



The Compliant Enterprise Active Control System

Release 2.0

**Policy Author
User's Guide**



May 2007

Copyright © 2005-2007 NextLabs, Inc. All rights reserved.
The information in this document is subject to change without notice.

NextLabs welcomes comments or suggestions regarding this manual or any of our product documentation. Please send an e-mail to info@nextlabs.com.

TRADEMARKS

Compliant Enterprise™, ACPL™ and the Compliant Enterprise logo are registered trademarks of NextLabs, Inc. All other brands or product names used herein are trademarks or registered trademarks of their respective owners.

LICENSE AGREEMENT

This documentation and the software described in this document are furnished under a license agreement or nondisclosure agreement. The documentation and software may be used or copied only in accordance with the terms of those agreements. No part of this manual may be reproduced, stored in a retrieval system or transmitted in any form or any means electronic or mechanical, including photocopying and recording for any purpose other than the purchaser's use, without the prior written permission of NextLabs, Inc.

Published in San Mateo, CA, by NextLabs, Inc.
www.nextlabs.com
info@nextlabs.com
650.577.9101

Document Revision Number: PAUG2.0-B03

Preface	7
What's New in Release 2.0	7
Product Documentation	8
Product Overview	8
Getting Started Guide	8
Implementation Guide	8
Administrator's Guide	9
Policy Author User's Guide	9
Enforcer Administrator's Guide	9
CE Reporter User's Guide	9
Solutions Guide	10
Current Versions	10
Release Notes	10
Feedback	10
1. Introducing Policy Author	11
Overview	11
About Views	12
Starting Policy Author	13
Logging In	13
Logging Out	13
Menu Commands	13
Interface Panes	15
Editor Pane Tools	16
The Preview Pane	17
About Components	18
Component Types	18
Defining Components	19
About Policies	20

Editing Tools	20
Handling Policies	22
Constructing Policies	22
Submitting Policies	22
Deploying Policies	22
Components and Policies: Life Cycle	23
2. Defining Components	25
Defining New Components	25
Component Types	26
Defining Object Components	26
Common Procedure	26
About Membership and Properties	27
Defining Membership	27
Multiple Memberships	29
Using the Lookup Feature	29
Membership Specification Syntax	29
Object Component Properties	30
Defining Properties	31
Defining Document Components	32
Document Components: Memberships	32
Removable Media	34
Special Keywords	34
Document Components: Default Properties	35
Defining User Components	37
User Components: Membership	37
The Member Search Tool	38
User Components: Default Properties	38
Defining Computer Components	39
Computer Components: Membership	39
Computer Components: Default Properties	40
Defining Application Components	40
Predefined Application Components	41
Defining Portal Content Components	43
Portal Content: Memberships	43
Portal Content: Default Properties	45
Defining Action Components	47
Basic Actions	48
File System vs. Portal Actions	49
3. Constructing Policies	51
Creating a New Policy	51
Defining a Policy Subject	52
Defining a Policy Action	54
Defining a Policy Object	54

Target Location	54
Define a Time Context	55
Defining Obligations	55
Add a Description	57
Using Obligations	57
Obligation Categories	57
Enforcement Event Logging	57
Notifications	58
Monitor Policies	60
Custom Obligations	61
About Enforcement Logging	61
Logging vs. E-Mail	62
Organizing Policies	62
Using Folders	62
Folder Permissions	62
Exporting and Importing Policies	62
4. Using Policies and Components	65
Deploying Policies and Components	65
The Deployment Sequence	65
Deploying All	66
The Full Deployment Sequence	67
Checking Dependencies	67
Setting Deployment Targets	68
Submitting for Deployment	69
Scheduling Deployment	70
Confirming Deployments	71
Management Tools	72
Deployed Components	72
Deployed Policies	73
Canceling Deployment	74
Deployments by Host	74
Deployment Version History	75
Managing Policies and Components	77
Searching for Policies or Components	77
Finding Out Where a Component is Used	77
Viewing Properties	78
Comparing to the Deployed Version	78
Modifying Policies and Components	79
Renaming an Object	80
Changing Access Rights	80
About Default Settings	81
Un-deploying Policies and Components	81
Cancelling	81
Deactivating	82

Deleting an Object	82
5. Using the Sample Policy Sets	85
The Sample Folder	85
Destiny Project	86
Sales Forecasting	86
Compensation Planning	86
Corporate	86
Using the Samples	87
Policy Set I: Destiny Project	88
Policy 1: Controlling Access	88
Effect	89
Subject	90
Action	90
Resource	90
Customizing the Sample	91
Policy 2: Duplication and Distribution Control	92
Subject	93
From and To Resources	93
Other Policy Sets	95
Policy Set 2: Sales	95
Policy Set 3: Compensation Planning	96
Policy Set 4: Corporate	97
Policy Life Cycle	98
Moving On	101
Appendix A: How Do I . . . ?	103
Index	107

Preface

Welcome to the Compliant Enterprise Active Control System, the information control platform that eliminates policy silos, controls information disclosure inside and outside the enterprise, and provides universal control over information access and use along with real-time enforcement. Only NextLabs delivers an Active Control System that can comprehensively enforce information access entitlements, protect end points while data is in use, and maintain reliable information barriers.

What's New in Release 2.0

Release 2.0 of Policy Author includes the following new features and improvements over the 1.6 release:

- Overall redesign of the Policy Author interface: look and feel, menu structure, generic controls, and organization of the components panels and action types. This redesign is geared toward increasing overall usability, and also supporting policy controls over collaboration portals.
- Support for policy enforcement on for collaboration portals such as SharePoint. In Policy Author, this support is reflected in the new Portal Content bin, and in several new basic actions.
- Support for the Linux File Server Enforcer, whose operation is essentially similar to the Windows file server enforcer.
- Support for Custom Obligations. You can write custom scripts or batch files to perform any kind of behavior you like, which can then be invoked as a result of policy enforcement. Examples of custom obligations might be sending a page message, encrypting some specified documents, or automatically scanning documents and flagging those with sensitive contents.
- Ability to export defined policies and components from one instance of Compliant Enterprise and import them into another.

Product Documentation

The Compliant Enterprise documentation set consists of eight titles: an introductory *Product Overview*; a *Getting Started Guide* with installation and configuration instructions; an *Implementation Guide* to help with strategies for auditing information use and designing policies; an administrator's guide for all enforcers and one for the system overall; user's guides for Policy Author and Reporter; and a guide to the predefined active control solutions available with release 2.0.

Product Overview

Because Compliant Enterprise is a powerful, distributed enterprise product, its components are likely to be used by a number of different users in any given organization. Even though various users may be engaged exclusively with individual components of the suite and may not be interested in any others, we strongly recommend that all users read the Product Overview carefully, in order to acquaint themselves with the high-level architecture and function of the platform as a whole.



Getting Started Guide

The *Getting Started Guide* provides instructions on planning your system architecture and installing the Control Center and Policy Author.

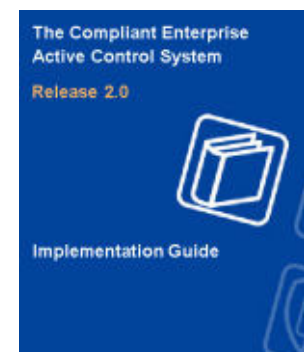
The installation procedures for all policy enforcers are provided separately, in the *Enforcer Administrator's Guide*.

Instructions on enrolling network entities, which is required after installation, are also provided separately, in the *System Administrator's Guide*.



Implementation Guide

The *Implementation Guide* provides a high-level approach to designing and implementing the information control policies that best suit your enterprise's needs. It offers generic advice on analyzing your needs through information use audits, approaches to designing appropriate policies, and optimizing those policies based on ongoing monitoring.



Administrator's Guide

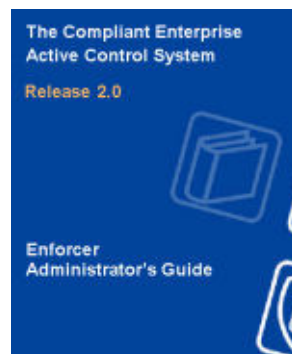
The *System Administrator's Guide* provides information required for managing and maintaining the Compliant Enterprise system once it is set up. It provides complete instructions on enrolling all kinds of network entities, which is required after the initial software installation. It also includes all user information for the administrative web application called Administrator, as well as for all utilities and other tools provided with the product. It is directed at the IT specialists who will be responsible for maintaining the Control Center after it has been installed.

**Policy Author User's Guide**

This user's guide provides complete information on how to use Policy Author, the user interface where you build, deploy, and manage your information control policies and the library of policy components they are built upon. It is intended for the Compliant Enterprise user who will be responsible for converting generically expressed information policy goals into the specific, ACPL-based policy controls that are actually distributed to enforcement points throughout the enterprise.

**Enforcer Administrator's Guide**

The *Enforcer Administrator's Guide* provides information on installing, using and maintaining all the types of enforcers currently available for Compliant Enterprise: for Windows file servers, Linux file servers, Windows desktops, and ShaerPoint servers. It is intended for the technical specialists who will be managing the enforcers; these may be the same as the Control Center administrators, or they may be different.

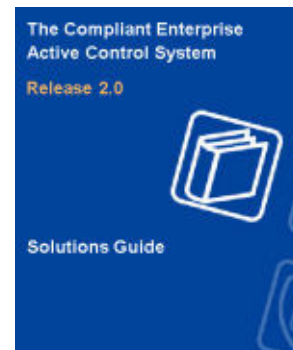
**CE Reporter User's Guide**

The *Reporter User's Guide* provides complete information on how to use Reporter, the web-based application that lets you easily generate reports on information use and access in your enterprise, and on the performance of your deployed policies. It is required reading for anyone with permission to generate or view Compliant Enterprise reports.



Solutions Guide

The *Solutions Guide* provides detailed information on customizing and using the pre-designed Active Control Solutions that are included with Compliant Enterprise: for Information Entitlements, for Endpoint Data Protection, and for Business Information Barriers.



Current Versions

Documents distributed in PDF format can become obsolete as subsequent versions are released. If you would like to check whether you are using the most current version of this or any manual, check the Document Control Number (DCN) at the bottom right of the inside cover, then click [here](#) to view a table of the most current versions of all Compliant Enterprise manuals. If the version listed in that table is later than the one in this manual, contact info@nextlabs.com to request the more recent version.

Release Notes

The release notes for each release of Compliant Enterprise are available directly on the installation CD, from the link on the splash screen or from the Docs directory. They describe any features or changes that could not be included in the documentation, and provide a list of known problems with the current version, along with suggested workarounds when appropriate.

Feedback

Feedback from Compliant Enterprise users is a valuable resource in helping our Product Information group provide you with the highest quality documentation as our product line develops. To this end, we would appreciate any comments you have on this manual or on any other Compliant Enterprise documentation; please send all feedback to info@nextlabs.com.

Introducing Policy Author

Implementing the information control policies you want to enforce in your environment involves, specifically, using the Compliant Enterprise tool called *Policy Author* to define a set of objects representing components in your environment, then using those objects to construct control policies, and then broadcasting those policies to all relevant Compliant Enterprise Policy Enforcers installed throughout the network.

The present chapter introduces some of the specialized terminology Policy Author uses, and then provides a very general description of the process of implementing your policies. Later chapters will delve more deeply into the details of defining components, constructing policies, and deploying them throughout your organization.

This chapter covers the following topics:

- Overview ([page 5](#))
- About Components ([page 12](#))
- About Policies ([page 14](#))

For a quick reference to many of the features of Policy Author, see [Appendix A](#), "How Do I . . . ?".

Overview

Policy Author is a graphical user interface that can be used to accomplish the following tasks:

Modeling: This refers to the process of creating an abstract model representing the various parts of the enterprise environment, such as users, applications, documents, and desktop PCs. The actual work of modeling consists of defining *policy components*. A component is a named definition that represents a category or class of entities, such as users, actions, or applications; or of actions, such as Open or Copy.

Components may be thought of as the parts of speech you use to construct your policy statements. For example, you might define "noun" policy components such as *All employees in the human resources department* or *Any file with an .xls extension*, and "verb" components like *Copy*, *Print*, and *Rename File*.

Constructing Policies: This refers to the process of building information control policies in Compliant Enterprise. Policies use components as building blocks to

represent rules that you wish to enforce to control information access and use in your organization.

For example, you might construct the following policy: “Allow only product managers to download or print spreadsheets.” The components used in this policy are “product managers,” “download,” “print,” and “spreadsheets.” When finished, policies are stored in a Policy Master database for distribution to the network components that enforce them.

In addition, you can qualify policies with contextual parameters such as time of day or day of the week. You can also specify notification actions, such as notifications to the end user or e-mail warnings to an administrator whenever a policy is enforced anywhere in the network.

Deployment: Deployment simply means the distribution of new or modified policies and policy components to the appropriate enforcement points on desktop PCs, laptops, and file servers throughout the organization. This means you can create, review and refine policies as long as you like, but they will not be enforced until you actually deploy them.

When you install Policy Author, an extensive set of pre-defined policies and components is loaded into the Policy Manager database, and is available for use in the Policy Tree. These pre-defined policies are organized into Solutions, each of which is designed to address a segmented class of information control challenges. If these solutions match your needs closely enough, you can simply deploy them (after minor modification to customize them to your environment), and may not need to define your own policies at all. For more details, refer to the *Compliant Enterprise Solutions Guide*.

About Views

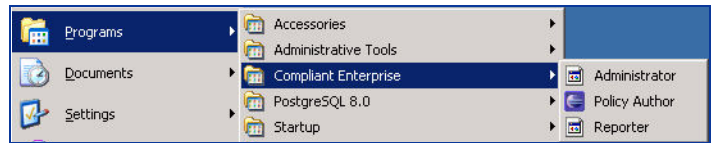
Depending on your configuration settings, you open Policy Author in one of three available views. A view is a set of rules about which controls and UI features are displayed, and which are hidden. The three views are:

- **File System View:** All controls and features relevant only to file system-based components are displayed and active. All those relevant only to collaboration portals—the Portals component pane, for example—are hidden.
- **Portal View:** All controls and features relevant to collaboration portal-based components are displayed and active to use. All those relevant only to file systems—the Attach to E-Mail and Paste basic actions, for example—are hidden.
- **Corporate View:** All controls for all six component types are displayed and active.

The default view is Corporate. Bear in mind, however, that if you do not see some of the features described in this manual, it may be because you are working in one of the other, more restricted views. If you need to change the view you are working in, consult your Compliant Enterprise administrator. Information on how to change views is provided in the *Compliant Enterprise 2.0 Administrator's Guide*.

Starting Policy Author

Policy Author is installed as a standard Windows application, and can be started from the Compliant Enterprise program group under Start, All Programs.



Logging In

When you start the application, you must specify a user name and password and a Control Center host name. The user name can be any user currently defined with permission to use Policy Author. The default user, *Administrator*, has a password assigned during the Control Center installation; other users may be defined by the system administrator after that.

The host is the machine where the Control Center (specifically, the Policy Server) you want to use is running; this is specified during Policy Author installation, and will display automatically. You also have the option of connecting to any other instance of Control Center, as long as you know its hostname and the valid username and password for it. (This is more likely in a development or testing environment than in a production setting.) The first time you connect to an alternate server you must type the hostname manually; after that both your default and the alternate will display in the combo-box list. The list will display all servers to which you have connected from this installation of Policy Author.

Logging Out

To stop Policy Author, use the Exit command in the File menu, or click the red **X** in the upper right corner of the window. Your work is always saved automatically when you exit the application.

Menu Commands

There are six menus available in Policy Author: File, Edit, Tools, Actions, Window, and Help. [Table 1-1](#) describes the functions of all commands in these menus.

Table 1-1: Policy Author Menu Commands

Menu	Command	Function
File	Save	Saves any recent changes in the currently selected policy or component. It is not necessary to manually save your changes; Policy Author does that automatically whenever any change is made in the editing pane.
	Change Password	Allows the current user to change his own login password.
	Properties	Displays the properties tab of the currently selected policy or component.
	Exit	Closes Policy Author.
Edit	Undo	Undoes the effect of the last action performed.
	Redo	Redoes the last undone action.
	Delete	Deletes the currently selected item.

Table 1-1: Policy Author Menu Commands (Continued)

Menu	Command	Function
Tools	Deployment History	Opens the Deployment History window; for details, see "Deployed Policies" (page 67).
	Deployment Status	Opens the Deployment History window, which displays the current deployment status of all components and policies that have been deployed. For details, see "Deployments by Host" (page 68).
Actions	Modify	Changes the status of the currently displayed component or policy to Draft. You must do this whenever you want to make any changes to a component or policy that has been submitted. Function is the same as the Modify button at the bottom of the Editing pane.
	Submit	Submits the currently displayed component or policy for changing from one status to another—for example, from Draft status to Submitted for Deployment. Function is the same as the Submit button at the bottom of the Editing pane.
	Deploy	Deploys the currently displayed component or policy. Function is the same as the Deploy button at the bottom of the Editing pane. As with individually deployed objects, you can specify a scheduled deployment, or choose Now.
	Deploy All	Deploys all currently submitted components or policies. Function is the same as the Deploy button at the bottom of the Editing pane.
	Deactivate	Changes the status of the currently displayed policy or component from Active to Deactivated.
	Show Policy Usage	For the currently displayed component, displays a list of all policies in which it is currently included, regardless of the policies' status.
	Show Deployed Version	For the currently displayed component or policy, displays the detailed definition of the version that is currently deployed.
	Check Dependencies	For the currently displayed component or policy, runs a value check and displays all the components that are required by (that is, referenced inside) the component or policy.
	Set Deployment Target	Allows you to manually specify which enforcement point the selected policy or component will be deployed to. You can use this feature as an alternative to Compliant Enterprise's Auto-deployment capability.
	Version History	Opens the Version History window for the currently displayed component or policy. This window displays a list of all previous versions, starting with the most recent one. Versions are identified by date and time stamp. For details, see page 11 . If there has only been one version of the component or policy, this command is disabled.
Window	Preview	Opens the Preview pane, at the right side of the Editor pane. The Preview pane allows you to test the actual content that would result from the current definition of a component. For details, see page 11 .
Help	Help	Displays the Policy Author User's Guide, in HTML, in the default Web browser.
	About	Displays the About tab in the Editor pane, with standard copyright information about Policy Author, the currently installed version, and links to product documentation and other resources.

Interface Panes

Chapter 2 of this manual presents a detailed description of how you use Policy Author to define components. Meanwhile, here is a brief overview of the tools available in Policy Author.

As [Figure 1-1](#) shows, the screen is divided into three panes:

- The **policy tree**, at the upper left, is where you store, organize, and retrieve policies once you have defined them.
- The **component panels**, at the lower left, are where you store and retrieve components once you have defined them. To construct policies, you simply drag components from one of these panels into the appropriate fields in the editing pane. (Click on one of the panels to expand it, if it is collapsed.) Note that some of the panels are further organized into several tabs for different component types.
- The **editing pane**, on the right, is where you work as you are defining new components or policies, or editing existing ones. You can use the buttons at the bottom of this pane to control the state of a policy: to submit, deploy, or modify it.

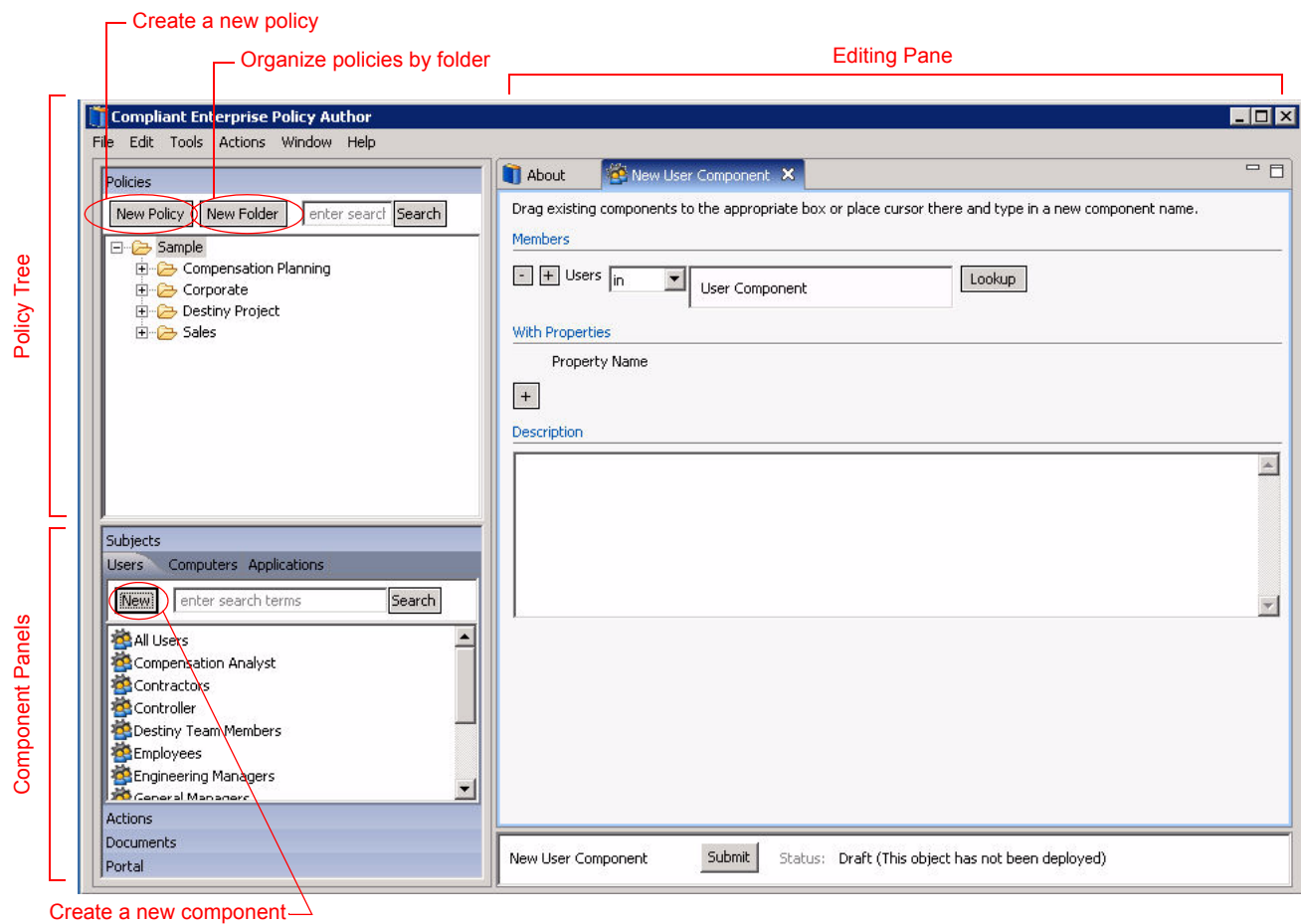


Figure 1-1: Component Editing Controls

Editor Pane Tools

Each object (component or policy) you open is displayed on a tab in this pane, and stays open until you close it. Each tab is dockable within this pane; you can grab it and reposition it below or to the right or left of other tabs. This allows you to view more than one object at once in a tiled layout as shown in [Figure 1-2](#).

When working with several docked tabs, you may find it convenient to use the maximize/minimize icon, at the upper right corner of each tab. Maximize expands the tab to fill the entire UI screen; minimize restores the previous view.

Maximize / Minimize Icon

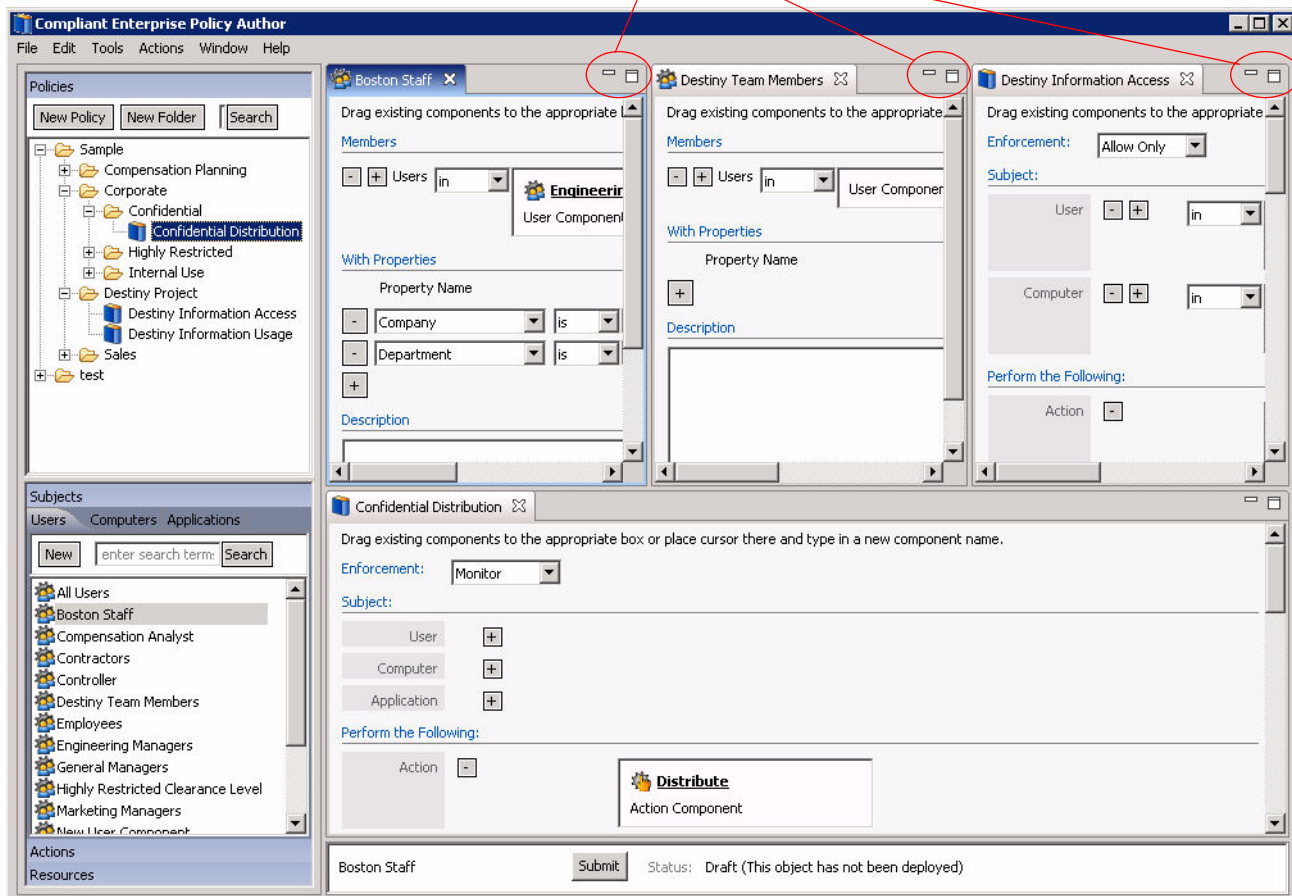


Figure 1-2: Viewing Tabs in Docked Layout

The Preview Pane

A fourth pane, Preview, can be opened as needed, using the Preview command in the Window menu. The Preview pane allows you to display all of the individual entities that would be included in a component as it is presently defined. It displays at the right side of the screen.

This feature is very useful for testing or debugging components. If the list contains entities you did not expect or intend to be included in the component you are defining, or does not contain entities you wanted to include, you know there is a problem with the definition of the component, and you can correct it before deployment.

The preview feature is available only for User, Computer, Document, and Application components. Note that for Document components the preview will generally display a list of file names, but if you have included an Include Only Directories property, it will also list network directories. (For details on this feature, see [page 30](#).)

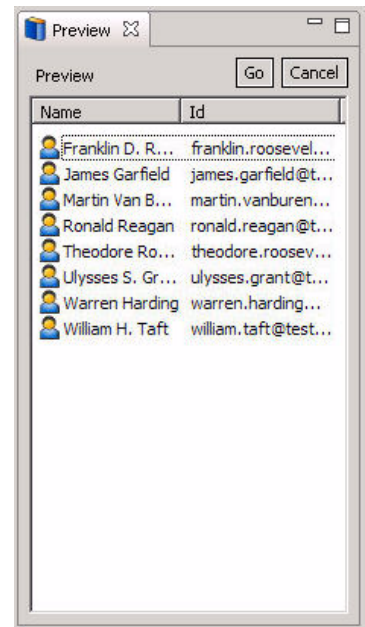


Figure 1-3: Previewing Component Contents

About Components

Components are abstract building blocks that you can define as the raw material for the information control policies you want to implement in your organization. They do not represent individual network entities, but rather categories or classes, either of physical entities in your organization that play some role in information control policies, or of actions that may involve those entities. Because components represent not individual real-world entities but the way an organization thinks about them, they provide a layer of abstraction that insulates an organization's information control policies from changes to the organization's environment.

Employees join and leave the organization, documents are created and deleted, computers are purchased and retired, and SharePoint sites or pages are created every day. In spite of these changes, you can write policies without needing to know specifically about the underlying physical-world users, documents, or servers—the policies will remain in effect, covering all appropriate network entities.

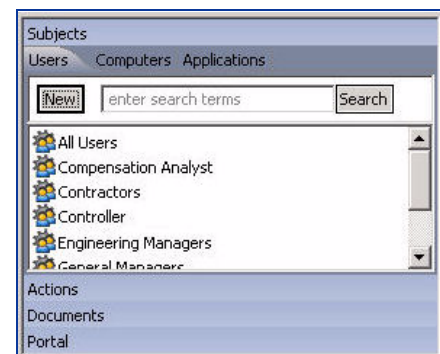
Component Types

Compliant Enterprise supports six broad categories of components:

- Users
- Computers
- Applications
- Actions
- Documents
- Portal Content

These categories are organized into collapsible component panels in the bottom left corner of the Policy Author screen.

Note that Users, Computers and Applications are combined as separate tabs on the Subjects panel; this is because these three components work together in a special way to define policy subjects.



You can define individual components by specifying one of these types, and narrow its definition by assigning properties. Once you define and save a component, it can be used to construct policies, which will govern not some *specific* user in your organization, but *any* user belonging to the category that the component describes.

For example, rather than listing every manager in your organization by name, you can create a user-type component that represents the category, “all users with the job title of Manager.” Then you can use this component as a building block to construct policies preventing (or allowing) *all* managers from using specified information in specified ways. This means that whoever designs policies using policy components does not need any direct knowledge of who the managers are, or of who will gain or lose that title in the future, but only of how as a class they should be allowed to access and use information.

Action components work a bit differently than the other two categories, in that they are not defined based on properties, but rather simply as a combination of one or more of the basic actions as listed in the table above. For example, you might define an action component called “E-Distribute”, that covers attaching to e-mail, inserting into e-mail, and attaching to instant messages.

You can use components in various combinations to construct the policies you need for controlling information. For example, you might define a user component “Independent Contractors,” and an action component “Copy, Move, Print, or Send via E-Mail or IM.” You could then combine these to implement an “eyes-only” rule: “All independent contractors may read company documents but may not transmit them in any way.” Once this rule is distributed to all Compliant Enterprise enforcement clients in your network, it will be enforced regardless of what individual contractor or document is involved.

We sometimes refer to the process of defining components in Policy Author as *modeling*, since it involves constructing a model of component categories that is comprehensive enough to represent all the actual entities in your organization that will need to be covered by policies. You can either make a component model based on your knowledge of the network entities, regardless of how they may need to be used in policies; or based on the requirements of a set of logical policies that have already been designed. Indeed, in practice you will often find it simplest to construct a set of policies first, then go back and define the components you need for those policies.

Defining Components

A policy component can be defined using one or more of the following:

- Imported Active Directory items, such as users, groups, applications, or computers, designated through the Browse tool
- Content currently defined in SharePoint other types of collaboration portals, designated through the Browse tool or typed directly
- Other existing components of the same type
- Properties of Active Directory items, such as employee job title or branch location, or file extension or saved date
- For a document-type component, you can also use the keywords [MyDocuments] and [MyDesktop] to represent those standard locations on all users’ PCs.

A network entity can belong to more than one policy component. For example, some of the same people might belong to both the “Home Office Employees” component and the “Human Resources Department” component.

About Policies

A *policy* is a statement that describes a document access or usage situation and defines what action Compliant Enterprise should take when that situation arises. It presents set of rules controlling how different categories of users in a given environment are allowed to use different categories of documents. You use Policy Author to construct policies as combinations of components that are linked together with operators and other logical constraints, and then further refined by contextual conditions, such as time of day. Typically, an organization will construct enough policies to cover all potential business situations where some kind of information control is required.

Each policy comprises a set of pre-defined building blocks strung together according to a precise syntax. [Figure 1-4](#) provides some examples.

Subject			Verb	Object	Context
USER Components	COMPUTER Components	APPLICATION Components	ACTION Components	RESOURCE Components	SCHEDULE Day, Date, Time
Contractors	Unmonitored PC	Any MS Office	Open	Any SP Library	Weekends
Temp Staff	Removable media	Word or Wordpad	Copy	Any Spreadsheets	Mon - Fri
Full-time Staff	Laptop	Approved E-mail	Attach to E-Mail	CAD Drawings	On 11 pm - 6 am
Managers	File server	Photoshop	Print	HR Portal Site	3rd Fri / month
Executives	Archive	Acrobat Reader	Rename	Product Specs	Every Mon 8 am
Sales Team	Contractor PC	Approved CRM	Modify	Payroll Docs	Nov 20 - Nov 27
Top Clearance	Top Clearance	Any Adobe app	Copy/paste content	Sensitive Docs	Tues - Thurs
IT Staff	IT Staff	Notepad	Change properties	Image Files	From/ To Every other Fri
Boston Branch	Boston Branch	Any bitmap editor	Import	Meeting Minutes	Fri after 6 pm

Figure 1-4: Components of a Compliant Enterprise Policy

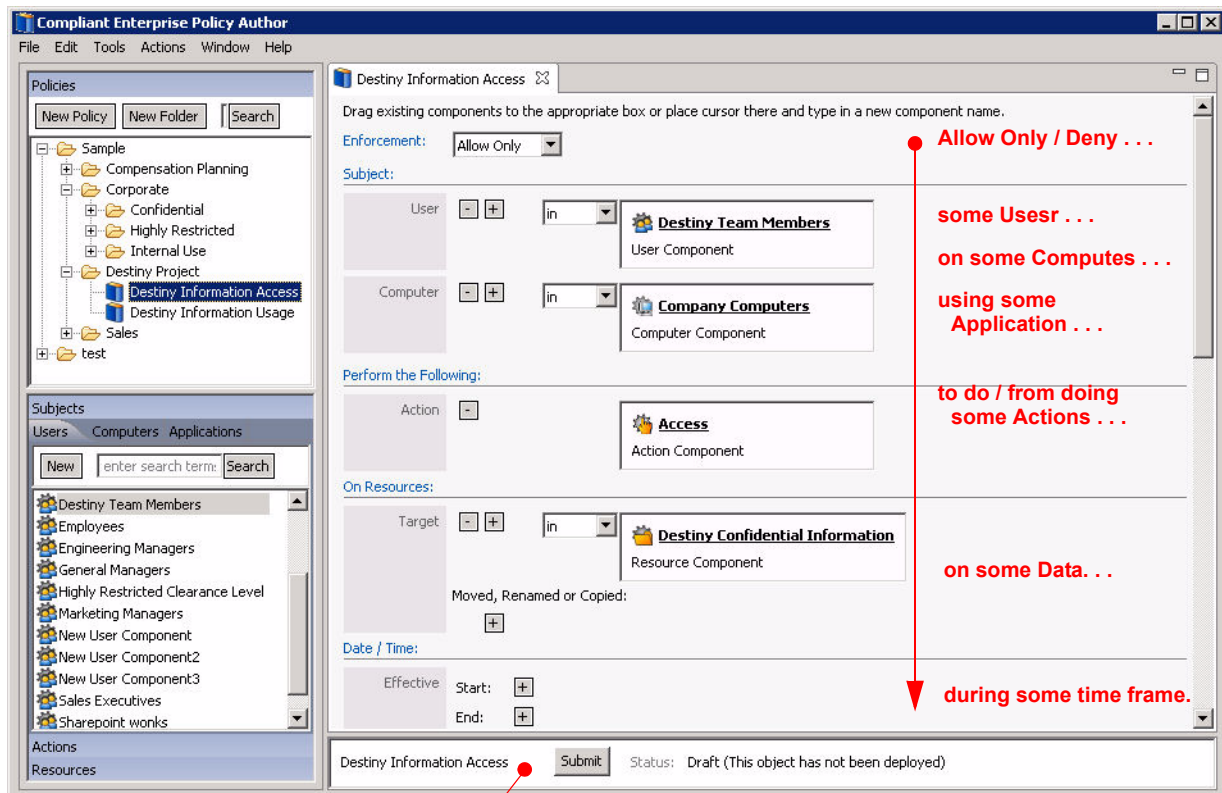
Once defined and deployed in Compliant Enterprise, policies are enforced continuously and automatically. Each enforcement event is caused by a single user performing a specific action on a particular document that is covered by a policy. For example, say a Human Resources assistant attempts to open a Word document that contains an offer letter. Because there is a policy in force allowing only the HR director access to this kind of document, Compliant Enterprise will enforce the policy by blocking access, displaying a message on the assistant's PC explaining that he is not authorized to open the file, and sending a notification e-mail to the HR director.

Editing Tools

A detailed description of how you use Policy Author to construct policies is presented in Chapter 3 of this manual. However, let's take a quick look at Policy Author's editing tools work.

As [Figure 1-5](#) shows, you use the policy editor pane, on the right, to define new policies and edit existing ones. This pane is divided into horizontal sections,

each of which provides tools for adding a specific “part of speech” of the policy: a subject, a verb, an object, or an adverb (context). To build these parts of speech, you use the components you have already defined—simply drag them up from the component panel at the lower left.



Policy lifecycle status bar

Figure 1-5: Policy Editing Controls

You work from the top of the pane down: first select the enforcement action—*Deny*, *Allow Only*, or *Monitor*—then define the user(s) the policy will apply to, then the applications, the computer resources, the actions themselves, the relevant document types, and a time frame. You do not need to include all these components in every policy; for example, many policies will not specify a time frame. Also, you may often specify *Any* for a given component: for instance, you may restrict certain users when they open certain documents *on any computer*, or block *any user* from access to certain documents between certain hours. (If you do not specify any component for an available section of the definition, it will have the effect of *Any*.)

The bottom area of the policy editor, *Obligations*, allows you to define what will happen when the policy is enforced: the event may be logged in the Activity Journal, a message may display to the person who violated the policy, an e-mail alert may be distributed to an administrator, or some combination of the three.

You can specify distinct sets of obligations that will occur when the policy blocks some action and/or when it allows the action.

Handling Policies

There are three main things you do with policies: **Construct** them in the first place (or edit existing ones), **Submit** them, and **Deploy** them. Let's take a quick look at each of these.

Constructing Policies

As we described above, you construct policies based on the available components already defined, combined using logical operators, and with optional context parameters added. A more detailed description of how you use Policy Author to construct policies is presented in Chapter 3 of this manual.

Submitting Policies

When you are finished defining a policy, click the button to Submit it for deployment. The submitted state simply means it is finished, and ready to be distributed throughout the system. In practice, there are two consequences:

- The policy is locked: all editing tools are disabled for it. To make further changes, you need to click the Modify button, which takes the policy out of submitted state.
- When you submit, Policy Author runs a dependency check to make sure that all the components required for this policy are valid and have been deployed. If a policy requires components that are not deployed yet, they must be deployed before or at the same time as the policy.

Deploying Policies

Deploying simply refers to the distribution of a finished policy throughout the system, so that it will be enforced on all specified users, computers, and so on. (Components are also deployed in this way, so that they are available for policies.) It is the point, so to speak, when the policy “goes live” in the network. You can deploy policies one at a time, or in batches. When you deploy them, you can specify whether they become active immediately, or at a specified time in the future. In addition, you can choose to deploy a policy to one or more specific enforcers, or you can use the *Autodeployment* feature. With Autodeployment, Compliant Enterprise automatically identifies all the enforcement points where the policy would be relevant, and deploys it to them.

Chapter 4 of this manual provides much more detail on submitting and deploying policies.

Components and Policies: Life Cycle

Policy Author provides a process that guides you through the life cycle of a policy or component, from initial creation through deployment and eventual retirement. Whenever you are working with a policy or component in the Policy Author editing pane, the current lifecycle stage of the policy or component is displayed in the control bar at the bottom of the window.

At any given time, every policy or policy component is in one of seven possible states, depending on what action you take while working in the Policy Author. The state of any component or policy is displayed in the lifecycle control bar, whenever the policy is open.

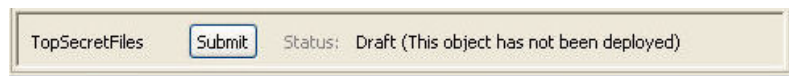


Table 1-2 provides descriptions of these states, listed in the sequence that represents their life cycle.

Table 1-2: Components and Policies: Life Cycle

State	Description
Draft	The object has just been created or is being modified.
Submitted for Deployment	Deployment has been requested for a new or modified object, as described in "Using Policies and Components" (page 59).
Pending Deployment	The request for deployment has been accepted and a time has been set for it to occur. After scheduling a deployment, you can also Cancel Scheduled Deployment; this returns the object to the Submitted state.
Deployed	Compliant Enterprise has finished deploying the object.
Submitted for Deactivation	The object has been marked for deactivation (see page 76).
Pending Deactivation	The request for deactivation has been accepted and a time has been set for it to occur. After scheduling a deactivation, you can also Cancel Scheduled Deactivation; this returns the object to the Submitted state.
Inactive	The object has been deactivated.

In the previous chapter, we provided a broad overview of Policy Author, and the two basic entities you use it to create: components and policies. In this chapter we take a closer look at the first of these: what components are, how you use Policy Author to build them, and how you use them to represent real entities in your physical environment.

The chapter is organized into the following sections:

- Defining New Components
- Defining Object Components ([page 20](#))
(Users, Computers, Applications, Documents, Portals)
- Defining Action Components ([page 41](#))

Defining New Components

As we have seen, you need to define components to represent various kinds of entities in your information environment. There are several times when you might want to define a new component:

- After setting up your Compliant Enterprise system, before constructing policies for the first time.
- When new classes of information or users come under the control of information policy.
- When a new policy requires a policy component that has not yet been created.
- When conditions at the organization change in any way that adds new items to be covered by information control policies. For example, if the company reorganizes and adds a new division, you might need a new policy component to represent the employees in that division.

When you are constructing a component, you do not need to save your work explicitly; it is automatically saved as you go. (However, you always have a standard Undo feature, if you need it.) If you are interrupted while working on a policy component, or you want to work on another task and return to constructing the policy component later, you can stop and continue the constructing process as desired; your work will be saved in Draft status. You can find the policy component later in the appropriate component panel.

Component Types

As we have seen, there are six categories of components:

- Users
- Applications
- Computers
- Actions
- Documents
- Portal Content

The components are organized into four panels in the lower left side of the Policy Author main window. Note that Users, Applications and Computers occupy separate tabs within the Subjects panel.

Defining Object Components

Regardless of which type of object component you want to define, the procedure is basically similar. For this reason, we'll describe in a general way the common procedure shared by all five types, then turn to examine the particulars of each type in more detail. Since action components are quite different, we will treat them as a separate topic.

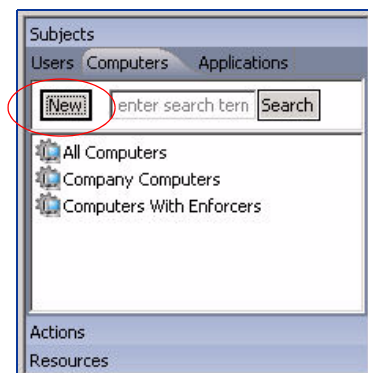
Our discussion of object components is divided into the following topics:

- Common Procedure
- Defining Document Components ([page 26](#))
- Defining User Components ([page 31](#))
- Defining Computer Components ([page 33](#))
- Defining Application Components ([page 34](#))
- Defining Portal Content Components ([page 37](#))

Common Procedure

To define any type of object component,

1. Expand the appropriate panel to display existing components of that type. For Subjects, click on the tab you want: Users, Computers or Applications.
2. Click the New button.
3. Type a name for the component, and click OK. Names cannot be longer than 128 characters, and may include any letters, numbers, and any special characters *except* square brackets. This opens the new component in the editing pane, as shown in [Figure 2-1](#).



4. In the Description field, provide a description of this component. Although this is not technically required, it is helpful to make this as descriptive as possible. It is also useful for recording the reasons for later changes to the definition of this component, if any.

About Membership and Properties

Besides a name and description, all object components are defined by one or more *memberships*, and one or more *properties*. (This is one of the characteristics that distinguish them from action components, which have neither membership nor properties.)

Membership identifies a superset or parent class of entities, of which this component is a subset. Properties are the metadata filters that define the subset. Thus: Component N = All members of parent class X that have properties A, B, and C. For example, a user component called Boston Staff = all members of parent class “All Employees” whose Status is “Current” and Location is “Boston.”

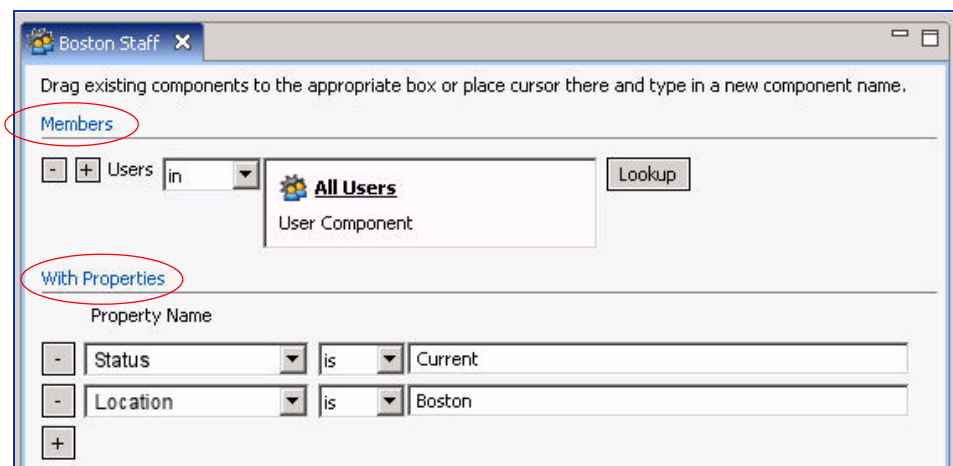


Figure 2-1: The Component Properties Pane

Defining Membership

The membership of a component will depend on what type of object it is. As a rule, it can be either an individual entity of the appropriate type (for example, an LDAP user group, for a user component), which you can either type in the field or select using the Lookup button; or an already defined component, which you can drag over from the component bin.

For the latter case, you will often use one of the predefined “All” components: All Users, All Computers, All Applications, or All Resources. If you like, however, you can also select a component you have already defined as the member of a new component, and further narrow it based on other properties.

If you do not define any membership for a component but only assign a property, the effect is the same as one of the “All” components; e.g., “All users with Location = Boston.”

Figure 2-2 describes the four Membership controls.

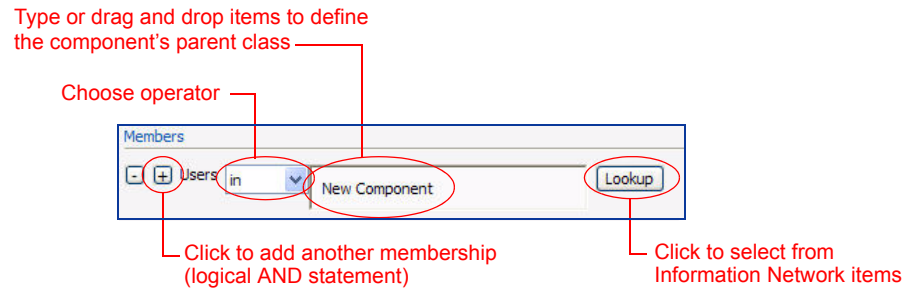



Figure 2-2: Object Component Properties: Membership Controls

To define a component’s membership,

1. In the Operator field, choose one of the available operators:
 - **In:** The component is represents a subset of the parent class or classes specified here.
 - **Not in:** The component represents entities outside the parent class or classes specified here.
2. To define the parent class, you have several alternatives:
 - Drag an existing component from the component bin on the left and drop it into this field. This parent component must be of the same type as the new component you are defining.
 - Type the name of an existing component.
 - Click the Lookup button to display all existing Information Network entities, and choose one or more from that list. (See "Using the Lookup Feature", below.)
 - Type the name of an Information Network entity. Note that if you manually type your membership criteria in this field, you must conform to the syntax requirements, as described below under "Membership Specification Syntax".
3. You have the option of leaving the Members field blank. If you do that, the membership will be All or Any—in our example, all users.

Multiple Memberships

You can specify as many memberships as you like for each component. To define multiple OR memberships, drag more than one parent entity into the Membership field. You do this to make the component's membership broader or more flexible; it represents a logical OR expression: "Employees *or* Contractors with the properties . . ."

To add multiple AND/BUT memberships, click the  icon to add a new field. You do this to make the component's membership more restrictive—for example, a group of users who are both interns and foreign nationals: "In Interns" *and* "In Foreign Persons." (A BUT expression is the same, but with a Not In membership—"In HR" *but* "Not In Contractors".) You can add as many membership condition fields to a component as you like; just bear in mind they are logical ANDs or BUTs—they will all have to hold true.

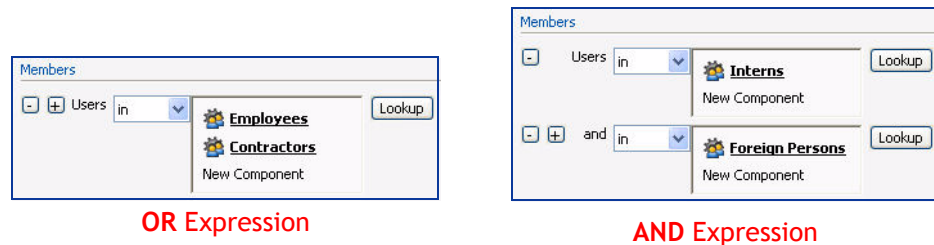


Figure 2-3: Multiple Memberships in Object Components

Using the Lookup Feature

If you click the Lookup button, a dialog box pops up, allowing you to choose users, desktops, applications, or documents, depending on which type of component you are defining. Items in the Users and Computers lists are imported from Active Directory; the icon next to each item identifies its source. For Document components, the lookup dialog contains a clickable folder and directory viewer that works like a standard browsing window. For Applications, the lookup window lists all applications that have been enrolled.

Note that for portal contents, the lookup feature supports only SharePoint portals. For other types, you must type the portal elements you want to add.

Membership Specification Syntax

Rather than use the lookup feature, you have the option of manually typing in the membership for a component. If you do that, you must take care to conform to the required syntax.

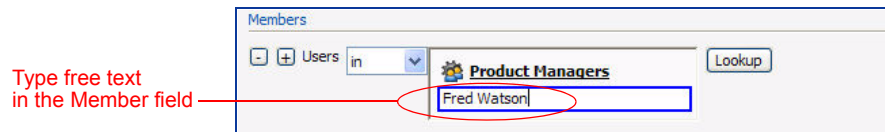


Figure 2-4: Object Components: Typing In Members

Bear in mind that the syntax rules differ depending on which type of component you are working on. User-, Desktop-, and Application-type components require either of the following:

- **Name** of an existing component of the same type. You can drag and drop a component from the component bin, or type the name directly.
- Unique **display name** of a user, host, or application. You can specify the name by clicking the Lookup button, or by typing it directly. If you type it, you must take care to enter the full name just as it was imported into your Information Network Directory.

Portal content components require the tree item representing the portal area you want to control. For more details, see the sections on individual component types, later in this chapter.

Object Component Properties

You use the Properties controls to specify the criteria that define the component as a subset of its parent class. Properties may be thought of as one or more filters that work on the metadata characteristics of the parent class. Specifically, you select a property from the list in the combo-box, type in a specific value for it, and combine them with an appropriate operator:

- String properties: *is* or *is not*
- Date properties: *on* or *after* or *before*
- Number properties: "=", ">", ">=", "<", "<=", or "!="

The available properties differ depending on which kind of component you are defining. For example, for a User component, you might specify *Job Title Is Vice President*; for a Computer component you might select *Site Is Not Boston*. Specific properties for each component type are described later in this chapter.

Different types of object components have different default properties, but for Users and Computers the list can be customized to include any properties defined in your Active Directory. If your administrator imports additional Active Directory attributes, they will display in the drop-down list in the Policy Author.

Note that Application-type components do not display any properties controls.

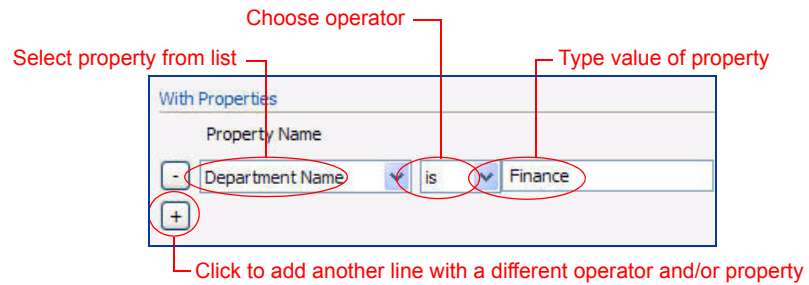


Figure 2-5: Object Component: Defining Properties

Defining Properties

To define an object component's properties:

1. Select the property name from the combo-box. This list displays all default properties, as well as any custom ones that may have been added.
2. Select an operator from the list in the combo-box. These will vary depending on what type of component you are defining.
3. Type the value to be compared to the current property value. If the expression is true for a particular person, file, etc., then it will be enforced. Note that you must have some knowledge of the actual values defined in your LDAP source, since this is free typing rather than a lookup function.
4. To verify that the definition you have created reflects the items you intended, you can use the Preview command in the Window menu to display the Preview pane, then click the Go button. The Preview pane will then display all of the individual entities that would be included in the component as it is presently defined.

This feature is very useful for testing or debugging components. If the list contains entities you did not expect or intend to include in the component you are defining, or does not contain entities you wanted to include, you know there is a problem with the definition of the component, and you can correct it before deployment.

Note that the Preview feature is only available for the four object component types—not for action components, or for policies.

Defining Document Components

Document-type components share all of the generic object component characteristics we discussed above. In addition, they have some special qualities as well, and they can be expanded to include custom attributes depending on your requirements.

Document Components: Memberships

To define the membership of a document component, you can specify the following entities:

- **An existing document component**, either dragged from the component bin or typed directly in the membership field;
- **A document name and/or path** typed directly in the membership field; or
- **A network path** selected the list displayed by the Lookup button, to the right of the Membership field.

You can use wildcard characters to the file name or the path. The table below explains all supported wildcards. With the exception of the standard * and **, all wildcards consist of a letter preceded by an escape character. Two escape characters are supported, a question mark and an exclamation point:

?<wildcard> means Matches the wildcard. It is useful in all kinds of general matching policies to existing files.

!<wildcard> means Does Not Match the wildcard. Beyond general matching purposes, it can be very useful in enforcing that names of newly-created files conform to required conventions.

You can also use the escape characters for themselves. That is, if you need to specify a file name that contains a literal question mark, you can represent it as “??”; for a literal exclamation point, use “!!”.

As [Table 2-1](#) shows, different letters represent different kinds of characters, and each has two versions: lower-case stands for a single character of a certain type, and uppercase stands for one or more of those characters. Note that this wildcard convention allows you to distinguish between upper- and lower-case characters. It is true that file names are not case-sensitive in Windows, but in a Linux environment they are, and the distinction is useful for that reason.

Table 2-1: Wildcards for Document-type Components

Wildcard Character	Description
*	Matches zero or more characters, <i>not</i> including the folder separator
**	Matches zero or more characters, including the folder separator
d	Matches any single-digit numeral, 0-9
D	Matches any numeral, one or more digits

Table 2-1: Wildcards for Document-type Components (Continued)

Wildcard Character	Description
a	Matches any single letter, either upper or lower case
A	Matches one or more letters, either upper or lower case
l	Matches any single lower-case letter
L	Matches one or more lower-case letters
u	Matches any single upper-case letter
U	Matches one or more upper-case letters
c	Matches any single alphanumeric character, excluding “\”
C	Matches one or more alphanumeric characters, excluding “\”
s	Matches a single space
S	Matches one or more spaces

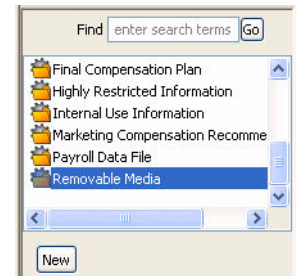
This wildcard convention enables you to define wildcard-based document components with a great deal of precision. The following are some examples:

- **c:\Program files*** matches any file in the c:\Program files\ folder.
- **c:\Program files**** matches any file in the c:\Program files\ folder or any of its subfolders.
- ****\.txt** matches any local file with a .txt extension, regardless of its place in the directory tree.
- ****\secret*** matches any local file with a name that starts with “secret”
- ****secret*** matches any local file with a name that contains the word “secret” anywhere. A more natural way to write the same thing is *****secret***, but the two expressions are equivalent.
- ****\Q?drevenue.doc** matches any local file with a name that starts with “Q”, ends with “revenue.doc”, and has exactly one character in between. For example, **\\documents\accounting\Q2revenue.doc** would match, while **\\documents\accounting\Q4-98revenue.doc** would not match.
- **\Personnel\?U?U?U*.*** matches any file in the Personnel directory that begins with three capital letters. For example, **OJAPerfReviews.doc** would match, but **OjaPerfRevs.doc** and **MO98PerfReviews.doc** would not.
- ****\?d?d*d*d*d*d*d/d*.*** matches any file that has a social security number anywhere in its name, either with hyphens or without.
- **\monthlySales\!d!d!d!d!d.xls** matches any file that does not conform to the required spreadsheet file-naming standard, **MM YYYY.xls**.
- **?c?c?c.*** matches any file with name containing exactly three characters of any kind, in either upper or lower case.

Lastly, it is important to note that these special wildcards are available only for document components—not for any other uses in Compliant Enterprise, such as defining subject components, or filtering queries in Reporter.

Removable Media

There is one predefined document component that has a special use: Removable Media. It can be dragged into policies to represent all peripheral devices such as thumb drives, other USB drives, and CD/DVD burners. It will cover any device that appears as a separate drive in the My Computer tree in Windows Explorer. It also can be used with subfolder paths and wildcard characters.



Special Keywords

For document components, two special keywords are available to help with commonly used components. To use these keywords, type them manually into the membership field, as shown in the example below. Be sure to include the square brackets. Note that the keywords are not case sensitive, but they may not contain spaces.

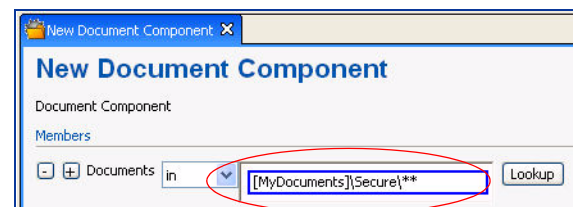


Figure 2-6: Document Component Special Keywords

Table 2-2 provides descriptions of these keywords and how you use them.

Table 2-2: Document Component Special Keywords

Keyword	Description
[MyDocuments]	<p>Creates a component representing the user's MyDocuments folder, wherever it happens to be physically mapped. You can specify subdirectories within My Documents, and/or specify document names using wildcard characters. For example:</p> <ul style="list-style-type: none"> [MyDocuments]** = All documents in the user's My Documents [MyDocuments]\Q4Sales*.xls = All Excel spreadsheets in the Q4 Sales subdirectory of My Documents <p>As a rule you will want to specify subdirectories, rather than have policies that control all files in users' MyDocuments folder.</p>
[MyDesktop]	<p>Creates a component representing the user's MyDesktop folder, wherever it happens to be physically mapped. Also can be used with subfolder paths and wildcard characters.</p>

Document Components: Default Properties

Document components have the following default properties. As with other object components, your Compliant Enterprise administrator can define additional custom properties; for more details, refer to the *Compliant Enterprise 2.0 Administrator's Guide* for more information. (If you will be working with custom properties, please note the caveat on [page 30](#).)

Table 2-3: Default Properties, Document Components

Property	Description	Allowed Operators
Access Date	The date and time the document was last opened.	on or after, before
Created Date	The date and time the document was created (if a document is a copy of another document, this property refers to the date and time the copy was created).	on or after, before
Directory	The full name of the directory where the document is located. Directory names are not case sensitive. For documents on network drives this property may have multiple values. For example, a document "\\docstore\\alldocs\\financial\\Q1results.xls" on the "docstore" server could be accessed as "X:\\Q1results.xls" if "\\docstore\\alldocs\\financial" is mounted as "X:", or as "Y:\\financial\\Q1results.xls" if "\\docstore\\alldocs" is mounted as "Y:". Allowed operations are "is" and "is not".	is, is not
Full Name	The name of the document, without the directory path but including the extension. For example, the name of "\\docstore\\alldocs\\financial\\Q1results.xls" is <i>Q1results.xls</i> . This value is not case-sensitive. You can use wildcard characters, both in the name and to express extensions. For example, the above file would be covered by a component with full name <i>Q1results*</i> or <i>Q?dresults.*</i>	is, is not
Include Only Directories	This is a special property that allows you to control how any policies with this component will handle directories. Valid values are Yes and No. For details, see below.	n/a
Modified Date	The date and time the document was last modified. Note that this includes modification of metadata as well as file content. For example, if you attach a custom property to a file, this date will be refreshed.	on or after, before
Owner	The owner of this file.	is, is not
Owner User Component	User component that contains the owner of this file.	is, is not
Owner LDAP Group	LDAP or Active Directory group that contains the owner of this file. This is the name of a group that has been enrolled with the Information Network Directory. The value should be specified as a principal name in the format company.com:Groups:Group Name.	is, is not
size	The size of the document, expressed in bytes.	=, !=, >, >=, <, <=
Type	The type of the document (i.e., its extension). This value is not case-sensitive, and you do not need to include the dot character. For example, for the file "\\docstore\\alldocs\\financial\\Q1results.xls" is <i>xls</i> or <i>XLS</i> . Note that for any component where you specified the Full Name property, you do not need to specify a Type, since the full name must contain an extension.	is, is not

About the Include Only Directories Property

All document components have a very special property, *Include Only Directories*, that allows you to control how policies with the component in them will handle directories. That is, it allows you to define policies controlling actions such as delete, copy, and move performed on directories themselves, as distinct from files.

This is a very useful feature because although a policy can control all files in a certain directory, it cannot stop a user from copying or moving a parent directory, thus circumventing the policy. This feature allows you to write policies that prevent this and similar problems.



Include Only has two available values, **Yes** and **No**. If you specify Include “Only Directories = Yes” for a component, only directories matched by the rest of the document component definition are considered. For example, if the location is set to `[MyDocuments]\financial**`, then all non-empty subfolders of the *financial* directory are matched by the component, but none of the individual files in them are matched. Similarly if you set the location to `[MyDocuments]\financial`, then just this single folder, rather than the files in it, would be matched.

If you specify Include Only Directories=No for a component, it has the opposite result: only files matched by the component definition are considered, and the directories will be ignored. This means that users will be able to move, copy, delete, etc. directories even if they are explicitly covered by the component definition. In effect, this property acts to override the explicit syntax of the component. For example, the location `[mydocuments]\financial**` would match all individual files at any level in the *financial* directory, but not any of the directories—policies would not prevent users from performing actions on directories.

If you do not select either—that is, do not assign this property to the component at all—then the component will not match directories except as explicitly required by the structure of its definition.

Note that empty folders will not be matched by the component no matter how it is set.

Practically speaking, you can use this property to create “directory components” rather than document ones—they represent directories only, not files. When used in a policy, they will control actions taken on folders, but not the individual files within them. Again, you would do this in order to prevent anyone from circumventing policies by manipulating parent directories.

Working with Custom Properties

If you intend to design policies that rely on document components defined by custom properties, it is very important to understand how they work in the context of policy enforcement. The critical principal is:

Any documents that do not have a custom property associated with them, will be considered as Not [x] by policies based on that custom property.

This point is particularly important if you are using a negative operator in your policy, such as when you assign a custom property Confidential, and write a policy based on “All Documents with [Confidential] [Is Not] [Yes].” In this case, *both* documents that have that property associated with them with a value other than Yes, *and* documents that do not have that custom property assigned to them at all, will belong to the component as defined. This is because an undefined value will still be evaluated as not equal to Yes.

To put this another way: according to Compliant Enterprise’s interpretation, not having a custom property at all is the same as having it but with a different value than the one specified in a policy. We can illustrate this with the following example. Say you want to assign a custom document property, *IsConfidential*, with Boolean values “Y” or “N”. In this case, there are three possible categories for all documents:

- **Category A:** Documents with this property assigned, with value “Y”
- **Category B:** Documents with this property assigned, with value “N”
- **Category C:** Documents without this property assigned at all

Table 2-4 shows all possible enforcement outcomes for this example.

Table 2-4: Custom Property Policy Enforcement

A policy based on all document components with:	Will be enforced for:
Property <i>IsConfidential</i> Is “Y”	Category A
Property <i>IsConfidential</i> Is Not “Y”	Category B and C
Property <i>IsConfidential</i> Is “N”	Category B
Property <i>IsConfidential</i> Is Not “N”	Category A and C

Defining User Components

As we mentioned, user-type components are organized on the Users tab of the Subjects component pane. They share all of the generic object component characteristics we discussed above. In addition, they have some special qualities as well.

User Components: Membership

To define the membership of a user component, you can specify the following entities:

- **An existing user component**, either dragged up from the component bin or typed manually
- **An enrolled LDAP user or group**, either selected using the search tool, or typed manually.

The Member Search Tool

Because your enrolled users and groups may number in the hundreds or even thousands, Policy Author provides the search tool shown in [Figure 2-6](#). This tool displays whenever you click the Members Lookup button.

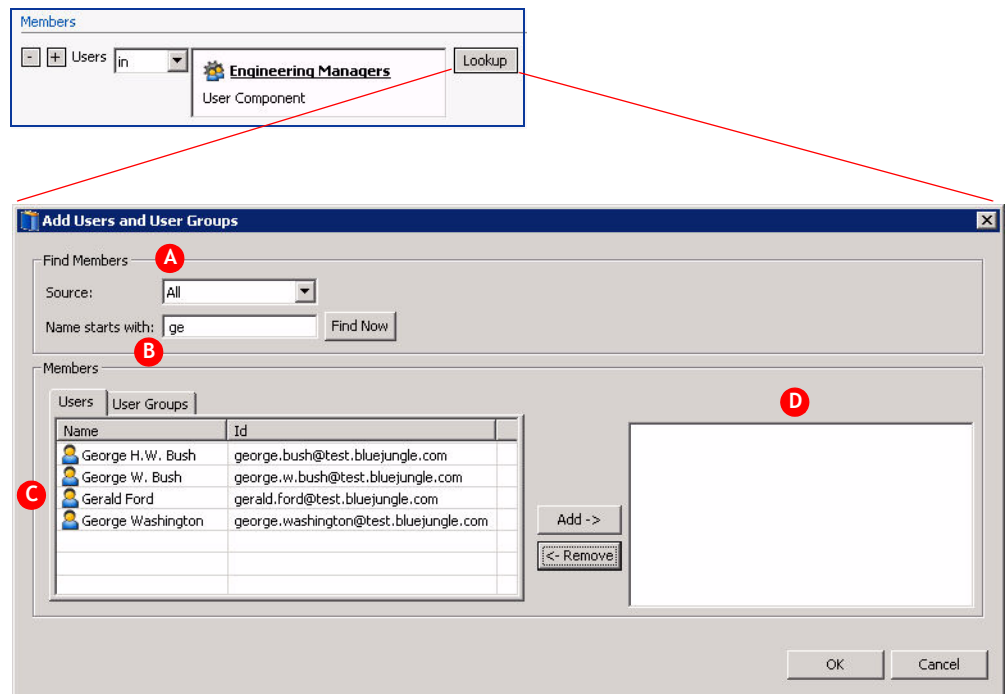


Figure 2-7: Using the Add Members Search Tool

To use this tool, first select an item from the Source combo-box (A), which displays a list of all enrollments performed so far. (If you do not know what enrollment you need, leave the default, All.) You can specify a string of initial characters in the Starts With field (B), to filter on users or groups in that way. (In the example, we filtered on all users whose names start with “ge”.) When you click the Find Now button, the results of the filter will display in the source list (C), which is organized in two tabs, for users and groups. Select the items you want from that list, then click the Add button to add them to the Members pane (D). You can use the Remove button to remove users or groups from the Members pane, if you make a mistake. Click OK when you have selected all the members you want.

User Components: Default Properties

[Table 2-5](#) displays the default properties of user components. As with other object components, your Compliant Enterprise administrator can define addi-

tional custom properties; for more details, refer to the *Compliant Enterprise 2.0 Administrator's Guide* for more information.

Table 2-5: User Components, Default Properties

Property	Description	Allowed Operators
Account Name	Account with which this user is associated.	is, is not
Company	Company with which this user is associated.	is, is not
Country Name	Country of this user.	is, is not
Department Name	Department with which this user is associated.	is, is not
First Name	This user's first name.	is, is not
Full Name	This user's whole name.	is, is not
ISO Country Code	The ISO code of this user's country.	is, is not
Last Name	This user's last name.	is, is not
Numeric Country Code	Numeric code of this user's country.	is, is not
Title	Title assigned to this user: Mr., Mrs., Ms., Dr., etc.	is, is not
User Principal Name	This user's LDAP principal name. This is a unique identifier for a user in Active Directory, typically in the form of an e-mail address; for example, jdoe@yourcompany.com.	is, is not

Defining Computer Components

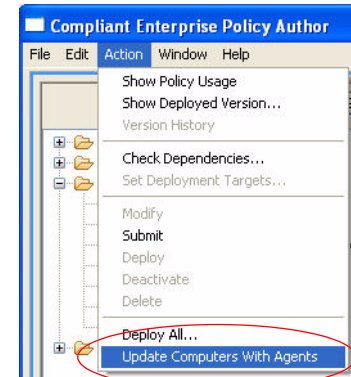
Computer-type components share all of the generic object component characteristics we discussed above. In addition, they have some special qualities as well.

Computer Components: Membership

To define the membership of a computer component, you can specify the following entities:

- **Any existing computer component**, either dragged up from the component bin, or typed manually
- **Any enrolled host(s) or host group(s)**, selected using the Member Search tool, which you open by clicking the Lookup button (see [page 32](#)).
- **All Computers with Agents**: This predefined component represents all desktop and laptop PCs where a Desktop Enforcer is installed. It can be very useful when you want to write a policy to ensure that only controlled computers are allowed to access sensitive documents. For example, you can use it to prevent anyone using a PC with no enforcer from opening or copying any files on your network's file servers, or from receiving any company files as e-mail attachments.

The actual membership of this component is a potentially long list of computer names. For convenience, the Actions menu has a special command, Update Computers with Agents, that lets you update the membership list all at once.



Computer Components: Default Properties

Table 2-5 displays the default properties of computer components. As with other object components, your Compliant Enterprise administrator can define additional custom properties; for more details, refer to the *Compliant Enterprise 2.0 Administrator's Guide* for more information.

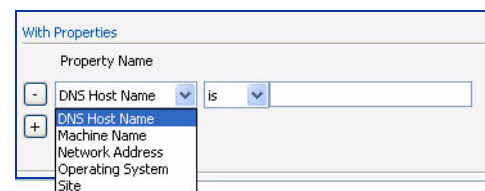


Table 2-6: Computer Component Properties

Property	Description	Allowed Operators
DNS Host	Name of this computer, which the DNS uses for resolving IP addresses in order to find this computer in the network.	is, is not
Machine Name	The permanent name assigned to this computer.	is, is not
Network Address	The network name assigned to the PC, which will be resolved by a DNS. You may also use the direct numeric IP representation.	is, is not
Operating System	Operating system of this computer.	is, is not
Site	The name of the site where the desktop is located. Sites are defined as one or more ranges of IP addresses. They must be enrolled manually, using the ImportLocations utility (see the <i>Compliant Enterprise 2.0 Administrator's Guide</i> for more information).	is, is not

Defining Application Components

Applications are unusual among object components in that they do not have any properties at all—just one or more memberships, which can be the following:

- An **existing application component**, either dragged up from the component bin or typed manually. May be one of two predefined components described below.
- An **application, selected from the Lookup list**. This list displays all applications currently enrolled into Compliant Enterprise.

When you define an application component, it is available for use as a part of the subject in policies. Bear in mind that applications are logically connected to the user component and the computer component; all these three work together as the single subject of the policy.

It is important to understand that application components refer specifically to applications, *not* to any file type or extension that may be associated with applications. For example, you might create an application component called *MS Word*. If you then use this component in a Deny policy, the policy will block users from using Word to open any documents using Word. It so happens that Word is associated with a .DOC file extension, but this does not mean that this policy applies to .DOC files generally. In fact, it will *not* block anyone from opening .DOC documents using another application, such as Notepad or Wordpad; the component refers to the application, not the file type.

Also, bear in mind that Compliant Enterprise considers different versions of an application as entirely independent of one another, and they must be enrolled separately. If you have more than one version of an application running in your environment and enrolled in Compliant Enterprise and you want to create application components to use them in policies, you will need to create a separate component for each, and make sure they are specified in the policies you write.

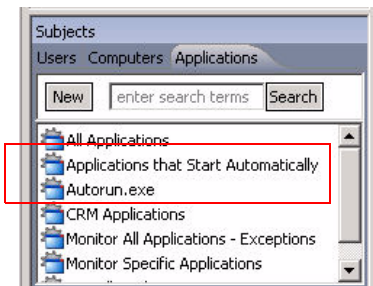
Predefined Application Components

Compliant Enterprise provides several predefined application components that allow you to easily set up commonly desired policies.

Autorun Prevention

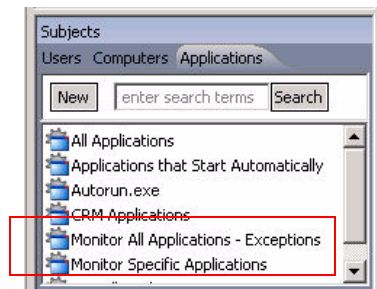
In many cases, security administrators may want to block any executables from automatically launching on network PCs. There are two predefined application components that can help with this:

- **Autorun.exe** represents the autorun executable. You can create a policy with this application as the subject, to prevent autorun from launching anywhere when a CD or other removable media is inserted or attached to a PC: “DENY All users, on all hosts, using AutoRun.exe, for all actions, on all documents.” This component is read-only.
- **Applications that Start Automatically** is a more generic component that you can define yourself. It contains the Autorun.exe component, but also allows you to add any other applications you have enrolled, that you want to be able to block.



Application Monitoring

Unless you specify otherwise, Compliant Enterprise applies each policy to all applications running on monitored PCs. However, there are two other predefined application-type components that allow you to define a list of applications you want to either exempt from monitoring, or explicitly include for monitoring.



- **Monitor All Applications—Exceptions:** If you want to exclude specific applications while applying to all others, add them to this list and deploy this component.
- **Monitor Specific Applications:** If you want your policies to monitor only specific applications and ignore all others, add them to this list and deploy this component.

They appear in the list in the application component bin, just like any other application component you might define. However, they work in a different way: they are global for all policies—that is, once defined and deployed, they will apply to all policies deployed in your network. They have the same effect as if you had included an application component in every deployed policy, so that it only works on some applications and ignores others.

You can define one or the other of these components (or neither of them), but you generally should not define both.

By default, Compliant Enterprise monitors all applications on desktops where Desktop Enforcers are running. If you do not define a list for either of these application components, this will be the expected behavior.

If you specify one or more applications to monitor, then all Desktop Enforcers will monitor *only those* for all deployed policies, and will ignore all others. This may be useful if you are launching a limited set of policies that you know will involve only a well-defined set of applications.

On the other hand, if you define one or more exceptions, Compliant Enterprise will monitor all applications on desktops where Desktop Enforcers are running, with the exception of the applications you specify. You may use this feature if you have some application—for example, an anti-virus security tool running on every desktop in your network—that you are sure you will never want any policy to block for any user under any circumstances.

If you do define and deploy both a Monitor Specific and an Exception application components, the latter will take priority over the former. That is, only those applications specified in the Monitor Specific list will be monitored, with the exception of any that are listed in the Exceptions list.

Note: You should not delete either of these predefined components for any reason. If you do delete one, you cannot simply recreate a new component with the same name; it will not function properly.

Windows Explorer

Note that for the purposes of excluding or including, Windows Explorer is considered an application if you are explicitly *excluding* applications, but not if you are explicitly *including* applications. That is, if you deploy the **Monitor Specific Applications** component, Windows Explorer will be monitored by default even if you do not include it in the list. In this case, it will be considered a part of the operating system rather than an application. As a result, Compliant Enterprise will enforce policies based on actions such as move, copy, delete, save, and so on.

On the other hand, if you deploy a **Monitor All Applications—Exceptions** component, you can list Explorer as an application. In this case, Compliant Enterprise treats Explorer as an application, and will not enforce policies governing any of Explorer's file handling functions. You should be very careful about doing this, since it will override any policies that include standard file-handling functions—copy, move, delete, save, etc.

Defining Portal Content Components

Portal components represent areas on a collaboration portal such as SharePoint. These resemble path-based document components in that they represent places on the portal where you may want to control access to specific users. You can use portal components to define controlled areas on a collaboration portal, just as you use document components to define controlled directories on the file system.

Specifically, portal content components can represent the following kinds of entities:

- portals
- sites
- web pages
- portlets (also called *Web parts*): libraries, lists, discussions, calendars and other structure elements smaller than sites or pages
- library items
- list items

Portal content components share all of the generic object component characteristics we discussed earlier in this chapter, but they also have some special characteristics.

Portal Content: Memberships

As with all object components, you have the option of defining portal content by typing an element name into the Membership field, dragging an existing component into that field, or using the SharePoint Lookup button. Note that this

lookup button supports only SharePoint portals; for other types you must either drag components, or type the memberships manually. As with document components, if you type the item manually you must take care to get the path and name exactly correctly, since there is no value check mechanism to detect errors. The following examples illustrate the required syntax for different types of SharePoint content:

- A site called “Alpha”:

```
//sharepoint2007.bluejungle.com/Alpha
```

- A sub-site called “abc” and all its children:

```
//sharepoint2007.bluejungle.com/Alpha/abc/**
```

- All items in a library “Docs 2”

```
SharePoint://sharepoint2007.bluejungle.com/Alpha/Docs 2/**
```

- A list “DealList”

```
SharePoint://sharepoint2007.bluejungle.com/Alpha/Lists/  
DealList
```

- A library Item “IND Design.doc”

```
SharePoint://sharepoint2007.bluejungle.com/Alpha/Docs 2/IND  
Design.doc
```

- A list item called “Test1”

```
SharePoint://sharepoint2007.bluejungle.com /Alpha/Lists/  
DealList/Test1
```

The SharePoint Lookup Button

If your resource is a SharePoint portal, you can use the Lookup button to open the Add Content window, as shown in [Figure 2-7](#) below. (This feature is not available for other types of portals.) At this point you have to connect to the SharePoint server where your resources reside. You can define resources based on any number of different SharePoint servers in your network. To connect, supply the following information:

- **Server:** The host name of the SharePoint server
- **User Name and Password:** A user name with privileges to access the SharePoint Server
- **Domain:** The domain where the SharePoint server is located.

Once you have provided the connection information, click the Display button. The left pane in this window lets you browse through the entire tree of items defined on the specified SharePoint server. From this tree, you can add as many items as you like to the component you are defining. To add items, select them in the tree, then click the Add button to add them to the Selection grid in the right-hand pane. As you work, bear in mind the following:

- You can add as many items as you want, but the Add button is disabled unless some item in the tree is selected.
- You can highlight any item and click the Remove button to remove it from the grid.
- For any item, you can click the Include All Children checkbox if you want all the sub-items included in the component. By default, this is not checked.

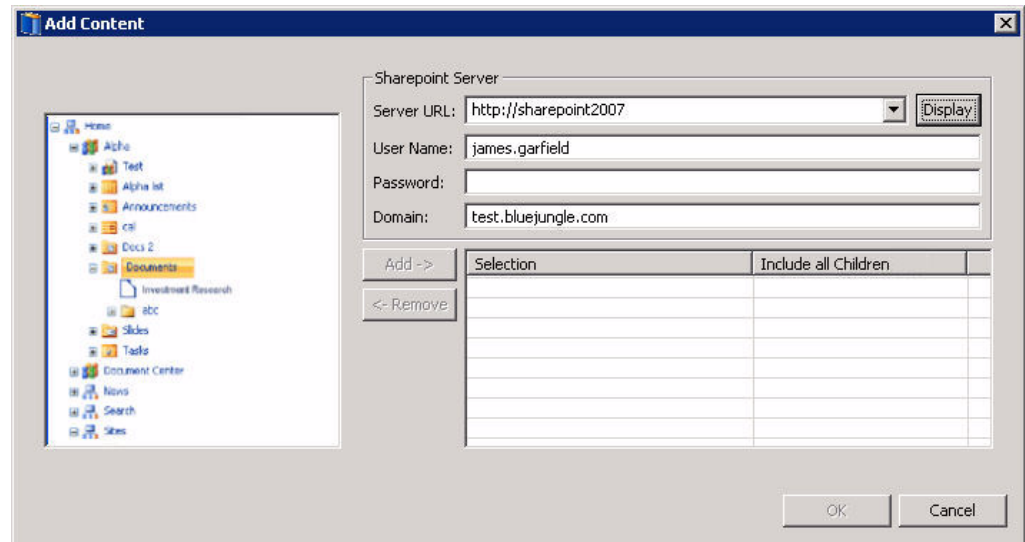


Figure 2-8: The SharePoint Add Content Window

- Once you add an item to the Add grid, it is disabled in the tree in the left-hand pane. If you select the Include All Children checkbox for an item, all its children are also disabled in the tree.
- When you have selected all the SharePoint content you want for the current component, click the OK button to return to the Policy Author main window, where the items have been added to the current policy. You now have the option of defining it further by filtering by one or more properties.

Portal Content: Default Properties

Table 2-7 lists the default properties that display in the Property Name list when you create a new portal content component. This list of properties is based on the attributes of content items in SharePoint; they may or may not be available for other collaboration portals. As with other object components, your Compliant Enterprise administrator can display additional custom properties; refer to the *Compliant Enterprise 2.0 Administrator's Guide* for more detailed information.

Note that the Property name field is an open text input; you can type the name of any property you like here, whether or not it displays in the drop-down list. The enforcers where this policy is deployed will then run a string match. If the property exists, the filter is enforced; if not, it is ignored.

For setting values for date-based properties, a calendar picker tool is provided.

Table 2-7: Portal Content Components: Default Properties

Property	Description	Allowed Operators
Created	Date when the content item was created.	'after', 'on or before'
Created by	User who initially created the content item.	'is', 'is not'
Description	Text description of the content item, if one has been supplied.	'matches', 'contains'
File Size	The file size of the content item.	'=', '>', '<', '>=', '<=', '!=', '=='
Modified	Date when the content item was most recently modified.	'after', 'on or before'
Modified by	User who most recently modified the content item.	'is', 'is not'
Name	The file name of the content element. Wildcards are supported.	'is', 'is not'
Sub-type	Sub-type of the content item; valid only if Type property is Portlet. Valid values include List and Library. Not mandatory; if type is portlet and no subtype is specified, the component will include both subtypes.	'is', 'is not'
Title	The title string that has been applied to the content item. Wildcards are supported.	'is', 'is not'
Type	Type of content item. Valid values include: <ul style="list-style-type: none"> • Portal • Site • Page • Portlet [i.e. Library or List] • Item [i.e. Library Item or List Item] 	'is', 'is not'

Defining Action Components

The procedure for defining action-type components is quite different than for object components, since they are much more simply defined. In fact they don't have either properties or memberships at all, as object components do. Rather, each action component consists of a name, a description, and one or more *basic actions* that you can combine to meet your requirements.

To create a new action component, select the Action Components panel. This will display a list of all currently defined action components.

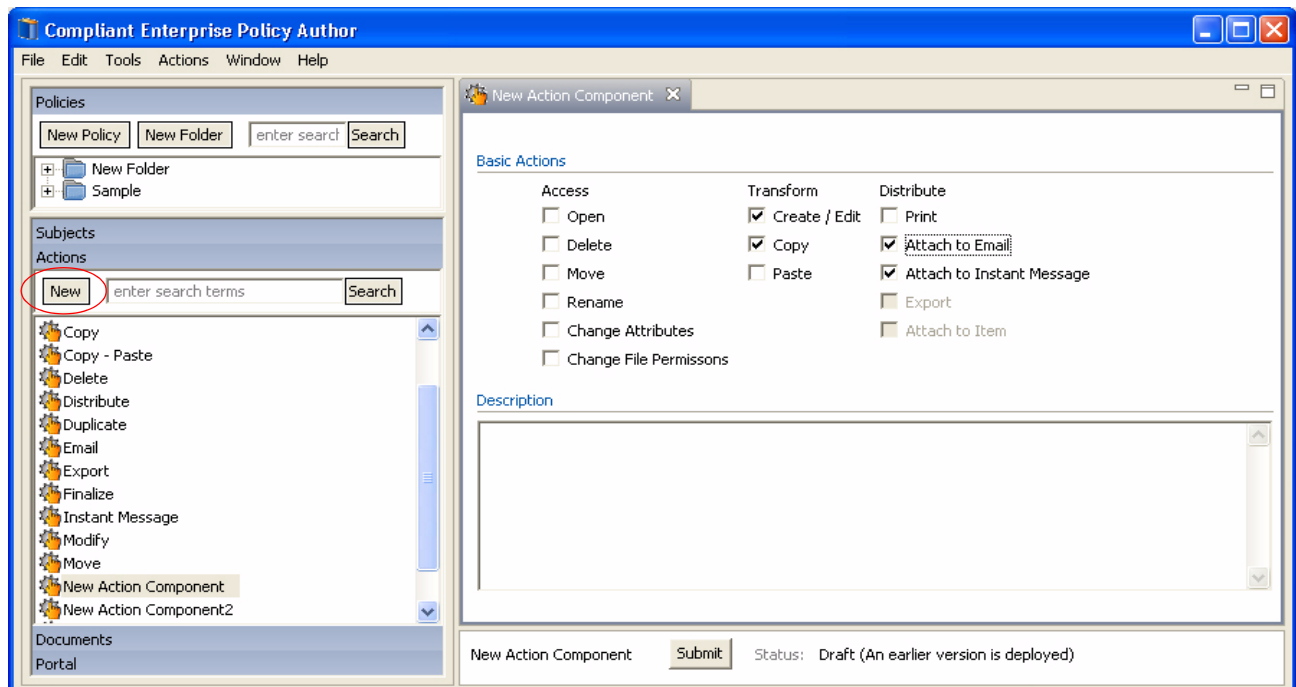


Figure 2-9: Defining an Action Component

1. Click the New button, and assign a name to the new action component. Since components often combine more than one basic action, it is important to assign names that are as specific and descriptive as possible.
2. Click OK to display the action editor controls, as shown at right.
3. Check the basic action checkboxes to combine as many of these as you need to define the new component; it could consist of one basic action, or several.
4. In the Description pane, provide a description of the action. This is helpful to remind yourself or other users of the purpose of combining these actions into a single component.
5. Click the Submit button, and close the tab for this new action.

Basic Actions As [Figure 2-9](#) shows, the basic actions available for building action components are organized into three columns in the pane, to reflect functional categories: Access actions, Transform actions, and Distribute actions.

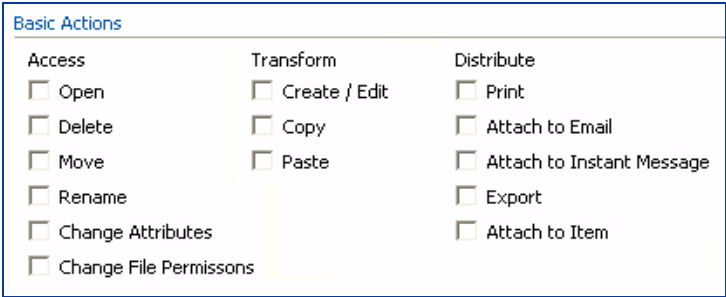


Figure 2-10: All Basic Actions

[Table 2-8](#) provides descriptions of all available basic actions. You should review these descriptions carefully, since many of them cover a number of actions that are related in ways that may not be self-evident at first.

Table 2-8: All Basic Actions

	Basic Action	Description
Access	Open	Open a file or portal item for viewing. Specific actions include: <ul style="list-style-type: none">• Open a file• Access a site, workspace, structure, list, or library item on a portal Enabled for both File System and Portal action components.
	Delete	Permanently remove a file or portal item from storage. Specific actions include: <ul style="list-style-type: none">• Delete a file• Delete a site, workspace, structure, list, or library item Enabled for both File System and Portal action components.
	Move	Delete a file from its current storage location and place it in a different location. Specific actions include: <ul style="list-style-type: none">• Move a file, within the file system• Move a site, workspace, structure, list, or library item, within the portal server Enabled for both File System and Portal action components.
	Rename	Assign a new name or extension to a file.
	Change Attributes	Modify file attributes, such as whether the file is read-only or hidden, using the Windows file property dialog. Disabled for Portal action components.
	Change File Permissions	Change permissions granted to users of a file, using the Windows security dialog. Disabled for Portal action components.

Table 2-8: All Basic Actions (Continued)

	Basic Action	Description
Transform	Create/Edit	<p>Create a new file or item, rename, or change the contents of an existing one. Specific actions include:</p> <ul style="list-style-type: none"> • Create a new document • Edit the content of an existing document • Overwrite an existing document by any means: copying a file with the same name from another location, renaming another file, using the Save As or Export command in another file, etc. • Create a portal site, page, list, library, library item, or column • Edit a portal site, page, list, library, library item, or column, by any means (in datasheet, in spreadsheet, etc.) • Upload any item to a portal <p>Enabled for both File System and Portal action components.</p>
	Copy	<p>Make a duplicate of a file or portal item. Specific actions include:</p> <ul style="list-style-type: none"> • Copy a file • Embed a file into another file, including attaching to e-mail from inside an application • Copy or move portal content within the portal server <p>Enabled for both File System and Portal action components.</p>
	Paste	<p>Copy or cut a portion of the file's contents and paste it to a location outside the file. Paste actions are enforced when a user selects the content and tries to copy it. That is, the user is prevented not from pasting content per se, but from even copying it to the clipboard.</p> <p>Disabled for Portal action components.</p> <p><i>Note:</i> Policies involving Paste actions do not support e-mail notifications or logging obligations. However, on-screen user notification obligations are supported.</p>
Distribute	Print	Print any file or portal content, either to a printer device for hard copy, or an output file of any format. Enabled for both File System and Portal action components.
	Attach to E-mail	Attach a file to an outgoing message in Microsoft Outlook. Note that in MS Word attaching to e-mail using the File, Send To command is considered a type of Embed, and will be governed by policies containing a Copy/Embed action. Disabled for Portal action components.
	Attach to Instant Message	Attach a file to an instant message. Disabled for Portal action components.
	Export	<p>Export a portal item. Specific actions include:</p> <ul style="list-style-type: none"> • Export list to datasheet • Export library to spreadsheet <p>Disabled for File System action components.</p>
	Attach to Item	Attach one portal list item to another. Disabled for File System action components.

File System vs. Portal Actions

As the table shows, some actions logically can be used in portal-based policies only, some can be used in file-system based policies only, and some can be used in both. The table below summarizes these restrictions.

As you select basic actions to define an action component, Policy Author does not allow you to combine portal-only actions and file system-only actions in a single component; as soon as you select a basic action of one type, all basic actions of the other type are disabled. Basic actions relevant to both types (the middle row in the table) may be combined with either in a single component.

Type	Basic Actions
File System Only	<ul style="list-style-type: none">• Change Attributes• Change Permissions• Attach to E-mail• Attach to IM• Paste
File System or Portal	<ul style="list-style-type: none">• Read• Delete• Move• Create/Edit• Copy• Print
Portal Only	<ul style="list-style-type: none">• Export• Attach to Item

Constructing Policies

In the previous chapter, we explained how to define various kinds of policy components. As we know, these are not an end in themselves, but rather are only useful as building blocks for your actual policies, which you also construct in Policy Author. The procedure for doing that is the subject of the present chapter. It is organized into the following sections:

- Creating a New Policy ([page 45](#))
- Using Obligations ([page 51](#))
- Organizing Policies ([page 56](#))

Creating a New Policy

When you are constructing a policy, your work is automatically saved as you go; you do not need to explicitly save your work. (However, you do have a standard Undo feature available, whenever you need it.) If you are interrupted while working on a policy, or you want to work on another task and return to authoring the policy later, you can stop and continue the authoring process as desired; your work will be saved as a draft. You can find the policy later by using the navigation pane.

To create a new policy,

1. In the policy tree, select the folder in which you want the new policy to be stored, then click the New Policy button. You can also right-click and select Add Policy.
2. Type a name for the new policy, and click OK.
3. In the enforcement type field at the top of the Subject area, choose one of the following to determine what Compliant Enterprise will do if the situation described in the policy occurs:
 - **Deny:** do not permit the listed Subjects to perform the task, but allow all others to do so.
 - **Allow Only:** permit the listed Subjects to perform the task specified in the rest of the policy, and prevent all others from doing so.
 - **Monitor:** always permit the specified action, and perform the specified obligations whenever the action occurs.

Constructing a policy involves supplying values in three sets of controls: the Subject, Object, and Action controls. These correspond to the required grammatical components of the ACPL statement that underlies each policy you construct.

Defining a Policy Subject

As [Figure 3-1](#) shows, you can use the controls in the Subject group to define the subject of your policy—that is, a defined category of entities whose behavior the policy will govern. It is important to understand that although *each* of the three components in the subject—user, application, and computer—may contain several specifications linked with either AND or OR operators, the three of them work together as a single whole. They are connected with logical AND operators, and must all be true for the policy to evaluate to true. For example, a policy whose subject specifies all members of the Marketing department **OR** the Sales department using laptop computers to open a document in MS Word, will not stop Marketing or Sales users on laptops from opening it in Notepad, or Marketing or Sales users on desktops from opening it in Word.

You can either fill in these controls, or leave one or more of them blank. If you leave a control blank, the effect is to apply the policy to all users, all computers, or all applications, respectively. (In practice, it is quite common to design policies that do not specify computer or application components; policies that do not specify any user are less common.)



The first line in User is already provided for you. To add another line to the Subject definition, click the Plus icon next to User, Computer, or Application.

1. From the drop-down list, choose one of the following operators:

- **In:** The policy subject includes any entity (for example, any user) that you specify in the editing box to the right of the drop-down list.
- **Not in:** The subject includes only those entities that are not included in the editing box.

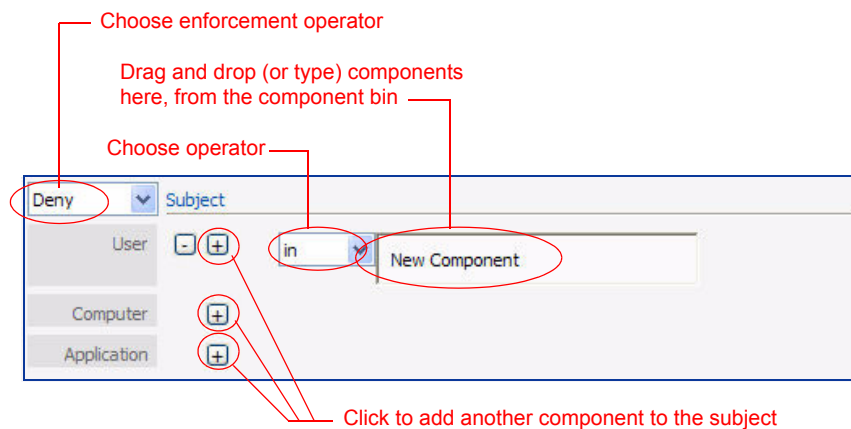


Figure 3-1: Defining the Policy Subject

- Click in the New Component area to the right of the drop-down list, and type the name of a policy component of the correct type (user, computer, or application). You can also click Components in the navigation pane and drag and drop a policy component name from there into the Subjects editing area.

For example, when filling in the User definition, search Components in the navigation pane to find a policy component that represents users. When filling in the Computers definition, use any of your computer components or use the predefined component Computers with Agents.

If you are not sure what value to enter in Subjects, use the Find field in the navigation pane.

If you type a name that does not match any existing policy component, a dialog box appears giving you the opportunity to create a new policy component with that name. Bear in mind, this creates a blank component with the specified name, but you will have to open it later to supply the properties that will actually define the component.

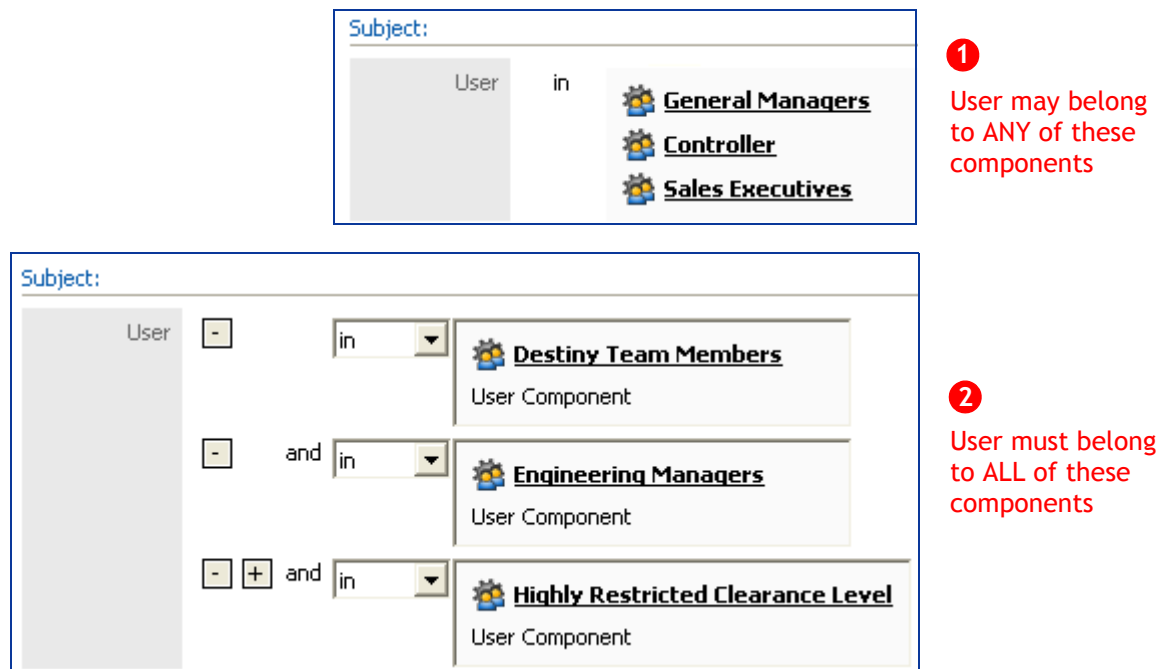


Figure 3-2: Multiple Components in Subjects

You can add more than one component to any of the fields in the Subject; when you do, an OR operator is understood (the first example in [Figure 3-2](#)). If you mean to use an AND operator, click the Plus icon again next to User, Computer, or Application to add a new line. You do this when you need to specify more than one component, separated by an AND or BUT logical operator: “in both

class A and class B” (the second example in [Figure 3-2](#)), or “in class X but not in class Y.” For example, you might want Users to include all employees in the Human Resources department, excluding contractors.

Defining a Policy Action

Just as every English sentence requires a verb, every policy requires an action. As [Figure 3-1](#) shows, you can use the controls in the Action group to define what each policy’s action is and how it will work.

1. In the Action controls group, drag an action component from the components bin into the Name field, or type a name manually. If you do not specify any action component, the policy will be in effect for all actions.
2. In the On Documents field, specify which documents are covered by the policy. The editing controls in this group are similar to those you used to define the Subject. If you leave these controls blank, the policy will be in effect for every document in your organization.
3. Choose an operator referring to the policy object: In or Not In.


Defining a Policy Object

Each policy must have an object, which is simply the resource component on which the policy will act whenever it is enforced. To do this, type the names of one or more policy components that represent files or other resources. As with Subject contents, you can drag and drop components from the navigation pane into the Resources area.

Target Location

For all actions that logically involve a From location and a To location—for instance Move, Rename, or Copy—you can specify a target location for the action of each policy. This control is used to specify a distinct location where you wish to either permit or prohibit copying or moving all resources covered by the policy. If you do not specify a target, the policy will apply whenever the specified From action occurs, without regard to a target location.

To specify a target location,

1. Click the Plus icon. 
2. In the drop-down list, select an operator: Into or Outside. Use Into when you want to specify a particular destination; for example, when the specified users attempt to copy a file into a particular directory. Use Outside when you want to specify anywhere but a particular destination.
3. In the editing area, type or drag and drop file names, directories, storage locations (file servers and folders), or document or portal content components.

For example, you can drag and drop the predefined Removable Media component into the target area to prevent copying or moving documents to devices such as USB thumb drives and CD burners.

Define a Time Context

In the Date and Time section, you can place once-only or recurring time restrictions on the enforcement of the policy. Once-only restrictions are defined in the Effective area; for example, you can specify a date range to enforce the policy for one month only, or specify a Start date if you want to make a policy effective starting on a specific date (leave the End date blank to continue enforcement permanently). Recurring restrictions are defined in the Schedule area; for example, specify a day of the week to enforce the policy once a week, such as to allow editing of certain files only during regularly scheduled maintenance.

Figure 3-3: Setting Time Context

Note that time is evaluated based on the clock running on the Policy Server (a key Control Center component), which is affected by the time zone where the Policy Server host is located. For example, if you are using Policy Author in your office in California, and the Policy Server is installed on a host at company headquarters in Boston, the time you specify will be interpreted as Boston time.

If you leave these controls blank, the actions specified in the policy are always restricted, starting when the policy is deployed.

Defining Obligations

In the last section of controls, Obligations, you can specify additional actions Compliant Enterprise will take when the situation described in the policy occurs. The obligation can be a message to the user, a log message written to the Activity Journal, or an e-mail. For more details, see “Using Obligations” on [page 51](#).

To set up the obligations for a policy, perform the following steps.

1. First, look for the label that describes the policy effect that you want to trigger the obligations: On Deny, or On Allow/Monitor as shown in [Figure 3-4](#). You can set up obligations for several different

effects if desired; for example, log an event and send an e-mail on Deny, but merely log an event on Allow.

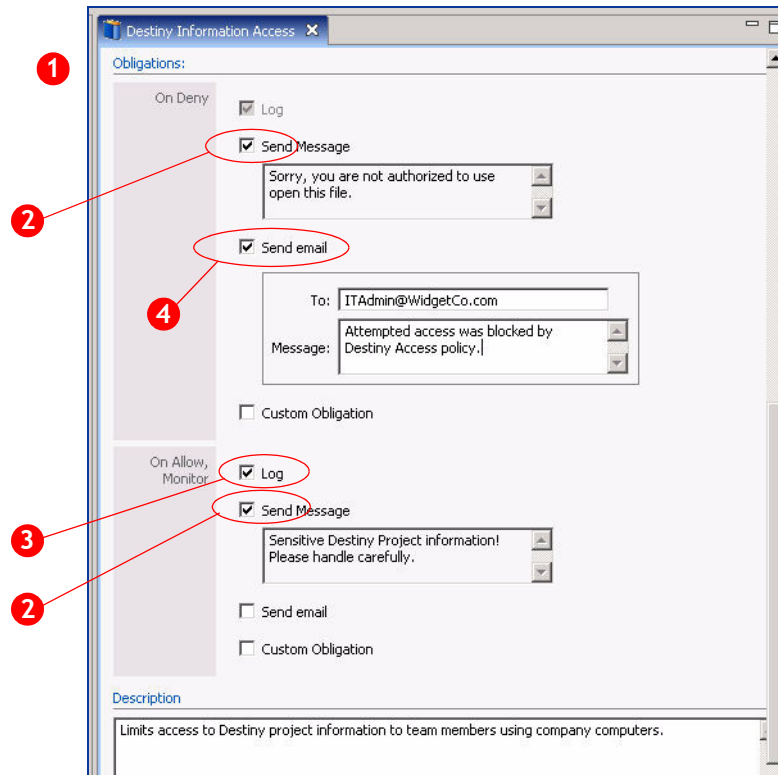


Figure 3-4: Defining Policy Obligations

2. To display a message on the subject's computer screen whenever the policy action is blocked, type the text in the On Deny Message field. To define a message that shows even if the action is allowed, type it in the On Allow/Monitor Message field. These message will appear only on computers where the Compliance Notifications feature has been enabled during installation of the policy enforcer software. You can specify messages for both On Deny and On Allow, if you wish.
3. To log the fact that this policy was triggered even for Allow, click the Log checkbox. (For On Deny obligations logging is always enabled, and this box cannot be unchecked.)
4. To send a notification e-mail: click the Send E-mail checkbox, type the recipient's e-mail address in the To field, and type the body of the e-mail in the Message field. To specify more than one recipient, separate them by commas.

The From address of these e-mails can be defined on the Control Center side, and will be the same for all policies. It is configured by your administrator.

Add a Description

In the Description field, type a meaningful description of the policy. You might use language from your company's policy manual, if appropriate. Although this field is optional, it is very helpful if several users are collaborating in constructing policies.

When you are ready to publish the new policy, click Submit to submit it for deployment. The procedure for deploying policies is described in Chapter 4.

Using Obligations

As we have just seen, for every policy you define in Policy Author you can specify various *obligations*, which are actions Compliant Enterprise will perform every time the policy is enforced. It is very important to understand that by “enforced” we mean not that some action was denied, necessarily, or that some action simply did happen somewhere in the network. Rather, it means that some action occurred that matched the context specified by *the entire definition* of the policy: the subject (user + computer + application) and the action, and the resource, and the time context, all evaluated to true; and so the specified effect (allow or deny) resulted. For the purposes of enforcement (and thus obligations), the important thing is not what that effect was, but whether or not the entire policy evaluated to true or not.

This distinction becomes particularly important in connection with Monitor policies and On Allow/Monitor obligations. For example, consider a Monitor policy that does not block Sales Managers from opening files on a certain SharePoint site, but does display a warning notification about the sensitivity of the data whenever they do. That is, the policy is enforced when Sales Managers access the site, even though it does not block them. By contrast, when other authorized users—say, the Sales VP—access the same site they are also not blocked either, but this does not mean the policy was enforced. The effect is the same, but in the VP's case the policy was not relevant since its entire definition did not evaluate to true: the action and resource did, but the user did not. For this reason, the VP does not see the notification message, and the event is not be logged in connection with this policy.

Obligation Categories

Practically speaking, obligations fall into three groups: event logging, notifications, and custom obligations. Let's take a closer look at the first two categories.

Note: Policies involving Paste actions do not support e-mail notifications or logging obligations. However, they do trigger on-screen user notification obligations, and all other actions support all types of notifications.

Enforcement Event Logging

Event logging means that details about an enforcement event are saved in the Activity Journal, where they are available for analysis in detailed reports. Technically, events are buffered in a temporary log file on the local host, and periodically copied to the Activity Journal in batches. The local buffer file is tamper-

resistant, and cannot be read, opened, changed or deleted by any user while the Enforcer is running.

Bear in mind that the Activity Journal gathers information about document access and use of all kinds throughout the network—not just those events connected with policy enforcement. This is what provides the platform’s valuable information use auditing features. However, policy-related events are our focus in this chapter.

Notifications

Compliant Enterprise can send two types of notifications: desktop notifications, which are text-balloon reminders that pop up on the PC where the policy was enforced, and e-mail notifications sent to one or more specified administrators, describing the incident. You can configure any policy to send either type, or both.

Bear in mind that you can only configure a policy to display desktop notifications on PCs where the Enforcer was installed with the Enforcement Notifications option enabled. This is generally the preferred option, and is the default whenever you do a complete installation. In addition, the subject who is using the PC has the option of disabling these text balloons by right-clicking on the CE icon and un-selecting Display Notifications.

E-mail notifications are a powerful tool enabling system administrators, information security personnel, or any other designated persons to know immediately whenever anyone in the network has performed some action that required a rule to be enforced—tried to e-mail a restricted internal document, say, or to print an “eyes-only” report.

Configuring E-Mail Notifications

As [Figure 3-5](#) shows, you enable this feature by adding an E-Mail Obligation to the policy itself. You can specify a different alert for On Deny, On Allow, and Monitor. These obligations have two E-Mail Details properties that control how they work (the fields only display when you check the Send E-Mail option):

- The **To:** field, where you supply the e-mail address to which the notification message will be sent. You can specify any number of addresses in this field; just use a comma to delimit them.
- The **Message:** field, where you supply a message that will go into the body text of the notification e-mail. In addition to whatever you specify here, the body of each e-mail will also include details about that instance of enforcement: the user, the location, the document name, and so on.

Note that the top Message field in both On Deny and On Allow is where you customize the text displayed in the pop-up balloon that appears on a subject’s desktop whenever the current policy is enforced with a Deny. It should not be confused with the Message field under E-Mail Details, which is for the e-mail body text.

As the figure shows, you can specify one message to be displayed to users who are allowed to perform the specified actions, and a different one to those who are denied. (The display of On Allow messages can be disabled at the desktop enforcer level, if the end-user wishes.)

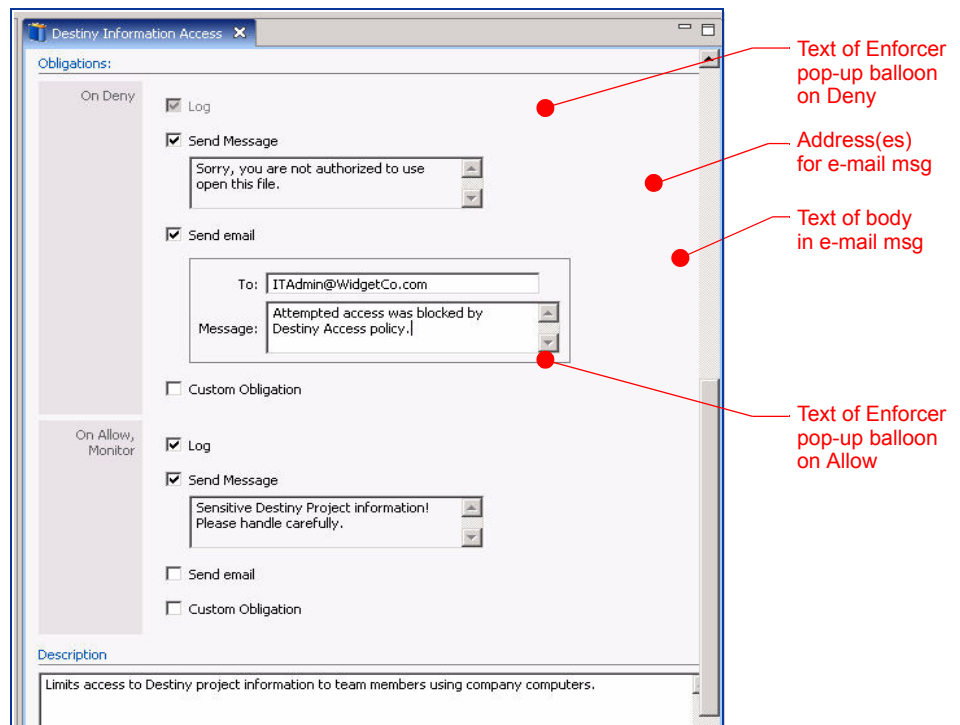


Figure 3-5: Configuring Administrator Notifications

Besides the messages you specify here, all notification e-mails will automatically include the following information:

- User who triggered the policy enforcement
- Host where user was working
- Application involved
- Source file being used or accessed
- Action attempted on the file
- Effect (allow/deny/when)
- Policy name
- Policy description
- Date and Time of enforcement

The From Address

By default, the From address field in all e-mail notifications will contain a dummy value that will not be valid. Your Compliant Enterprise administrator can replace this with a real address by manually editing the Control Center configuration file; for details, see the Configuration Tools chapter of the *Compliant Enterprise 2.0 Administrator's Guide*. Note that currently you can only define one From address, which will be included in all e-mails for all policies.

Monitor Policies

Recall that there are three available effects for any policy:

- **Deny:** Prevent the activity defined by the policy from happening;
- **Allow Only:** Permit the activity defined by the policy to happen, but only for that specifically defined case;
- **Monitor:** Do not restrict the activity defined by the policy at all.

The third effect, Monitor, is designed to support “audit policies”—that is, cases where you do not need to block a certain action, but you do want to track the details about it, every time it occurs. This is a powerful tool for any kind of auditing: either during initial network auditing to discover what kinds of policies you need to implement, or in ex-post facto cases such as proving that certain employees did or did not use specific documents in certain ways.



In many cases you may want to define pure notification policies with a logging obligation only, since real-time e-mails may not be required for the purposes of establishing an audit trail. Similarly, in such cases you likely will not need a notification message to pop up on the subject's PC. However, all three of these obligation options are available in any case.

Note that you can use Monitor policies in creative ways, such as to remind users about policies, approaching deadlines, or any other matter that may or may not be connected with resource use and access. Especially when used in policies with time-based contexts, this feature offers great flexibility in sending automatic notifications in response to specific user actions.

Again, it is important to bear in mind that just because a user is allowed to open or use a resource that is the subject of some monitor policy, that does not necessarily mean the policy was enforced in that case. For a policy to be enforced, *all* of its parts must evaluate to true: subject, action, resource, and time context.

Custom Obligations

The Custom Obligation controls allow you to specify one or more custom obligations for any policy. Custom obligations may be any kind of executable code that can be triggered by the enforcement of a policy. Your Compliant Enterprise administrator can create them as independent executable files; you can then specify them in the Name field in any policy you want to use them. (Note that this field does not display until the Custom Obligation box is checked.) When the policy is enforced, the executable will be invoked and the resulting behavior will occur.

Here are some examples of custom obligations:

- **User Prompt:** A policy prevents some user from accessing the files in a directory, then pops up a web page describing the security policy that is involved, and offering the user an e-mail link to request access if he think the policy has been incorrectly applied.
- **Removable Media Encryption:** A policy allows some users to copy sensitive data to Removable Media, but automatically encrypts the data before copying.
- **Call Pager:** A policy denies some users access to project files, and sends a call to the project owner's pager to notify him of any attempts to open the files.



The Name field does not offer a combo-box list; you must know the display name of any custom obligation you want to use, and type it exactly.



You can use the Plus sign icon to add more than one custom obligation to a policy. Use the minus sign icon to remove custom obligations from a policy, if necessary.

Custom obligations are created and configured entirely outside Policy Author. For details on creating, configuring and using custom obligations, refer to the *Compliant Enterprise 2.0 Administrator's Guide*.

About Enforcement Logging

As we have mentioned, enforcement logging can be configured as an obligation, associated with each individual policy you construct. By default it is enabled for On Deny for all policies and cannot be disabled, since in practice you will want every instance of a Deny enforcement to be logged.

By default, no obligations are enabled for On Allow and Monitor obligations for all policies. You can enable one or all for any given policy, if you wish.

Logging vs. E-Mail

Take care not to enable e-mail notifications for policies unless there is a real need to do so. (This is a commonsense point, simply in order to reduce e-mail traffic and clutter in administrator mailboxes.) For the same reason, you should consider how often a policy is likely to be enforced, before adding e-mail notifications to it.

Logging is another matter; it should not present a problem to enable logging obligations for every policy you construct. These are stored in the Activity Journal, which should have adequate storage capacity for extensive logging. In fact, you are very unlikely ever to define a policy whose enforcement you do *not* want recorded in the Activity Journal.

Organizing Policies

You can organize policies into folders in the navigation pane. This is useful for managing large numbers of policies.

To create a folder:

1. In the Policies area in the navigation pane, if you want to create a folder inside an existing folder, find and open the existing folder.
2. Click New Folder.
3. Type a name for the folder.
4. Click OK.

Using Folders



The folders work like standard Windows folders in Windows Explorer. You can use the Plus and Minus icons to expand or collapse the list of folder contents. To put a policy into a folder, either drag and drop an existing policy from another part of the tree, or open the folder and click Add Policy to create a new policy in the folder.

Folder Permissions

For each folder, you can define a an access control list (ACL) that governs who can access its contents. For information about the access control permissions and how to set them, see “Changing Access Rights” on [page 74](#).

Exporting and Importing Policies

Components and policies require time and effort to design and construct. However, there may be cases where you want to distribute the components and policies defined in one implementation of Compliant Enterprise and deploy them in another. For example, you may have a separate test environment where you have fine-tuned your policies, and you now need to transfer them to your production network.

To avoid having to manually redefine all the components and policies in the second system, Compliant Enterprise provides a CLI-based export/import utility. It allows you to export any policies in one Control Center, along with their folder structure, and then import them elsewhere. Although all components required by exported policies are automatically included, you have the option of manually specifying components to export as well. All objects are exported to an XML file, which you can then import into the policy repository of another Compliant Enterprise system.

The Export utility works entirely independently from Policy Author. For details on using it, consult the “Routine Management” chapter of the *Compliant Enterprise 2.0 Administrator’s Guide*.

Using Policies and Components

In the two previous chapters, we described how you can use Policy Author to create components and policies. As we have seen, these two kinds of objects have a hierarchical logical relationship—components are the building blocks for policies—but in practice, there are many operations you will perform on both kinds of objects the same way. These are the subject of the present chapter, and they include the following:

- Deploying Policies and Components ([page 59](#))
- Management Tools ([page 66](#))
- Managing Policies and Components ([page 71](#))

Note that in this chapter we will often use the generic term *object* to mean either components or policies, since the discussion applies to both equally.

Deploying Policies and Components

Once a policy or policy component is created, it must be distributed to the Policy Enforcers installed on PCs or file servers wherever policy enforcement is desired. An organization may have specific requirements for controlling changes to computer systems, so one person is typically assigned to control access to policies and the distribution of policy changes to enforcement points. The process of distributing and organizing the distribution of policies and policy components is called *deployment*.

The Deployment Sequence

The deployment process can be broken down into a sequence of several tasks:

1. **Checking Dependencies:** see whether deployment depends on completion of other components.
2. **Setting Deployment Targets:** specify which machines should receive the deployment.
3. **Submitting for Deployment:** place a policy or component into the list of objects awaiting deployment.
4. **Scheduling Deployment:** specify when deployment should occur.
5. **Confirming Deployment:** check to see that deployment occurred according to the specified targets and schedule.

We will describe each of these in more detail later in this section.

Deploying All

If you have submitted several policies or components for deployment, you can use the Deploy All feature to deploy them all at the same time. This feature is also useful if you want to allow Compliant Enterprise to set the deployment targets and deployment time. You can also deselect some of the components and policies if you do not want to include them in the deployment.

To deploy some or all policies and policy components that have been submitted for deployment:

1. From the Actions menu, select the Deploy All command. Compliant Enterprise identifies all policies and policy components that have been submitted but not yet scheduled for deployment. Compliant Enterprise performs a dependency check to see whether you must finish working on supporting policy components before deploying the policies and components that are already submitted. If everything is finished, the Deploy All window appears with all submitted policies and components selected.

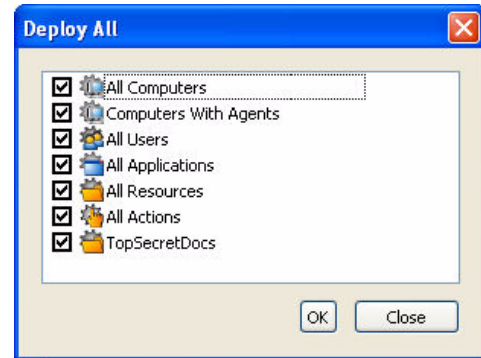


Figure 4-1: The Deploy All Window

2. Uncheck any items that you don't want to deploy with this batch.

3. Click OK. The Deploy window appears, with all dependent components checked.

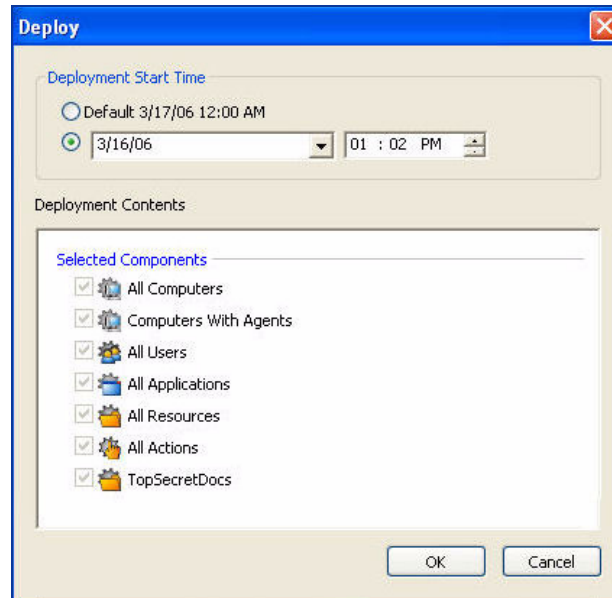


Figure 4-2: The Deploy Window

4. For Deployment Start Time, choose one of the following:
 - **Default:** Starts deployment at the default date and time, as displayed. By default this is set at midnight of the current day, but you can set a different time if you wish. (This might be delayed slightly by the heartbeat mechanism, which is set to 60 seconds by default.)
 - **Date and Time controls:** display the current time by default, but you can change them to set any specific moment you want deployment to begin.
5. Click OK. The policies and components are automatically deployed to all required targets, as identified by Compliant Enterprise. (For information on setting targets manually, see [page 62](#)).

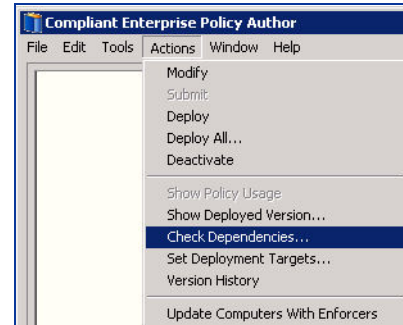
The Full Deployment Sequence

As we mentioned, the full deployment sequence—as distinct from the quick deployment described above—comprises five steps. Let’s go through them in detail now.

1. Checking Dependencies

Whenever you finish defining a component or constructing a policy and are considering deploying it, it is important to see whether it depends on completion of other components. This is called a *dependency check*. Compliant Enterprise automatically performs a dependency check when you submit an object for deployment, but you can also perform this check yourself at any time.

1. In the navigation pane, or using the tabs along the top of the editing pane (if you are editing multiple objects at the same time), select the policy or policy component whose dependencies you want to check. If you are editing a policy or component, it is already selected.
2. From the Actions menu, select the Check Dependencies command. If dependencies are found, the Check Dependencies dialog box appears with the results. For each dependent component an explanation of the dependency is shown.
3. Read the information in the dialog box. If it shows that you must take action, such as finish editing another component before you can deploy the current one, make a note of the required task. If the dependent component is greyed out, you do not have permissions to perform the required task yourself.



The dependent component might be in any of the following states:

State	Description
Missing	The component has never been submitted for deployment. If it is required by the object you are trying to deploy, you will need to submit the missing component first.
Required	The component has been submitted for deployment and will automatically be deployed along with the current component.
Modified	An existing version of the component has been deployed, but someone has created an edited version more recently that has not yet been deployed. The existing deployed version will be used until the modified version is deployed.

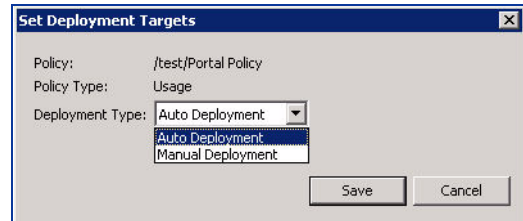
However, if you want your new object to use the more recently modified version, you can open that object, see why it has not yet been deployed, and move the process forward if possible. For example, perhaps the modified component needs to be submitted for deployment; if it has been submitted, perhaps it has not yet been scheduled; or perhaps it has been scheduled, but its deployment time is too far in the future.

4. When you are finished reading the information, click OK.

2. Setting Deployment Targets

Before a policy or component is submitted for deployment, you can specify which Policy Enforcers you want it deployed to. To do this,

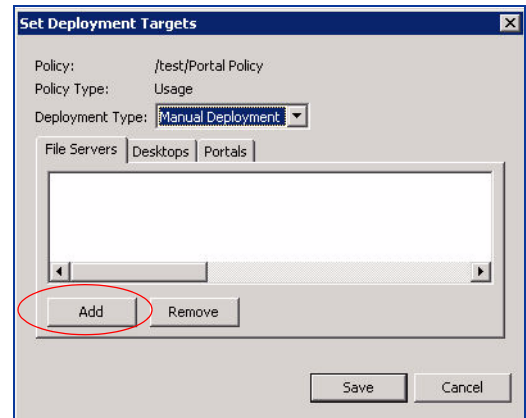
1. From the Actions menu, select the Set Deployment Targets command. The Set Deployment Targets dialog box appears.
2. In Deployment Type, choose one of the following:



- **Autodeployment:** Compliant Enterprise automatically chooses which machines should receive the policies. (This is the default behavior.)
- **Manual Deployment:** You can specify which hosts you want to receive the policies.

If you choose Manual Deployment, the File Servers, Desktops, and Portals tabs appear. You can use them to specify which enforcers of each type you want this policy deployed. Each tab displays a list of groups of enforcer hosts or individual host names. (Bear in mind that the File Servers tab will show both Windows and Linux hosts, without distinction.) If no deployment targets have been set up yet, this list is blank.

3. To add a deployment target to the list, click the Add button. A pop-up dialog appears where you can select individual enforcer hosts or groups where you want to deploy policies.



4. To delete a deployment target from the list, select the desired group or machine name and click Remove. This will prevent the currently selected policies and components from being deployed to this target.

Note that you cannot select items within groups, only the group as a whole.

5. When finished, click Save.

3. Submitting for Deployment

Submitting a policy or policy component for deployment places it in a state where it is ready to be sent to policy enforcers and take effect within your organization, but it will not be deployed until someone takes the step of scheduling the actual deployment.

1. In Policy Author, display the policy or component you want to deploy. Typically, it is already displayed, because you have been editing it.

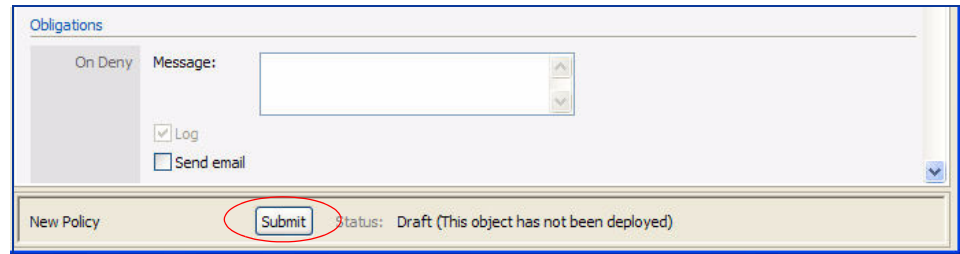


Figure 4-3: Submitting for Deployment

2. Click Submit. Compliant Enterprise automatically performs a dependency check to see whether any additional policy components must be finished and deployed before you can continue. If there are no dependencies, you are ready to schedule deployment.

If Compliant Enterprise discovers one or more dependencies, it displays a dialog box describing which new or modified components are involved. You will not be able to proceed with deployment until these components have also been submitted; they will display as Missing.

4. Scheduling Deployment

After a policy or component is submitted for deployment, you can specify when deployment should occur.

1. In the navigation pane, or using the tabs along the top of the editing pane (if you are editing multiple objects at the same time), select the policy or policy component for which you want to schedule deployment.
2. Click the Deploy button. Compliant Enterprise performs a dependency check. If it finds that the deployment depends on the completion of other policy components, a dialog box appears to explain why the deployment could not be completed.

If there are no dependent components that require attention before deployment, the Deploy dialog box appears. This dialog box shows which policies and components will be deployed: the one you are working on, plus any dependent components detected in the dependency check.

3. For Deployment Start Time, choose one of the following:
 - **Default:** Starts deployment at the default date and time, as displayed. By default this is set to midnight of the current day, but you can change it if you wish. If you set it to deploy now, there may be a short delay due to the heartbeat mechanism, which by default is set to 60 seconds.

- **Date and Time** controls: display the current time by default, but you can change them to set any specific moment you want deployment to begin.
- 4. In Deployment Contents, look for the Modified Components category. If present, it shows policy components that are needed by the object you're trying to deploy and that have been modified and resubmitted for deployment. Check the box next to any of these components that you want to schedule for deployment at the same time.
- 5. Click OK.

5. Confirming Deployments

Once you finish defining and deploying a policy or a component, you may want to confirm that it has indeed been deployed in the way you intended. In addition, when you construct and deploy a large number of components and policies, it is often helpful to be able view summary of where they are deployed.

Policy Author offers several convenient tools that allow you to monitor the status of deployed policies and components; these are described in the following section.

Management Tools

You can use Policy Author to monitor the current status of your deployed objects from four perspectives:

- The **component** perspective: which policies incorporate a particular component
- The **policy** perspective: what is going on with all currently deployed policies (see [page 68](#))
- The **enforcer** perspective: which policies each enforcer in your network is using, and how (see [page 69](#))
- The **version history** perspective: which version of each policy or component is currently deployed, and what other alternative versions exist

Let's examine each of these in turn.

Deployed Components

One of the ways you can monitor current deployment, is by component. That is, from time to time, you may need to find out what a particular component is doing—that is, which policies it is used in. To do this, select the component you are interested in, and select Show Policy Usage from the Actions menu. Policy Author will then display a list of all policies that use the selected component.



You can also use the Find tool—the search field at the top of the policies tree—for this purpose. If you type a text string in the field and click the Go button, Policy Author will display a list of policies containing the string you typed. If you type a component name, it will display a list of policies containing that component. The tool is not case-sensitive.

The example in [Figure 4-4](#), below, shows the search results for the string “Access”). To return to the policy tree view, click the X next to the Find button.

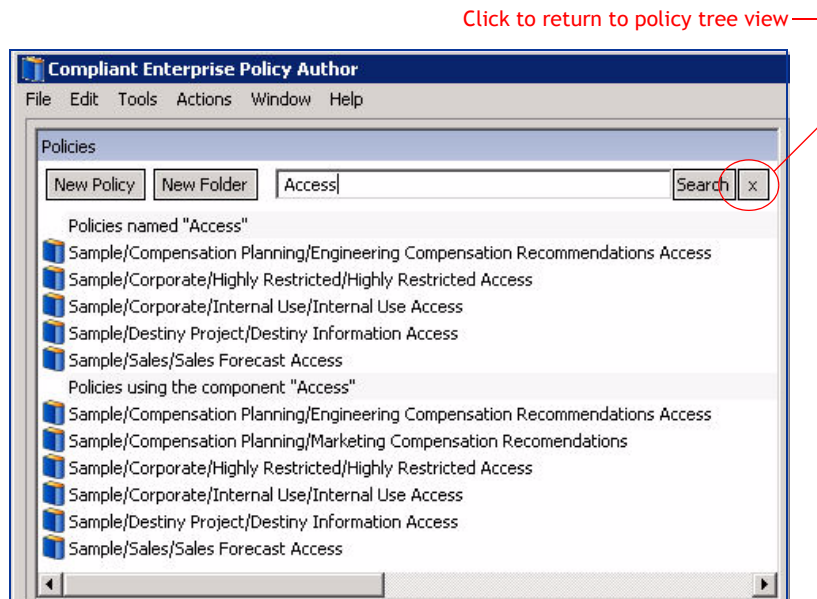


Figure 4-4: Using the Find Tool

Deployed Policies

To view the current state of all your defined policies, select the Deployment History command from the Tools menu. This will display a list of all recent policy deployments in chronological order, as shown in Figure 4-5, below. This list shows all deployments during the past eight days.



Each row in this list represents a single a *deployment package*. A deployment package simply refers to one policy and any associated components that were deployed together. (Although this is by no means required, in practice, administrators will generally define several policies and then deploy them at the same time.) Table 4-1 describes the information presented in the Deployment History window.

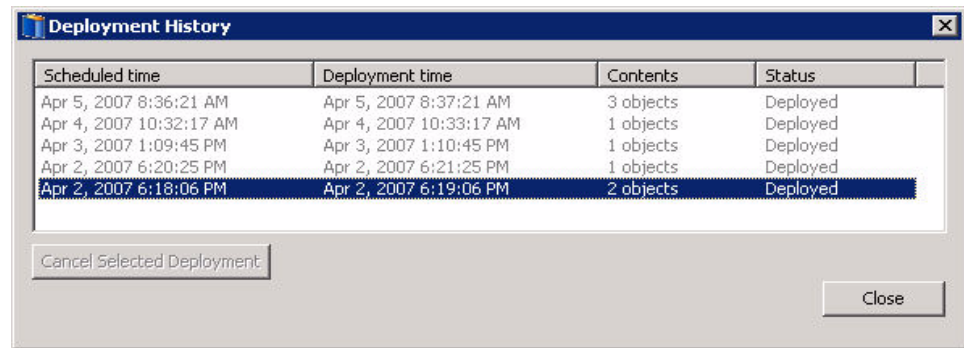


Figure 4-5: The Deployment History Window

Table 4-1: Deployment History Information

Column	Description
Scheduled Time	The time at which this package was actually submitted for deployment, regardless of any specified delay before the scheduled deployment.
Deployment Time	The time at which this package is supposed to take effect. This may be the same as the Scheduled At time, or some point thereafter.
Contents	The number of items (policies + components) in this package.
Status	<p>The current status of this package. There are seven possible statuses:</p> <ul style="list-style-type: none"> • Scheduled: The request for deployment has been accepted and a time has been set for it to occur. After scheduling a deployment, you can also Cancel Scheduled Deployment; this returns the object to the Submitted state. • Deployed: Compliant Enterprise has finished deploying the object. • Pending Deactivation: The request for deactivation has been accepted and a time has been set for it to occur. • Inactive: The object has been deactivated.

Canceling Deployment

You can use the Deployment History window to cancel deployment packages that are still in Scheduled state (see [page 75](#)). Note that the Cancel Selected Deployment button is not enabled unless a Scheduled item is selected in the list.

Deployments by Host

To check which policies and components are currently deployed on which Policy Enforcers in your network, select the Deployment Status command from the Tools menu.



As [Figure 4-6](#) shows, this window is organized into three tabs, for Desktop Enforcers, File Server Enforcers and Portal Enforcers. Each tab displays a list of all hosts—file servers or PCs—where policies have been

deployed. For each machine in the Host list, the total numbers of policies and of policy components deployed are also displayed.

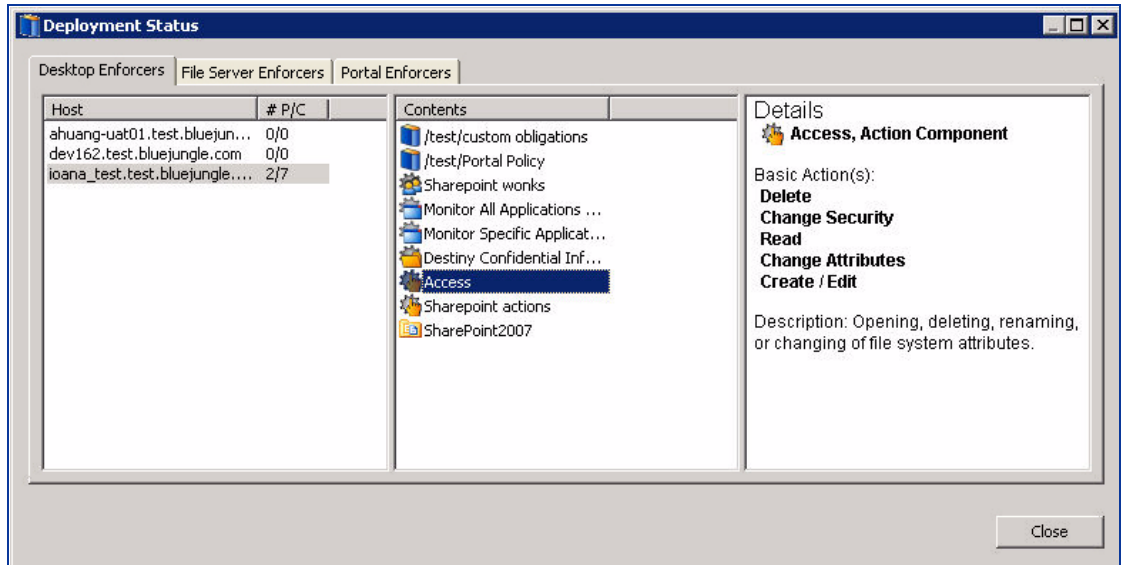


Figure 4-6: Deployment Status by Policy Enforcer

On either tab, when you select an enforcer in the Host list, the Contents list (in the middle pane) displays all policies and components deployed on it.

When you select a policy or component in the Contents list, the detailed definition of it appears in the Details pane, at the right.

Deployment Version History

As a fourth alternative, you can check which policy enforcers have received deployment of a particular policy or component, when the deployment occurred, and whether more than one version of a policy has been deployed. This can happen when you make some changes to a policy and then deploy it to the network, but the new version cannot be deployed to one or more enforcers. This may happen, for example, if the host is a laptop that is temporarily disconnected from the network.



To view this information, select a policy or component, then select Version History from the Actions menu. This command opens the Version History window, as shown in Figure 4-7. (If the Version History command is disabled in the menu, it means you have selected an object that has never been deployed.)

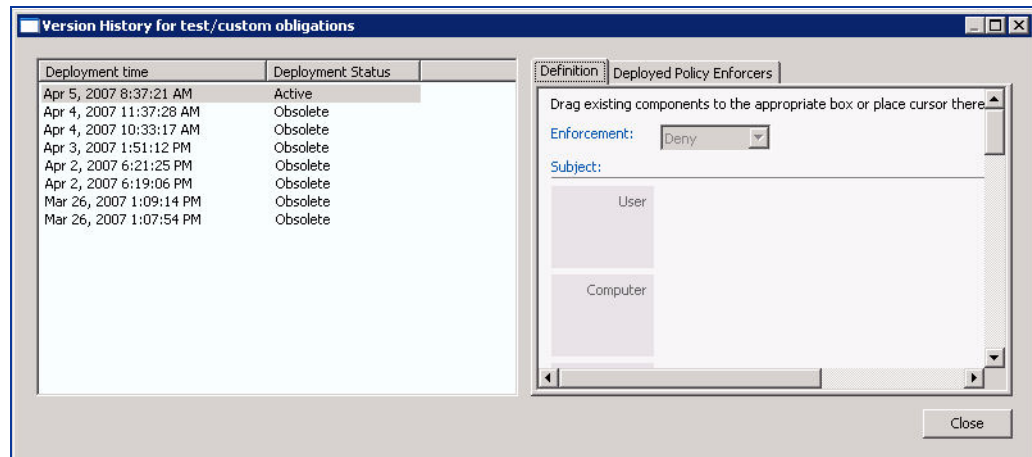


Figure 4-7: Version History Window

The left pane of this window displays a list of currently scheduled and past deployments of the selected policy or component. Each item in this list represents a version of the selected policy or component; the top one will be current, and all others will be obsolete. You can select any item in this list and view details, in the information pane on the right. That pane is organized on two tabs:

- The **Definition** tab displays the detailed definition of the selected version.
- The **Deployed Policy Enforcers** tab displays a list of hosts where the selected version has been deployed, and the type of each host: desktop, file server, or portal. Here too, file servers will include both Windows and Linux, without distinction.

Managing Policies and Components

Once you have deployed objects, you may still need to manage them in various ways. For example, you may need to adjust how a policy is defined, in order to improve its effectiveness.

Policy Author lets you perform the following standard management functions

- Searching for Policies or Components
- Modifying Policies and Components ([page 73](#))
- Changing Access Rights ([page 74](#))
- Un-deploying Policies and Components ([page 75](#))

Searching for Policies or Components

To find a particular object, you can click Policies or Components in the navigation pane, click a category, and scroll through the lists. However, if you have a large number of items to search through, a more efficient way to find what you want is to use the search tool.

To search for an object:

1. In the Find field above the policies navigation tree, type a search term. The type of search term you can use, and the results you will get, depend on whether you are searching for an object.

You can find **components** based on any of the following criteria:

- Name of the policy component; for example, “Human Resources Department.” You can type part or all of the name.
- Name of a policy that refers to the policy component; for example, you might have a policy called “HR Offer Access” that says anyone in the Human Resources department can open documents that contain offer letters to prospective employees.

You can find **policies** based on any of the following criteria:

- Policy name
- Name of a policy component that is used in the policy you are trying to find

2. Click Go.

Finding Out Where a Component is Used

From time to time you may need to find out which policies use a particular policy component. To do this,

1. In the component bin, click the panel for the component type, and select the component you are interested in.
2. From the Actions menu, select Show Policy Usage. This will display a list showing which policies refer to this policy component. If this

command is disabled, it means the selected component is not used in any policy.

Viewing Properties

To view or modify the properties and access permissions of a folder, policy, or policy component, use the Properties dialog box.

To view the properties, either right-click on the object and select Properties, or select the object, open the File Menu, and click Properties. This will open the Properties window, as shown at right.

The Object Properties window is organized into two tabs, General and Access Control.

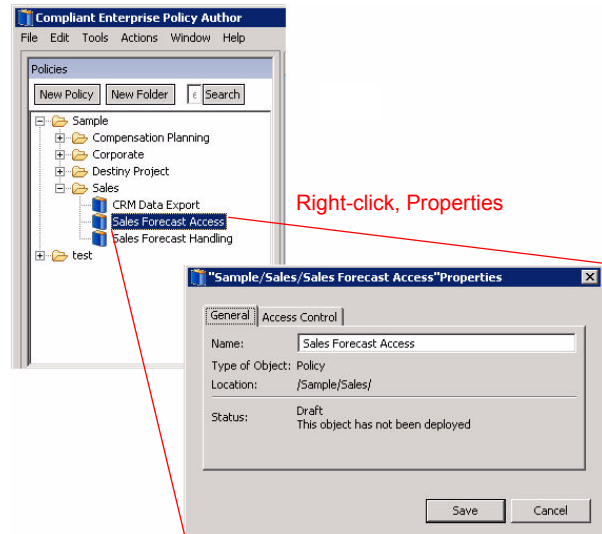


Figure 4-8: The Properties Window

On the **General** tab, you can:

- View information about the folder, policy, or component: name, current status, type (component, policy or folder), and location (if it is a policy).
- Rename the object. If you do not have permission to change the name, it is greyed out.

On the **Access Control** tab, you can

- See who owns the object, and
- View the object's currently defined access rights.

The Owner field is read-only and cannot be changed by anyone; the access rights can be edited by the owner or by any user with *Set Access* rights for that object. For a detailed discussion of access control feature, see [page 74](#).

Comparing to the Deployed Version

When you are editing an object that has already been deployed, you might want to see this deployed version that is currently in live use in your Compliant Enterprise system, so that you can compare it to the changes you are making.

1. Use the navigation pane to locate the object.
2. From the Actions menu, select Show Deployed Version. This displays a screen showing a non-editable view of the currently deployed version of the object definition, as well as other versions currently

pending deployment, if any. As long as this screen is displayed, you cannot continue editing.

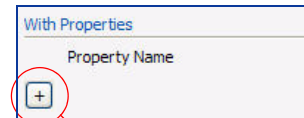
3. To continue editing the current object, close the Deployed Version screen.

Modifying Policies and Components

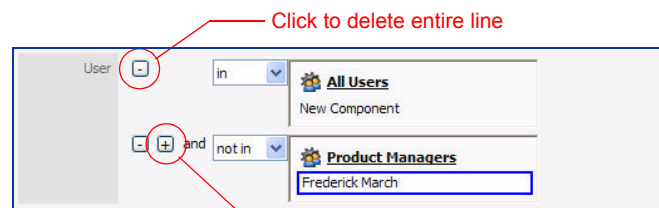
As company conditions change over time, you might want to change an existing object. For example, if you have defined a policy that allows only Human Resources personnel to view offer letters, and the company CEO expresses an interest in viewing such letters, you would need to modify the policy to include the CEO.

To modify an object,

1. Display the object in the editing pane.
2. If the object has already been submitted for deployment, the editing controls are disabled. If you still want to modify the component, click **Modify**; this removes the object from the deployment queue so you can edit it.
3. Make the desired changes:
 - Drag and drop new items into the editing areas, add new lines to the definition, or make any other desired changes.
 - If a line in the definition has not yet been filled in, you will see only the line label. To expand an entry so you can edit it, click the Plus icon.
 - To remove a line, click the Minus icon.



Click to expand and edit



Click to delete entire line

Click to add another line

4. While you are working, you might want to see the object version(s) currently in use, so you can compare your changes to what is already in use. To do that, use the Version History command in the Actions menu.
5. When you are ready to publish the changed object, you can deploy it as you wish.

Renaming an Object

The name of a component or a policy is not available among the properties displayed in the editor pane. If you want to assign a different name to an existing object, right-click the object in the policy tree or the component bin, and select Properties to open the Properties window (see [page 72](#)). You can assign a different name in the name field, and save your changes.

Note that if the name field is disabled (greyed-out), that means you do not have sufficient privileges to rename the object.

Changing Access Rights

As we mentioned earlier, you can right-click any object to view its Properties window. One of the tabs in this window, Access Control, displays the currently set access rights for that object. Access rights govern how specific users or groups can do things like read, deploy or delete the object. [Table 4-2](#), below, provides descriptions of all available permissions.

Users with Set Access permission can use this tab to redefine these rights, including changing the rights of an existing user or group, and also granting access rights to additional users or groups. The settings you make in this tab will override the default access control settings.

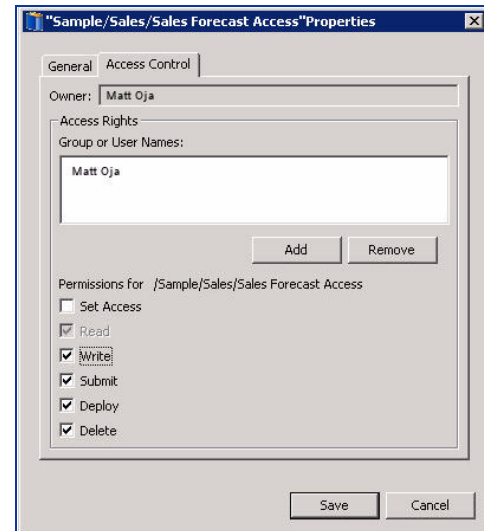


Figure 4-9: The Access Control Tab

Three kinds of setting changes can be made on this tab:

- Check or uncheck permissions for an individual user or group;
- Click the Add button to add a new user or group, and set permissions for it;
- Click the Remove button to delete a user's or group's permission from this object. (This is the same as unchecking all permissions, except that it is harder to undo.)

Be sure to click the Save button when you are finished, to save your changes.

Table 4-2: User and Group Permissions

Permission	Description
Set Access	Users can make changes in the Access Control tab.
Write	For policies and components: Users can edit the object, including setting deployment targets. For folders: Users can create policies or subfolders in the folder.
Submit	Users can submit for deployment or deactivation.

Table 4-2: User and Group Permissions (Continued)

Permission	Description
Read	For policies and components: Users can view the object name in the navigation pane, and can view but not modify the definition in the editing pane. For folders: Users can read the contents of the folder. Note that if policies inside a folder have ACLs that allow them to be read, they will be visible if a filter result returns those policies, even if the containing folder does not allow read access.
Deploy	Users can click the Deploy button.
Delete	Users can delete the object.

About Default Settings

The settings you make in this tab will override the default access control settings that were granted when the object was first defined. By default, whenever you create a new policy or component in Policy Author, users and groups are assigned access rights to it based on their roles as defined in Administrator—specifically, on which boxes are checked in the Policy Author Editing Privileges, in the Roles pane of the Users and Roles tab. If a user belongs to more than one group, his access rights will be determined by whichever group is specified in the Default Access Control Group setting; this also set in Administrator, in the Users pane of the Users and Roles tab.

Un-deploying Policies and Components

There are two procedures for un-deploying policies and components once they have been deployed, depending on the state they are in. Those that have been deployed on a schedule but are still waiting for the appointed time, are in a scheduled state, and may be cancelled. Those that have been deployed immediately, or whose scheduled deployment time has already passed, are in an active state, and must be deactivated.

If you want to delete a deployed policy or component, you must first un-deploy it one way or the other.

Cancelling

If a package of policies and/or components has been scheduled for deployment but is still awaiting the scheduled time, you can cancel it if you wish. You may want to do this if you change your mind about deploying a new version of a policy or policy component, for example, or if you clicked “Deploy” by mistake. To do this,

1. From the Window menu, select Deployment History to open the Deployment History window, as shown in [Figure 4-5](#), above.
2. In this list, select the deployment package you want to cancel. (If you are not sure which package to choose, you can double-click on any line to see a detailed list of the contents of the deployment package.)

3. Click the Cancel Selected Deployment button. A dialog box appears requesting you to confirm the cancellation. This dialog box shows a list of the policies and components in the deployment package. You can only cancel them as a group, not individually.
4. Confirm the cancellation by clicking the Cancel This Deployment button.

Deactivating

When you *deactivate* an object, it is removed from deployment and is no longer in active use, but you can always deploy it again later. You can only deactivate an object if it is not in use (that is, if it is not referenced in any currently deployed policy). You might want to deactivate:

- A policy component that was never used in any policy.
- A policy component that represents a concept that is no longer useful in your company; for example, it represents the employees in a division that has been sold.
- A policy that is no longer useful—for example, one covering documents that were confidential during project development, but have been made public since the project finished.

If the policy is already actively deployed—that is, was deployed either immediately or on a schedule whose time has passed—you use the deactivate command to un-deploy it. To do this,

1. Display the policy or component you want to deactivate.
2. Right-click and select Deactivate, or use the Deactivate command in the Actions menu. This marks the object for deactivation. Note that if you are deactivating a component that is in use by any existing policies or other policy components, you will not be allowed to proceed.
3. Click the Deploy button to distribute the new deactivated state of this policy out to all enforcers where it is currently active.
4. Once a policy is deactivated, you can delete it by simply selecting it and clicking the Delete button.

Note the following important points about deactivating policies:

- When you deactivate a policy, it is removed from all enforcers in the network where it was deployed. Before you do this to any policy, be sure you understand what it is doing, and where.
- The deactivation can only be communicated to Desktop Enforcers if they are connected to the network. Enforcers on any laptops that are not connected will continue enforcing the policy until they reconnect and have their policy profiles updated.

Deleting an Object

After you cancel or deactivate a policy or component, it remains in Policy Author's policy tree pane, in case you want to redeploy it in future. If you are

sure you will not want to redeploy an entity, you can delete it permanently by selecting it and clicking the Delete button.

When you *delete* an object, its definition is no longer displayed in Policy Author. You can only delete an object if it was never deployed or it has already been deactivated. (Technically, once an object is deleted it remains in the database in Deleted state, but is not visible or usable in any way.)

To delete an object,

1. Display the object using one of the following techniques:
 - If you are already working with multiple objects in the editing pane, a row of tabs appears at the top with the names of the objects. Look at the tab labels to see whether the object you want is already there. If so, click the tab.
 - In the navigation tree, locate the object you want to modify. Scroll through the lists or use the Find box. When you see the name you want, click it.
2. The details about the selected object appear in the right portion of the window.
3. From the Edit menu, select Delete.

Note: This command is available only if the object was never deployed or has been deactivated. If it is greyed out in the menu, the object is still active—use the Deactivate command and then deploy the object, and the command will become available.

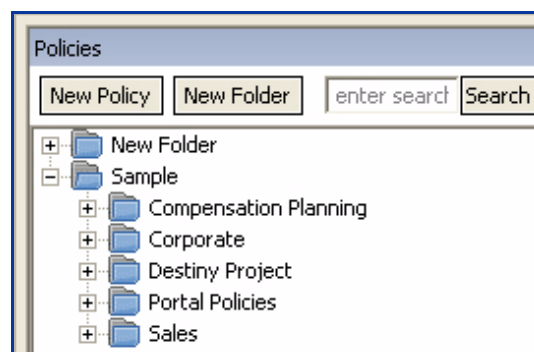
Using the Sample Policy Sets

In this chapter we will present four use cases that provide illustrations of the way you can define sets of policies to solve specific business problems. The policies and components for these cases are predefined, and after you install Policy Author they are available in the policy tree, for you to view and experiment with. The chapter is organized into the following sections:

- The Sample Folder
- Policy Set I: Destiny Project ([page 82](#))
- Other Policy Sets ([page 89](#))
- The Policy Life Cycle ([page 92](#))

The Sample Folder

When you install Policy Author, the policy tree is automatically populated with a folder called *Sample*, containing a number of predefined policies that provide examples of the types of policies that can be implemented using Compliant Enterprise. These policies are organized into four subfolders, each of which represents the solution to a specific business problem. (A group of policies that are designed to work together to solve a single distinct business problem is referred to as a *policy set*. For more on this, refer to Part I of the *Compliant Enterprise Implementation and Solutions Guide*.)



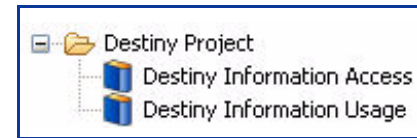
The four use cases are:

- Compensation Planning
- Corporate
- Destiny Project
- Sales

Let's examine each of them in turn, focusing on the nature of the problem at hand, and the way the policies work to solve it. We'll approach them from the simplest case to the most complex.

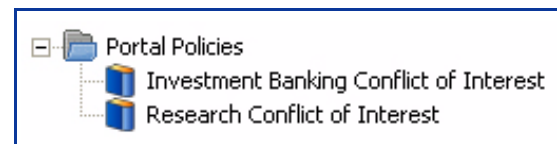
Destiny Project

This policy set is designed around a hypothetical top-secret project called Destiny. It demonstrates how policies can be used to limit access and usage of confidential project information to authorized team members. It consists of two policies; one allows only project team members to access a set of project files; the other prevents duplication or distribution of project files to uncontrolled locations.



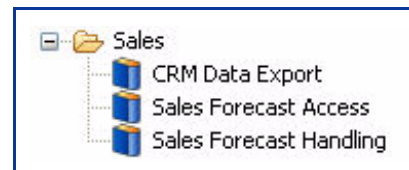
Conflict of Interest on SharePoint

This policy is designed to segregate one group of users from another. Each has a separate site on a SharePoint collaboration portal where they need to share documents among their colleagues within the group, but the members of each group must be reliably blocked from accessing the document of the other. This case is easily covered by two policies; one allows only the research group to access its site, the other allows only the IB group to access its site.



Sales Forecasting

This policy set is designed around a typical sales forecasting process, in which sales data is exported from sales applications and then analyzed and tracked using a set of spreadsheets. It includes three policies that control data exported from sales applications, control access to sales forecasting spreadsheets, and prevent duplication or distribution of non-public sales data.



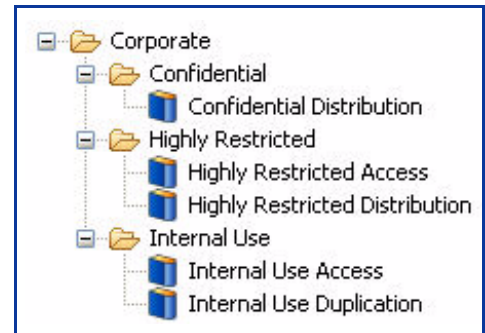
Compensation Planning

This policy set is designed to control data exported from payroll applications, control access to specific spreadsheets, prevent duplication or distribution, and ensure that finalized compensation plans are not deleted or modified. It consists of seven policies, which address data use in a more sequential or work-flow based way.



Corporate

This policy set implements a three-tiered data classification structure and places controls on information access and use based on its classification.



Using the Samples

By default, the sample applications are accessible only to the Administrator user account, and are in the Submitted state. To make the samples accessible to a larger set of users, change the access control settings for the Sample folder, policies, and policy components. To begin editing a sample policy or policy component, click the *Modify* button, which will return the policy or component to the Draft state.

Each sample application provides a set of policies and the corresponding policy components. The policy components are often blank, since the details required to make these components function correctly depend on your environment. For example, the definition of a user component depends on the users or groups defined within your organization. Therefore, before deploying or enforcing policies based on these examples, you must define the details of the required policy components as follows:

- User components must be defined to represent users or user groups within your environment
- Computer components must be defined to represent hosts or host groups within your environment
- Application components must be defined to represent applications within your environment
- Document components must be defined to represent file resources within your environment
- Portal content components must be defined to represent collaboration portal items

Policy Set I: Destiny Project

The Destiny Project sample set shows how two policies can be used to prevent unauthorized access to and use of confidential project information. In this example, members of a project team collaborate on a set of documents as they design and develop a new product. Each team member works on a part of the project and shares files with the team on a file server. In addition, team members often work with documents offline from local copies of these documents.

The information control requirements for the Destiny project are:

- There is a team, called the Destiny Team, that needs access to confidential information
- Destiny documents are shared within the directory \\fileserver\share\destiny on a shared file server
- Team members will often need to copy documents to their PCs or laptops, work on them locally, then update the version on the server
- Destiny documents must not be copied to unprotected locations, sent to recipients outside the team, or accessed from non-company computers

To ensure that unauthorized persons do not gain access to this information and that information is not inadvertently leaked, two policies were implemented.

Policy 1: Controlling Access

The first policy, Destiny Information Access, states:

“Allow only Destiny Team Members using Company Computers to Access Destiny Confidential Information”

In terms of the ACPL policy grammar, the sample policy consists of the following “parts of speech:”

- Effect = Allow Only
- Subject = Destiny Team Members on Company Computers using any application
- Action = Access
- Resource = Destiny Confidential Information
- Obligation = Log On Deny

Figure 5-1 shows how this policy appears in Policy Author.

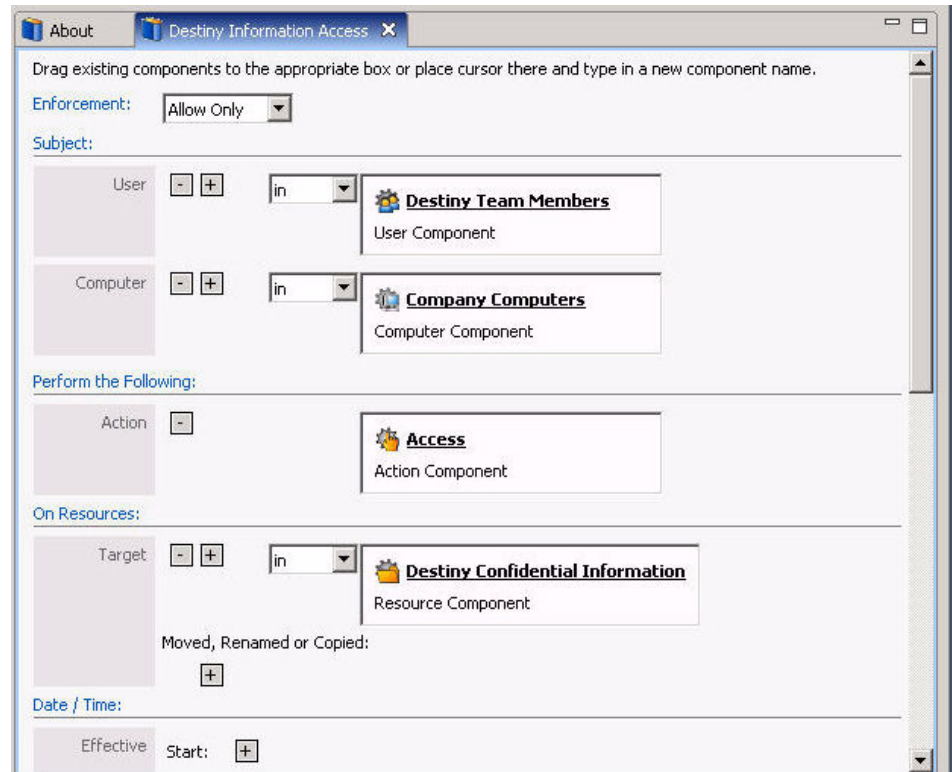


Figure 5-1: Controlling Destiny Access

Effect

The policy's effect determines the active enforcement behavior—that is, whether to permit or prevent the specified action. The Policy Author application's Policy Editor provides the following options (the first option, Allow Only, is used in the Destiny Information Access example policy):

- **Allow Only:** Only the specified subject is allowed to perform the specified action on the resource; all other subjects will be blocked from performing the action. In this example policy, the subject is Destiny Team Members on Company Computers. Therefore, only users who are within the Destiny Team using a computer that is a Company Computer will be allowed to perform this action; both conditions must be present.
- **Deny:** The specified subject is not allowed to perform the specified action on the resources.
- **Monitor:** The action is allowed in any case, but all specified obligations—a log entry, a warning to the policy subject, and/or an e-mail notification—will occur whenever the policy is enforced.

Compliant Enterprise is a “deny override” system: policies are written either to prevent subjects from performing actions or to perform obligations when a trigger occurs. Policies are not used to grant access. In this example policy, the Allow Only effect implicitly denies all users not covered by the definition of the policy’s subject.

Subject

The policy’s subject refers to the users, computers, and/or applications that perform the action. In this example, the subject includes both a user component and a computer component; since no application is specified, it includes all applications. These components are evaluated using a logical AND: the policy applies to Destiny Team Members (any application) on Company Computers, and all three must be true for the policy to evaluate to true. (Since Any Application will always be true, it need not actually be specified.) For example, if Jane Smith, who is a Destiny Team Member, attempts to access this information while using a Company Computer she is allowed to do so, but if she tries to access this information from a home computer, she is denied.

Action

In this example, the action is Access, a predefined action component that contains several basic actions within it. [Figure 5-2](#) shows the action component as defined for this example, as it appears in Policy Author.

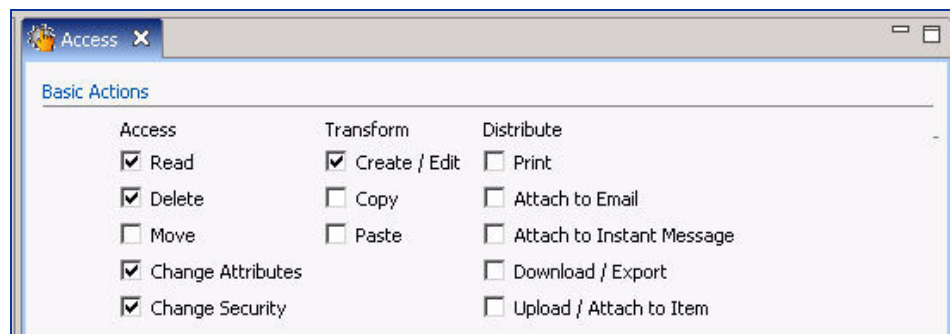


Figure 5-2: Sample Action Component: Access

Resource

The resource in this sample policy, Destiny Confidential Information, is a document component that describes a class of documents, defined by the network paths where they may be stored. In this example, Destiny Confidential Information may reside on a number of server or desktop systems and move between the multiple locations. With the Destiny Confidential Information component, you can write policies about this class of information that apply to all of these locations. In our example, team members often need to work on local copies of Destiny Confidential Information. While these documents exist on local desktop systems, we want to ensure that policies are still appropriately applied.

Figure 5-3 shows the definition of this document component in Policy Author, illustrating how it applies to documents on file servers and on local desktops within the My Documents folder.

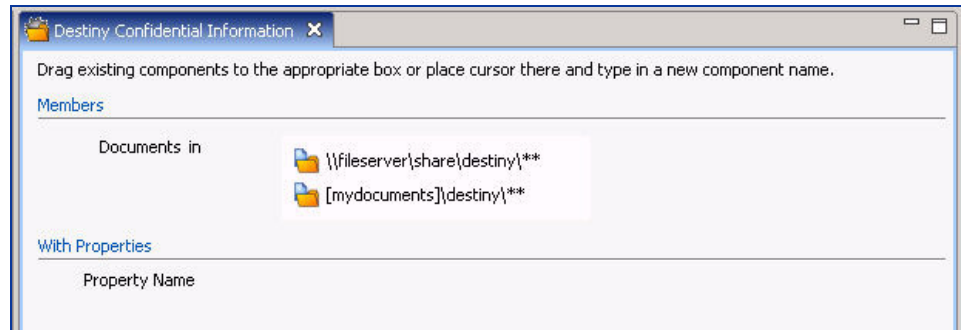


Figure 5-3: Destiny Confidential Information

Customizing the Sample

Sample policy 1 defines one each of four component types, which you can actually use in your real policies, after customizing the definitions so match your needs. To view the details of how each component is defined, double click it to open it in the editing pane. The following steps walk through the process of mapping these components to your environment.

1. First, map the user component Destiny Team Members to users defined in your environment. You will then be able to see the effect of the policy on Destiny Team Members as compared to other users.
 - A. Open the Destiny Team Members user component.
 - B. Add one or more members to the team either by selecting them from the lookup list, or by typing their names directly. You can add either LDAP users or groups. At this point, do not include yourself in this user component.
2. Next, map the component Company Computers to computers within your environment. You will then be able to see the effect of the policy when accessing documents from different computers.
 - A. Open the Company Computers component.
 - B. Add one or more LDAP hosts or host groups to this component definition. Include the computer that you will be using in this computer component.
3. Last, update the Destiny Confidential Information document component to map to a folder on a file server in your environment. You will then be able to see the effect of accessing documents that are considered Destiny Confidential.

- A. Open the Destiny Confidential Information document component.
- B. Update the existing value “\\fileserver\share\” to point to a file server location in your environment.
- C. For testing the sample policy, you will need to create a folder called “Destiny” on both the file server and within your My Documents folder. Create one or more files in each of these directories. For help on the required syntax, see [page 23](#).

Note that there is no need to make any changes to the Action component; it can be used as is.

Policy 2: Duplication and Distribution Control

The second policy in the Destiny Project example application is designed to control the duplication and distribution of Destiny confidential information. The Destiny Information Usage policy states:

“Deny Duplication and Distribution of Destiny Confidential Information outside of Destiny Confidential Information”

[Figure 5-4](#) shows how this policy appears in Policy Author.

This sample policy consists of the following ACPL “parts of speech:”

- Effect = Deny
- Subject = None Specified (implicit All)
- Action = Duplicate, Distribute
- From Resource = Destiny Confidential Information
- To Resource = outside Destiny Confidential Information
- Obligation = Log on deny

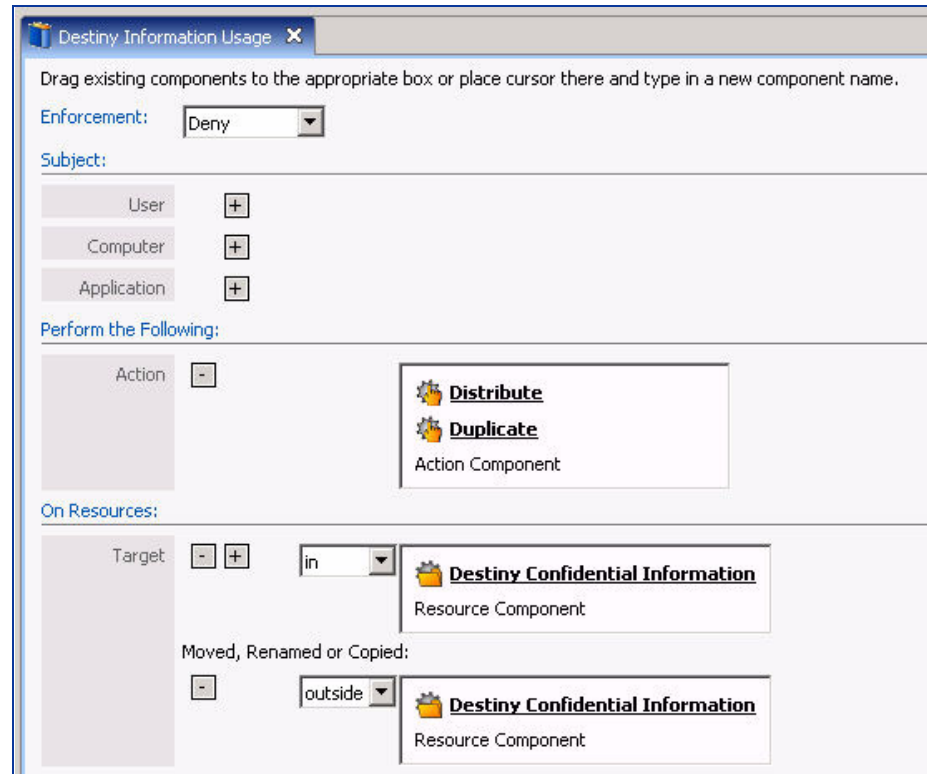


Figure 5-4: Destiny Information Usage Policy

Subject

In this example, no subject has been specified. By default, this means that the effect will apply to any subject that attempts to perform the specified action on the specified resource. Because this policy works together with the first example policy (Destiny Information Access), in practice, only Destiny Project Members will be able to access this information; the Destiny Information Usage policy will only affect those users. To be complete, however, the policy does not specify a specific subject; this makes the policy apply to all subjects.

From and To Resources

Some system actions support a concept of both source and destination resources. For example, you can move a file from one location to another or rename a file from one name to another. For all such actions, an optional destination or “To” resource can be specified. This allows you to create policy that can prevent this action from occurring based on either the source or the destination, or both.

In this example, preventing duplication of Destiny Confidential Information outside of Destiny Confidential Information will allow the user to move or copy the files to locations that are still within the definition of the Destiny Confidential Information document component but prevent them from moving or copying the documents to locations that are outside of the Destiny Confidential Information document component definition. For example, a user will be able to copy a file from Destiny folders on the server share to Destiny folders on a local hard drive and back, but will not be able to move or copy that document anywhere that they please.

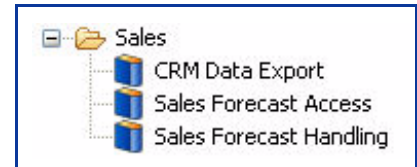
By implementing this type of policy, you can continue to control information as it moves or is duplicated as required by the business users, while maintaining the same control of the document and document copies.

Other Policy Sets

The other three policy sets are more complex, and contain more policies, than the Destiny Team Members set. However, once you familiarize yourself with the the first sample set, it should be clear just by examining the definitions of the others and of their components, how they are designed to work. For that reason, we will provide a less detailed summary of their design and purpose here.

Policy Set 2: Sales

The purpose of this set is to control what users can do with information related to sales activity and forecasts. It consists of three policies:



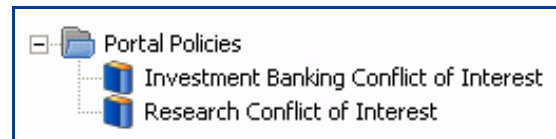
Sales Forecast Handling: Defines the basic information barrier within which sales forecasts documents will be contained, by preventing any user from duplicating (moving, copying or cutting & pasting) or distributing (e-mailing or IM-ing) any documents in a designated directory, to any location outside that same directory. You will find that a policy using this standard syntax—“Deny All Users to Duplicate or Distribute from In location [resource] X to Outside location X”—will commonly be required, in all kinds of policy sets. (For example, compare this policy to the Compensation Data Duplication policy, under Compensation Planning.)

CRM Data Export: Controls how users can export data from the company’s CRM application. Specifically, it allows users to export CRM report files from the CRM application to areas within the controlled sales areas, but blocks them from exporting them to any other location. These files are defined by the CRM Export Files component, as any file whose name starts with the string “report”. Since the CRM application automatically uses this naming convention when exporting data, this will cover all required files. This policy provides a good example of the use of an application in the subject of a policy, and of how to use different source and target resources to define locations.

Sales Forecast Access: Prevents anyone but General Manager, the Controller, and all Sales Managers from accessing any documents in the controlled sales area, during a specified six-week blackout period. It also requires that all access during this period can be only on company computers. This policy provides a good example of using computer components in a policy subject, and of setting a time window during which the policy will be enforced.

Policy Set 3: SharePoint Conflict of Interest

The purpose of this set is to define two segregated user sites on a single SharePoint server where members of the Investment Banking group and the Research group can post their



documents and other information and collaborate with one another in each group, but are reliably blocked from accessing any information on one another’s sites. It represents a standard information barrier scenario consisting of just two

policies, but is of interest because it demonstrates how you can define information barriers on SharePoint portals.

Investment Banking Conflict of Interest. This policy explicitly denies all members of the Research team from accessing any content on the Investment Banking site. This access will always be blocked, regardless of any permissions users in the IB group may grant for the content they post on their site. This eliminates the most vexing security problem of SharePoint site access—namely, that owners of site content have the authority to grant access to other users, which can easily lead to breaches of information use policies.

Research Conflict of Interest. This policy allows only members of the Research team to access the Research site on the SharePoint portal. Here, too, the policy overrides any permissions granted by individual members of any team, that might otherwise open the site or some of its content to access by unauthorized users.

Policy Set 4: Compensation Planning

The purpose of this policy set is to define several information barriers around all documents containing information regarding employee compensation. This is useful for controlling access to sensitive data per se, but it also illustrates how the policies in a set can work together to enforce a sequential work flow—in this case, the processing of annual compensation reviews. The set consists of seven policies:



Compensation Data Access: Restricts exporting data from the payroll application database to any location but the controlled area. Note the similarity to the Sales policy, CRM Data Export. In this case, the payroll data documents are defined as *payroll*.csv*—integrating with the naming convention used by the application being controlled.

Compensation Data Duplication: Defines the primary information barrier—the controlled area where compensation documents may be stored. (Note that it uses the same standard syntax as the Sales Forecast Handling.)

Engineering Compensation Recommendations Access: A standard access policy, which blocks everyone but Engineering Managers from all files in the Engineering Compensation Recommendations directory, and requires that they do so from company computers only. In addition, it defines a ten-week time window, which is helpful in imposing a strict deadline for finishing recommendations.

Marketing Compensation Recommendations: Identical to the previous policy, but applies to Marketing Managers.

Compensation Plan Approval: Deployed simply to send a notification e-mail when any authorized user finalizes a compensation plan document, which is defined as moving it from the Recommendations directory to the Final Plan directory. This is a good example of how you can use Monitor policies to help enforce a standard workflow: the Engineering and Marketing Managers work on recommendations in their own separately defined controlled areas, then pass them on to the Compensation Analyst when they are finished. Once they do this, they are blocked from making any more changes, by the following policy.

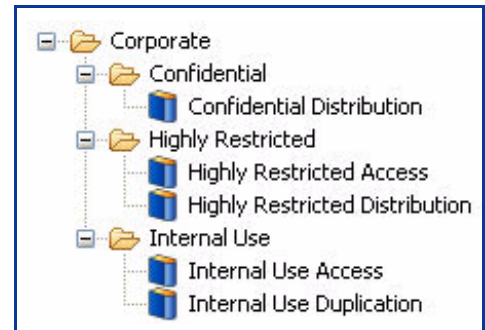
Final Compensation Plan Updates: A standard access policy, which blocks everyone but Compensation Analysts from making any modifications to any files in the Final Compensation Plan directory. This creates a lock-box mechanism: a place where Engineering and Sales Managers can place their final recommendations, but lose any access once they do. The previous policy sends a notification to the compensation analyst when this hand-off occurs. Again, this is related to enforcing workflow.

Final Compensation Policy Retention: Permanently blocks any user from deleting a finalized compensation plan. Note that Compensation Analysts is permitted to modify these plans, but are blocked by this policy from deleting them. This is useful in enforcing document retention requirements.

Policy Set 5: Corporate

This set defines access and use boundaries around three classes of information: Confidential, Highly Restricted, and Internal Use. These categories fall into the following logical hierarchy, from least to most sensitive:

- **Confidential:** May be accessed by authorized users and distributed by e-mail or IM, but all instances of distribution will be written to a log for auditing purposes.
- **Internal Use:** May not be copied, moved, or cut & pasted outside a protected location on the network, where it can be accessed only by employees or contractors.
- **Highly Restricted:** May not be copied, moved, or cut & pasted outside a protected location on the network, where it can be accessed only by users with highly restricted clearance.



It consists of five policies, and provides a good example of how you can use folders to organize policies within a set, according to their function:

Confidential Distribution: A Monitor policy, simply logs all instances when users distribute (attach to e-mail or IM) confidential information.

Internal Use Access: Prevents anyone but employees and contractors from accessing documents in the Internal Use protected location.

Internal Use Distribution: Prevents anyone from distributing (by e-mail or IM) documents in the Internal Use protected location.

Highly Restricted Access: Prevents anyone but highly restricted authorized employees from accessing documents in the Highly Restricted protected location.

Highly Restricted Distribution: Prevents anyone from duplicating (moving, copying, or cut & pasting) or distributing (by e-mail or IM) documents in the Highly Restricted protected location.

The Policy Life Cycle

As we have seen, every policy and component has a life cycle. In this section we will use Sample Policy 2 to walk through all possible stages of the policy lifecycle.

1. Draft your new components and policies.

When you create a new policy or policy component, it begins in the Draft state. Draft indicates that the object is being worked on and is not yet ready for deployment. While in the Draft state, changes that you make to an object are automatically saved so that you can safely pause and resume your work without losing changes.

2. Submit your new components and policies.

Once you have reviewed the details of the two policies within the Destiny Project sample and also provided detailed definitions of the policy components, you can indicate that they are ready for deployment by *submitting* them. The access control defined for a policy or policy component (collectively referred to as objects) determines which users have permission to submit the object.

3. View the access control settings for the Destiny Information policy.

In this step, you will view the policy properties, including the Access Control settings, so that you are aware of the permissions that have been granted to various types of users for this object. To do this,

- A. Select the Destiny Information policy.
- B. From the File menu, choose Properties. (You can also right-click on the policy.)
- C. Click the Access Control tab.

4. View deployment targets.

In this step, you will view the deployment targets that have been set up for the policy to understand which policy enforcers will receive the policy when it is deployed. To do this,

- A. From the Actions menu, choose Set Deployment Targets.
- B. If you have the required access permission to submit an object, you will see a “Submit” button in the status bar at the bottom of the screen.

When you submit an object, Compliant Enterprise performs a dependency check and a set of warnings are presented if any of the required policy components are not submitted as well. These warnings can be ignored, but eventually you will not be able to deploy the object until they are resolved. As a convenience, you can check off dependent objects that you would also like to submit at this time.

5. Submit the policy.

In this step, you will submit the policy and related policy components for deployment, indicating that you have completed your edits and the policy is ready to be sent to the policy enforcers.

- A. Click the Submit button in the status bar.
- B. This will perform a dependency check on the deployed object.
- C. Check each of the required components and click the Submit button.

6. Deploy the policy.

Once the policy is submitted, if you have the permission to deploy the object, the Deploy button will display in the status bar. When you deploy an object, you can either deploy the object at the next scheduled system deployment time (default deployment) or select a specific date and time at which the policy should be deployed.

You can also deploy objects by using the “Deploy All. . .” command from the Actions menu. This will provide a list of all objects that are currently in the submitted state and eligible for deployment. This feature can be helpful if you want to deploy a large number of policies or components at the same time.

7. Deploy the Destiny Information policy

In this step, you will schedule the deployment of the policy and related components. At the specified time, the new policy will be automatically distributed to the relevant policy enforcers.

- A. Click the Deploy button in the status bar, or choose Action - Deploy. This will bring up a dialog that allows you to schedule the deployment.
- B. Choose the current date and time and click the OK button.

8. View Deployment Status

In this step, you will view deployment status to see whether the deployment has reached all of the policy enforcers.

- A. Choose Windows - Deployment Status.
- B. Select one of the deployed policy enforcers to view the current set of policies and policy components that are deployed on that policy enforcer.
- C. Check to see whether the Destiny Information policy has been deployed.

It may take several minutes for a deployment to reach all of the targeted policy enforcers. This delay depends on the scheduled deployment time and the configured heartbeat frequency of the policy enforcers (by default, 60 minutes).

9. Test Policy Enforcement

In this step, you will attempt to perform actions on Destiny Confidential Information in order to observe the active enforcement of the policy.

- A. Attempt to access a document within one of the Destiny folders that you created earlier.
- B. You should not be able to access these files. Look for a message from the Desktop Enforcer running in your system tray.

10. Modify

Once you deploy an object, you can come back later to create a modified version of that same object. This is useful in cases where you want to change a policy or modify the definition of a policy component. To do this you move the object into the Modify state. Let's say for example that you want to remove individuals from the Destiny Team.

11. Modify and deploy the Destiny Team Members user component

In this step, you will begin to edit a new version of the previously deployed Destiny Team Members user component. Add yourself to the user component so you can see how your access to information changes.

- A. Open the Destiny Team Members user component.
- B. Click the Modify button in the status bar.
- C. This will make the policy component editable.
- D. Add yourself as a member of the team.
- E. Click the Submit button in the status bar.
- F. Deploy the change by clicking the Deploy button and choosing the current date and time for deployment.
- G. Then confirm that the updated user component has been deployed by choosing Windows - Deployment Status. Remember that this may take several minutes.

12. Test Policy Changes

In this step, you can see the effect of having added yourself to the Destiny team.

- A. Again, try to access documents from one of the Destiny folders.
- B. This time, you should be able to access the documents normally.

13. Deactivate

If at some point a particular policy or policy component is no longer valid, you can *deactivate* that policy so that it is no longer enforced. To deactivate an object, you follow a process that is identical to deploying an object: submit a request to deactivate the object and then deploy the deactivation of the object. Deactivating an object removes it from all of the policy enforcers. If you want to remove a policy from only a subset of policy enforcers, remove targets from the deployment target list.

Moving On

A careful analysis of the sample sets provided in Policy Author, combined with the descriptions provided in this chapter, can give you considerable insight into the various approaches to designing policy sets and components that will exactly meet your information control requirements. However, much more advanced information on how you can proceed with policy design strategies, and how you can take advantage of the available predefined Compliant Enterprise solutions, is provided in the *Compliant Enterprise 2.0 Implementation and Solutions Guide*.

A

How Do I . . . ?

This appendix provides quick reminder of how to perform commonly required actions or procedures. It is organized in tables, focusing on the following areas:

- Working with Components ([page 97](#))
- Working with Policies ([page 98](#))
- Monitoring the System ([page 99](#))

Table A-1: Working with Components

How do I . . .	In Policy Author:
Create a new component?	In the component bin, select one of the five component types, click the New button, provide a name, and define the component in the component editor pane.
Delete an existing component?	Right-click on the component and select Deactivate (if it is active), then right-click and select Delete. These commands are also available in the Actions and Edit menus, respectively.
View more than one component at a time?	Open two components, click to grab the tab of one, and drag it to the bottom or to either side of the editing pane. You can tile multiple components this way.
Rename a component?	Right-click on the component and select Properties. On the General tab, type the new name, then click Save. (page 74)
Deploy a new or changed component?	Individual components: click the Submit button if not yet submitted; then the Deploy button. Or, deploy all submitted components with the Actions menu, Deploy or Deploy All commands.
Manually control where I deploy a component?	Right-click on the component and select Set Deployment Targets. This command is also available in the Actions menu.
Find out which policies a component is used in?	Right-click on the component and select Show Policy Usage. This command is also available in the Actions menu.
Find out which enforcers a component is deployed to?	Select the component, then open the Deployment Status window (under the Tools menu). This window has three tabs, listing the File Server Enforcers, Desktop Enforcers, or Portal Enforcers where the component is deployed.
Control who can use a component in their policies?	Right-click on the component and select Properties. On the Access Control tab, add or remove users or user groups from the list. For each user or group, you can set individual permissions. (page 74)
Control how others can use a component in their policies?	Right-click on the component and select Properties. On the Access Control tab, choose the user or group you want to edit, then select or deselect any of the six available Permissions. (page 74)
Find out what version of a component is deployed?	Right-click on the component and select Show Deployed Version. This command is also available in the Actions menu. (page 72)
View previously deployed versions of a component?	Select the component, then Actions menu, Version History command. (Disabled if only one version of the component was ever deployed.)

Table A-1: Working with Components (Continued)

How do I . . .	In Policy Author:
Know whether any applications are being globally ignored?	Right-click on the special Monitor All Applications—Exceptions component, and select Show Deployed Version. Any applications displayed in this list will be ignored by all currently deployed policies. If this component is not deployed, no applications are being ignored.
Test whether I have defined a component correctly?	Select the component, open the Preview window (Window --> Preview), then click the Go button. Not available for action components.
Enroll new entities to use as components?	These procedures must be performed by a system administrator, using a set of special Compliant Enterprise utilities. Details are provided in the <i>Compliant Enterprise 2.0 Administrator's Guide</i> .
Set up automatic synchronization with my LDAP directory?	
Add or delete LDAP attributes to my enrolled entities?	
Define a site ?	
Define a component to cover a directory in MyDocuments ?	Type the keyword [MyDocuments] in the membership field of a document-type component. See page 28 .
Exclude an application from all policy enforcement?	Use the special application component, Monitor Specific Applications (see page 35).
Monitor activity on only specified applications , ignoring all others?	Use the special application component, Monitor All Applications—Exceptions (see page 35).

Table A-2: Working with Policies

How do I . . .	In Policy Author:
Create a new policy?	Select a folder in the policy tree, and click the New Policy button. Provide a name, and define the policy in the editor pane.
Delete an existing policy?	Right-click on the policy in the policy tree and select Deactivate (if it is active), then right-click and select Delete. Both commands are also available in the Actions menu.
View more than one component at a time?	Open a two components, click to grab the tab of one, and drag it to the bottom or to either side of the editing pane. You can tile multiple components this way.
Rename a policy?	Right-click on the policy and select Properties. On the General tab, type the new name, then click Save.
Manually designate where I want to deploy a policy?	Right-click on the policy and select Set Deployment Targets. This command is also available in the Actions menu. See page 62 .
Deploy a new or changed policy?	For individual policies, click the Submit button if not yet submitted; then the Deploy button. Or, deploy all submitted policies with the Actions menu, Deploy All command.
Cancel a submitted policy?	From the Tools menu, select Deployment History. In the Deployment History window, select the deployment containing the policy you want to delete, and click the Delete Selected Policy button. See page 68 .
Find out which enforcers a policy is deployed to?	Select the policy, then open the Deployment Status window (Window --> Deployment Status). This window has two tabs, listing the File Server Enforcers and Desktop Enforcers where the policy is deployed.
Control who can use a policy?	Right-click on the policy and select Properties. On the Access Control tab, add or remove users or user groups from the list. For each user or group, you can set individual permissions. See page 74 .
Control how others can use a policy?	Right-click on the policy and select Properties. On the Access Control tab, choose the user or group you want to edit, then select or deselect any of the six available Permissions. See page 74 .

Table A-2: Working with Policies

How do I . . .	In Policy Author:
Find out what version of a policy is deployed?	Right-click on the policy and select Show Deployed Version. This command is also available in the Actions menu.
View previously deployed versions of a policy?	Select the policy, then Actions menu, Version History command. (Disabled if only one version of the policy was ever deployed.)
Construct a policy covering all removable media ?	Use the predefined document component RemovableMedia, for either the source or target document of your policy. See page 28 .
Write a policy that controls access and use of file directories , rather than files?	Create a Document component based on the special Include Only Directories property (see page 30).

Table A-3: Monitoring the System

How do I . . .	In Administrator:
Check how many policies are deployed in my network?	Status tab, Status Overview link, Policies statistic (lower left).
Find out what hosts my Control Center server components are installed on?	Status tab, Status Overview link, Server Status pane (at right): Host column.
Check the health of all my Control Center server components?	Status tab, Status Overview link, Server Status pane at right: Last Heartbeat value turns red if expected interval has been exceeded.
Check how many enforcers are currently running?	Status tab, Status Overview link, Policy Enforcers statistics (middle left): shows File Server Enforcers, Desktop Enforcers, and Total.
Find out which enforcers are currently running?	Status tab, Policy Enforcer Status link: set Show filter to <i>All Policy Enforcers</i> .
Check if any enforcers are disconnected ?	Status tab, Status Overview link, Policies Statistics (lower left). See REF.
Find out which enforcers are disconnected ?	Status tab, Policy Enforcer Status link: set Show filter to <i>All Policy Enforcers with Warnings</i> ; then check the leftmost column for enforcers with Warning icons (exclamation points). See REF.
Check for enforcers that are using obsolete policies ?	Status tab, Status Overview link, Policy Consistency statistic (upper left). See REF.
Find out which enforcers are using obsolete policies ?	Status tab, Policy Enforcer Status link: set Show filter to <i>Policy Enforcers with Warnings</i> , look for red X icon in Policy Up-to-Date column. See REF.
Tell how often policies are being enforced?	Status tab, Status Overview link, Policies statistic (lower left). See REF.
Find out which profile an enforcer is using?	Status tab, Policy Enforcer Status link: set Show filter to <i>All Policy Enforcers</i> , check Profile Name column. See REF.
Find out current heartbeat setting for an enforcer?	1. Find which profile the enforcer is using (see above). 2. Policy Enforcer Configuration tab, Desktop Enforcers or File Server Enforcers link, locate the profile in the list at left, click the Settings tab, check the Heartbeat Frequency setting.



Index

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Symbols

- ! (wildcard character)
in document components 32
- * (wildcard character)
in document components 32
- ** (wildcard character)
in document components 32
- ? (wildcard character)
in document components 32

A

- abstraction 18
- access control levels
for sample applications 87
in example 98
setting 78
- ACLs, setting 78
- ACPL
parts of speech 88
- action components
adding to policies 54
- Action menu
Check Dependencies 68
Set Deployment Targets 69
Version History 79
- Add button 69
- Add Users or Groups 38
- adding
users
to a policy 38

- Administrator user
and sample applications 87
- All Computers with Agents 39
- Allow Only 51
- application components
defining 40
predefined 41
- applications
blocking different versions 41
excluding from monitoring 42
monitoring 42
- Applications that Start Automatically (predefined
component) 41
- attach to e-mail (basic action) 49
- attach to item (basic action) 49
- Autodeployment 22, 69
- autorun prevention 41

C

- canceling deployment 81
- canned components
all computers with agents 39
Removable Media 34
- case sensitivity
of document component names 35
of Find tool 72
- change attributes (basic action) 48
- change file permissions (basic action) 48
- Check Dependencies command 68
- compliance notifications 56
- component panels
description of 15

- components
 - allowed characters in names 26
 - deactivating 82
 - deleting 83
 - ignored applications 42
 - membership
 - OR / AND expressions 29
 - permissions 78
 - possible states 23, 74
 - renaming 80
 - viewing properties 78
- computer components
 - properties of 40
- connecting to Control Center 13
- copy (basic action) 49
- Corporate View
 - definition 12
- create/edit (basic action) 49
- creating
 - folders 62
 - policies 51
 - policy components 25
- custom obligations 61
- custom properties
 - of document components
 - working with 36

D

- Date and Time (policy context) 55
- deactivating 82
 - sample application 101
- default
 - deployment time 70
 - properties
 - of computer components 40
 - of document components 35
 - of portal content components 45
 - of user components 39
- Delete
 - user permission 81
- delete (basic action) 48
- deleting
 - policies and components 83
- Deny 51
- deny override 90
- dependency checking 66, 67, 70

- Deploy
 - user permission 81
- Deploy All command 66
- Deploy All feature 66
- Deploy button 70
- Deploy dialog box 67
- Deployed Policy Enforcers tab 76
- deployed status 23, 74
- deployment 22
 - automatic 69
 - canceling 81
 - default time 67, 70
 - definition 65
 - manual 69
 - overview 65
 - sample application 99
 - scheduled 70
 - scheduling 67
 - targets, setting 68
 - types 69
 - viewing past 75
- Deployment History window 73, 81
- Deployment Start Time 70
- Deployment Status command 74
- deployments
 - status 74
- Destiny Confidential Information 90
- development process 23
- docking
 - Policy Author tabs 16
- document components
 - custom properties
 - working with 36
 - default properties 35
 - keywords 34
 - memberships 32
- draft (status) 23

E

- effects 89
 - Allow Only 89
 - Deny 89
 - Monitor 60, 89
- e-mail 55
 - notification
 - configuring 56, 58

- contents of 59
 - setting the From field 60
- e-mail (basic action) 49
- excluded applications 42
- Explorer
 - as excluded application 43
- export (basic action) 49
- exporting
 - policies 62

F

- features, new in 2.0 7
- File System View
 - definition 12
- folders
 - access control 78
 - creating 62
 - Sample 87
 - viewing properties 78
- for components
 - ACLs, setting 78
- From field (notification e-mail) 56, 58
 - configuring 60
- Full Name (document property) 35

G

- General tab (properties) 78
- groups
 - searching for 38

I

- ignored applications 42
- IM 49
- importing
 - policies 62
- In (operator) 28
- inactive status 23, 74
- Include Only Directories (document property) 36
 - in the Preview Pane 17
- Information Network Directory 30
- instant messaging 49

K

- keywords 19
 - for document components 34

L

- lifecycle 23, 98
 - control bar 23
- Linux file servers
 - in Set Deployment Targets window 69
 - in Version History window 76
- Log checkbox 56
- logging 55
- logging in
 - Policy Author 13
- logging out
 - Policy Author 13

M

- Management 63
- manual deployment 69
- member search tool 38
- membership
 - description of 27
 - multiple
 - OR /AND expressions 29
- Message (in obligations) 56, 58
- message to user 55
- minus icon 79
- Missing (dependent component) 68
- Modified (dependent component) 68
- modified components 71
- Monitor (obligations) 55
- Monitor All Applications--Exceptions 42
- Monitor policies 51, 89
- Monitor Specific Applications 42
- move (basic action) 48
- Moved, Renamed, or Copied (policy field) 54
- MyDesktop keyword 19, 34
- MyDocuments keyword 19, 34

N

name
 document component property 35
 new features in 2.0 7
 new policy component 26
 Not in (operator) 28
 notification e-mail
 configuring 56, 58
 contents of 59

O

object
 of policies 54
 object components
 membership and properties 27
 properties of 39
 obligations
 custom 61
 defining 55
 On Allow (obligations) 55
 On Deny (obligations) 55
 on-allow message 56, 58
 on-deny message 56, 58
 open (basic action) 48
 operators
 component definition 28
 overview
 editing policy components 15, 20

P

parts of speech 88
 subject 90, 93
 Paste
 limited obligation support 49, 57
 paste (basic action) 49
 pending
 deactivation 23, 74
 deployment 23
 peripheral devices, controlling 54
 permissions
 setting 78
 plus icon 52, 79

policies
 ACLs, setting 78
 deactivating 82
 deleting 83
 effects 89
 importing and exporting 62
 life cycle 98
 modifying 79
 permissions 78
 possible states 23, 74
 renaming 80
 statuses of 23
 submitting 69
 viewing properties 78
 where used 77

Policy Author
 definition 11
 docking tabs 16
 editing tools 20
 lifecycle control bar 23
 logging out 13
 UI panes 15

policy components
 creating 25
 deactivating 82
 modifying 79
 overview of editing 15, 20
 sample application 91
 submitting 69

policy tree 15

portal content components
 and lookup button 29
 default properties 45
 defining 43
 membership 43
 properties of 46

Portal View
 definition 12

predefined components
 applications 41

preventing autostarting applications 41

Preview pane 31
 using 17

print (basic action) 49

properties
 General tab 78
 of computer components 40
 of document components
 custom
 working with 36

- default 35
- of portal content components 46
- of user components
 - default 39

Properties window 78

R

Read

- user permission 81

Removable Media 54

Removable Media component 34

rename (basic action) 48

Required (dependent component) 68

resource

- from and to 93

S

Sample folder 87

- Compensation Planning policies 96
- Corporate policies 97
- Destiny Project policies 88
- Sales policies 95

scheduled status 74

scheduling deployment 70

searching

- for objects 77
- for users and groups 38

Send E-mail 56

Send in E-mail (basic action) 49

Send in IM 49

Set Access

- user permission 80

Set Deployment Targets command 69

SharePoint Lookup button 43

SharePoint lookup button 44

Show Deployed Version 78

Show Policy Usage command 72, 77

source and destination 93

starting

- Policy Author 13

status of components and policies 23, 74

statuses (components and policies) components

- statuses of 23

stopping

- Policy Author 13

subject 90, 93

- of a policy, defining 52

Submit 69

- user permission 80

submitted

- for deactivation 23
- for deployment 23

syntax

- policy component definition 29

T

target

- for deployment 68

time

- context in policies 55
- time zones 55

To field (notification e-mail) 56, 58

To Resource 54

U

Update Computers with Agents command 40

user components

- default properties 39

user messages 55, 56

User Principal Name 39

users

- searching for 38

V

version

- of enrolled applications
 - blocking 41

version history 79

- viewing 75

viewing

- deployment 75
- deployment history 75

views (in Policy Author)

- description 12

Index

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

W

- walkthrough of example applications 88
- wildcards
 - escape characters for 32
 - in document components 32

- Window menu
 - Deployment History 81
- Windows Explorer
 - as excluded application 43
- Write
 - user permission 80