chapter describes how to create and manage these resource types:

- "Web Servers" on page 37
- "Applications" on page 39
- "Application Resources" on page 39
- "Application Functions" on page 43

For information on creating security policy to protect these resources, see Chapter 5, "Managing Security Policy" and the Entitlements Manager online help files.

Authorization Mode: Active and Passive Modes

Before using the Entitlements Manager to create resources, you should be aware of whether your RSA ClearTrust environment is configured for active or passive mode. Access to resources not defined as part of an application nor protected by a security policy will depend on this configuration.

- **Active Mode**: If the system is configured in active mode, users are denied access to only those resources that have been explictly added to the RSA ClearTrust system. All other resources on the Web server will be available for general access.

  In Active Mode, you must take specific action to protect resources; that is, you must define them as application resources, URIs, or functions and create policy to deny access to them.

- **Passive Mode**: If the system is configured in passive mode, every resource is protected by default, whether or not it is part of an application. To provide access in this mode, you must expressly define each resource as part of an application, and grant access to users with an entitlement or SmartRule.

  In Passive Mode, all resources are protected by default; you must define resources and create policy to allow access.

From an Administrator's standpoint, passive mode involves more configuration work. You will have to explicitly create applications for anything you want your users to be able to access, or they will be denied access. On the other hand, if you intend to protect all of your resources, passive mode will ensure that this occurs.

Case for Active Authorization Mode

For example, let us assume that the Human Resources department of a large enterprise will implement RSA ClearTrust to protect its Web servers containing all HR information. Most of this information should be freely available to all authenticated clients; all employees should have free access to benefits information, the holidays schedule, company events, and so on.

However, the HR Web servers also contain a few pages of sensitive salary and personal information for each employee of the enterprise. This information should be available only to the RSA ClearTrust administrators (who are HR administrators) and the individual employee.

Since the administrators desire to leave 75% of the Web server resources unprotected, they choose to configure the system in active authorization mode. Then they perform the RSA ClearTrust administrative tasks necessary to actively protect the sensitive resources (adding the URIs to protect and creating user entitlements).

Case for Passive Authorization Mode

Many organizations implementing ClearTrust have high security needs, in which the majority of Web-based resources must be closely protected. For instance, an online stock trading site might maintain only a few directories of public material freely available to all users. All other information regarding user portfolios and accounts is highly sensitive.

In fact, even within the directories of public material, there is only a small portion the company wants to make available to any user -- prospective clients should be able to view the home page and demo pages, but all analysis of stocks and mutual fund information should be available only to members.

Since the administrators desire to leave only 5% of the Web-based resources unprotected, they choose to configure the system in passive authorization mode. Their needs still require a significant effort to allow appropriate access to the remaining 95% with appropriate SmartRules and entitlements, but because security is so crucial to their enterprise, this choice makes sense.

Configuring the Authorization Mode

Important: When the system is configured in passive authorization mode, any objects associated with logon, password change, or self-registration forms are automatically protected. This may prevent graphics or CGIs in such forms from displaying or functioning properly.

Active or passive mode is set in the `authserver.conf` file, by this parameter:

```
cleartrust.aserver.authorization_mode
```

By default, the parameter is set to active. See the *RSA ClearTrust Configuration and Implementation Guide* for information on changing the setting.

Web Servers

Before you can protect an *Application, Application Resource or Application Function*, you must identify to the system the *Web Server* (or Web servers) on which it resides. These Web servers must have the RSA ClearTrust Web Server Agent installed and running, as noted in Chapter 1, "Getting Started with RSA ClearTrust Administration".

Web servers in the RSA ClearTrust system are identified by name, not domain name. The name you choose has important dependencies in the setup of the RSA ClearTrust Web Server Agents. When you create a Web server name in the Entitlements Manager, you must define basic information including the port number and administrative group:

Table 4.1 Web Server information

Server Information	**Description**	**Comments**
Name	The name by which the Web server is known to the RSA ClearTrust system	Each Web server in the system must have a unique name. The name must match the value of the `cleartrust.plugin.Web_server_name` parameter in the Web Server Agent's `webagent.conf` file. (The default name in that file is "WebServer").
Hostname	This must match the actual, fully-qualified name of the Web server	For example, *hostname.domain.com*, or the Web server's IP address.
Port number	This is the port address on which the Web server advertises its HTTP services.	The default is port 80, but this can be changed if the Web server is configured with a different port number.
Owner	The administrative group that owns this Web server.	By default, this is the administrative group of the Administrator account that created the Web server. A Super User can transfer ownership to another administrative group.

Server Trees

Access to a Web Server is controlled by the administrative group that owns the server. Usually, this is the administrative group that created the Web Server. To facilitate delegated administration, you can divide the directories on a Web Server into *Server Trees*. Server trees, and therefore the resources in the directories which they contain, can be owned by different administrative groups. Administrators from these administrative groups can then create the applications made up of these resources, and can control access to these applications.

See the online help files for detailed instructions on dividing your Web Servers into server trees.

Mirror Sites

If an organization has several load-balanced Web servers that are acting as mirror sites (they all serve the same content, and have the same directory structure and files), those servers can all share the same name in the Entitlements Manager. In addition to having the same name in the Entitlements Manager, the servers must all have the same name specified in their Web server Agent's configuration file (`webagent.conf`). URIs will only need to be assigned once for mirrored resources.

Note: If servers are configured this way it will not be possible to track activity on each individual Web server separately in the RSA ClearTrust activity logs. To track activity separately for each server, they must be named differently and URIs assigned on each one.

Applications

An *Application* is a collection of resources in the RSA ClearTrust system that have been logically grouped together and named. The grouping of resources into applications allows you to apply security policy to logically related resources of different types. Resources included in an application can include Web Servers, URIs for Web pages, CGI files, directories, GIF or JPG files, and application Functions:

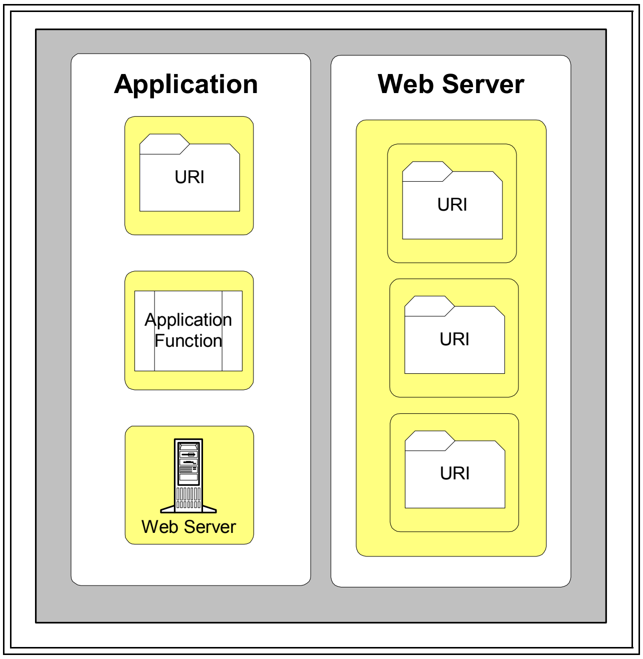


Figure 4.1 Example of an Application and its Included Resources

Application Resources

You can create security policy for specific resources at the application resource level. Application resources are inclusive - on a system where the `/marketing` directory contains the `/applications` directory, for example, the URI `/marketing/*` contains everything in the
`/applications/` directory.

Administrators can only manage application resources in their own administrative group. Trying to manage or create policy for another administrative group's URI will generate an error message.

Defining URIs as Application Resources

When entering URIs as application resources in the Entitlements Manager, take care with slash characters, both at the beginning and end of the URI. The URI must begin with a slash, and must end with either a fully specified URI or a /*. In all cases of protecting a directory, `/*` syntax is required. These are the basic rules for URI syntax:

- Begin all URI definitions with a "/"
- When protecting a directory, end the URI definition with "/*"

Important: When protecting *entire* Web servers, use "`/*`" with caution. Using "`/*`" to protect the entire Web server will block access to graphics and objects associated with logon or self-registration forms.

Failing to observe these URI syntax rules may result in security holes. If an administrator wishes to protect all resources under the `Finance_Server/Projections` directory and defines the URI in this way

```
#(error example -- do not define directories using this syntax)
/Finance_Server/Projections
```

then unauthorized users will still be able to gain access to this URI with a trailing slash:.

```
/Finance_Server/Projections/
```

The directory is securely protected only by defining the URI with a /* at the end:

```
/Finance_Server/Projections/*
```

Limitations on Application Resources

Resources under an application are subject to the following limitations:

- A specific URI can be defined as a member of only one application at a time. The same URI cannot be added to more than one application simultaneously.
- Since a URI can be a directory, URIs in different applications can overlap.
- A single application can contain URIs on more than one Web server, as long as the administrative group that owns the application also owns the all the Web servers or server trees and all the included URIs.

URIs in Overlapping Applications

A URI can only be part of one application at a time, but because URIs can overlap it is possible for resources to belong to more than one application. When determining if a user has access to a resource, RSA ClearTrust will apply the most specific URI available for that resource.

Figure 4.2 on page 41 includes the following example applications:

- The *Profit Projections* application, which contains the URI `/projections/profits/*`
- The *Salaries* application, which contains the URI `/salaries/*`
- The *Finance Server* application, which contains `/*` (all files on that Web server)

Both the *Profit Projections* and *Salaries* applications overlap the boundaries of the *Finance Server* application. When a user tries to access a resource that matches more than one application, RSA ClearTrust will enforce the security policy controlling access to the most specific URI:

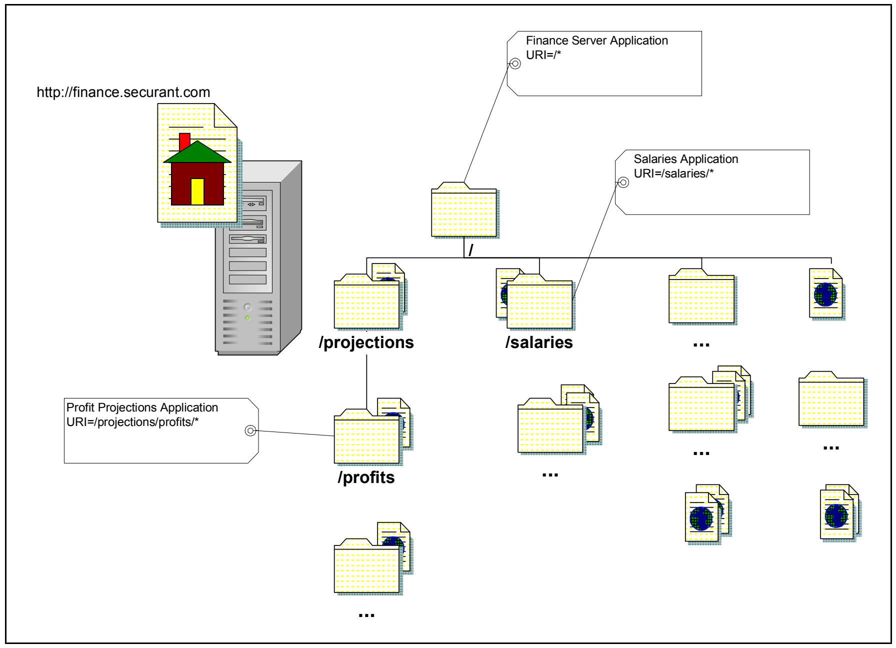

Finance Server Application
URI=/*
http://finance.securant.com
Salaries Application
URI=/salaries/*
/
/projections
/salaries
...
Profit Projections Application
URI=/projections/profits/*
/profits
...
...
...
...

Figure 4.2 Example of a Web Server with Multiple Application URIs

Table 4.2 describes how some access attempts will be handled by the RSA ClearTrust system.

Table 4.2 Access to overlapping URIs

Resource	Most Specific URI	Application Determining Access
`/projections/profits/`	`/projections/profits/*`	Profit Projections
`/projections/spending/`	`/*`	Finance Server
`/salaries/exec.html`	`/salaries/*`	Salaries
`/salaries/slack/bob.gif`	`/salaries/*`	Salaries
`/salaam.html`	`/*`	Finance Server

Dynamic Content

Within RSA ClearTrust, server-side programs and scripts are treated like any other web content, definable as an application resource. CGIs, Active Server Pages (ASP), or similar files can be defined by a URI. For example, the following are all valid RSA ClearTrust application resources:

```
/projections/spending/today.cgi
/projections/sales/db_update.pl
/receivables/aging_report.asp
```

Dynamically created URIs, however -- pages using a CGI or any other server-side program that appends data to the URL at submission time -- should be defined using a wildcard. For example, the URI to designate a Web page such as:

```
http://yourserver.domain.com/budget/today.cgi?day=tuesday
```

would be defined as:

```
/budget/today.cgi/*
```

It is also possible to define automatically generated pages and images that are all in one directory by designating the directory as a URI using a wildcard. For example, if all of the automatically generated content will be in a directory called `/results/daily/`, everything in that directory, including the automatically generated content, can be defined using the URI `/results/daily/*`.

Application Functions

In addition to defining resources at the application and application resource level, you can define specific *Application Functions* to protect. Application functions are useful for access checking in situations that do not involve URI requests.

To take advantage of application function-level resources, you must work closely with your software developers. Your role as an administrator is to define the application function by naming it, and then apply policy to it; the developer's role is to design the appropriate function call out to the Authorization Server, which processes your security policy to determine whether to allow or deny access to the function.

Application Functions in Context

Access control at the application function level is useful in special situations requiring granular control of actions or functions, especially in non-Web applications. For example, if you want to create policy for a non-Web Java application that has a sensitive method that you do not wish to make available to all users, you can protect it with an application function.

For a method called `updateBalance()` you could create an application function entry in the Entitlements Manager with this specific name, description and Policy Evaluation Order:

- **Name**: updateBalance
- **Description**: Allows users with the appropriate access to update account balances.
- **Policy Evaluation Order**: Deny-Allow

  This setting determines access priority for conflicting entitlements and SmartRules. See the next section of this chapter, as well as Chapter 5, "Managing Security Policy" for more information regarding policy evaluation order.

Note: Software Developers must use the exact value for the application function **Name** when making the appropriate function call to the Authorization Server from the application function. See the RSA ClearTrust *Developer's Guide* for more information.

Assuming that the updateBalance function has been integrated with RSA ClearTrust through the API, you may apply SmartRules or basic entitlements to this application-level resource using the defined name.

Setting Policy Evaluation Order (Allow/Deny - Deny/Allow)

For each resource you create in RSA ClearTrust you can set a policy evaluation order -- either **Allow/Deny** (Allow *before* Deny) or **Deny/Allow** (Deny *before* Allow). You may switch this toggle control, which is set to Allow/Deny by default, any time that you create or edit applications, application resources, or application functions.

This setting becomes important when the system must make access control decisions. If one rule allows access and another rule denies access to a given resource (and the rules are of equivalent specificity), then the system checks your Allow/Deny setting to decide which rule takes priority. See Chapter 5, “Managing Security Policy” for more detail on security rules.

Therefore, when setting policy evaluation order for a given resource, you must consider the sensitivity of the resource and the manner in which you would prefer to resolve conflicting access rules to the resource. If you would like to resolve any potential conflict by denying access, you would set each resource to Deny/Allow; to allow access in case of conflict, you would choose Allow/Deny.

Note: The policy evaluation order affects system access control decisions in cases of conflicting entitlements or multiple SmartRules on a given resource. It is not a global or ubiquitious setting governing all attempts to gain access to the resource -- rather, it is important only in cases of conflicting security rules as described in “Resolving Multiple or Contradictory Entitlements” on page 50.

In addition to using the Entitlements Manager Create/Edit pages for each resource type, you can use the Administrative API to set policy evaluation order. See the *RSA ClearTrust Developer’s Guide* for more information.

5 Managing Security Policy

This chapter describes the two types of RSA ClearTrust security policy: *Basic Entitlements* and *SmartRules*. Basic entitlements control access to system resources based on the identity of a given user or by membership in a group. SmartRules control access based on the value of specific user properties.

Creating security policy with basic entitlements and SmartRules is discussed below in the following sections:

- “SmartRules”
- “Basic Entitlements”

SmartRules and basic entitlements can be applied to your system resources at three distinct levels: application functions, application URLs and applications. See Chapter 4, “Managing Resources” for details on these resource types and how to add them to the system.

SmartRules

SmartRules are dynamic access control policies that protect your system resources based on User Properties. SmartRules allow you to apply security policies immediately and pervasively throughout the system.

For example, if you implement a user property called *Employment Status* (a Boolean value), then when employees are terminated, the SmartRule based on Employment Status can instantly revoke all access privileges as soon as that one property is changed.

Important: SmartRules decide a user’s access to a specified application ***only*** if no relevant basic entitlement exists at any level (User or Group). Basic entitlements ***always*** take precedence over SmartRules.

Forms of SmartRules

A SmartRule compares the value of a user's user property to a specified value (a *comparison criterion)* according to a *comparison operator*. SmartRule operators are described in the following table:

Table 5.1 Smart Rule Operators

User Property	Operator
Date	Before, After
Boolean	Is Not, Is
String	Does Not Contain, Ends With, Equals, Starts With, Contains (you can also apply numeric operators to Strings).
Integer, Float	>=, <, =, >, <=, !=

In this *Account Balance* SmartRule, for example, the comparison operator is the integer > and the comparison criterion is the user property **`Account Balance`**:

ALLOW if Account Balance > $500.00

Types of SmartRules

SmartRules can be created in one of three types — `ALLOW`, `DENY`, or `REQUIRE`. These types are explained further in Table 5.2.

Table 5.2 SmartRule types

Type	Processing Order (default)	Logic for Multiple Rules of This Type	Usage Notes
Deny	First	OR	If the value of the user property meets the condition, the user is denied access immediately and no further rules are evaluated. When access is denied, the user is presented with HTTP 404 "File not Found" form.
Allow	Second	OR	If the user property meets the condition, the user can access the application function immediately; no further rules are evaluated
Require	Last	AND	If the value of the user property meets the condition, RSA ClearTrust evaluates the next Require rules. If all Require rules are fulfilled and Allow and Deny rules allow access, the user is granted access.

The three kinds of SmartRules can be combined in various ways to implement business rules and control access to an application.

Important: If a User has a user property of value "N/A" (not yet entered), a Deny rule based on that user property will consider the condition not met (will not Deny) if the Authorization Server is set to *active* mode, but will consider the condition met (will Deny) if the Authorization Server is set to *passive* mode. For more information see "Authorization Mode: Active and Passive Modes" on page 35.

Combining SmartRules

An individual SmartRule applies a single condition to one particular user property. To create more complex conditions for access, you can define and combine multiple SmartRules. For example, say a URI in a particular application has these two SmartRules associated with it:

- ALLOW if State = CA
- DENY if Age < 21

and user *Joe* has values set for each of these properties, as follows:

- State = CA
- Age = 18

If the default processing order (**Deny ▸ Allow**) is in effect, then the SmartRules are evaluated as follows:

1. DENY if Age < 21
2. ALLOW if State = CA

Joe will be denied access to the specified URI because of the value in his user property for Age. The second SmartRule is never evaluated.

Order of Processing for SmartRules

Deny ▸ Allow is the default processing order of SmartRules. Administrators can reverse the order—so that Allow Rules are processed before Deny Rules. In the previous example, access was ultimately denied. Changing the processing priority to **Allow ▸ Deny** in the example changes the processing order as follows:

1. ALLOW if State = CA
2. DENY if Age < 21

Joe would be granted access to the URI because he meets the ALLOW condition. The DENY condition is never evaluated.

SmartRules on Multivalued User Properties

SmartRules can be based on user properties with more than one value. In such cases, the relevant user properties will be mapped to multivalued LDAP attributes, or database entries that hold more than one value.

Note: Though you can use the Entitlements Manager to create SmartRules based on multivalued user properties, you cannot create the multivalued properties themselves. If your users contain multivalued properties, you must use your existing, non-RSA ClearTrust user management tool (such as an LDAP browser) to save and edit users.

For instance, each user may have an attribute entry for "Phone Number" that contains multiple values for home phone, mobile phone, work phone, and so on. If your security needs require that you give access to users when any one of these numbers contains a certain area code prefix, you could create a SmartRule based on "Phone Number" with the operator "contains":

- ALLOW if Phone Number contains 415

When processing this SmartRule, the Authorization Server will consider all values for phone number, and allow access if any one of them contains 415.

Testing SmartRules

Before implementing a security policy based on SmartRules, it is a good idea to test the rules to see if they are operating as intended. The testing tool in the Entitlements Manager allows administrators to simulate a specific user attempting to access a resource in order to test SmartRules. See "Testing Security Policy" on page 53.

Further Examples of SmartRules

For further clarification of the use of SmartRules in business applications, two examples are provided below.

Example One

In this example, the organization using the RSA ClearTrust system is an insurance company with customers throughout the south and northwest United States. At the beginning of its fiscal year, the company decides to make a special offer available to residents of California, Texas, and Oregon via its web site. To accomplish this, the insurance company creates three SmartRules to control access to the area of the web server containing the special offer (having already created a user property called *State*, normally used as part of the customer's mailing address):

- ALLOW if State = CA
- ALLOW if State = TX

- ALLOW if State = OR

This simple setup accomplishes the desired goal. Residents of California, Texas, and Oregon can access the special offer, and everyone else is denied access.

A month later, the insurance company decides it must limit the offer to Users with good credit ratings. Since there is already another user property called *Bad-Credit*, (a Boolean which is set to *yes* if the account has been flagged for non-payment), adding another SmartRule is straightforward:

- DENY if Bad-Credit = true
- ALLOW if State = CA
- ALLOW if State = TX
- ALLOW if State = OR

Since the priority for this Application Function is set to its default value, DENY - ALLOW, Deny rules will be evaluated first. Now only users with good credit from California, Texas, or Oregon can access the insurance company's special offer web page.

Example Two

It is also possible to combine SmartRules of the `REQUIRE` type. In this example, a company wants to limit access to an area of its web site to retail customers that have account balances over $100. In this case, both parts of the condition must be met or the user will be denied access. The RSA ClearTrust Administrator creates two SmartRules:

- REQUIRE Account Balance> 100
- REQUIRE Account Type = Retail

At runtime, only Retail users with account balances in excess of $100 will be allowed access to the site.

Basic Entitlements

A Basic Entitlement explicitly *allows* or *denies* access to a specific resource (typically an application). Basic entitlements can be specified at the user or group level.

- Basic entitlements assigned at the user level affect only that user.
- Basic entitlements assigned at the group level affect all of the users contained in that group and its member groups.

If a group is granted access rights to a resource through a basic entitlement, all the users in the group have access rights to the resource unless they are specifically excluded at the user level. When users are denied access to a resource, the system returns an HTTP 404 "File not Found" form.

Resolving Multiple or Contradictory Entitlements

Since entitlements can exist at several levels, users can belong to multiple groups, and groups can belong to other groups as member groups, it is possible to create contradictory basic entitlements. If basic entitlements are in conflict, access to resources is resolved according to the specificity of the resource and entity as well as the policy evaluation order of the resource. These are the criteria for resolving conflicts, in order of priority:

- "Specificity of the Resource": application functions/resources take priority over applications.
- "Specificity of the Entity": users take priority over groups, and nested groups take priority over parent groups.
- "Policy Evaluation Order (Allow/Deny - Deny/Allow)": depending on the order set for the resource, *Allow* policies or *Deny* policies will take priority.

The fundamental principle is "most specific match wins", and when specificity is equal, the policy evaluation order for the resource determines the resolution of conflicting entitlements.

Specificity of the Resource

RSA ClearTrust applies a processing logic of "most specific match wins" to resources, where specific-to-general is defined this way:

Application Function/Application Resource < Application

This means that security policies based on application resources/functions take priority over policies based on applications that include them.

Additionally, the system evaluates the specificity of the URI defining an application resource. If a user has entitlements at various levels in a Web server directory, the most specific entitlement determines user access. In this example, Entitlement I grants group access to the Profits directory while Entitlement II denies access to the more specific Executive directory:

- Entitlement I: Allow the group "Junior Analysts" access to `Finance_Server/Projections/Profits/*`
- Entitlement II: Deny the group "Junior Analysts" access to `Finance_Server/Projections/Profits/Executive/*`

These rules allow the group full access to all resources under `Profits`, except the more specifically defined `Executive` resources such as, for instance, the URI `Projections/Profits/Executive/Q2_Exec_Summary.html`. See "URIs in Overlapping Applications" on page 41 for more information regarding resource specificity.

Specificity of the Entity

RSA ClearTrust applies a processing logic of "most specific match wins" to entities, where specific-to-general is defined this way:

User < Group < "Parent" Group < "Parent" Group

This means that user entitlements take priority over group entitlements, and "child" group entitlements take priority over entitlements on groups that include them. In these examples, users 1 and 2 belong to the Gold group and, by nested group membership, in the Silver and Bronze groups. Due to the principle of entity specificity, users 1 and 2 will have access to `index.html` as described in these three separate case examples:

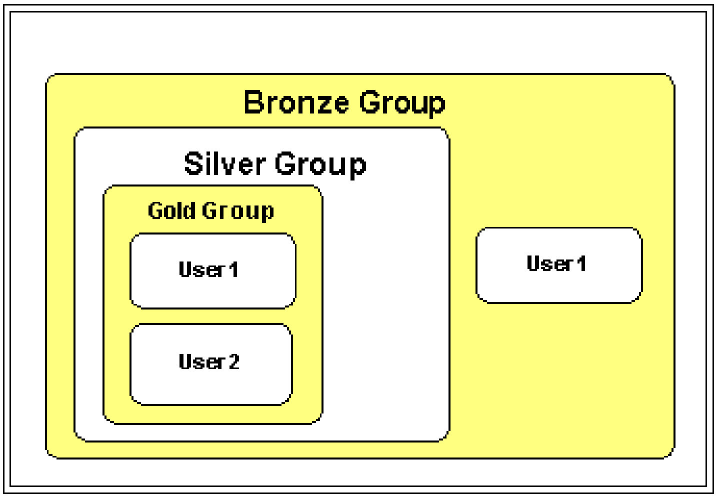


Figure 5.1 Entitlements on Nested Groups

Case I: User-level specificity wins over group-level specificity.

- Entitlement I: Deny User 1 access to index.html
- Entitlement II: Allow Gold Group access to index.html

User 1 is denied access based on specific user entitlement (despite group allow).

User 2 and all other members of Gold are allowed access based on group allow rule.

Case II: Child group-level specificity wins over parent group-level specificity.

- Entitlement I: Deny Silver Group access to index.html
- Entitlement II: Allow Gold Group access to index.html

Both users are allowed access because Gold Group is more specific than its parent Silver -- and the user has no specific deny rule at the user level (please consider the user deny rule in case I as entirely separate with no effect on case II).

Case III: Conflicting entitlements -- must be resolved by policy evaluation order

- Entitlement I: Deny Bronze Group access to index.html
- Entitlement II: Allow Gold Group access to index.html

User 2 is allowed access because Gold group is more specific than Bronze group.

User 1, however, is a member directly of both groups, and the groups are thus of equal "specificity" to the user. In this case, the system must consider the **Policy Evaluation Order** specified for the application resource index.html (as described in the next section).

Policy Evaluation Order (Allow/Deny - Deny/Allow)

For each resource you protect in RSA ClearTrust you can set a policy evaluation order -- either **Allow/Deny** (Allow *before* Deny) or **Deny/Allow** (Deny *before* Allow). You may switch this toggle control, which is set to Allow/Deny by default, any time that you create or edit applications, application resources, or application functions.

This setting becomes important when the system must make access control decisions for SmartRule Processing or for conflicting basic entitlements. If one rule allows access and another rule denies access to a given resource (and the rules are of equivalent specificity), then the system checks your Allow/Deny setting to decide which rule takes priority.

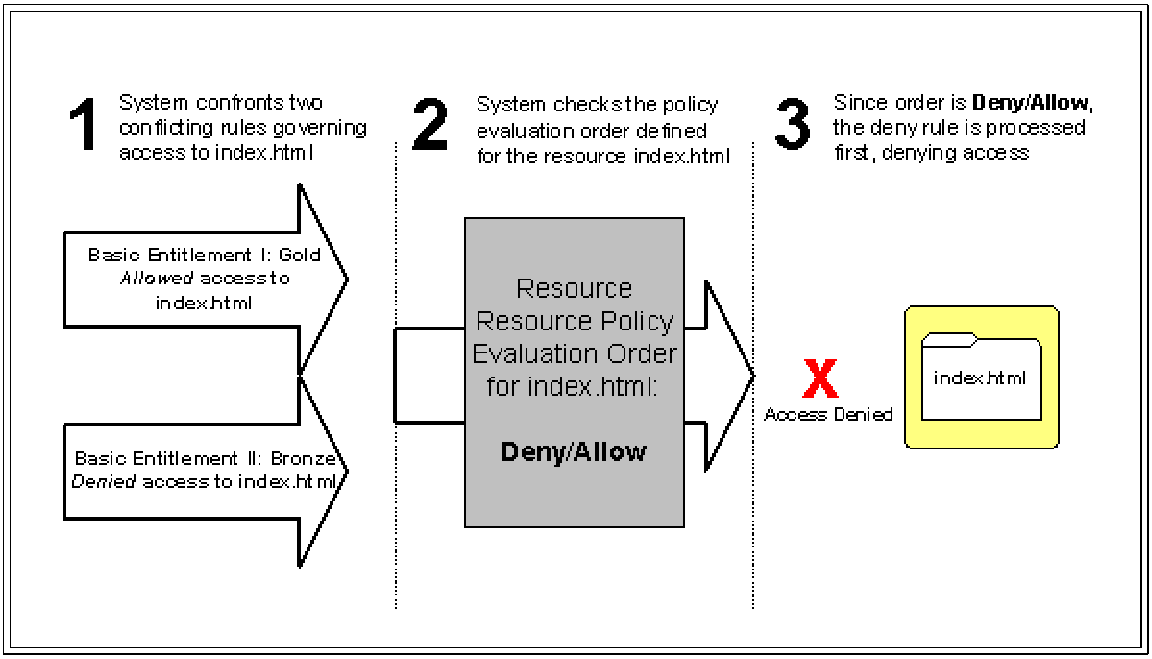

1
System confronts two conflicting rules governing access to index.html
Basic Entitlement I: Gold *Allowed* access to index.html
Basic Entitlement II: Bronze *Denied* access to index.html
2
System checks the policy evaluation order defined for the resource index.html
Resource Resource Policy Evaluation Order for index.html:
Deny/Allow
3
Since order is **Deny/Allow**, the deny rule is processed first, denying access
X
Access Denied
index.html

Figure 5.2 Policy Evaluation Order in Resolving Conflicting Entitlements

In this example, the two equivalent entitlements to index.html are resolved by checking the policy evaluation order specified by the administrator when the URI was entered in the system. See "Application Resources" on page 39 for more information about entering URIs as application resources.

Policy Enforcement Priority

Though the scenarios for policy processing conflicting basic entitlements or multiple SmartRules may become highly complex, the policy enforcement results can be accurately summarized in their priority ordering. For applications, application resources and functions, the ordering of policy precedence is as follows:

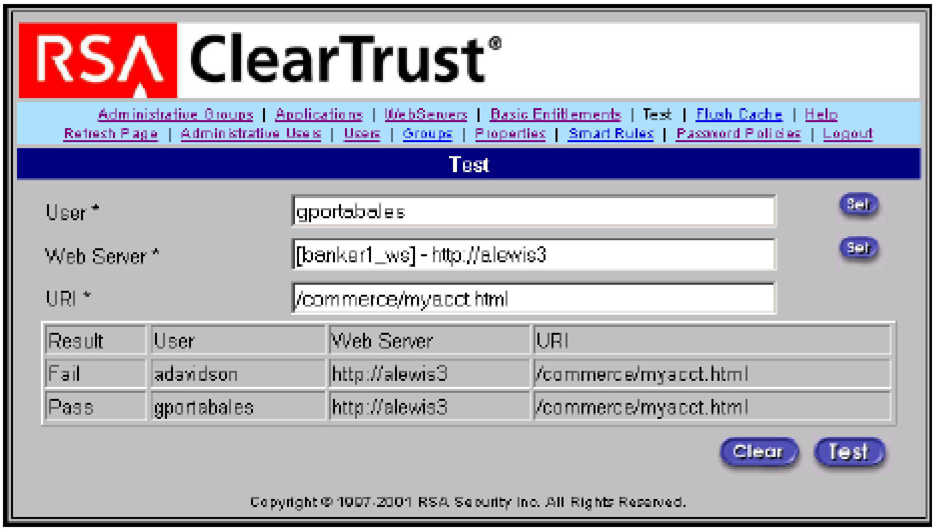
1. User Entitlements on the URL/application function
2. Group Entitlements on the URL/application function
3. Smart Rules on the URL/application function
4. User Entitlements on the application that includes the URL or function
5. Group Entitlements on the application that includes the URL or function
6. Smart Rules on the.application that includes the URL or function

Testing Security Policy

Before you apply security policies, it is a good idea to test them with the Entitlements Manager **Test** page. By setting up a test on this page, you can simulate a user trying to access a resource and determine whether your policy denies or allows access as desired.

Note: The testing tool tests the integrity of your security policy, but not the system setup. RSA ClearTrust servers and agents must be properly installed for your security policies to protect resources at runtime.

In this example, the user "adavidson" is denied access to the "My Account" page on the Web server "banker1_ws". However, user "gportabales" belongs to a group that has been granted access to this page by a basic entitlement, so his test passes:


RSA ClearTrust®
Administrative Groups | Applications | WebServers | Basic Entitlements | Test | Flush Cache | Help
Refresh Page | Administrative Users | Users | Groups | Properties | Smart Rules | Password Policies | Logout
Test
Sel
Sel
Clear
Test

User * gportabales

Web Server * [banker1_ws] - http://alewis3

URI * /commerce/myacct.html

Result	User	Web Server	URI
Fail	adavidson	http://alewis3	/commerce/myacct.html
Pass	gportabales	http://alewis3	/commerce/myacct.html

Copyright © 1997-2001 RSA Security Inc. All Rights Reserved.

Figure 5.3 Test Page

A Glossary of Terms

A

active mode

If the Authorization Server is configured in *active mode*, all Resources not explicitly protected by the RSA ClearTrust system are accessible. This means that users can access any resource on a protected Web server, unless they are explicitly denied access in the Entitlements Manager. Active mode is the default setting. See also, passive mode.

Adapter

See Data Adapter.

Administrative API

This is the RSA ClearTrust application programming interface (API) that developers can use to manipulate user, administrator, resource and policy information in the RSA ClearTrust data stores. The Administrative API uses the Entitlements Server to write to the user, policy and administrator data stores on your configured LDAP directory server or relational database management servers (RDBMS).

Administrative Group

In the RSA ClearTrust data model, an Administrative Group is a collection of objects (Users, password policyGroups, Web servers, Server Trees, Applications, and User Properties) that are *owned* by the administrators in that group. This allows you to delegate administration responsibilities by creating these logical Administration Groups and assigning ownership of these objects to the group's administrators. Each administrator in RSA ClearTrust belongs to an Administrative Group, and is also assigned an Administrative Role, which determines the actions that he or she can perform on the objects in that group.

Administrative Groups are also referred to as *Virtual Business Units* (VBUs).

Administrative Role

In the RSA ClearTrust data model, an Administrative Role is a defined set of *privileges* that administrators can perform. Administrative Roles are given to administrators within an Administrative Group, and usually are named to reflect the real-world role of the administrator (for example, Help Desk or HR). The Administrative Role controls what actions the administrator can perform on the objects (Users, password policyGroups, Web servers, Server Trees, Applications, and User Properties) owned by their Administrative Group.

Privileges include such actions as creating users, changing user properties, changing passwords, restricting or allowing access to resources, and so on.

Administrative User

In the RSA ClearTrust data model, an Administrative User is a user that has been assigned one or more Administrative Roles, and is assigned to an Administrative Group.

In RSA ClearTrust 4.7, Administrator accounts are kept separate from the regular user data store. This means that you must create a new, separate Administrator account in RSA ClearTrust for each user that you want to assign an Administrative Role. This allows you to maintain your existing user data stores as read-only, only using this data to check authentication credentials and access control at runtime. You cannot designate users in your existing user data stores as RSA ClearTrust Administrators. You can, however, point to existing Administrator data in your native user data to map to RSA ClearTrust Administrative Roles. See the *RSA ClearTrust Installation and Configuration Guide* for more information.

Also refered to as an *Administrator*.

Agent

A software module that adds a specific feature or service to a larger system. The RSA ClearTrust Agents augment the native security features of a Web server or application server to protect resources served from those servers. The RSA ClearTrust Agents interface with the RSA ClearTrust Authorization Server to perform authentication and access control. In order to protect a Web server or application server in RSA ClearTrust using the Entitlements Manager, you must have the appropriate Agent installed on that server.

See also Application Server Agent and Web Server Agent.

API

See application programming interface.

Application

In the RSA ClearTrust data model, an Application or *Resource Group* is a collection of resources that are logically grouped together. An Application can consist of URIs and Application Functions. For example, you can create an *Application* called *HR*, which contains various resources such as benefit plans, the employee handbook, and so on. These resources may reside in different directories or on different Web servers, but are logically grouped together in RSA ClearTrust. Access to Applications can be controlled via SmartRules and Basic Entitlements.

Application Function

In the RSA ClearTrust data model, an Application Function is a customized set of rules that specify how other resources can be accessed and manipulated. Application Functions are useful for access checking in situations that do not involve URI requests. For example, if you want to create policy for a non-Web Java application that has a sensitive method that you do not wish to make available to all users, you can protect it with an Application Function. Application Functions are developed using the RSA ClearTrust Administrative API, and then named and specified in the Entitlements Manager. Access to Application Functions can be controlled via SmartRules and Basic Entitlements.

application programming interface

A set of routines, protocols, and tools for building software applications that will interface with the RSA ClearTrust Server components. RSA ClearTrust provides an Administrative API, a Runtime API, and a Web Server Agent Extention (WAX) API.

application server

A program run on a mid-sized machine that handles all application operations between browser-based computers and a company's back-end business applications or databases. Because many databases cannot interpret commands written in HTML, the application server works as a translator, allowing, for example, a customer with a browser to search an online retailer's database for pricing information.

To run RSA ClearTrust's Web-based administration tool, the Entitlements Manager, you will need an application server.

See also, Application Server Agent.

Application Server Agent

This RSA ClearTrust component extends authentication, access control and single sign-on functionality to application servers. The Application Server Agents utilize the RSA ClearTrust Runtime API to control the protection of all objects served up by the application servers. JSPs, Servlets, and EJBs can be protected independently or in abstract groupings.

authentication

The process of identifying an individual, usually based on a username and password. In security systems, authentication is distinct from authorization, which is the process of giving individuals access to system objects based on their identity. Authentication merely ensures that the individual is who he or she claims to be, but says nothing about the access rights of the individual.

RSA ClearTrust supports multiple types of authentication, including RSA SecurID, X.509 certificates (such as RSA Keon), NT authentication and username/password. In addition, the RSA ClearTrust Web Server Agent can be extended using the WAX API to use other authentication schemes, including Kerberos.

authorization

The process of granting or denying access to a network resource. Most computer security systems are based on a two-step process. The first stage is authentication, which ensures that a user is who he or she claims to be. The second stage is authorization, which allows the user access to various resources based on the user's identity.

Authorization Server

A runtime component of the RSA ClearTrust system. The Authorization Server takes requests from the RSA ClearTrust Web Server Agent or Application Server Agent, verifies these requests against the data in your RSA ClearTrust Data Stores, and performs authentication of the user and then authorization of the request.

B

base DN

When searching for entries in an LDAP directory hierarchy, the Base DN determines the node where to begin the search. For example, if you wanted to search for all entries in the organization "RSA", you would specify the Base DN as "o=RSA".

Basic Entitlement

A Basic Entitlement allows you to either grant or deny access to a specific resource for a specific user or group of users. A Basic Entitlement can be applied to an Application, Application Function or to an individual URI.

browser

An application program that provides a way to look at and interact with information on the World Wide Web. Technically, a Web browser is a client program that uses the Hypertext Transfer Protocol (HTTP) to make requests of Web servers throughout the Internet on behalf of the browser user.

C

cache

A place to store data temporarily, especially data or files that are frequently requested by the user or system. Caching can be implemented for Internet content by distributing it to multiple servers that are periodically refreshed.

In RSA ClearTrust, the Authorization Server caches user entitlement data as well as resource and security policy data. Runtime requests are then verified against the cache, instead of against the data stores. Caching improves the response time.

When you update records in the Entitlements Manager, those selected records will be refreshed in the cache automatically. You do not need to use the **Flush Cache** command. This command clears out the entire cache on all Authorization Servers.

ClearTrust

See RSA ClearTrust.

cn

A commonly used LDAP attribute name that stands for *common name*.

D

Data Abstraction Layer (DAL)

This is an architectural component of the RSA ClearTrust Servers that allows the Servers to access and translate data in a variety of data sources, such as LDAP directories, SQL databases and so on. The DAL integrates with the Data Adapter component.

Data Adapter

The Data Adapter is a vendor-specific and platform-specific component that allows the RSA ClearTrust Servers to access data in its native format (LDAP, SQL, XML and so on), and translate that data for use by RSA ClearTrust. The Data Adapter is essentially an *agent* for your data server that allows you to map your data to RSA ClearTrust. The Data Adapter integrates with the Data Abstraction Layer (DAL).

dc

A commonly used LDAP attribute name that stands for *domain component*.

Delegated Administration

The practice of distributing administrative powers among multiple Administrative Users and business groups. Using delegated administration, organizations can establish individual administrative hierarchies responsible for managing specific resources and users. RSA ClearTrust uses the Administrative Group model to enable delegated administration for various administrative users ranging from help desk personnel to Extranet partners.

Directory Replication Manager

The RSA ClearTrust administrative tool for transferring user and policy data to the Entitlements Database from LDAP directories or other environments. This is a component of RSA ClearTrust 4.6 that is not supported in RSA ClearTrust 4.7.

Dispatcher/Key Server

The RSA ClearTrust Server responsible for two different pieces of functionality: providing a route for the Web Server Agent and Authorization Server to contact one another and establish communications; and, when Single Sign-on is being used, rotating through the encryption keys to maintain robust security between Web clients and the RSA ClearTrust system.

DN

An X.500 distinguished name, which is unique name for a node in an LDAP directory tree. Usually a DN is used to provide a unique name for a person or any LDAP directory entry. A DN is the concatenation of selected attributes from each node in the tree along the path leading from the root node down to the named entry's node (for example, the person). In LDAP notation, the DN for a woman named Rosanna Lee working at RSA's US office would be “cn=Rosanna Lee, ou=People, o=RSA, c=us” and in Microsoft Active Directory notation, this would be “/c=us/o=RSA/ou=People/cn=Rosanna Lee”. See also, base DN and RDN.

E

Entitlements Database

The Entitlements Database is not supported in RSA ClearTrust 4.7. All user, resource, administrative, and policy data is now stored in native LDAP or SQL data stores. The RSA ClearTrust schema is added to an existing LDAP directory or SQL database. The new Data Abstraction Layer (DAL) component allows customers to map their existing user and group data, so it can be used by the RSA ClearTrust Servers.

In RSA ClearTrust 4.6, the Entitlements Database is a proprietary data repository where all of the information needed by RSA ClearTrust is stored. RSA ClearTrust 4.6 supports both Oracle and Sybase databases out-of-the-box.

Entitlements Manager

The Entitlements Manager is the administrative tool used to define and edit data in the RSA ClearTrust Data Stores. This tool is a Web-based application that interfaces with your back-end data server (via the Entitlements Server). Administrators use the Entitlements Manager to define protected resources and access control policy in RSA ClearTrust.

User and group data can also be managed through the Entitlements Manager. However, RSA Security recommends using your existing user management tool if you already have an established LDAP directory for your user data.

Entitlements Server

The central server for the administrative side of the RSA ClearTrust system. Changes to the RSA ClearTrust policy data store (Adds, Deletes and Modifications) can only be made via the Entitlements Server. However, you can manage your user and group information in your native LDAP administration tool.

G

password policyGroup

In the RSA ClearTrust data model, a group is a collection of users that are logically grouped together. Groups can be nested. For example, you can create a group called *Sales*, which contains a group called *West Coast Sales*, which contains a group called *Product Sales*, and so on.

H

hashing

The process of producing hash values for accessing data or for security. A hash value (or simply hash) is a number generated from a string of text. The hash is substantially smaller than the text itself, and is generated by a formula in such a way that it is extremely unlikely that some other text will produce the same hash value.

Hashes play a role in security systems where they are used to ensure that transmitted messages have not been tampered with. The sender generates a hash of the message, encrypts it, and sends it with the message itself. The recipient then decrypts both the message and the hash, produces another hash from the received message, and compares the two hashes. If they're the same, there is a very high probability that the message was transmitted intact.

L

LDAP

Lightweight Directory Access Protocol. A set of protocols for accessing information in directories. LDAP supports TCP/IP, which is necessary for any type of Internet access. Because it's a simpler version of the X.500 standard, LDAP is sometimes called X.500-lite.

The current version of LDAP is version 3 (v3), which is the version supported by most directory vendors and by RSA ClearTrust 4.7.

LDAP attributes

LDAP attributes (also referred to as *attribute types* or *attribute definitions*) hold a specific data element such as a name, business phone number, or email address. Attributes also have information about that data element, such as the data type, maximum length, and whether it is single-valued or multivalued. The attribute type provides the real-world meaning for a set of values, and specifies the rules for creating and storing specific pieces of data, such as a name or a phone number.

LDAP attribute name

Attribute names are usually short, and are limited to ASCII letters. An attribute name must be unique across your entire directory service, because LDAP applications generally refer to an attribute using its name. Some common attribute names are: dc, ou, uid, cn, and sn.

LDAP attribute values

Attribute values are the actual data contained within an attribute. For example, for the attribute type *email*, an attribute value might be *bsmith@rsasecurity.com*.

LDAP filter

An LDAP filter is an expression that defines the types of entries to be returned from a search on your directory.

For example, you may want to create a search filter that looks for all users in your LDAP directory who are over 21 (age>21).

LDAP namespace

The LDAP directory namespace provides the basic means by which directory data is named and referenced. A namespace is required to reference entries, and provide features such as groups.

The LDAP namespace model is inherited from the X.500 directory standard, which was intended to be a fairly rigid, worldwide, hierarchical directory service. The X.500 namespace starts with Countries (c=us) at the top of the hierarchy, followed by States (st=CA), then Organizations (o=rsasecurity), and so on.

The LDAP namespace model is more flexible, allowing you to design a hierarchical, tree-structured namespace that meets the needs of your environment.

LDAP referral

A referral is a piece of information returned by an LDAP server that indicates to the requesting client machine that it must contact other LDAP servers to fulfil the request. The client then contacts the other LDAP servers to which it was referred, resubmits the original request, and presents the results to the user.

Referrals allow an organization to distribute its directory accross multiple LDAP servers. RSA ClearTrust 4.7 supports LDAP referrals.

LDAP schema

An LDAP schema is the collection of LDAP attributes, object classes and other information which an LDAP directory server uses to determine how records are created and stored. RSA ClearTrust has its own proprietary LDAP schema for storing resources, administrator information and policy information in an LDAP directory. RSA ClearTrust's schema was designed based on LDAP v3 and IETF standards, and includes some proprietary object classes and LDAP attribute type definitions.

LDIF

LDAP Data Interchange Format. A common, text-based format for exchanging directory data between systems. There are two different types of LDIF files. The first form describes a set of directory entries, such as a corporate directory. The other type is a series of LDIF update statements, which describe changes to be applied to directory entries.

O

o

A commonly used LDAP attribute name that stands for *organization*.

object classes

In LDAP, object classes are used to group information. Typically an object class models a real-world object such as a person, department or server. Each directory entry belongs to one or more object classes. The object class determines the attributes that make up an entry. One object class can be derived from another, thereby inheriting some of the characteristics of the other class.

OID

Object identifiers. In LDAP, this is a unique identifier assigned to an object. OIDs are used to uniquely identify many different types of things, such as the X.500 directory object and LDAP attributes.

ou

A commonly used LDAP attribute name that stands for *organizational unit*.

P

passive mode

If the Authorization Server is configured in *passive mode*, all Resources not explicitly protected by the RSA ClearTrust system are not accessible. This means that users cannot access any resource on a Web server, unless they have been explicitly granted access in the Entitlements Manager. See also, active mode.

Password Policy

A specified set of requirements for user passwords. For example: minimum length, frequency of change, avoidance of common words.

policy

A rule defining access to system resources. RSA ClearTrust provides dynamic permissions and access control by means of SmartRules and Basic Entitlements, defined by RSA ClearTrust Administrators.

private

In the Entitlements Manager, an object (such as a user, Web server, and so on) that is defined as private is visible only to members of the Administrative Group who own the object. See also public.

public

In the Entitlements Manager, public objects (such as users, resources, and so on) and their associated information are visible to all Administrative Users. See also private.

R

RDN

Relative Distinguished Name. In an LDAP directory entry's DN, the left most component is considered the RDN. If the DN for a woman named Rosanna Lee working at RSA's US office is “cn=Rosanna Lee, ou=People, o=RSA, c=us”, then the RDN is “cn=Rosanna Lee”. Among a set of peer entries, each RDN must be unique. See also, DN.

Realm

In previous releases of RSA ClearTrust, a realm was a specified group of groups. Realms are not supported in 4.7, since RSA ClearTrust now supports nested password policyGroups.

Replication Manager

See Directory Replication Manager.

Resource

In the RSA ClearTrust data model, this refers to a protected item or a group of protected items. A resource can be a Web server, directory or file, including Web pages, graphic files, generated files and software programs.

RSA ClearTrust

RSA ClearTrust® is a Web access management solution that was designed to flexibly integrate into your existing information technology (IT) environment. RSA ClearTrust allows organizations to secure applications, Web sites, and other Web-based resources via intranets, extranets, and business-to-business (B2B) and business-to-consumer (B2C) infrastructures.

RSA ClearTrust version 4.7 allows you to quickly and easily implement unified access management by natively connecting to your existing user data stores, such as LDAP directories.

RSA ClearTrust Data Stores

This is either the LDAP directory or the SQL database that contains your RSA ClearTrust user and policy data. The policy data store is proprietary to RSA ClearTrust. The policy schema is required to be inserted into an existing LDAP directory or SQL database.

User data can be kept in your own, native LDAP directory structure or SQL database tables, and simply mapped to RSA ClearTrust using the Data Adapter configuration file.

RSA ClearTrust Servers

The RSA ClearTrust Servers are the core of the RSA ClearTrust system, and are responsible for validating users and their access control privileges. This consists of the Authorization Server, the Dispatcher/Key Server and the Entitlements Server.

S

SecureDetector

The threat detection and auditing component of RSA ClearTrust version 4.6. SecureDetector is not supported in RSA ClearTrust 4.7.

Server Tree

In the RSA ClearTrust data model, a Server Tree is a protected directory on a Web server. What differentiates a Server Tree from a URI is that it can be owned and administered by an Administrative Group.

SmartRules

Clear Trust's dynamic access control policies. After Administrators define Smart Rules in the Entitlements Manager, the rules and policies are applied and updated automatically during runtime operations. Smart Rules are based on User Properties.

sn

A commonly used LDAP attribute name that stands for *surname*.

Super User

A system administrator with the highest levels of access in the Entitlements Manager. Some functions can only be done by the Super User.

U

uid

A commonly used LDAP attribute name that stands for *unique identifier*.

URI

Uniform Resource Identifier. A URI is the way to identify any point of content on the Web, whether it be a page of text, a video or sound clip, a still or animated image, or a program. The most common form of URI is the Web page address, which is a particular form or subset of a URI called a URL. A URI typically describes:

- The mechanism used to access the resource
- The specific computer that the resource is housed on
- The specific name of the resource (a file name) on the computer.

URL

Uniform Resource Locator. A URL is the address of a file (resource) accessible on the Internet. The resource can be an HTML page, an image file, a program such as a Java applet, or any other file supported by HTTP. The URL contains the name of the protocol required to access the resource, a domain name that identifies a specific computer on the Internet, and a hierarchical description of a file location on the computer.

User

An individual logon account in the RSA ClearTrust system.

User Properties

A User Property is a custom piece of information (or attribute) for an RSA ClearTrust user that is not part of the existing RSA ClearTrust data model. User Properties in RSA ClearTrust map to your existing user LDAP attributes or database columns. The values stored in a User Property are used for dynamic access control, also known as SmartRules.

V

VBU

Virtual Business Unit. See Administrative Group.

virtual host

This term refers to a single physical Web server machine that is hosting more than one Web site or domain. For example, in Microsoft IIS and Apache, you can configure your Web server to host multiple Web sites on the same Web server. For example, *www.company.com* and *intranet.company.com* can be hosted from the same Web server machine.

RSA ClearTrust supports virtual hosts on Apache and IIS Web servers, meaning that each virtual Web site can have its own RSA ClearTrust configuration. If you are using virtual hosts, you must create separate Web servers in the Entitlements Manager for each virtual host.

W

Web form

An HTML document used to collect information from the user. This information is usually submitted to a server-side program for processing.

Web server

A Web server is a program that, using the client/server model and the World Wide Web's Hypertext Transfer Protocol (HTTP), serves Web page files to browser clients. Many of RSA ClearTrust's security features are enabled by Web Server Agents, which are integrated into Web servers to extend and enhance their security functionality.

Web Server Agent

An RSA ClearTrust component that is installed on supported Web servers. The Web Server Agents interface with the RSA ClearTrust Authorization Server to perform authorization and authentication requests.

X

X.500

An ISO (International Organization for Standardization) and ITU (International Telecommunications Union) standard that defines how global directories should be structured. X.500 directories are hierarchical with different levels for each category of information, such as country, state, and city. See also, LDAP.

X.509

The most widely used standard for defining digital certificates. X.509 is actually an International Telecommunication Union (ITU) recommendation, which means that has not yet been officially defined or approved. As a result, companies have implemented the standard in different ways. For example, both Netscape and Microsoft use X.509 certificates to implement SSL in their Web servers and browsers. But an X.509 Certificate generated by Netscape may not be readable by Microsoft products, and vice versa.

Index