Oracle® Universal Content Management
Managing Security and User Access
10*g* Release 3 (10.1.3.3.1)

May 2010

Managing Security and User Access, 10*g* Release 3 (10.1.3.3.1)

Contributing Authors: Karen Johnson

Contributors: Peter Walters, Samuel White

Table of Contents

Chapter 1: Introduction

Chapter 2: Designing a Security Model

Chapter 4: Internal Security: Using Accounts

Chapter 5: Internal Security: Assigning User Logins and Aliases

Chapter 6: External Security: Active Directory

Chapter 7: External Security: LDAP

Chapter 8: Proxied Connections

Appendix A: Third Party Licenses

Chapter

1

INTRODUCTION

OVERVIEW

This section covers the following topics:

- ❖ About This Guide (page 1-1)
- ❖ What's New (page 1-2)
- ❖ Audience (page 1-2)
- ❖ Other Administrator Guides (page 1-3)
- ❖ Understanding Content Server (page 1-4)
- ❖ Administration Page (page 1-5)
- ❖ Launching Applications (page 1-6)
- ❖ Conventions (page 1-8)

ABOUT THIS GUIDE

This document discusses tasks related to user administration, such as planning and implementing a security model, adding and deleting users, and implementing accounts. Additionally, it explains how to integrate external user bases with Content Server. The two most common security integrations—Active Directory and LDAP—are described in detail.

WHAT'S NEW

For Content Server 10*g*R3, these features are new:

- **Create Similar Users**: There is a new Create Similar button on the User Admin applet that enables you to create a user login based on the roles and accounts assigned to an existing user. See Assigning Roles to Create Similar Users (page 3-11).
- **Proxy Connections:** Proxy connections provide additional levels of security for Content Server through the following functions:
  - Security credentials mapping from one content server to another content server.
  - Secured "named" password connections to content servers (password protected provider connections).
  - HTTP protocol communication between content servers (HTTP-based proxy servers).

  For more information, see Proxied Connections (page 8-1).
- **Filtering of data for illegal or corrupt HTML constructs:** The encodeHtml Idoc script functions enables the filtering of data for illegal or corrupt HTML constructs. It is especially useful in a WCM environment. See Data Input Filter (page 2-7).
- **Logging in through different web server front ends (HTTP/HTTPS)**: The BrowserUrlPath component is now bundled. If you wish users to be able to access Content Server using different web server front ends, and one server front end is HTTPS and the other is HTTP, the BrowserUrlPath component is recommended. You will need to enable this component to make use of the functionality.
- **Cookie-based logouts:** The ExtranetLook component is bundled with version 10*g*R3 to provide you with the opportunity to set configuration entries to easily add logout functionality. You will need to enable this component to make use of the accompanying functionality. See Login/Logout Customization (page 2-5).

AUDIENCE

This guide is intended for administrators who are responsible for managing network security and content security for Content Server.

OTHER ADMINISTRATOR GUIDES

Administrators set up, maintain, and manage Content Server users, content, and system configurations. Common tasks for an administrator include configuring the system to manage and index files, archiving and replicating information, working with content server security, adjusting system properties, reviewing log files, and so forth.
.i.administrator documentation;
.i.system administrators:documentation for --;
.i.administrators:documentation for --;
.i.documentation:documentation for system administrators;

Documentation for administrators and sub-administrators of Content Server software includes the following:

- *Getting Started (PDF and HTML)*
  This document provides an overview of the Oracle suite of products and general guidelines for their setup and implementation.
- *Managing Security and User Access Guide (PDF and HTML)*
  This document discusses tasks related to user administration, such as planning and implementing a security model, adding and deleting users, and implementing accounts. Additionally, it explains how to integrate external user bases with Content Server. The most common security integrations—Active Directory and LDAP—are described in detail.
- *Managing Repository Content Guide (PDF and HTML)*
  This guide discusses tasks that affect how the content is displayed or handled, such as creating customized content types, using schemas, building a web site, or moving content through a workflow.
- *Managing System Settings and Processes Guide (PDF and HTML)*
  This guide describes tasks that are impact system configuration on an ongoing basis such as managing revisions and indexing, configuring providers, and working with system properties.
- *Administration Tutorials (PDF and HTML)*
  This document contains administration tutorials for people who need to administer (part of) a Content Server-based content management solution.
- *Enterprise Search Administration and User Guide (PDF and HTML)*
  This document provides management and administration information for Enterprise Search. This enables multiple content server instances to be searchable as if they were a single instance.

- *Troubleshooting Guide (PDF and HTML)*
  This document contains general information about troubleshooting a Content Server environment and how to diagnose issues, also provides more in-depth information about troubleshooting in specific areas.

- *Release Notes (hardcopy and PDF)*
  Content Server software is shipped with release notes, which list new and enhanced features of each new software release, and also provide special, up-to-the-minute considerations for installing and using the software. The release notes are important documents. Always make sure you read them before installing or updating Content Server software.

Note: The optional add-ons to Content Server generally have their own administration documentation, which is included as PDF files on the add-on distribution media, typically in a /documentation directory.

UNDERSTANDING CONTENT SERVER

This section covers these topics:

- Purpose (page 1-4)
- Users (page 1-5)

Purpose

Content Server is used for sharing, managing, and distributing business information using a web site as a low-cost access point.

Designed for the web, this software is considered the unrivaled solution for medium to large companies for building secure business libraries with content check in, check out, revision control, and automated publishing in web-ready formats. Current information is available to authorized users anytime, anywhere. You can link virtually any type of file—letters, reports, engineering drawings, spreadsheets, manuals, sales literature, and more—under one powerful system of knowledge distribution.

Users

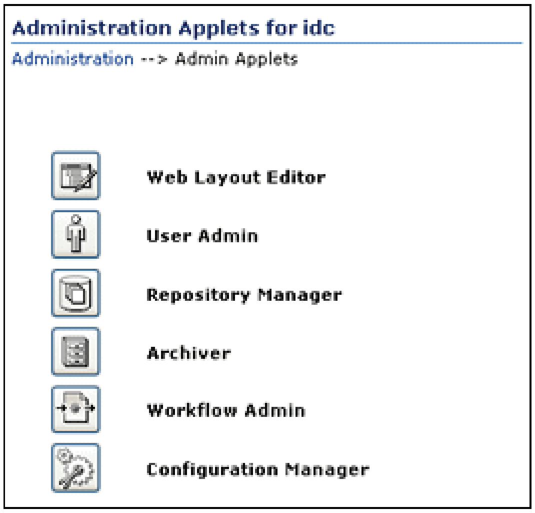
Content Server is designed for two types of users and two types of administrators:

- **Consumers**: Users who just need to find, view, and print files.
- **Contributors**: Users who need to create and revise files.
- **Administrators**: Administrators who oversee an entire instance.
- **Sub-administrators**: Administrators who oversee a subset of an instance.

In a typical system, the majority of the users are consumers. These users do not need a user name and password to access the content server system unless security is placed on the files. To safeguard the integrity of the files, the contributors need a user name and password to check files in and out of the system.

Typically, the majority of administrators are sub-administrators. They administer portions of the software that correspond to the rights that the system administrator assigns to them.

Administration Page


Administration Applets for idc

Administration --> Admin Applets

This page provides access to administration applets and configuration tools. To access this page, log in as an administrator or sub-administrator, and click the Administration tray in the portal navigation bar. Then, click the Admin Applet link.

Note: You may experience problems if you start any Java applets (such as a Content Server administration applet or the multiple-file upload applet) from a browser that is using Sun’s JDK 1.3/1.4 Java plug-in. These issues are related to authentication when launching an applet for the first time and applets closing when the parent window is changed.

Launching Applications

You can launch Content Server’s administration applications using these methods:

- Running Applications as Applets (page 1-6)
- Running Applications in Stand-alone Mode (page 1-7)

Running Applications as Applets

You can run several of the Content Server administration applications as applets from any browser with access to the content server. Applets are convenient for remote administration.

The Batch Loader, Component Wizard, System Properties, and Content Server Analyzer utilities cannot be run as applets; for security reasons, they must be run in stand-alone mode from the computer where the content server is installed. See Running Applications in Stand-alone Mode (page 1-7) for details.

Some functions that are available in the stand-alone version of an application are not available from the applet version. See the documentation for each application for more information.

To run an administration application as a Java applet within a Java-enabled browser:

1. Open a browser window.
2. Log in to the content server as an administrator.
3. Click the Administration tray link in the portal navigation bar.
4. Click the Admin Applets link.

Running Applications in Stand-alone Mode

You can run many Content Server administration applications in stand-alone mode from the computer where Content Server is installed. The method required to start these programs differs slightly between Windows and UNIX installations.

Running the stand-alone version of an application offers greater security than browser applets, and enables you to send passwords without having them captured or copied from the web or a network.

On Windows Systems

To run a stand-alone administration application on a Windows operating system:

1. Select the application from the Windows Start menu:
   - Select **Start—Programs—Content Server—*instance*—Applications—*application***.
   - To run one of the administration Utilities, select **Start—Programs—Content Server—*instance*—Utilities—*utility***.

   For all applications except for Component Wizard and System Properties, a login screen is displayed. For Component Wizard and System Properties, the main screen of the application is displayed.

Tech Tip: It may take several seconds for the login screen or the application screen to appear, and the screen may be hidden by other windows.

2. Enter the administrator login name and password.
3. Click **OK**.

   The main screen of the application is displayed.

On UNIX Systems

To run a stand-alone administration application on a UNIX operating system:

1. Navigate to the <*Install_Dir*>/bin/ directory.
2. Executable applications are listed. Enter ***/application_name***, where *application_name* is the name of one of the executable files. If an application is not listed, it can be entered as a parameter to the IntradocApp application, as in this example:

   ```
   %<Install_Dir>%/bin/intradocApp workflow
   ```

3. For all applications except for Component Wizard and System Properties, a login screen is displayed. For Component Wizard and System Properties, the main screen of the application is displayed.
4. Enter the administrator login name and password.
5. Click **OK**.

   The main screen of the application is displayed.

CONVENTIONS

The following conventions are used throughout this guide:

- The notation *<Install_Dir>/* is used to refer to the location on your system where the content server instance is installed.
- Forward slashes (/) are used to separate the directory levels in a path name. A forward slash will always appear after the end of a directory name.
- Notes, technical tips, important notices, and cautions use these conventions:

Symbols	Description
💡	This is a note. It is used to bring special attention to information.
⚙	This is a technical tip. It is used to identify information that can be used to make your tasks easier.
❗	This is an important notice. It is used to identify a required step or required information.
✖	This is a caution. It is used to identify information that might cause loss of data or serious system problems.

Chapter

2

DESIGNING A SECURITY MODEL

OVERVIEW

This section covers the following topics:

LEVELS OF SECURITY

Oracle Content Server offers two levels of content security: *security groups* (which are required) and *accounts* (which are optional). Every content item is assigned to a security group, and if you have enabled accounts, content items can also be assigned to an account. Users are assigned a certain level of permission (Read, Write, Delete, or Admin) for each security group and account, which enables them to work with a content item only to the extent that they have permissions to the item's security group and account.

SECURITY OPTIONS

Content Server provides these security options:

- Internal Security (page 2-2)
- External Security (page 2-3)
- Additional Security Options (page 2-4)

Internal Security

You can set up user security within Content Server using the User Admin application. You assign each user to one or more *roles*, which in turn are assigned specific permissions to security groups. If you have enabled accounts, you can assign each user specific permissions to certain accounts, which then limits the permissions they might otherwise have through their assigned roles.

The following components also can be used to provide additional internal security:

- You can customize security for user access by using the ExtranetLook component, which can be installed and enabled when you install Content Server. See Login/Logout Customization (page 2-5) for details.
- You can customize security for user access and search results by using the Need to Know component. This component enables you to further configure user access restrictions, modify the display of search results, alter search behavior, and set up *hit list* roles.

Note: MS Internet Explorer 7 supplies a rather threatening message to users logging in with basic authentication without a secure connection: *Warning: This server is requesting that your username and password be sent in an insecure manner.* The message is new. The behavior (sending usernames and password in text) is not new for basic authentication and does not cause problems.

See Also

– *Chapter 3 (Internal Security: Security Groups, Roles, and Permissions)*

External Security

User logins, passwords, and permissions are derived from one of the following external user bases:

- **Active Directory—**User information is stored in a Microsoft Active Directory user base. See Chapter 6 (*External Security: Active Directory)*.
- **LDAP—**User information is stored in an LDAP-compliant user base, such as iPlanet. See Chapter 7 (*External Security: LDAP)*.
- **Active Directory with LDAP—**User information is stored in a Microsoft Active Directory user base, which the Content Server accesses using an LDAP provider. The Active Directory LDAP component is required for this type of security integration, and it is provided with the Content Server installation. See Chapter 7 (*External Security: LDAP)*.

External Users

Users who are authenticated through external security (Active Directory or LDAP) are considered *external* Content Server users. The first time an external user logs in they are added to the database, and administrators can view external user information through the Repository Manager. However, external users are not included in user lists, such as the Author field on a content Check In page.

Note: If an Override check box is selected on a user's User Profile page, any user information defined in the Content Server database overrides the user information derived from the external user base.

By default, external security integrations map only a limited set of user information (user name, password, roles, and accounts) from the external user base to the content server. If you are using Active Directory or LDAP integration, additional user information, such as e-mail address or user locale, can be mapped from a Content Server administration page. More sophisticated mapping techniques, such as mapping a domain prefix to a specific account name, can be provided through use of proxy connections. Chapter 8 (*Proxied Connections)* provides information on how to map credentials from one Content Server to another.

Microsoft Login

If you integrate Active Directory security with Content Server, a **Microsoft Login** button on the portal navigation bar enables users to log in to Content Server without having to re-enter their user name and password. Clicking the standard Login button will prompt the user for a user name and password.

See Also

– *External Security: Active Directory (page 6-1)*

Additional Security Options

A content management system can combine authentication methods. For example, you can define some local users in Content Server, allow some users to log in using their Microsoft domain identity, and grant other users Content Server access based on their Active Directory or LDAP credentials. The following options can be used to provide additional security:

- ❖ You can customize Content Server to filter data input for illegal or corruptive HTML constructs. For more information see Data Input Filter (page 2-7).
- ❖ You can use proxied content servers, which can have different security integrations from the master content server. For example, you could define only internal users for a master contribution server, but set up an LDAP security integration for the proxied consumption server. You also could support enterprise searching on multiple content servers by using proxy connections to set up credential mapping and HTTP protocol communication between the master and proxied servers. For more information see Chapter 8 (*Proxied Connections)*.
- ❖ If you wish users to be able to access Content Server using different web server front ends, and one server front end is HTTPS and the other is HTTP, you can customize your configuration with the BrowserUrlPath component, which can be installed and enabled when you install Content Server. This component supports a web server front end using HTTPS and a load balancer that forwards itself as the HTTP Host header. If you only use one access method (only HTTPS, or only HTTP), or you are not using a load balancer that blocks the “Host” parameter from the browser, then this component is unnecessary. For more information see Browser URL Customization (page 2-9).
- ❖ You can customize security to support web communication encryption and authentication by using the Security Providers component. This component enables a Secure Socket Layer (SSL) provider, which can be configured to use certificates for socket or server authentication.

Note: If you use SSL and HTTPS to connect to Content Server, and are unable to connect through WebDAV, try connecting to the content server through the browser using the same URL you used in your WebDAV connection string. This will let you see if there is a problem with the certificate, which is used to encrypt communications. If you get a dialog box stating a problem with the certificate, resolve the issue and then try to connect through WebDAV again.

Tech Tip: In all environments, a comprehensive understanding of your organization's security needs and a thorough planning phase is crucial to a successful security integration. See the *Planning and Implementation Guide* and *Getting Started Guide* for more information on content server security. We also recommend that you contact Consulting Services for further assistance when planning and implementing your security model.

Login/Logout Customization

Note: This functionality is only available if you installed and enabled the ExtranetLook component when you installed the Content Server.

The ExtranetLook component, available with Content Server, can be used to customize user access in two ways: by enabling cookie-based login forms and pages, and by suppressing the interface for users who are not authenticated by the Content Server via error and challenge pages issued by the web server. This section discusses setting up cookie-based authentication and logouts. See *Modifying the Content Server Interface* for details on changing the interface.

The default user authentication does not allow for customization or logout functionality. In order to end a Content Server session, users close the browser session. A built-in web server plug-in, `CookieLoginPlugin.dll`, can monitor requests and determine if a request is authenticated based on cookie settings. When user authentication is cookie-based, logout functionality can be implemented without the need to end the browser session.

Set the following configuration variables in the <*Install_Dir*>/config/config.cfg file in order to enable cookie-based authentication:

- `CookieProxyPassword`: Used to encrypt the passwords when users log in to the web site using cookie-based authentication. You can change it to any value.
- `CookieAuthTimeoutInDays`: If set to a positive value, the cookie does not time out for that number of days. A value of zero (0) or less specifies that the cookie lasts only for

the browser session. If the value is set to a positive value, then a logout option is required that clears the cookie.

If you need a much shorter time out, the configuration variable `CookieAuthTimeoutInMins` can be used to specify time in minutes. This variable is available only with version 10.1.3.3.0 or later of the `CookieLoginPlugin.dll` web server plug-in; however, it is usable by earlier versions of Content Server.

- `IsWebServerPagesOnly`: Used to set basic authentication, which does not allow customization or logout functionality. You can reduce the functionality of the web server plug-in by setting `IsWebServrePagesOnly` to `TRUE`.

If you rename the `CookieLoginPlugin.dll` plug-in, you must change the `iapFileNameRoot` entry, `CookieLoginPlugin`, to the new name. The `iapFileNameRoot` entry is the name of the `.dll`, `.so`, or `.sl` file without the extension or the directory path. Both the extension and the directory are computed by the web server filter based on its knowledge of the native operating system. The `iapFileNameRoot` entry is located in the extranetlook_resource.hda file in the <*Install_Dir*>/custom/ExtranetLook/resources directory.

The following files can be used to customize your login/logout configuration:

- access_denied.htm
- login_page.htm
- prompt_login.htm
- redirect_after_url.htm
- report_error.htm

These files have parameters of the form `<!--$ParameterName-->`, which allow HTTP headers, computed variables in the filters, and configuration parameters to be referenced.

Note: Only substitution and simple IF conditionals are allowed in these files. The conditionals can test only one variable and do not support either ELSE or ELSEIF constructions. The plug-in filter does not support any other Idoc Script functionality for cookie-based logins; specifically, it does **not** support includes.

Note: Cookie-based authentication does not work satisfactorily in certain situations. For example, if you log in as sysadmin using cookie-based authentication and try to upload a component to install on the content server, the admin server uses a different host address to upload the component than the host address used when you logged in, so the upload is not performed.

For further information about variables and enabling the component, see the *Idoc Script Reference Guide* and the *Content Server Installation Guide* for your operating system. See *Modifying the Content Server Interface* for details about altering the look and feel of the Content Server interface.

Data Input Filter

The Content Server can be customized to filter data input for illegal or corruptive HTML constructs by using the `encodeHtml` Idoc Script function and a filter hook to automatically scrub all input data for dangerous HTML constructions. The `encodeHtml` function can be applied to a specific string. The `HtmlDataInputFilterLevel` configuration variable can be used to apply a level of encoding to filter all data input to the Content Server.

See the following topics:

- encodeHtml Function (page 2-7)
- HtmlDataInputFilterLevel Configuration Variable (page 2-8)

encodeHtml Function

The `encodeHtml` Idoc function can be used to filter data input for illegal or corrupted HTML constructs. The output is an encoded string. The `encodeHtml` function is applied by default to the discussions in the Threaded Discussions component.

The `encodeHtml` function is generally used at the `exceptsafe` or higher level of encoding because the `HtmlDataInputFilterLevel` configuration variable will already have been encoded as `unsafe` (assuming it uses the default configuration).

The `encodeHtml` function is defined as follows:

`encodeHtml` (*string*, *rule*, *wordbreakrules*)

- **string**: The string to encode.
- **rule**: The rule to apply when encoding HTML constructs. The following values are allowed:
  - `none`: No conversion is done to HTML constructs.
  - `unsafe`: Only well-known unsafe script tags are encoded. The list includes: script, applet, object, html, body, head, form, input, select, option, textarea.
  - `exceptsafe`: Only well-known safe script tags are **not** encoded. The list includes: font, span, strong, p, b, i, br, a, img, hr, center, link, blockquote, bq, fn, note, tab, code, credit, del, dfn, em, h1, h2, h3, h4, h5, blink, s, small, sub, sup, tt, u, ins,

kbd, q, person, samp, var, ul, li, math, over, left, right, text, above, below, bar, dot, ddot, hat, tilde, vec, sqrt, root, of, array, row, item.

- `lfexceptsafe`: (Recommended where extended comments are entered by a user and they want to preserve the line feed breaks of the original text.) Similar to `exceptsafe`, however, line feed (ASCII 10) characters are turned into HTML break tags (br). Line feeds inside of HTML tags are **not** turned into break tags. The following script tags that are safe with `exceptsafe` are **not** safe with `lfexceptsafe`: br, p, ul, li.

Except for the rule `none`, all the rules have special HTML comment handling. In particular, all HTML comments are allowed through the filter. However, when inside an HTML comment, all less than (<) and greater than (>) symbols are encoded. This does not apply to the HTML closing signature (-->). Also, if there is an unterminated comment, the encoding function appends the HTML comment close signature (-->).

Additionally, except for the rule `none`, any attribute value located inside a tag has any parenthesis encoded to `%28` (for '(') or `%29` (for ')'). Otherwise, if any character is escaped it is escaped using the XML (`&xxxx;`) type encoding.

wordbreakrules: This is an optional parameter that specifies if long strings without space characters are to be broken up and what maximum word size to apply. Either the string `wordbreak` or `nowordbreak` can be specified. This parameter can be used with any of the `encodeHtml` rules. The default is to turn on `wordbreak` if the rule `lfexceptsafe` is specified, and to use a `maxlinelength` of 120 characters.

The additional parameter `maxlinelength=`*xxx* can be used with the `wordbreak` parameter to specify a desired maximum line length. For example:

```
encodeHtml ("exceptsafe", "<bad> text", "wordbreak, maxlinelength=80")
```

The `wordbreak` functionality is only usable by the `encodeHtml` function because the function is used for display and not applied before the data is stored.

For information about Idoc Script see the *Idoc Script Reference Guide*.

HtmlDataInputFilterLevel Configuration Variable

The `HtmlDataInputFilterLevel` configuration variable can be used to apply a level of encoding to filter all input data to the Content Server for bad HTML constructions. The HtmlDataInputEncodingRulesForSpecialFields table in the std_resources.htm file is used for special case encoding rules and may override this configuration entry for certain parameters.

Note: If you change the HtmlDataInputFilterLevel value, you need to restart the Content Server.

Using the HtmlDataInputFilterLevel variable has no effect on the behavior of the Idoc Script encodeHtml function.

You can set the HtmlDataInputFilterLevel configuration variable to the following values:

- none—(Not recommended.) All filtering is turned off.
- unsafe—(Default. Recommended.) Protects against bad HTML constructions. Examples of bad constructions include: script, applet, object, html, body, head, form, input, select, option, textarea.
- exceptsafe—(Not recommended.) Allows only well known safe constructions through the filter. If exceptsafe is chosen, then the unsafe option will be applied to requests using GET style requests. Doing a higher level of encoding on GET requests breaks Content Server operation because <$...$> and other tags are routinely passed in as part of the parameter data or URLs. The higher level of filtering is only applied to non-scriptable services (those services that are usually called with POST).

  Examples of well known safe constructions include: font, span, strong, p, b, i, br, a, img, hr, center, link, blockquote, bq, fn, note, tab, code, credit, del, dfn, em, h1, h2, h3, h4, h5, blink, s, small, sub, sup, tt, u, ins, kbd, q, person, samp, var, ul, li, math, over, left, right, text, above, below, bar, dot, ddot, hat, tilde, vec, sqrt, root, of, array, row, item.

See the encodeHtml Function (page 2-7) rule description for information about HTML comment handling, which also applies to HtmlDataInputFilterLevel configuration values.

Note: The value lfexceptsafe is not supported for the HtmlDataInputFilterLevel configuration variable. It is only supported with the encodeHtml function.

Browser URL Customization

Note: This functionality is only available if you installed and enabled the BrowserUrlPath component when you installed the Content Server.

The BrowserUrlPath component provides support for determining URL paths used in certain configurations of Oracle Content Server and web servers.

The following topics are discussed in this section:

- About BrowserUrlPath Customization (page 2-10)
- Affected Idoc Script Variables and Functions (page 2-11)
- Determining the URL Path (page 2-12)
- Changing Absolute Full Path Computation (page 2-14)
- Changing Administration Path Computation (page 2-14)

About BrowserUrlPath Customization

This component overrides certain Idoc Script variables and functions, adds computation to certain variables, and provides additional configuration entries for determining URL paths.

- You can configure a system with different web server front ends. One front end can use HTTP and the other can use HTTPS so that the Content Server can be accessed simultaneously by websites using HTTP and HTTPS. You then need to apply the BrowserUrlPath component to enable the Content Server to handle both types of access.
- If you are using a load balancer that forwards itself as the HTTP host header, then you need to apply the BrowserUrlPath component.

BrowserUrlPath configuration variables are located in the <*instance*>/custom/components/browserurlpath/config.cfg file.

Caution: The BrowserUrlPath component requires extensive configuration using the variables. You may wish to back up your configuration before modifying variables.

Note: The BrowserUrlPath component is only supported with Trays and Top Menu layouts for the Content Server user interface. It is not supported with the Classic layout.

In typical scenarios, the web server will forward to the Content Server two critical pieces of information:

- HTTP_HOST: The host header that the browser sends, identifying the host as it appears to the user in their browser address bar.
- SERVER_PORT: The port the browser uses in connecting to the Content Server.

The browser-based full address is used for two critical pieces of functionality:

1. Automatic creation of URLs in the left-hand frame of the Trays layout for the Content Server. In particular, the left-frame mini-search requires a prediction of the full URL, not just the relative URL.
2. The secondary URL (the #xml-http... piece following the PDF URL) that does highlighting for PDF documents.

Without any additional configuration, the BrowserUrlPath component augments the functionality of certain variables, so if SERVER_PORT has the value 433, then the component assumes the protocol is HTTPS instead of HTTP. Likewise, if SERVER_PORT does not have the value 433, then the component assumes the browser issued the request using HTTP and not HTTPS. This enhancement allows both a SSL (HTTPS) and non-SSL web server (HTTP) to access the same Content Server.

This component also has special functionality for WebDAV access. The configuration entry WebDavBaseUrl is augmented so that its usage is dynamic (its host and protocol vary using the "absolute" path rules).

Caution: The functionality for WebDAV access alters the behavior of CHECKOUT and OPEN functions on some Content Server pages, and alters some behavior in the Site Studio client.

Affected Idoc Script Variables and Functions

The BrowserUrlPath component overrides the computation of the following Idoc Script variables and functions:

- HttpBrowserFullCgiPath
- HttpWebRoot
- HttpCgiPath
- HttpEnterpriseWebRoot
- HttpEnterpriseCgiPath
- HttpAdminCgiPath
- HttpImagesRoot
- proxiedCgiWebUrl
- proxiedBrowserFullCgiWebUrl

The BrowserUrlPath component adds computation for the following variables:

- HttpBrowserFullWebRoot: Defines the full URL path to the web root of the current Content Server using values supplied from the user's current browser's address bar. This variable is similar to HttpBrowserFullCgiPath except it is for the web root instead.
- HttpAbsoluteWebRoot: Defines the universal full URL path to the web root of the current Content Server. It can have a different protocol or hostname than the path in HttpBrowerFullWebRoot. For example, if the user specifies an IP address for the hostname, the HttpBrowserFullWebRoot variable might pick up the IP address, but the HttpAbsoluteWebRoot variable would ignore it and use the appropriate internally configured hostname.
- HttpAbsoluteCgiPath: Defines the universal full dynamic root URL for the current Content Server. This is the path that executes the plug-in code in the web server that makes calls for dynamic content from the Content Server. It can have a different protocol or hostname than the path in HttpBrowserFullCgiPath. For example, if the user specifies an IP address for the hostname, the HttpBrowserFullCgiPath variable might pick up the IP address, but the HttpAbsoluteCgiPath variable would ignore it and use the appropriately internally configured hostname.
- HttpAbsoluteEnterpriseWebRoot: Defines the universal full URL path that includes only protocol, hostname, and optionally the port number. It can have a different protocol or hostname than the path in HttpEnterpriseWebRoot. For example, if the user specifies an IP address for the hostname, the HttpEnterpriseWebRoot variable might pick up the IP address, but the HttpAbsoluteEnterpriseWebRoot variable would ignore it and use the appropriate internally configured hostname.

In the case of the browser path variables HttpBrowserFullCgiPath and HttpBrowserFullWebRoot, the implementation code determines what the user is currently using for protocol (HTTP versus HTTPS), port number, and hostname in the browser. It bases this determination on what the web server receives in its request.

Determining the URL Path

The BrowserUrlPath component supports the following configuration entries for guessing the URL path as the browser determines it:

- HttpIgnoreWebServerInternalPortNumber: When set to true, this disables the use of the SERVER_PORT parameter. This entry is useful in a load balancing scenario where SERVER_PORT is not the port used by the browser, but is the port used by the load balancer to communicate with the web server. Enabling this entry will make it impossible (without the BrowserUrlPath component) for the Content Server to determine which port the browser used to access the web server. Without additional BrowserUrlPath configuration, this variable makes it impossible to both support an SSL and non-SSL address to the same Content Server. Using this variable prevents a load balancing configuration problem in which the load balancing server is using a different port number than the internal web server actually delivering the response to the request.

- `HttpIgnoreServerNameForHostName`: When set to true, this disables the fallback logic where if the HTTP_HOST parameter is missing, the Content Server will typically look for the parameter SERVER_NAME (the web server's self identification).
- `HttpBrowserSSLPort`: Only use this configuration entry if the SERVER_PORT entry is forwarded to the web server that communicates to the Content Server. This entry is used to decide whether a request is HTTPS or HTTP by comparing it with the SERVER_PORT parameter. The default SERVER_PORT value is 443. If you use HTTPS, but use a port other than 443, you need to use this entry to set the expected HTTPS port number.
- `HttpBrowserUseIsSslCookie`: If you want to look in the cookie to see if it indicates whether to use SSL or not, set this entry to true.
- `HttpBrowserIsSslCookieName`: Only use this entry if the `HttpBrowserUseIsSslCookie` entry is enabled. Set the entry to the name of the cookie used to determine whether the server believes the browser is using SSL or not. The default is the cookie name `UseSSL`. The value of the cookie can be 1 or 0 (zero). If a cookie with this name is present, then it will supersede other rules for determining whether to use SSL.
- `HttpBrowserUseHostAddressCookie`: When set to true, this specifies to use a cookie to determine the full hostname of the browser (the part between the protocol and the relative web address).
- `HttpBrowserHostAddressCookieName`: This entry is enabled only if `HttpBrowserUseHostAddressCookie` is enabled. Use this entry to specify the name of the cookie used to determine what the server believes is the browser's current hostname. The hostname part of the protocol can include the port number. For example, `HttpbrowserHostAddressCookieName=myhost:81` would specify the host `myhost` using the webport `81`. If you do use this cookie, then it is unlikely that you need to enable `HttpBrowserUseIsSslCookie`, because if you use `myhost:433`, that will translate to `https://myhost/%rest-of-url%`.

Changing Absolute Full Path Computation

The BrowserUrlPath component supports the following configuration entries for changing how the absolute full path is computed. This is useful for e-mail, where it is better to use a specific hostname and protocol, even if the browser shows a different URL. This path is considered the *absolute* or *universal* path.

- `HttpBrowserAbsoluteUrlHasRelativeSSL`: When set to true, this variable allows a URL computed on the Content Info page to change from HTTP to HTTPS (or the other way if `UseSSL` is enabled in the config.cfg file), depending on what the Content Server determines as the current use in the user's browser. The change between HTTP and HTTPS also changes the computation of the URL for creating the e-mail body for the "email to" links. This configuration has no effect on automatically generated e-mail.
- `HttpBrowserAlternateWebAddress`: Specifies an alternate absolute host web address (hostname plus optional port number). For example, `HttpBrowserAlternateWebAddresss=<host_name>:447`. This web address is used for the absolute path computation if the current SSL choice is different from the default for the Content Server. This configuration has no effect on automatically generated e-mail.
- `HttpBrowserAbsoluteUrlUsesBrowser Path`: When set to true, if browser path information can be computed, then the absolute path will use the browser path. This essentially turns off the absolute path except for background activities (such as sending notification e-mail).

Changing Administration Path Computation

The BrowserUrlPath component supports the following configuration entries for changing how paths are computed for Administration tray or top menu links. For example, the variable `HttpAdminCgiPath`, which retrieves the Admin Server CGI as a relative URL to the Admin Server, computes an administration path.

- `HttpBrowserAdminUsesAbsolutePath`: When set to true, instead of using the browser-based path (which is the default with the BrowserUrlPath component), the absolute path is used as the basis for computing administration paths, with the exception of the protocol that is dictated by the configuration variable `HttpBrowserUseAdminSSL`.
- `HttpBrowserUseAdminSSL`: This configuration entry is only relevant if the `HttpBrowserAdminUsesAbsolutePath` variable is set. When set to true, this variable dictates the protocol in the administration paths (HTTP or HTTPS) even if `HttpBrowserAbsoluteUrlHasRelativeSSL` is set. The default value of

`HttpBrowserUseAdminSSL` is the opposite of `UseSSL`. This allows the administration path to be non-standard from the default URL constructions for all other paths. The variable `HttpBrowserAlternateWebAddress`, if set, can be used to also give the administration path a different web address in the case that `HttpBrowserUseAdminSSL` is set to the opposite of `UseSSL`.

For further information on variables and enabling the BrowserUrlPath component, see the *Idoc Script Reference Guide* and the *Content Server Installation Guide* for your operating system.

SECURITY RECOMMENDATIONS

The following is a series of recommendations for improving overall security on a Content Server instance. We recommend using four types of security to completely secure Content Server:

- ❖ **Access to the directory structure**: secure the file system to allow access to only those operating system accounts that require access.
  - Read Access: specify Read access for the system administrator to check log files and for those who need to perform regular backups and periodic disaster recovery backups.

    Also set Read access for the account that runs the web server to access and deliver files from the content server web site to the user's browser. The web site includes the files stored under the weblayout directory, the data directory, and the idcplg directory. In the case of Netscape Enterprise Server or IIS running without MS network security integration, this is a single operating system account. If Content Server is running with MS network security integration, IIS will assume the account of the user that is accessing the system. The web server does not need access to other directories.
  - Write Access: specify Write access for the system administrator to install new software and perform customization. Set Write access for the Content Server and optionally the Inbound Refinery (all the same account).

  There should be no need to grant any other account access to the Content Server directory structure (unless you are running some other process to access data directly).

- **Network Access**: configure the network to only allow access to the Content Server directory structure through the Content Server application. If file sharing is required to allow an optional Inbound Refinery on another box to access the Content Server directories, then that sharing should be set up to allow only the Inbound Refinery to access the directories.

  The data directory and the config directory contain user names and passwords. These directories should not be shared on the network.

  For extra security, transmissions to and from the web server should be encrypted by using Secure Socket Layer (SSL).

- **Database Access**: Content Server uses a single database account to access data stored in the database. The database user name and password should be chosen so that they are hard to break and should be updated periodically.
- **Physical Access**: keep the server that is running Content Server in a locked room.

TYPES OF USERS

You can set up users to access your Content Server instance in a variety of ways. Specifically, there are three user login types available:

- Local Users (page 2-16)
- Global Users (page 2-17)
- External Users (page 2-18)

Local Users

Local users are defined by an administrator or sub-administrator within the Content Server system. Administrators assign these users one or more roles, which provide the user with access to security groups. Undefined users are assigned the guest role. The following chapters focus on local users:

- Chapter 3 (*Internal Security: Security Groups, Roles, and Permissions)*
- Chapter 4 (*Internal Security: Using Accounts)*
- Chapter 5 (*Internal Security: Assigning User Logins and Aliases)*

The following is a list of common characteristics of local users:

- **Logins are Created By**: Administrator/sub-administrator in the Content Server instance. Credentials may extend to multiple Content Server instances.
- **Access is Determined By**: Content Server roles, which provide access to security groups.
- **User Login**: Content Server is not required to run for users to log in to the Admin Server. Users can log in to a proxied server using a server-relative URL. For example:

  username<proxied_server>/<local_user_on_proxied_server>
- **User Password**: Users can change their passwords.
- **Interface Issues**: User names appear in the content check-in lists. Users can specify whether to change full name, e-mail address, and user type.
- **Recommended for**: 1000 or fewer users

Note: Because of performance issues, do not configure more than 1000 local users.

A content server can handle approximately 1000 local users; performance issues might become a problem with more local users. For large enterprise user bases, the *global* type of user, whose validation is always performed dynamically, was created. Global user credentials are not published to the web server security filter, so the master server always validates the credentials by querying the database tables. Because of this, the master server must be set up to log in as a global user.

Follow these steps to set up local users:

1. Set up security groups. See Adding a Security Group (page 3-5).
2. Establish roles. See Creating a Role (page 3-10).
3. Arrange permissions. See Adding and Editing Permissions (page 3-12).
4. Assign user logins. See Adding a User Login (page 5-4).
5. (Optional) Use accounts. See Enabling Accounts (page 4-7).

Global Users

Global users are lightly managed users whose credentials extend to multiple content servers. Only a master content server can store global user information.

The concept of global users was created to handle the scalability issues of local users. If you have more than 500 users, the Author option list becomes too long to effectively select a name from it. You then need a way to filter the list of users. This is done using the Organization Path field, which can contain additional information assigned to a user so you can filter on it. See Add/Edit User Screen: Info Tab (Global User) (page 5-13) for information about an Organization Path.

The following is a list of common characteristics of global users:

- **Login is Defined By**: Credentials extended to multiple content servers. Self-registered users are global users.
- **Access is Determined By**: Content Server roles set on the master instance, which provide access to security groups across multiple instances.
- **User Login**: Master content server must be running for users to log in.
- **Interface Issues**: User names do not appear in content check-in lists. Users can specify where to change full name, e-mail address, and user type.
- **Recommended for**: Enterprise situations with over 1000 users, or situations where self-registration is implemented.

To set up global users:

1. Set up a master server and proxied server configuration. See Chapter 8 (*Proxied Connections)*.
2. Have your users log in to the system using their master server login.

External Users

External users are defined outside the Content Server system and authenticated through external security. External users who are automatically registered in the system but are not manually set up by an administrator might use a Microsoft login or some other type of provider (such as LDAP) login.

Generally, these are users in a trusted domain to whom you grant access and do not manage through Content Server. Their password is owned by the Microsoft network domain or other type of provider. Unlike local users, undefined external users are **not** assigned the guest role.

The following is a list of common characteristics of external users:

- **Login is Defined By**: Participation in an external user database,
  - Trusted domain/Microsoft
  - LDAP
  - Other database
- **Access is Determined By**: Credentials from a trusted domain or other user base (such as LDAP)
- **User Login**: Content Server must be running for users to log in.
- **User password**: Users cannot change their passwords.
- **Interface Issues**: User names do not appear in the content check-in lists. However, users can participate in workflows.
- **Recommended For**: Integration with external user base. For example:
  - Trusted domain/Microsoft login
  - Active Directory Server
  - LDAP
  - Other user database

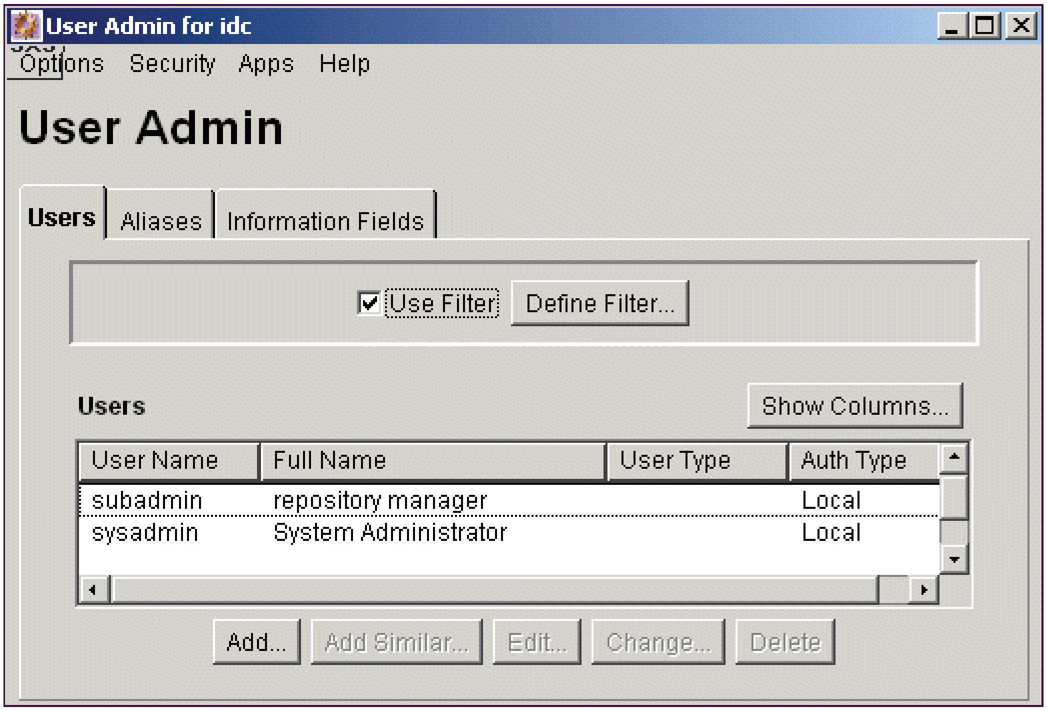
See Chapter 6 (*External Security: Active Directory)*, and Chapter 7 (*External Security: LDAP)* for details about adding external users.

SECURITY ADMINISTRATION INTERFACE

The following are the main screens used when managing security:

- User Admin Application (page 2-20)
- Define Filter Screen (page 2-22)
- Show Columns Screen (page 2-24)

User Admin Application



User Name	Full Name	User Type	Auth Type
subadmin	repository manager		Local
sysadmin	System Administrator		Local

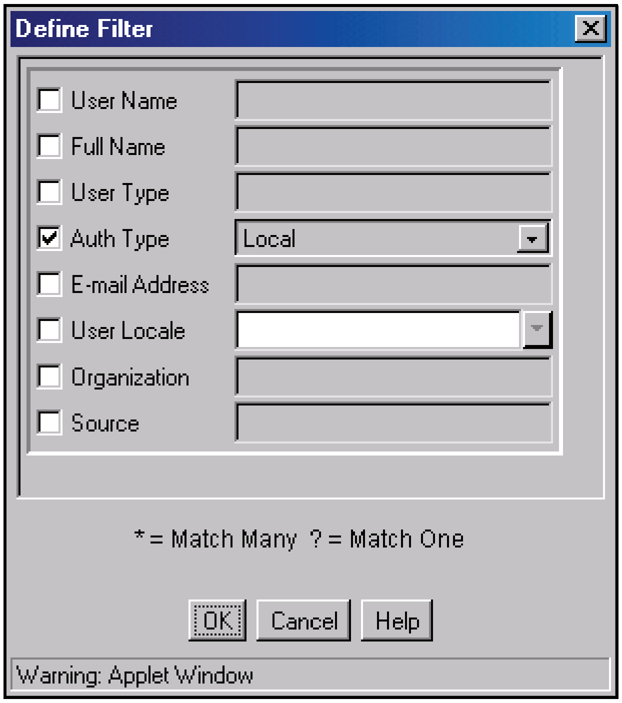
The User Admin application is an administration application used to set up and manage users, security groups, and accounts. You can run this application by accessing it on the Administration page or in stand-alone mode. See Running Applications as Applets (page 1-6) or Running Applications in Stand-alone Mode (page 1-7) for details.

Note: If you run the User Admin application by accessing it in stand-alone mode, it might cause ADSI authenticated users to lose their credentials.

Feature	**Description**
Options menu	**Tracing**—Opens the Tracing Configuration screen, from which you can perform features related to system-wide tracing. **Exit**—Closes the User Admin application.

Feature	Description
Security menu	Displays options to set: **Permissions by Group**—Displays the Permissions By Group Screen (page 3-13). **Permissions by Role**—Displays the Permissions By Role Screen (page 3-15). **Predefined Accounts**—Displays the Predefined Accounts Screen (page 4-11). This option is available only if accounts are enabled. See Enabling Accounts (page 4-7).
Apps menu	Used to open other administration applications. The other applications open in the same mode (applet or stand-alone) as the current application.
Help menu	**Contents**—Displays the Content Server online help. **About Content Server**—Displays version, build, and copyright information for the Content Server.
Users tab	Used to add, edit, and delete user logins. See User Admin Screen: Users Tab (page 5-8).
Aliases tab	Used to add, edit, and delete user aliases. See User Admin Screen: Aliases Tab (page 5-19).
Information Fields tab	Used to add, edit, and delete user information fields. See User Admin Screen: Information Fields Tab (page 5-31).

Define Filter Screen



Define Filter

☐ User Name

☐ Full Name

☐ User Type

☑ Auth Type: Local

☐ E-mail Address

☐ User Locale

☐ Organization

☐ Source

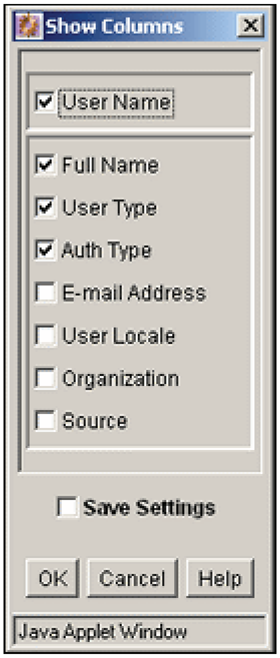
* = Match Many ? = Match One

The Define Filter screen is used to narrow the list of information that is displayed on several administration application screens. The Define Filter screen displays a series of fields that are applicable to the administration application screen. Check the box next to the field to activate that field as a filter.

This screen can be accessed from a variety of other administration screens. For example, a Define Filter button is displayed on the Users tab part of the User Admin screen.

Feature	Description
Check boxes	Select one or more check boxes to activate the filter fields.
Fields	The Users list on the original screen will be filtered based on the criteria entered. The following wildcards can be used in these fields: • With MS Access or MSDE: * = one or more characters ? = single character • With all other databases: % = one or more characters _ = single character
User Name field	The user login.
Full Name field	The full name that corresponds to the user login.
User Type field	An attribute defined by the system administrator as a way to classify users.
Auth Type field	User authorization type, either Local, Global or External.
E-Mail Address field	The e-mail address associated with the user. This is used for workflow and subscription notifications.
User Locale field	The user’s locale, which specifies the language of the user interface and date/time format.
Organization field	The user’s Organization Path value, which can be defined by the system administrator as a way of classifying global users.
Source field	The LDAP user provider used to retrieve user information. Also, this field specifies if the user came from an NTLM or ADSI integration with the value: MSN.
Custom fields	Any custom user information fields will be available as filter fields.

Show Columns Screen



☑ User Name

☑ Full Name

☑ User Type

☑ Auth Type

□ E-mail Address

□ User Locale

□ Organization

□ Source

□ Save Settings

The Show Columns screen is used to specify the columns that are displayed on several administration application screens. The Show Columns screen displays a series of fields that are applicable to the administration application screens. Check the box next to a field to have that field displayed as a column in the administration screens.

This screen can be accessed from a variety of other administration screens. For example, a Show Columns button is displayed on the Users tab part of the User Admin screen.

Feature	Description
Check boxes	**Selected—**The field is displayed in the Users list on the original screen. **Clear—**The field is not displayed on the Users list. **Note:** See Define Filter Screen (page 2-22) for field descriptions.
Save Settings check box	**Selected—**The column settings are applied every time the original screen is displayed. **Clear—**The column settings apply only until the original screen is closed.0

SECURITY ARCHITECTURE

Several aspects of system architecture should be considered when setting up Content Server security. See *Managing System Settings and Processes* for details about web servers, web server filters, and filter plug-ins.

Chapter 3

Internal Security: Security Groups, Roles, and Permissions

Overview

This section covers these topics:

Concepts

Tasks

- Assigning Roles to a User (page 3-11)
- Adding and Editing Permissions (page 3-12)

Interface

- Permissions By Group Screen (page 3-13)
- Add New Group Screen (page 3-14)
- Edit Permissions Screen (page 3-16)
- Permissions By Role Screen (page 3-15)
- Add New Role Screen (page 3-16)

About Security Groups

A security group is a set of files grouped under a unique name. Every file in the content server repository belongs to a security group. Access to security groups is controlled by the permissions, which are assigned to roles, which are assigned to users.

Security groups enable you to organize content files into distinct groups that can be accessed only by specific users. For example, files could be assigned to a security group with the name HRDocs, which could represent documents under the Human Resources designation, and could be accessed only by people who worked in the Human Resources department. There are two predefined security groups:

Public	By default, any user can view documents in the Public group without logging in.
Secure	System files are stored in the Secure group and are available only to the system administrator.

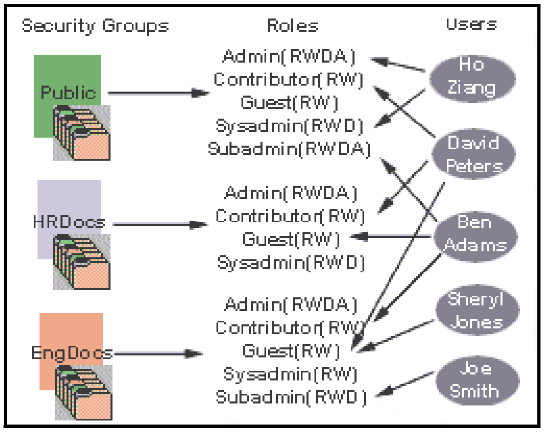
Tips for Working with Security Groups

Keep these considerations in mind when you define security groups:

Important: Define security groups **before** anyone checks in files that must be secure.

- The number of security groups should be kept at a minimum to provide optimum search performance and user administration performance. If your security model requires more than 50 security classifications, you should enable accounts and use them to control user permissions. This number varies depending on Search Performance (page 3-4) and User Admin Performance (page 3-4).
- Put all files that share the same access into one security group.
- Set up a logical naming convention for your security groups. For example, use department names if you are setting up an intranet, and use levels of security (internal, classified, and so forth) if you are setting up an extranet.

Figure 3-1 Example of Defining Security Groups



Performance Issues

Your user access choices for security groups and roles can affect the following system performance areas:

- Search Performance (page 3-4)
- User Admin Performance (page 3-4)

Search Performance

Search performance is affected by the number of security groups a user has permission to access. To return only content that a user has permission to view, the database WHERE clause includes a list of security groups. The WHERE clause either includes all of the security groups the user has permission to access, or it includes all of the security groups the user does *not* have permission to access. Which approach is taken depends on whether the user has permission to more than 50% or fewer than 50% of the defined security groups.

For example, if 100 security groups are defined, and a user has permission to 10 security groups, the 10 security groups will be included in the WHERE clause. In contrast, for a user with permission to access 90 security groups, the WHERE clause includes the 10 security groups the user does *not* have permission to access.

Therefore, if a user has permission to almost 50% of the security groups, the search performance is less efficient. If a user has permission to all or none of the security groups, the search performance is more efficient.

User Admin Performance

The total number of security groups multiplied by the total number of roles determines the number of rows in the *RoleDefinition* database table, which affects the performance of the User Admin application for operations involving local users. To determine the approximate time required to perform an operation in the User Admin application, such as adding a security group or changing permission for a role, use the following formula:

(# of security groups) X (# of roles) / 1000 = Time of operation in seconds

For example, using a PC with a 400 MHz processor, 128 MB of RAM, it took approximately 10 seconds to add a security group and/or role using the User Admin application when the *RoleDefinition* table has 10,000 rows.

As the number of security groups increases, administration performance is affected more than consumer search performance.

Managing Groups

The following tasks are used to manage security groups:

- Adding a Security Group (page 3-5)
- Deleting a Security Group (page 3-5)

Adding a Security Group

To create a security group and assign permissions:

1. From the User Admin Application (page 2-20), select **Security—Permissions by Group**.

   The Permissions By Group Screen (page 3-13) is displayed.

2. Click **Add Group** to display the Add New Group Screen (page 3-14).
3. Enter a group name and description.
4. Click **OK**.
5. Set permissions for the security group:
   a. Select the security group.
   b. Select the role to edit.
   c. Click **Edit Permissions**.
   d. After enabling the permissions that you want the role to have for the group, click **OK** to close the Permissions by Group screen.

Deleting a Security Group

To delete a security group:

1. Make sure that no content items are assigned to the security group you want to delete. You cannot delete a security group if content still exists in that security group.
2. From the User Admin Application (page 2-20), select **Security—Permissions by Group**.

   The Permissions By Group Screen (page 3-13) is displayed

3. Select the group you want to delete.
4. Click **Delete Group**.

   A confirmation screen is displayed.

5. Click **Yes**.

   The security group is deleted.

6. After you have deleted the security group, click **OK** to close the Permissions by Group screen.

About Roles and Permissions

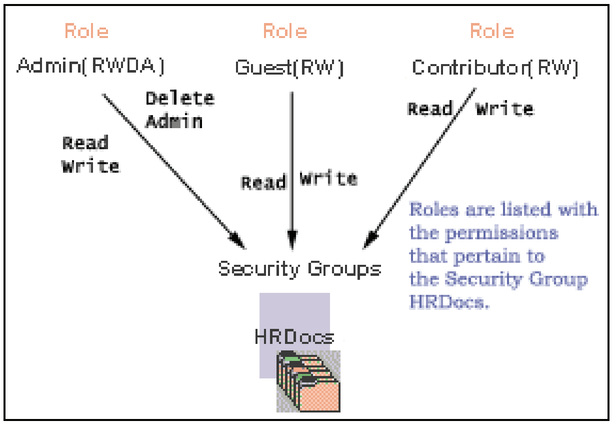
A role is a set of permissions (Read, Write, Delete, Admin) for each security group. You can think of a role as a user's job. Users can have different jobs for various security groups. Users can also have different jobs to identify the different teams in which they participate. You can:

- Assign multiple roles to a user.
- Set up multiple users to share a role.
- Set the role's permissions to multiple security groups.

Note: For information about editing rights, see Setting up a Sub-Administrator (page 5-26).

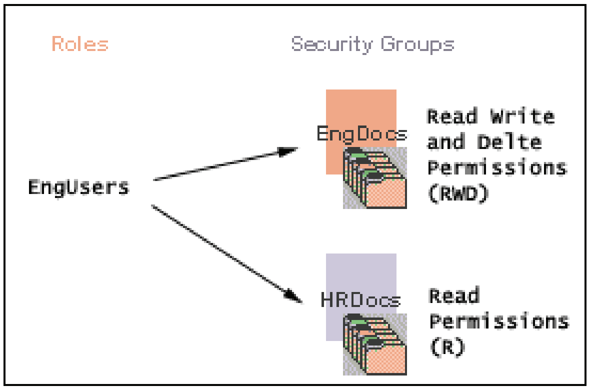
For example, Figure 3-2 shows three roles and the permissions those roles have to the same security group.

Figure 3-2 Example of Roles and Their Permissions



Roles are assigned to one or more users by the system administrator to provide access to the security groups. Figure 3-3 shows the EngUsers role with only Read permission to the HRDocs security group. However, this role provides Read, Write, and Delete permissions to the EngDocs security group. This provides an added measure of security, ensuring that only users who need access to certain documents can modify them.

Figure 3-3 Example of Roles and Security Group Access



Predefined Roles

The following roles are predefined:

Roles	Description
admin	The *admin* role is assigned to the system administrator. By default, this role has Admin permission to all security groups and all accounts, and has rights to all administration tools.
contributor	The *contributor* role has Read and Write permission to the Public security group, which enables users to search for, view, check in, and check out content.
guest	The *guest* role has Read permission to the Public security group, which enables users to search for and view content.
sysmanager	The *sysmanager* role has privileges to access the Admin Server.

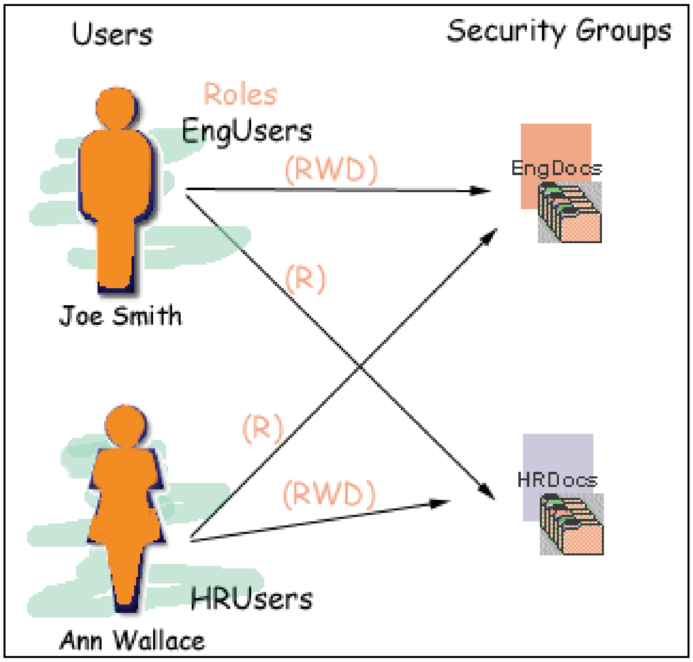
About Permissions

Each role allows the following permissions for each security group: Read (R), Write (W), Delete (D), or Admin (A). The permission that a user has to access the files in a security group is the **highest permission defined by any of the user’s roles.** This means that if a user has the guest and contributor roles, where guest is given Read permission and contributor is given Write permission to the Public security group, the user will have Write permission to content in the Public security group.

In the following figure, Joe Smith and Ann Wallace have permissions to two security groups:

- Joe Smith has Read, Write, and Delete permission to the EngDocs security group, but only Read permission to the HRDocs security group. As a member of the EngUsers role, he has been given Read, Write, and Delete access to Engineering Documents, but only Read access to Human Resource documents.
- Ann Wallace has Read, Write, and Delete permission to the HRDocs security group, but only Read permission to the EngDocs security group. As a member of the HRUsers role, she has been given Read, Write, and Delete access to Human Resource documents, but only Read access to Engineering documents.

Figure 3-4 Example of Assigned Permissions



Predefined Permissions

Each role allows the following permissions to be assigned for each security group:

Permission	Description
Read	Allowed to view files in that security group.
Write	Allowed to view, check in, check out, and get a copy of documents in that security group. The author can change the security group setting of a document if the non-author has Write permission in the new security group.
Delete	Allowed to view, check in, check out, get a copy, and delete files in that security group. The configuration setting `AuthorDelete=true` adds delete permission to all security groups to which the author has Write permission.
Admin	Allowed to view, check in, check out, get a copy, and delete files in that security group. If this user has Workflow rights, they can start or edit a workflow in that security group. Users are also allowed to check in documents in that security group with another user specified as the Author. Non-authors can change the security group setting of a document if the non-author has write permission in the new security group.

MANAGING ROLES AND PERMISSIONS

The following tasks are used to manage user roles.

- Creating a Role (page 3-10)
- Deleting a Role (page 3-10)
- Assigning Roles to a User (page 3-11)
- Assigning Roles to Create Similar Users (page 3-11)
- Adding and Editing Permissions (page 3-12)

Creating a Role

To create a role and configure permissions:

1. From the User Admin Application (page 2-20), select **Security—Permissions by Role**.

   The Permissions By Role Screen (page 3-15) is displayed.

2. Click **Add New Role**.

   The Add New Role Screen (page 3-16) is displayed.

3. Enter a Role Name.

4. Set permissions for the role:

   a. Select the role.

   b. Select the security group to edit.

   c. Click **Edit Permissions**.

   d. Edit the permissions.

   e. Click **OK** and close the Permissions By Role Screen (page 3-15).

Deleting a Role

To delete a role:

1. Make sure that no users are assigned to the role to delete. (You will not be able to delete a role if any users are assigned to it.)

2. From the User Admin Application (page 2-20), select **Security—Permissions by Role**.

   The Permissions By Role Screen (page 3-15) is displayed.

3. Select the role to delete.

4. Click **Delete Role**.

   A confirmation screen is displayed.

5. Click **Yes**.

Assigning Roles to a User

To assign roles to a user:

1. From the User Admin Application (page 2-20), select the user.
2. Click the **Edit** button.

   The Add/Edit User Screen (page 5-10) is displayed.
3. Click the Roles tab.

   The Add/Edit User Screen: Roles Tab (page 5-15) is displayed.
4. Click **Add Role**.

   The Add New Role Screen (page 3-16) is displayed.
5. Select the role from the Role Name list.
6. Click **OK**.

   The role is added to the Roles list.

Assigning Roles to Create Similar Users

To create a user login that has similar access to that of another user login, perform these tasks:

1. From the User Admin Application (page 2-20), select the user.
2. Click the **Add Similar** button.

   The Add/Edit User Screen (page 5-10) is displayed.
3. Click the Roles tab.

   The Add/Edit User Screen: Roles Tab (page 5-15) is displayed. Notice that the roles tab is populated based on the roles and accounts assigned to the selected user.
4. Provide new user-specific data on the Info tab.
5. Click **Add Role**.

Adding and Editing Permissions

To add permissions to a role or edit existing permissions:

1. From the User Admin Application (page 2-20), select **Security—Permissions by Role**.

   The Permissions By Role Screen (page 3-15) is displayed.

2. Either select an existing role, or add a new role.

   The permissions associated with the security groups are displayed.

3. Select an item in the Groups/Rights column.
4. Click **Edit Permissions**.

   The Edit Permissions Screen (page 3-16) is displayed.

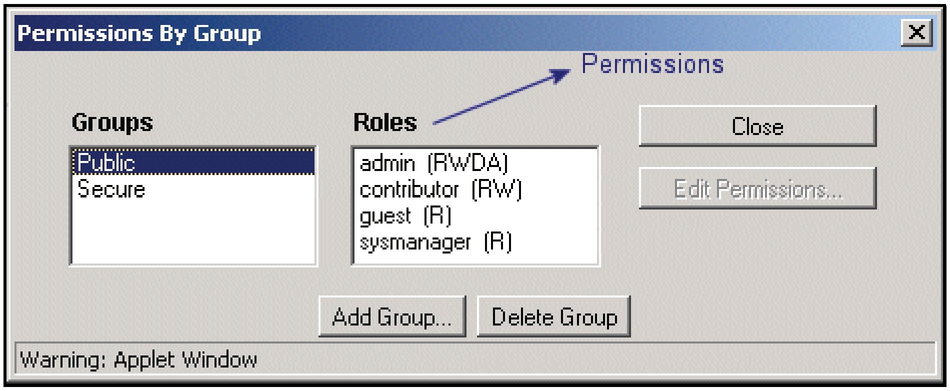
5. Specify the permissions to associate with this role and security group. See Predefined Permissions (page 3-9).
6. Click **OK**.

GROUPS, ROLES, AND PERMISSIONS INTERFACE SCREENS

The following screens are used when creating groups and roles and establishing permissions:

- Permissions By Group Screen (page 3-13)
- Add New Group Screen (page 3-14)
- Permissions By Role Screen (page 3-15)
- Add New Role Screen (page 3-16)
- Edit Permissions Screen (page 3-16)

Permissions By Group Screen



Groups:
Public
Secure

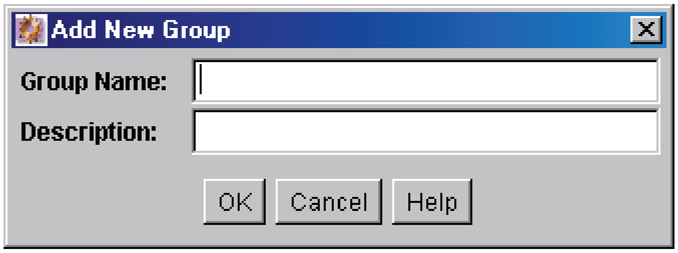
Roles:
admin (RWDA)
contributor (RW)
guest (R)
sysmanager (R)

The Permissions By Group screen is used to add security groups, delete security groups, and edit permissions associated with existing security groups. To access this screen, select **Security—Permissions by Group** in the User Admin Application (page 2-20).

Caution: Security Group names cannot contain square brackets. This is due to limitations in the search engine technology.

Feature	Description
Groups list	Lists existing security groups
Roles list	Lists the roles associated with existing security groups.
Edit Permissions button	Enables you to edit permissions for the security group.
Add Group button	Displays the Add New Group Screen (page 3-14).
Delete Group button	Enables you to delete an existing security group. (You will not be able to delete a security group if content still exists in that security group.)

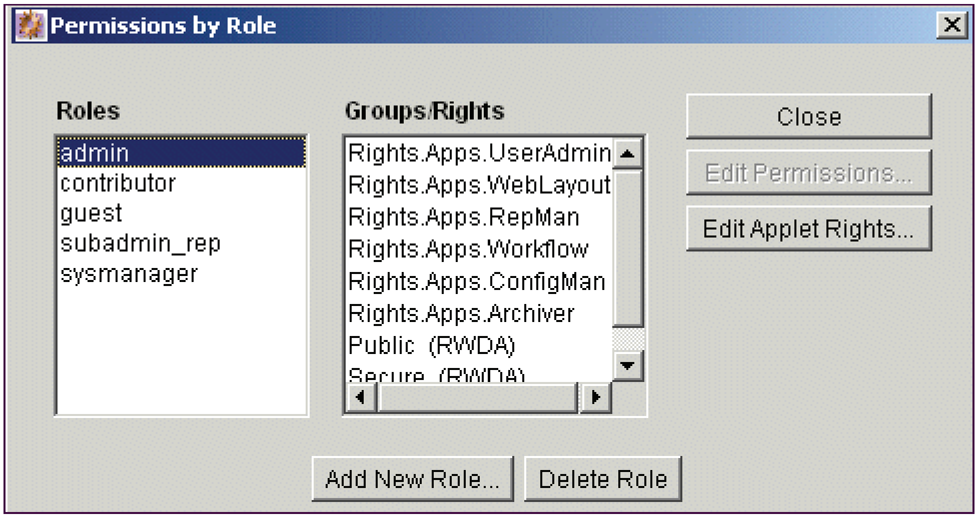
Add New Group Screen



The Add New Group screen is used to define the name and description for a new security group. To access this screen, click **Add Group** on the Permissions By Group Screen (page 3-13).

Feature	Description
Group Name field	• The Group Name is limited to 30 characters. • The following characters are not allowed: spaces, tabs, linefeeds, returns, and ; : ^ ? & + " # % < * ~ | • Uppercase accented letters are not allowed; lowercase accented letters are acceptable. (For example, *Älvdalsån* will not work, but *älvdalsån* will.)
Description field	A brief description of the security group. • The Description is limited to 80 characters. • This field is displayed only in the User Admin application.

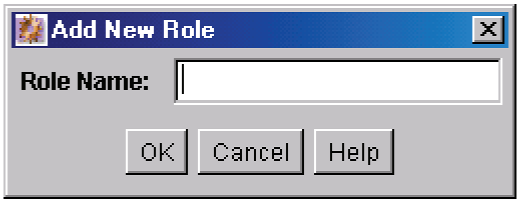
Permissions By Role Screen



The Permissions By Role screen is used to add roles, delete roles, and edit rights and permissions associated with roles. To access this screen, select **Security—Permissions by Role** in the User Admin Application (page 2-20).

Feature	Description
Roles list	Lists existing roles.
Groups/Rights list	Lists the security groups and the rights associated with the selected role.
Edit Permissions button	Enables you to edit permissions for a security group and role. This button is available when you select a role and a group/right.
Edit Applet Rights button	Enables you to edit rights for the role. This button is available when you select a role.
Add New Role button	Displays the Add New Role Screen (page 3-16), on which you can set up a new role for users. Add the role name, and click **OK**.
Delete Role button	Enables you to delete the selected role. (You will not be able to delete a role if any users are assigned to that role.)

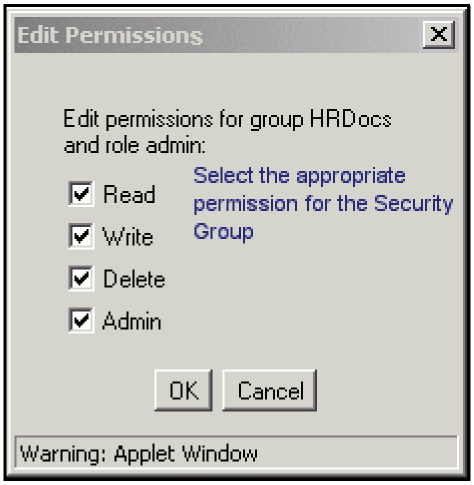
Add New Role Screen



The Add New Role screen is used to define the name of a new role. To access this screen, click **Add New Role** on the Permissions By Role Screen (page 3-15).

Feature	Description
Role Name field	• The Role Name is limited to 30 characters. • The following characters are not allowed: spaces, tabs, linefeeds, returns, and ; : ^ ? & + " # % < * ~ | • Initially, a role is assigned Read (R) permission to the Public security group and no permissions to any other security groups.

Edit Permissions Screen


Edit Permissions
Edit permissions for group HRDocs and role admin:
☑ Read
☑ Write
☑ Delete
☑ Admin
Select the appropriate permission for the Security Group
OK
Cancel
Warning: Applet Window

The Edit Permission screen is used to change permissions to a specific security group for a specific role. To access this screen, do one of the following:

- In the Permissions By Group Screen (page 3-13), select a security group, select a role, and click **Edit Permissions**.

- In the Permissions By Role Screen (page 3-15), select a role, select a security group, and click **Edit Permissions**.

Feature	**Description**
Read check box	Allows users to view files.
Write check box	Allows users to view, check in, check out, and obtain a copy of files.
Delete check box	Allows users to view, check in, check out, get a copy, and delete files.
Admin check box	Allows users to view, check in, check out, get a copy, and delete files, and check in files for other users. In addition, if the user has Workflow rights, they can start or edit a workflow.

Chapter

4

INTERNAL SECURITY: USING ACCOUNTS

OVERVIEW

This section covers these topics:

Concepts

Tasks

Interface

- Predefined Accounts Screen (page 4-11)
- Add New Predefined Account Screen (page 4-12)
- Add/Edit Account Permissions Screen (page 4-12)

Examples

- An Accounts Case Study (page 4-13)

About Accounts

Accounts enable you to obtain greater flexibility and granularity in your security structure than security groups alone provide. Account permissions are assigned to users similar to the way role permission are assigned, through the User Admin tool. An account can also be assigned to each content item. To access a content item that has an account assigned to it, the user must have the appropriate permission to the account.

There are three ways accounts can be created:

- The system administrator creates *predefined accounts* using the User Admin tool. See Creating Predefined Accounts (page 4-7).
- The system administrator creates an account while assigning accounts to a user (using the User Admin tool). See Creating Accounts During User Administration (page 4-8).
- A user administrator creates an account while checking in content. See Creating Accounts When Checking In Content (page 4-8).

Note: You must enable accounts to be able to use them. See Enabling Accounts (page 4-7) for more information.

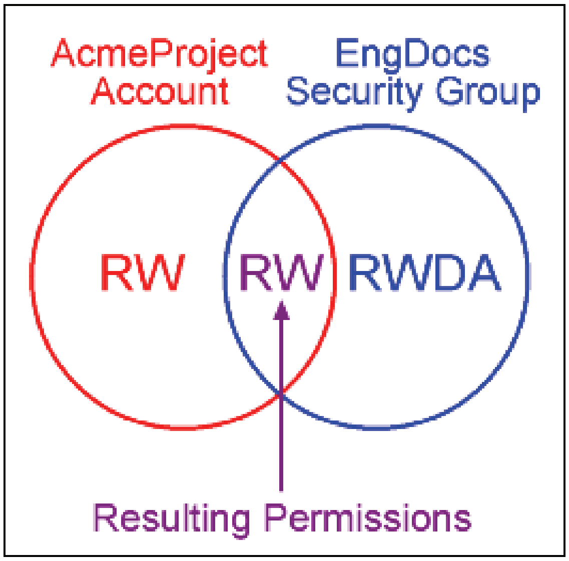
Accounts and Security Groups

When accounts are used, **the account becomes the primary permission to satisfy before security group permissions are applied**. You can also think of a user's access to a particular document as the *intersection* between their account permissions and security group permissions.

For example, the *EngAdmin* role has Read, Write, Delete, and Admin permission to all content in the *EngDocs* security group. A user is assigned the *EngAdmin* role, and is also assigned Read and Write permission to the *AcmeProject* account. Therefore, the user has only Read and Write permission to a content item that is in the *EngDocs* security group and the *AcmeProject* account.

Figure 4-1 shows the intersection of the *AcmeProject* account and *EngDocs* security group permissions.

Figure 4-1 Example of Security Group Permissions

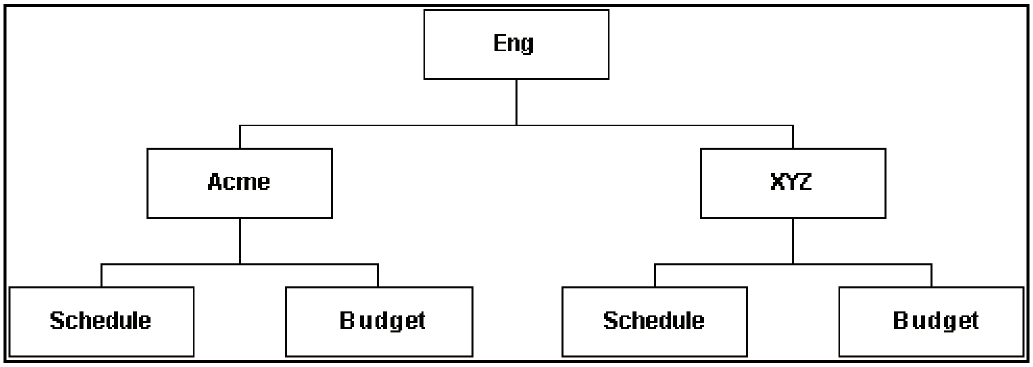


Note: Security group permissions are ignored if the account does not permit access to any content. Remember that the account acts as a filter that supersedes the permissions defined by the user’s roles.

Hierarchical Accounts

Accounts can be set up in a hierarchical structure, which enables you to give some users access to entire branches of the structure, while limiting permissions for other users by assigning them accounts at a lower level in the structure. Figure 4-2 shows a typical hierarchical account structure.

Figure 4-2 Example of Hierarchical Account Structure



Important: Because account names form part of the directory path for the URL of a content item, account names cannot exceed 30 characters.

- If you use slashes to separate the levels in account names (for example, *Eng/Acme/Budget*), Content Server creates a weblayout directory structure according to your account structure. (However, each actual directory will not be created until a content item is assigned to the account during the check-in process.) Each lower level in the account name becomes a subdirectory of the upper level, with an @ symbol prefix to indicate that the directory is an account level.
- If a user has permission to a particular account “prefix”, they have access to all accounts with that prefix. For example, if you are assigned the “Eng/XYZ” account, you have access to the *Eng/XYZ* account and any accounts that begin with the *Eng/XYZ* prefix (such as *Eng/XYZ/Schedule* and *Eng/XYZ/Budget*).

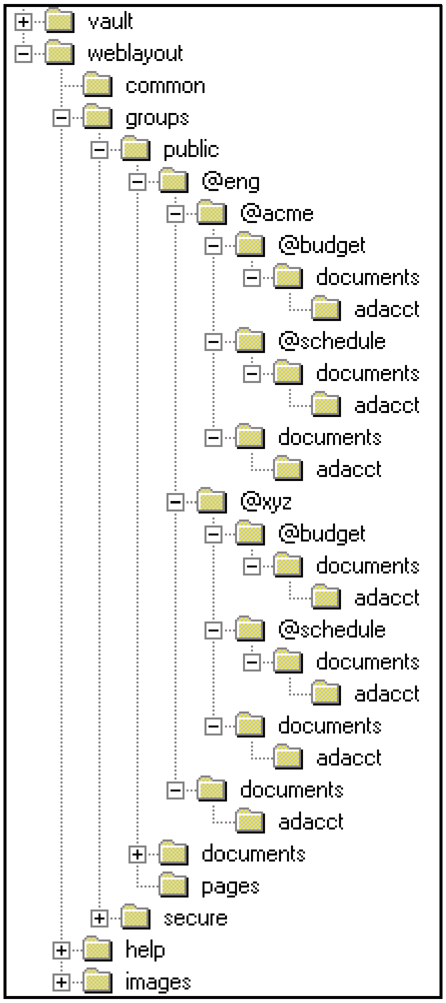
Important: The account prefix does not have to include slashes. For example, if you have accounts called *abc*, *abc_docs*, and *abcdefg*, all users who have access to the *abc* account will have access to the other two accounts as well.

To handle the security structure depicted above, you would create the following accounts:

- Eng
- Eng/Acme
- Eng/XYZ
- Eng/Acme/Schedule
- Eng/Acme/Budget
- Eng/XYZ/Schedule
- Eng/XYZ/Budget

Figure 4-3 illustrates the file structure that Content Server would create in the *<Install_Dir>*/weblayout/ directory.

Figure 4-3 Example of a Security File Structure



Performance Considerations

Consider the following performance issues when using accounts in your security model:

- Theoretically, you can create an unlimited number of accounts without affecting content server performance. A system with over 100,000 pieces of content has only limited administration performance problems at 200 accounts per person; however, there is significant impact on search performance with over 100 accounts per person. (Note that these are explicit accounts, not accounts that are implicitly associated with a user through a hierarchical account prefix. A user can have permission to thousands of implicit accounts through a single prefix.)
- For performance reasons, do not use more than approximately 50 security groups if you enable accounts.
- Ensure that your security groups and accounts have relatively short names.

External Directory Server Considerations

Accounts are available whether or not your content server is integrated with an external directory server (such as ADSI or LDAP). If you use accounts with an external directory, ensure that you follow these additional guidelines:

- Set up a global group with the appropriate users in it to match the account.
- Map group names to either a role or an account by configuring mapping prefixes.

Managing Accounts

The following tasks are involved in managing accounts.

- Enabling Accounts (page 4-7)
- Creating Predefined Accounts (page 4-7)
- Creating Accounts During User Administration (page 4-8)
- Creating Accounts When Checking In Content (page 4-8)
- Deleting Predefined Accounts (page 4-9)
- Assigning Accounts to a User (page 4-9)

Enabling Accounts

Caution: If you enable accounts and use them, you cannot disable them without losing data. DO NOT enable accounts unless you are certain that you want to use them.

To enable accounts:

1. On the Administration Page (page 1-5), click the **Admin Server** link.
2. Click the <*name_of_instance*> button.
3. Select **General Configuration** in the left navigation bar.
4. Add the following line to Additional Configuration Variables field, which shows the contents of the <*Install_Dir*>/config/config.cfg file:

   ```
   UseAccounts=true
   ```
5. Save the changes.
6. Stop and restart the content server.

Creating Predefined Accounts

To create a predefined account:

1. From the User Admin screen, select **Security—Predefined Accounts**.

   The Predefined Accounts Screen (page 4-11) is displayed.
2. Click **Add**.

   The Enabling Accounts (page 4-7) is displayed.
3. Add the name of the new account. Keep the names short and consistent. For example, set up all of your accounts with a three-letter abbreviation by location or department (MSP, NYC, etc.). Account names can be no longer than 30 characters, and the following are not acceptable: spaces, tabs, line feeds, carriage returns, and the symbols: ; ^ ? : & + " # % < > * ~.
4. Click **OK**.
5. If you already have content checked into the content server and you are using Verity, FAST, or a database with full text indexing, rebuild your search index.

   If you are using only the metadata database search indexer engine, you do not need to rebuild your search index.

See Also

– *Creating Accounts During User Administration (page 4-8)*

– *Creating Accounts When Checking In Content (page 4-8)*

Creating Accounts During User Administration

Tech Tip: Generally, you should create predefined accounts rather than creating an account during user definition. See Creating Predefined Accounts (page 4-7).

To create an account while you are assigning accounts to a user, be logged in as *sysadmin*, and perform these tasks:

1. Display the Add/Edit User Screen (page 5-10) for the user you want to create the account for.
2. Click the Accounts tab.
3. Click **Add**.
4. Enter a new account name.
5. Select or clear the check boxes to specify the account permissions.
6. Click **OK**.

   The new account is assigned to the user.

See Also

– *Creating Predefined Accounts (page 4-7)*

– *Creating Accounts When Checking In Content (page 4-8)*

Creating Accounts When Checking In Content

Tech Tip: Generally, you should create predefined accounts rather than creating an account during the content check-in process. See Creating Predefined Accounts (page 4-7).

To create an account at the time you check in a content item, you must have User Admin rights, and perform these tasks:

1. Display the Content Check In Form page.
2. Enter all required and optional information.
3. Type an account name in the Account field.
4. Click **Check In**.

   The new account is assigned to the content item.

See Also

– *Creating Predefined Accounts (page 4-7)*

– *Creating Accounts During User Administration (page 4-8)*

Deleting Predefined Accounts

To delete a predefined account:

1. Select **Security—Predefined Accounts**.

   The Predefined Accounts Screen (page 4-11) is displayed.
2. Select the account to delete.
3. Click **Delete**.

   The account is deleted immediately.

Note: You can delete an account even if content with that account still exists. The account value will remain assigned to the content item, but will be considered a user-defined account.

Assigning Accounts to a User

To assign accounts to a user:

1. From the User Admin screen, select the user.
2. Click the **Edit** button.

   The Add/Edit User Screen (page 5-10) is displayed.
3. Click the Accounts tab.

   The Add/Edit User Screen: Accounts Tab (page 5-16) is displayed.

4. Click **Add**.

   The Add/Edit Account Permissions Screen (page 4-12) is displayed.

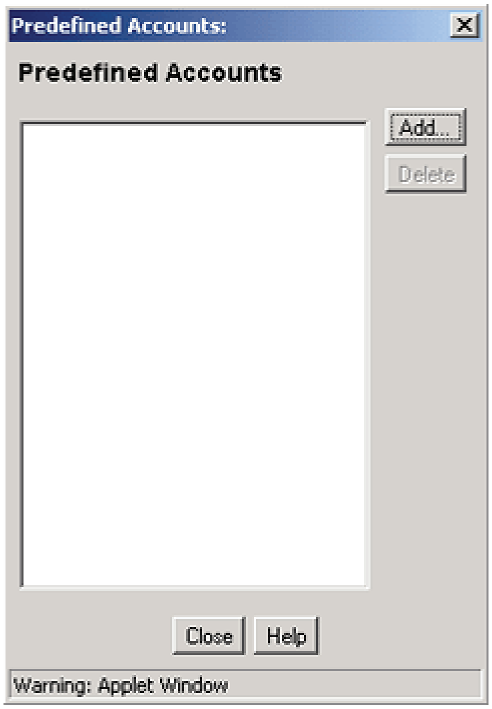
5. Select the account from the **Account** list.
6. Select or clear the check boxes to specify the account permissions.
7. Click **OK**.
8. If necessary, specify a default account. This is the account that will show by default on the Content Check In Form page.
9. Click **OK**.

ACCOUNTS INTERFACE SCREENS

The following screens are used when adding accounts.

- ❖ Predefined Accounts Screen (page 4-11)
- ❖ Add New Predefined Account Screen (page 4-12)
- ❖ Add/Edit Account Permissions Screen (page 4-12)

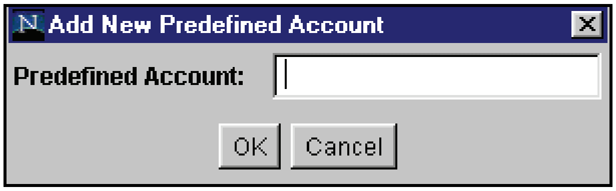
Predefined Accounts Screen



The Predefined Accounts screen is used to add and delete predefined accounts. To access this screen, select **Security—Predefined Accounts** in the User Admin Application (page 2-20).

Feature	Description
Predefined Accounts list	Shows the predefined accounts.
Add button	Displays the Enabling Accounts (page 4-7).
Delete button	Deletes the selected account. **Note:** You can delete an account even if content with that account still exists. The account value will remain assigned to the content item, but will be considered a user-defined account.

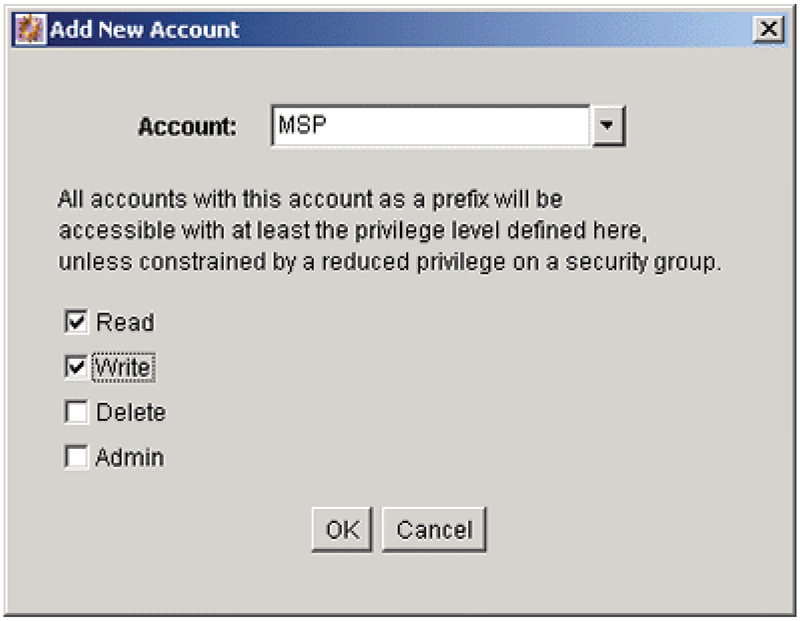
Add New Predefined Account Screen



The Add New Predefined Account screen is used to name a new predefined account. To access this screen, click **Add** on the Predefined Accounts Screen (page 4-11).

Feature	Description
Predefined Account field	Enter the name of the account to be added. Keep the names short and consistent. For example, set up all of your accounts with a three-letter abbreviation by location or department (MSP, NYC, etc.). Account names can be no longer than 30 characters, and the following are not acceptable: spaces, tabs, linefeeds, carriage returns, and the symbols: ; ^ ? : & + " # % < > * ~.

Add/Edit Account Permissions Screen



Account: MSP

All accounts with this account as a prefix will be accessible with at least the privilege level defined here, unless constrained by a reduced privilege on a security group.

☑ Read
☑ Write
☐ Delete
☐ Admin

The Add New Account screen/Edit Permissions for Account Screen is used to assign account permissions to users. To access this screen, click **Add** or **Edit** on the Add/Edit User Screen: Accounts Tab (page 5-16).

Feature	Description
Account list	Select a predefined account from the list, or enter a user-defined account.
Permissions check boxes	Set the Predefined Permissions (page 3-9) that the user will have to the account.

AN ACCOUNTS CASE STUDY

In this example, Xalco is a world-wide software company with offices in London, New York, and Paris. They have a content server hosted in the London office, with access from the other offices via the corporate WAN. At the same time, Xalco is replicating some files out to an area of their public web site. Initially, the Sales and Finance departments at each location want to use their instance to publish files. The New York office is small and has no Sales department.

The following sections provide sample information for the Xalco case study:

❖ Xalco Security (page 4-13)

❖ Xalco Accounts (page 4-14)

❖ Xalco Roles (page 4-15)

❖ Roles and Permissions Table (page 4-15)

❖ Roles and Users Table (page 4-16)

❖ Accounts and Users Table (page 4-16)

Xalco Security

❖ Xalco staff and security levels:

- **London**: David Smith, Worldwide CFO, and Jim McGuire, UK Sales Manager
- **New York**: Catherine Godfrey, Regional Finance Manager
- **Paris**: Helene Chirac, Finance Clerk (Europe)

- Xalco levels of security clearance (security groups) for Xalco content:
  - **Public**: Files suitable for consumption by members of the public (public content is replicated to the Xalco web site)
  - **Internal**: Files which have unrestricted access internally, but are not suitable for public consumption
  - **Sensitive**: Files which are commercially sensitive, and restricted to middle managers and above
  - **Classified**: Highly-sensitive files, suitable only for board members
- Xalco staff access:
  - **David Smith**: As Worldwide CFO, he requires full access to all files held in the instance.
  - **Jim McGuire**: As UK Sales Manager, he needs to have full control of Sales files in London, and have visibility of sales activities in Paris. As a manager, he has clearance to the Sensitive level.
  - **Helene Chirac**: Based in the Paris office, she needs to view only files relating to Finance in Europe, and she has clearance only to the Internal level.
  - **Catherine Godfrey**: As a Regional Finance Manager based in New York, she needs to contribute Finance files for New York and view all other Finance documents. As a manager, she has clearance to Sensitive level.

Xalco Accounts

Access varies by location and job function, so this is reflected in the account structure:

- London has Finance and Sales departments, so it needs two accounts:
  - London/Finance
  - London/Sales
- New York has only a Finance department:
  - NewYork/Finance
- Paris has both Finance and Sales departments:
  - Paris/Finance
  - Paris/Sales

This results in three top-level accounts (London, NewYork, Paris) and five lower-level accounts.

Xalco Roles

We need to create two roles for each security group (one for Consumers and one for Contributors)

- PublicConsumer
- PublicContributor
- InternalConsumer
- InternalContributor
- SensitiveConsumer
- SensitiveContributor
- ClassifiedConsumer
- ClassifiedContributor

Roles and Permissions Table

Role	Public	Internal	Sensitive	Classified
PublicConsumer	R			
PublicContributor	RWD			
InternalConsumer		R		
InternalContributor		RWD		
SensitiveConsumer			R	
SensitiveContributor			RWD	
ClassifiedConsumer				R
ClassifiedContributor				RWD

Note: To give specific users the ability to start workflows, we would need to add Admin permission and Workflow rights to the Contributor roles.

Roles and Users Table

Role	David Smith	Helene Chirac	Jim McGuire	Catherine Godfrey
PublicConsumer		X		
PublicContributor	X		X	X
InternalConsumer		X		
InternalContributor	X		X	X
SensitiveConsumer				
SensitiveContributor	X		X	X
ClassifiedConsumer				
ClassifiedContributor	X		X	X

Accounts and Users Table

Account	David Smith	Helene Chirac	Jim McGuire	Catherine Godfrey
London/Finance	RWDA	R		R
London/Sales	RWDA		RWDA	
NewYork/Finance	RWDA			RW
Paris/Finance	RWDA			R
Paris/Sales	RWDA		R	

Note: It would be sufficient to give David Smith RWDA permission on London, NewYork, and Paris accounts.

Chapter

5

Internal Security: Assigning User Logins and Aliases

Overview

This chapter covers these topics:

Concepts

Tasks

- Setting up a Sub-Administrator (page 5-26)
- Setting Up Self-Registration (page 5-38)

Interface

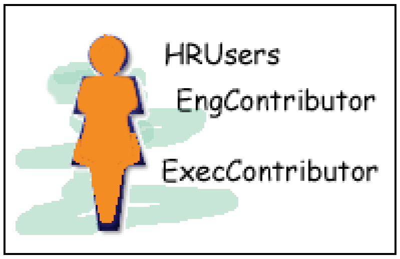
- User Admin Screen: Users Tab (page 5-8)
- Choose the Authorization Type Screen (page 5-9)
- Add/Edit User Screen (page 5-10)
- Add/Edit User Screen: Info Tab (Global User) (page 5-13)
- Add/Edit User Screen: Roles Tab (page 5-15)
- Add/Edit User Screen: Accounts Tab (page 5-16)
- Option List Screen (page 5-17)
- User Admin Screen: Aliases Tab (page 5-19)
- Add New Alias/Edit Alias Screen (page 5-20)
- Select Users Screen (page 5-21)
- Sub-Administration Interface: Edit Rights Screen (page 5-27)
- User Admin Screen: Information Fields Tab (page 5-31)
- Add Metadata Name Field Screen (page 5-32)
- Add/Edit Custom Info Field Screen (page 5-34)
- Update Database Design Screen (page 5-37)

ABOUT USER LOGINS AND ALIASES

User logins are the names associated with the people who access the content server. The system administrator or a subadminstrator assigns one or more roles to each user. A role provides the user access to files within the security groups. Undefined users are assigned to the *guest* role, which allows viewing of documents only in the Public security group by default.

You can also create a group of users that can be then referenced by a single name, or *alias*, in workflows, subscriptions, and projects. For example, it is much easier to add an alias called *Support* to a workflow than it is to add *user1*, *user2*, *user3*, and so on.

Figure 5-1 Example of a User With Roles



Important: User logins are case-sensitive.

Tech Tip: If you log in to multiple browser windows on the same computer using different login methods (such as standard login, Microsoft login, or self-registered login), the content server can become confused as to which user is logged in to each window. Remember to close any open browser windows while testing different login methods.

Predefined User Logins

The following user logins are predefined, and should not be changed or deleted:

User Login	Description
sysadmin	This login is the system administrator, and is assigned the *admin* and *sysmanager* roles by default. The default password is `idc`. This login is a local user.
user1	This login is assigned the *contributor* role. The default password is `idc`. This login is a local user.

Important: For security reasons, you should change the default password of the *sysadmin* user.

Managing Logins and Aliases

This section discusses the tasks involved in managing user logins.

- Adding a User Login (page 5-4)
- Editing a User Login (page 5-5)
- Deleting a User Login (page 5-5)
- Creating an Alias (page 5-5)
- Editing an Alias (page 5-6)
- Deleting an Alias (page 5-6)

Adding a User Login

To add a user login:

1. From the User Admin Screen: Users Tab (page 5-8), click **Add**.
   - If the content server is a master server, the Choose the Authorization Type Screen (page 5-9) is displayed.
   - If the content server is a proxied server, go to step 4.
2. Set the Authorization Type from the drop-down list. See Types of Users (page 2-16) for more information.
3. Click **OK**.

   The Add/Edit User Screen (page 5-10) is displayed.
4. Enter information about the user.
   - If you enter a password, you must re-enter the same password in the Confirm Password field.
   - Keep in mind that the user name and password are case-sensitive.
5. Assign roles to the user. See Assigning Roles to a User (page 3-11).
6. If accounts are enabled, assign accounts to the user. See Assigning Accounts to a User (page 4-9).
7. Click **OK**.

Editing a User Login

To edit a user login:

1. From the Users tab of the User Admin Application (page 2-20), double-click the user name, or select the user name and click **Edit**.

   The Add/Edit User Screen (page 5-10) or Add/Edit User Screen: Info Tab (Global User) (page 5-13) is displayed.

2. Edit the user login as necessary.

Note: If you change the user locale for a user who has the *sysmanager* role, you must restart the Admin Server service for the Admin Server interface to appear in the user's locale language.

Deleting a User Login

To delete a user login:

1. From the Users tab of the User Admin Application (page 2-20), select the user name.
2. Click **Delete**.

   A confirmation screen is displayed.

3. Click **Yes**.

Note: If you delete a user who is involved in a workflow, you are prompted to confirm the deletion. You will need to adjust the workflow and remove the user from the list of workflow reviewers.

Creating an Alias

To define an alias:

1. Display the User Admin Screen: Aliases Tab (page 5-19).
2. Click **Add**.

   The Add New Alias/Edit Alias Screen (page 5-20) is displayed.

3. In the **Alias Name** field, enter a name that identifies the group of users.
4. In the **Description** field, enter a detailed description of the alias.

5. Click **Add**.

   The Select Users Screen (page 5-21) is displayed.

6. Select the user names from the list.

   - To narrow the list of users on the Select Users screen, select the **Use Filter** check box, click **Define Filter,** select the filter criteria, and click **OK**.
   - To select a range of users, click one user login and then hold down the Shift key while clicking another user login.
   - To select users individually, hold down the Ctrl key while clicking each user login.

7. Click **OK**.

8. Close the User Admin screen.

Editing an Alias

To edit an alias:

1. Display the User Admin Screen: Aliases Tab (page 5-19).

2. Highlight an alias and click **Edit**.

   The Add New Alias/Edit Alias Screen (page 5-20) is displayed.

3. Alter the information as needed.

4. In the **Description** field, enter a detailed description of the alias.

5. Click **OK**.

6. Close the User Admin screen.

Deleting an Alias

1. Display the User Admin Screen: Aliases Tab (page 5-19).

2. Highlight the alias to be deleted and click **Delete**.

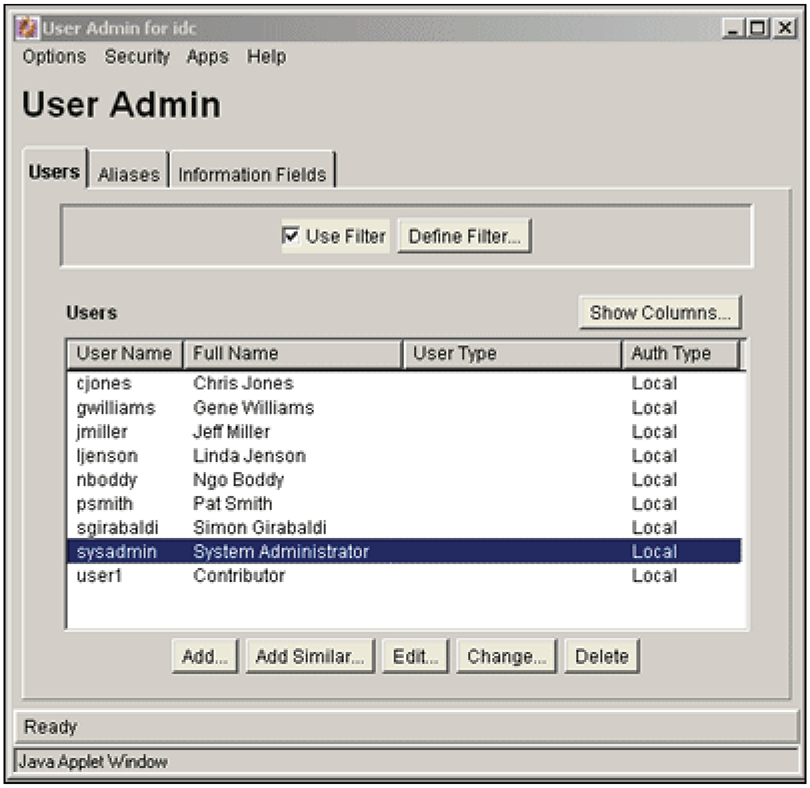
   A screen appears, asking you to confirm the deletion. Click **Yes** to delete the entry or **No** to retain it.

3. Close the User Admin screen.

USER LOGIN AND ALIAS INTERFACE SCREENS

The following screens are used to create user logins and aliases.

User Admin Screen: Users Tab



☑ Use Filter

Users

User Name	Full Name	User Type	Auth Type
cjones	Chris Jones		Local
gwilliams	Gene Williams		Local
jmiller	Jeff Miller		Local
ljenson	Linda Jenson		Local
nboddy	Ngo Boddy		Local
psmith	Pat Smith		Local
sgirabaldi	Simon Girabaldi		Local
sysadmin	System Administrator		Local
user1	Contributor		Local

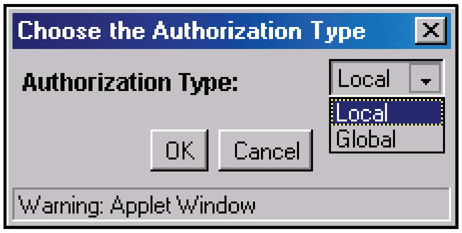
Ready

The Users tab of the User Admin screen is used to add, edit, and delete user logins. To access this tab, display the User Admin Application (page 2-20).

Feature	Description
Use Filter check box	Select this check box to narrow the Users list as defined by the Define Filter Screen (page 2-22).
Define Filter button	Displays the Define Filter Screen (page 2-22).
Show Columns button	Displays the Show Columns Screen (page 2-24).
Users list	Shows the users that match the filter settings. Double-clicking a user displays the Add/Edit User Screen (page 5-10) for that user.
Add button	Displays the Choose the Authorization Type Screen (page 5-9).

Feature	Description
Add Similar button	If you highlight a user and click **Add Similar**, the system displays the Add/Edit User Screen (page 5-10) with some fields already populated.
Edit button	Displays the Add/Edit User Screen (page 5-10) for the selected user.
Delete button	Enables you to delete a user login.

Choose the Authorization Type Screen



The Choose the Authorization Type screen is used to specify the user authorization type when adding a new user. To access this screen, click **Add** on the User Admin Screen: Users Tab (page 5-8).

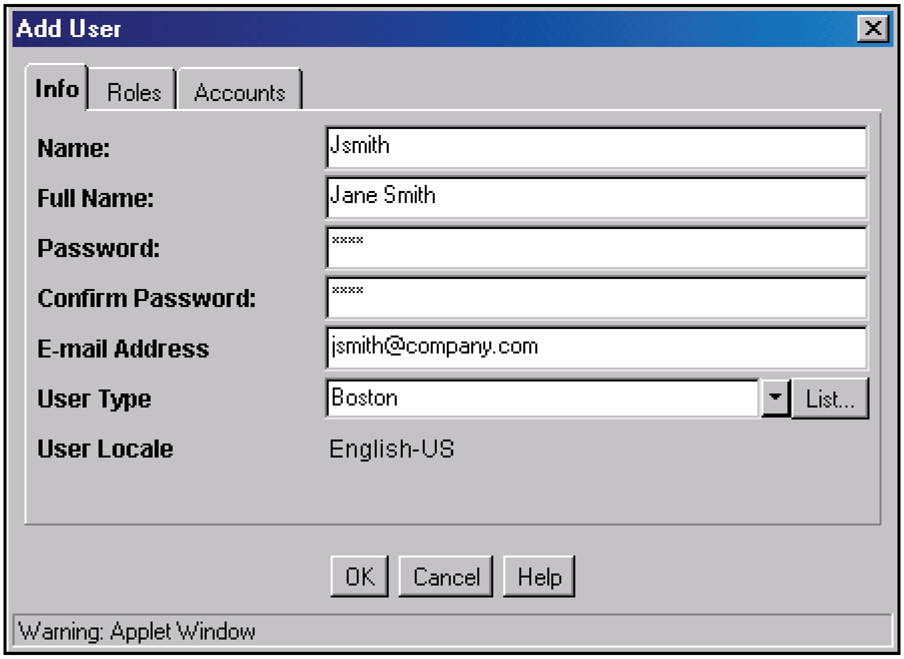
Note: Global users can be created only for a master content server, so this screen is not available from a proxied content server. See Types of Users (page 2-16) for more information.

Feature	Description
Authorization Type list	The type of user. **Local**—Users defined by an administrator or sub-administrator within the Content Server system. Administrators assign these users one or more roles, which provide the user with access to security groups. Undefined users are assigned the *guest* role. Most of the first portion of this guide focuses on local users. **Global**—Lightly-managed users. Both local and global user credentials can extend to multiple content servers.

Feature	Description
OK button	Displays the Add/Edit User Screen: Info Tab (Local User) (page 5-11) or the Add/Edit User Screen: Info Tab (Global User) (page 5-13), depending on which Authorization Type is selected.

Note: External users are created automatically when they are granted content server access using an external user repository.

Add/Edit User Screen



Name: Jsmith

Full Name: Jane Smith

Password: ****

Confirm Password: ****

E-mail Address jsmith@company.com

User Type Boston

User Locale English-US

The Add/Edit User screen is used to define user information, assign roles, and assign account permissions for a user. To access this screen, do one of the following:

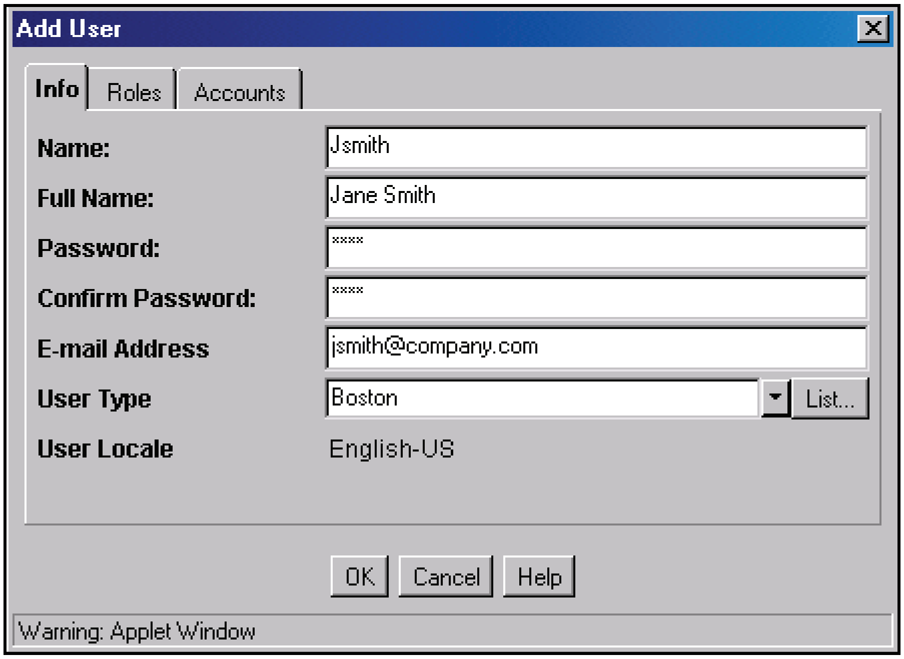
- Select a user type and click **OK** on the Choose the Authorization Type Screen (page 5-9). The Add User screen is displayed.
- Select a user and click **Edit** on the User Admin Screen: Users Tab (page 5-8). The Edit User screen is displayed.

The information that appears on this screen may be different than that on your system if custom metadata fields have been added. The fields shown in this screen shot are the defaults installed with Content Server. See Managing User Information Fields (page 5-29) for details about setting up custom fields.

The tabs on this screen depend on which type of user is selected and whether accounts are enabled:

- Add/Edit User Screen: Info Tab (Local User) (page 5-11)
- Add/Edit User Screen: Info Tab (Global User) (page 5-13)
- Add/Edit User Screen: Roles Tab (page 5-15)
- Add/Edit User Screen: Accounts Tab (page 5-16)

Add/Edit User Screen: Info Tab (Local User)



Name: Jsmith

Full Name: Jane Smith

Password: ****

Confirm Password: ****

E-mail Address jsmith@company.com

User Type Boston

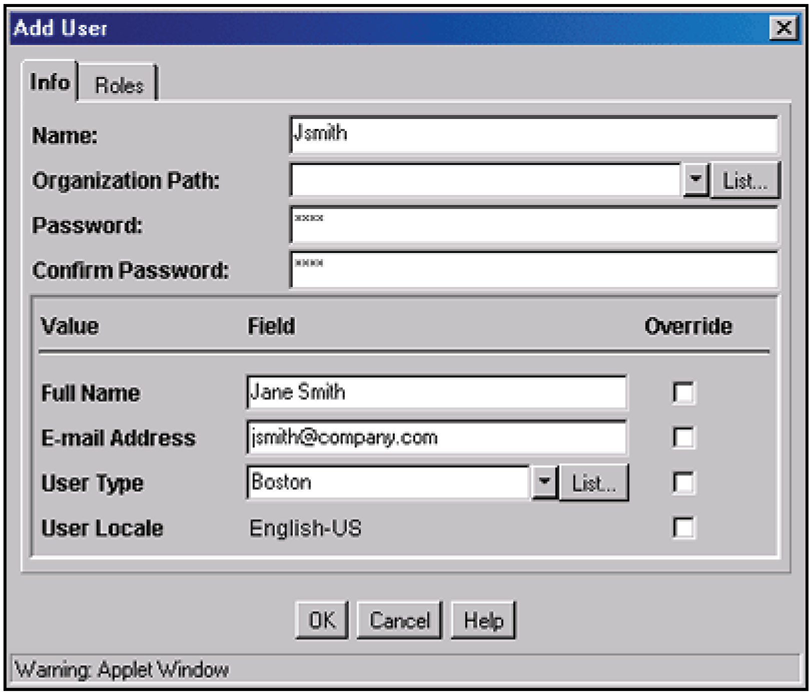
User Locale English-US

The Info tab of the Add/Edit User screen is used to add a user. To access this tab for a local user, do one of the following:

- Select **Local** and click **OK** on the Choose the Authorization Type Screen (page 5-9).
- Select a local user and click **Edit** on the User Admin Screen: Users Tab (page 5-8).

Feature	Description
Name field	The name of the new user. • This field has a 50-character limit. • User names are case-sensitive.
Full Name field	The entire name of the new user. This field has a 50-character limit.
Password field	The password for the new user login. • This field has a 50-character limit. • Passwords are case-sensitive.
Confirm Password field	Reenter the password from the previous field to confirm the spelling.
E-mail Address field	The e-mail address associated with the user. This is used for workflow and subscription notifications.
User Type list	A list of attributes that can be defined by the system administrator as a way to classify users.
List button	Displays the Option List Screen (page 5-17).
User Locale field	The user's locale, which specifies the language of the user interface and date/time format. Locale options must be enabled by the system administrator. See *Using Content Server in International Environments* for more information. **Note:** If you change the user locale for a user who has the *sysmanager* role, you must restart the Admin Server service for the Admin Server interface to appear in the user's locale language.

Add/Edit User Screen: Info Tab (Global User)



Name: Jsmith

Organization Path:

Password: ****

Confirm Password: ****

Value	Field	Override
Full Name	Jane Smith	□
E-mail Address	jsmith@company.com	□
User Type	Boston	□
User Locale	English-US	□

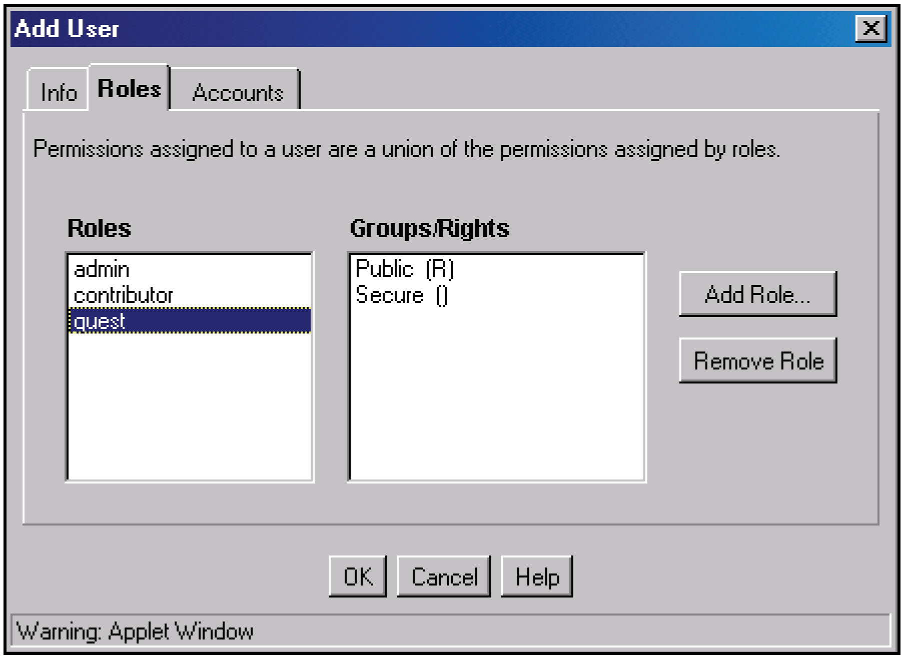
The Info tab of the Add/Edit User screen is used to add a user. To access this tab for a global user, do one of the following:

- Select **Global** and click **OK** on the Choose the Authorization Type Screen (page 5-9).
- Select a global user and click **Edit** on the User Admin Screen: Users Tab (page 5-8).

Feature	Description
Name field	The name of the new user. This field has a 50-character limit.
Organization Path list	A list that can be defined by the system administrator as a way of classifying users.
List button	Displays the Option List Screen (page 5-17).
Password field	The password for the new user login. This field has a 50-character limit.
Confirm Password field	Reenter the password from the previous field to confirm the spelling. The same limit applies.

Feature	Description
Full Name field	The entire name of the new user. This field has a 50-character limit.
E-mail Address field	The e-mail address associated with the user. This is used for workflows and subscriptions.
User Type field	A list of attributes that can be defined by the system administrator as a way to classify users.
User Locale field	The user's locale, which specifies the language of the user interface and date/time format. Locale options must be enabled by the system administrator. See *Using Content Server in International Environments* for more information. **Note:** If you change the user locale for a user who has the *sysmanager* role, you must restart the Admin Server service for the Admin Server interface to appear in the user's locale language.
Override check boxes	These settings apply only if the user is changed from a global user to an external user, or if user information is "auto-assigned" by a custom plug-in to the content server. **Selected**—The user information assigned in the Add/Edit User screen overrides any externally assigned user information (such as user attributes from an LDAP server). **Clear**—The user information assigned in the content server is overridden by any externally assigned user information.

Add/Edit User Screen: Roles Tab

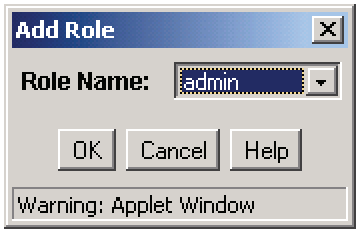


Permissions assigned to a user are a union of the permissions assigned by roles.

The Roles tab of the Add/Edit User screen is used to assign roles to a user. To access this tab, click **Roles** on the Add/Edit User Screen (page 5-10).

Feature	**Description**
Roles list	These roles are displayed in the Roles field.
Groups/Rights list	Lists the security group permissions associated with the selected role.
Add Role button	Displays the Add Role Screen (page 5-16), on which you can select a role from a drop-down list.
Remove Role button	Removes the selected role from the user login.

Add Role Screen

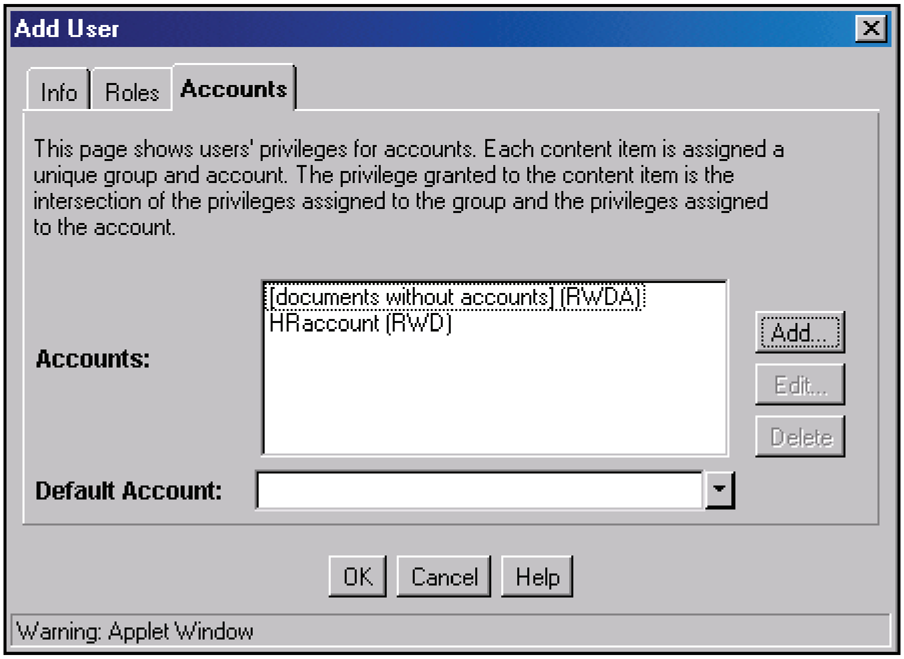


Role Name: admin

The Add Role screen is used to assign a role to a user. To access this screen, click **Add Role** on the Add/Edit User Screen: Roles Tab (page 5-15).

Feature	Description
Role Name field	Select a role to assign to the user.

Add/Edit User Screen: Accounts Tab



This page shows users' privileges for accounts. Each content item is assigned a unique group and account. The privilege granted to the content item is the intersection of the privileges assigned to the group and the privileges assigned to the account.

Accounts:
[documents without accounts] (RWDA)
HRaccount (RWD)

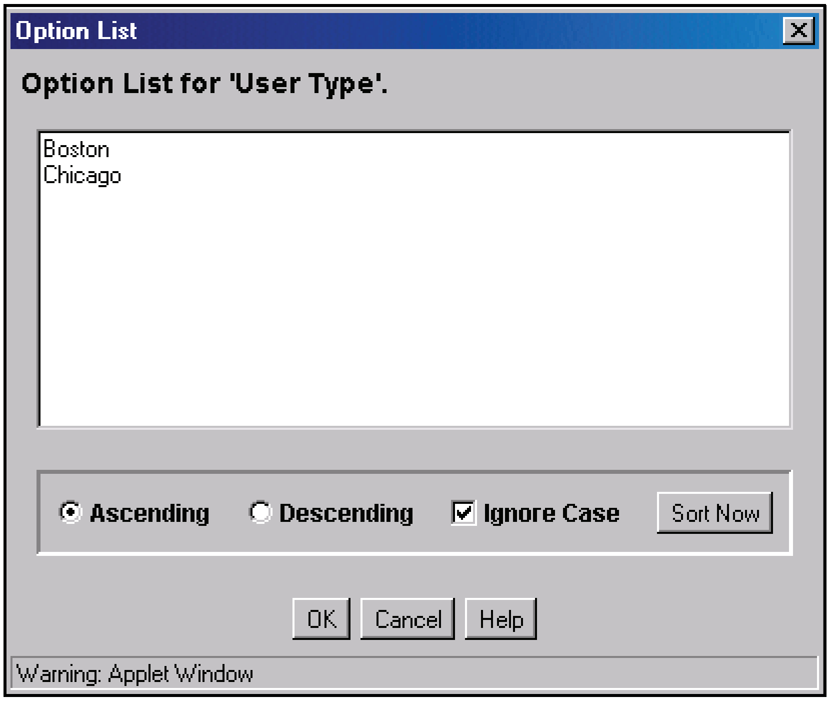
Default Account:

The Accounts tab of the Add/Edit User screen is used to assign accounts to a user. To access this tab, click **Accounts** on the Add/Edit User Screen (page 5-10).

Note: This tab is available only if accounts are enabled. See Chapter 4 (*Internal Security: Using Accounts)* for details.

Feature	Description
Accounts list	Lists the accounts that are assigned to this user login. By default, all new users are assigned Read, Write, Delete, and Admin permission to documents that are not in an account.
Add button	Displays the Add/Edit Account Permissions Screen (page 4-12).
Edit button	Displays the An Accounts Case Study (page 4-13).
Delete button	Enables you to delete a new account.
Default Account list	Select the account that will be entered as the default value on the Content Check In Form page for this user. All accounts for which the user has at least RW permission are listed.

Option List Screen



Option List for 'User Type'.

Boston
Chicago

● Ascending ○ Descending ☑ Ignore Case

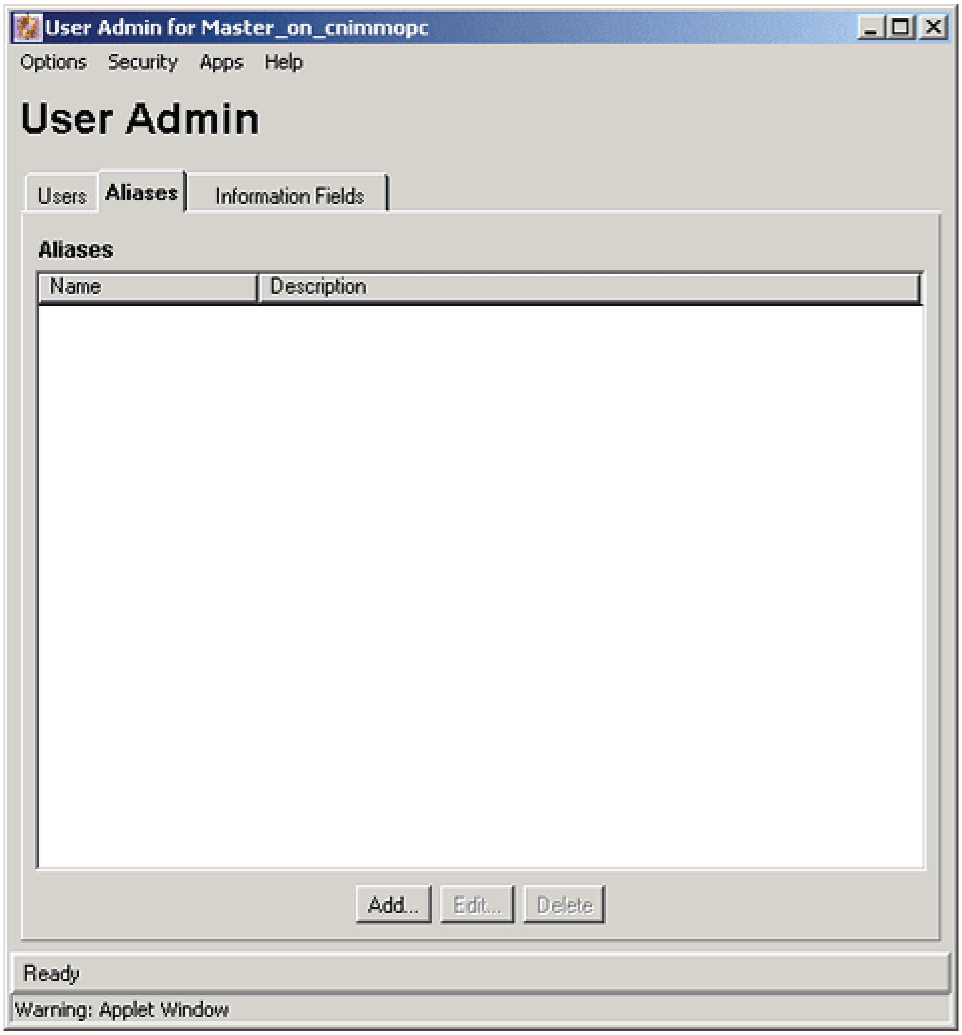
Warning: Applet Window

The Option List screen is used to create a list of options that can be used to group users. This screen can be accessed from a variety of interface locations. This screen is accessed by using the pulldown menu for User Type on the Add/Edit User Screen: Info Tab (Local User) (page 5-11) and Add/Edit User Screen: Info Tab (Global User) (page 5-13).

Note: These option lists do not have any security functionality in the content server; they are simply a means by which you can group users.

Feature	Description
Option list	Enter the values that can be selected for the User Type or Organization Path. Each value must be on a separate line, with a carriage return between values.
Ascending option	Sorts the list in alphabetical order.
Descending option	Sorts the list in reverse alphabetical order.
Ignore Case check box	**Selected**—Sorts the list in alphabetical order, regardless of case. **Clear**—Values that start with uppercase letters are grouped separately from values that start with lowercase letters.
Sort Now button	Sorts the list in the manner specified by the Ascending, Descending, and Ignore Case options.

User Admin Screen: Aliases Tab

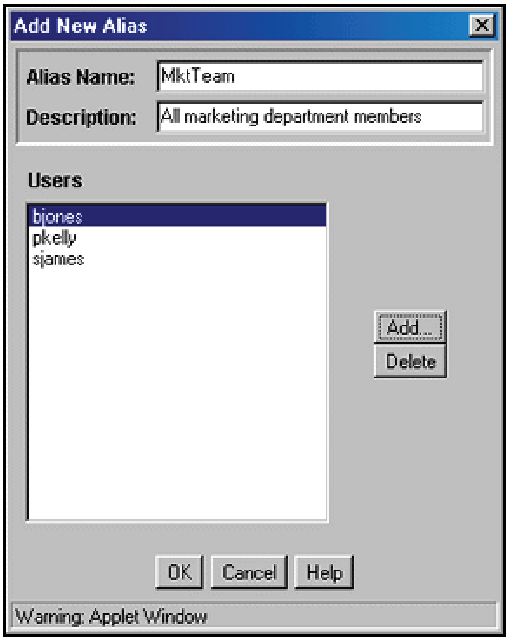


The Aliases tab of the User Admin screen is used to add, edit, and delete aliases. To access this tab, display the User Admin Application (page 2-20), and click **Aliases**.

Feature	Description
Name column	Lists the alias names.
Description column	Description of each alias.
Add button	Displays the Add New Alias/Edit Alias Screen (page 5-20).

Feature	Description
Edit button	Displays the Add New Alias/Edit Alias Screen (page 5-20).
Delete button	Enables you to delete the selected alias.

Add New Alias/Edit Alias Screen



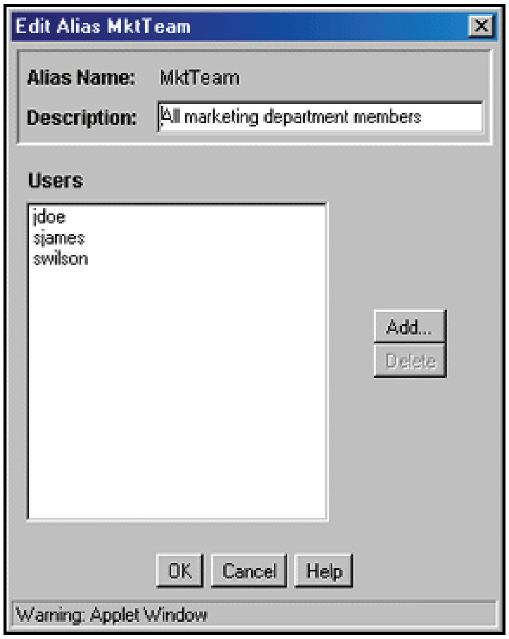
Add New Alias

Alias Name: MktTeam

Description: All marketing department members

Users

bjones
pkelly
sjames



Edit Alias MktTeam

Alias Name: MktTeam

Description: All marketing department members

Users

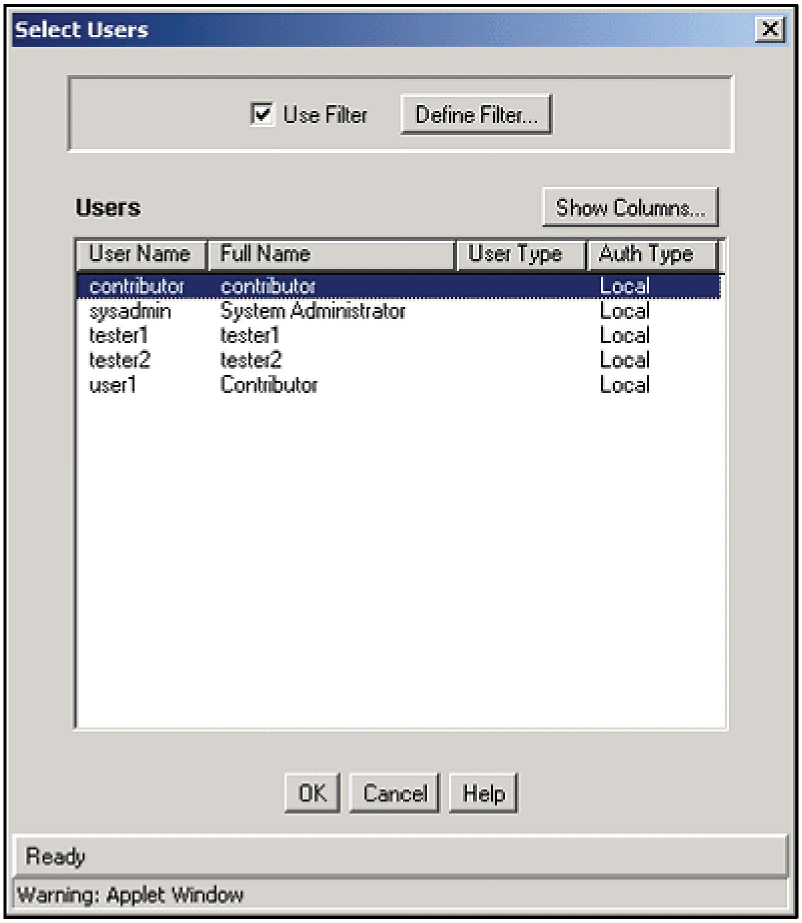
jdoe
sjames
swilson

The Add New Alias/Edit Alias screen is used to add, edit, and delete user logins for an alias. To access this screen, click **Add** or **Edit** on the User Admin Screen: Aliases Tab (page 5-19).

Feature	Description
Alias Name field	The alias name is limited to 30 characters. The following are not allowed: spaces, tabs, line feeds, returns and ; : ^ ? @ & + " # % < * ~ |
Description field	Maximum 80 characters.
Users list	Lists the user logins that are included in the alias.

Feature	Description
Add button	Displays the Select Users Screen (page 5-21).
Delete button	Deletes the selected user login from the alias.

Select Users Screen



☑ Use Filter

User Name	Full Name	User Type	Auth Type
contributor	contributor		Local
sysadmin	System Administrator		Local
tester1	tester1		Local
tester2	tester2		Local
user1	Contributor		Local

Ready

Warning: Applet Window

The Select Users screen is used to add user logins to an alias. To access this screen, click **Add** on the Add New Alias/Edit Alias Screen (page 5-20).

Feature	Description
Use Filter check box	Select this check box to narrow the Users list as defined by the Choose the Authorization Type Screen (page 5-9).
Define Filter button	Displays the Choose the Authorization Type Screen (page 5-9).
Show Columns button	Displays the Show Columns Screen (page 2-24).
Users list	Shows the users that match the filter settings. See Choose the Authorization Type Screen (page 5-9) for column descriptions.

SUB-ADMINISTRATORS

This section covers these topics:

Concepts

- About Sub-Administrators (page 5-22)

Tasks

- Setting up a Sub-Administrator (page 5-26)

Interface

- Sub-Administration Interface: Edit Rights Screen (page 5-27)

About Sub-Administrators

A *sub-administrator* is a user who has been assigned specific rights by a system administrator to one or more administration tools (User Admin, Web Layout Editor, Repository Manager, and Workflow Admin) to administer portions of the software that correspond to those rights. Users with the *admin* role are the only users who can access the Configuration Manager and Archiver tools.

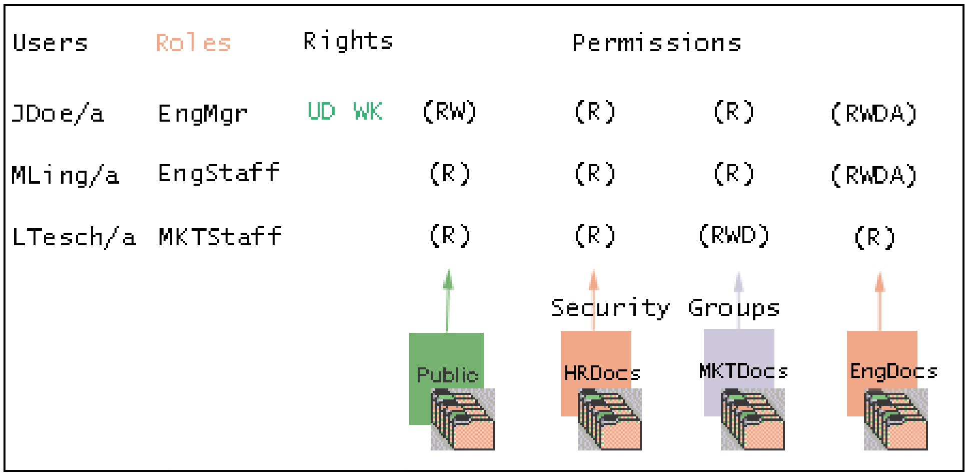
Sub-administrators need to have Admin permission to at least one security group to be able to perform administrative tasks. Typically, a sub-administrator is assigned to be the administrator for a particular security group or account.

The following table describes the functions that a sub-administrator can perform:

Right	Description
UserAdmin	Add, edit, and delete users whose roles and accounts are a subset of the roles and accounts of the sub-administrator.
WebLayout	Add, edit, and delete Library pages to which the sub-administrator has Admin permission for the corresponding security groups and accounts.
RepMan	View documents and execute Repository Manager functions (update, approve, delete, and so forth) on documents to which the sub-administrator has Admin permission.
Workflow	Add, edit, or delete workflows that are in a security group to which the sub-administrator has Admin permission.

Consider JDoe's sub-administrator rights as identified in Figure 5-2 and the corresponding explanation.

Figure 5-2 Example of Sub-Administrator Rights



Users	Roles	Rights		Permissions		
JDoe/a	EngMgr	UD WK	(RW)	(R)	(R)	(RWDA)
MLing/a	EngStaff		(R)	(R)	(R)	(RWDA)
LTesch/a	MKTStaff		(R)	(R)	(RWD)	(R)
			Public	HRDocs	MKTDocs	EngDocs

- **UserAdmin:** User MLing's roles and accounts are a subset of user JDoe's roles and accounts, so JDoe has the authority to add, edit, and delete user MLing. However, user

LTesch has higher permissions for the *MKTDocs* security group than does JDoe, so JDoe cannot add, edit, or delete LTesch.

- **WebLayout:** JDoe cannot perform any Web Layout Editor administrative tasks because JDoe does not have WebLayout rights.
- **RepMan:** JDoe can execute Repository Manager functions on documents that are in the *EngDocs* security group account *a*, or documents that are in the *EngDocs* security group and are not assigned an account. However, JDoe cannot execute Repository Manager functions on documents that are in the *EngDocs* security group in account *b*, *c*, or documents in any other security group because JDoe does not have Admin permission for these documents.
- **Workflow:** JDoe can add, edit, and delete workflows that are in the *EngDocs* security group.

UserAdmin Rights

A user with *UserAdmin* rights can perform the following tasks:

- Add new users, but only assign roles to which the sub-administrator belongs and accounts to which the sub-administrator has permission.

  For example, suppose the sub-administrator belongs to the roles called *EngAdmin*, *EngContributor* and *EngUser*. These three roles are the only roles this sub-administrator can assign to a new user. This sub-administrator cannot assign the role called contributor.
- Edit and delete users that possess a subset of the roles the sub-administrator belongs to and subset of the accounts to which the sub-administrator has permission.

  For example, suppose the sub-administrator belongs to the role called *HTAdmin*, *HTContributor* and *HTUser*. The sub-administrator can only edit or delete a user that belongs to roles that are a subset of the sub-administrator’s roles. If a user belongs to the role called *MKTUser*, this sub-administrator could not edit or delete this user.

A user with *UserAdmin* rights cannot perform the following tasks:

- Create a new user with more permissions than the sub-administrator
- Add, edit, or delete roles
- Add, edit, or delete security groups
- Create aliases

WebLayout Rights

A user with *WebLayout* rights can perform the following tasks:

- Create directory pages for groups and accounts if they have permissions for those groups and accounts.

Note: Only administrators can create a local page that is a report. Administrators or sub-administrators with appropriate rights can create a local page that is a directory.

- Sub-administrators do not have access to the Query Result Pages function in the Options menu nor to any applications for which they do not have rights. Additionally, sub-administrators have viewing, editing, and deleting rights that are restricted as described in these sections:
- For sub-administrators to see a page in the weblayout pane, they must be able to view its parent. For sub-administrators to delete a page, the page must be a directory page and the sub-administrator must have access to that page and all of its children.
- For sub-administrators to see the contents of a page, they must have Read access to that page and all of its parents. This prevents the sub-administrators from seeing a page that they cannot get to through the Library link.
- Adding a query results page/Editing a query results page/Deleting a query results page: These tasks are available only for administrators, not for sub-administrators with WebLayout rights.

RepMan Rights

A user with *RepMan* rights can perform the following tasks:

- Administrators and sub-administrators with RepMan rights can display a list of content item revisions in the Repository Manager. Administrators can display all content items; sub-administrators with RepMan rights can display only content items for which they have Admin permission to the security group and account (if applicable). The revision list can be “searched” by specifying metadata fields and revision status as filter criteria.

- ❖ The Indexer tab on the Repository Manager screen enables administrators (not sub-administrators) to:
  - **Update the Search Index**: Incrementally updates the index database. This is usually not necessary because the index is automatically updated approximately every five minutes by the server.
  - **Rebuild the Collection**: The search index is entirely rebuilt, and the old index collection is replaced with a new index collection.

Workflow Rights

A user with *Workflow* rights can perform the following tasks:

- ❖ Add, edit, or delete workflows that are in a security group to which the sub-administrator has Admin permission.

Setting up a Sub-Administrator

When you set up sub-administrators, remember to grant Admin permission to the security groups and sub-administrator role that you want them to use to perform administrative tasks. A sub-administrator's rights are useless without Admin permission associated to at least one security group.

Also, if your sub-administrators have *UserAdmin* rights, remember to assign multiple roles to them because when they add users, they can assign only their roles to these users. If you assign only a *subadmin* role to your sub-administrator, this is the only role that they can assign to users whom they administer.

To set up a sub-administrator:

1. Add a sub-administrator role, such as *HRsubadmin*. See Creating a Role (page 3-10).
2. From the Permissions By Role Screen (page 3-15), assign permissions and rights to the sub-administrator.
   a. Select the sub-administrator role.
   b. Click **Edit Permissions**.

      The Edit Permissions Screen (page 3-16) is displayed.
   c. Select the security group for which you want sub-administrators to perform administrative tasks.
   d. Select **Admin** permission.
   e. Click **OK**.

f. Click **Edit Rights**.

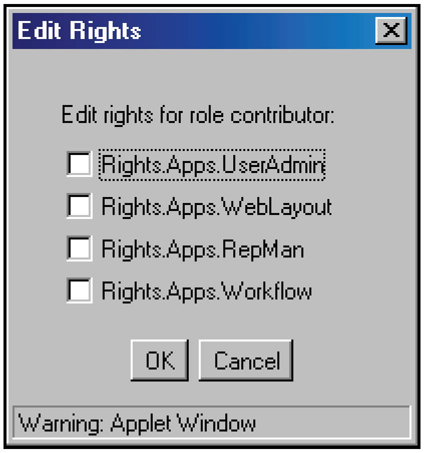
The Sub-Administration Interface: Edit Rights Screen (page 5-27) is displayed.

g. Select one or more rights for the sub-administrator.

h. Click **OK**.

3. Assign the sub-administrator role to users. See User Information Fields (page 5-28).

Sub-Administration Interface: Edit Rights Screen



The Edit Rights screen is used to assign sub-administration rights to a role. To access this screen, select a role and click **Edit Rights** on the Permissions By Role Screen (page 3-15).

Feature	Description
Rights.Apps.UserAdmin check box	Assigns sub-administration rights to the User Admin application. See UserAdmin Rights (page 5-24) for details.
Rights.Apps.WebLayout check box	Assigns sub-administration rights to the Web Layout Editor application. See WebLayout Rights (page 5-25) for details.
Rights.Apps.RepMan check box	Assigns sub-administration rights to the Repository Manager application. See RepMan Rights (page 5-25) for details.
Rights.Apps.Workflow check box	Assigns sub-administration rights to the Workflow Admin application. See Workflow Rights (page 5-26) for details.

USER INFORMATION FIELDS

This section covers these topics:

Concepts

Tasks

Interface

About User Information Fields

User information defines the unique attributes of a user, such as full name, password, and e-mail address. User information fields describe a user in the same way that metadata fields describe a content item. User information is stored in the content server database, and can be used to sort users, display user information on content server web pages, or customize the display of web pages based on user attributes.

The following user information fields are predefined in the system. These fields cannot be deleted, and the field name and type cannot be changed.

Name	Type	Caption	Is Option List
dFullName	Long Text	Full Name	False
dEmail	Long Text	E-mail Address	False

Name	Type	Caption	Is Option List
dUserType	Text	User Type	True
dUserLocale	Text	User Locale	True

Managing User Information Fields

This section describes the tasks involved in managing user information fields.

- Adding a New User Information Field (page 5-29)
- Editing an Option List (page 5-29)
- Editing a User Information Field (page 5-30)

Adding a New User Information Field

To add a new user information field:

1. On the User Admin Screen: Information Fields Tab (page 5-31), click **Add**.

   The Add Metadata Name Field Screen (page 5-32) is displayed.

2. Enter a new field name. Duplicate names are not allowed. Maximum field length is 29 characters. The following are not acceptable: spaces, tabs, line feeds, carriage returns and ; ^ ? : @ & + " # % < * ~ |

3. Click **OK**.

   The Add/Edit Custom Info Field Screen (page 5-34) is displayed.

4. Configure the properties for the field, and click **OK**.

5. Click **Update Database Design**.

Editing an Option List

To edit an option list key:

1. On the Add/Edit Custom Info Field Screen (page 5-34), select the Enable Option List check box.

2. Click **Edit**.

   The Option List Screen (page 5-36) is displayed.

3. Add, edit, or delete option values.
   - Each value must appear on a separate line.
   - A blank line will result in a blank value in the option list.
4. To sort the list, select sort options and click **Sort Now**.
5. Click **OK**.

Editing a User Information Field

To edit a user information field:

1. Double-click the field, or select the field and click **Edit**.

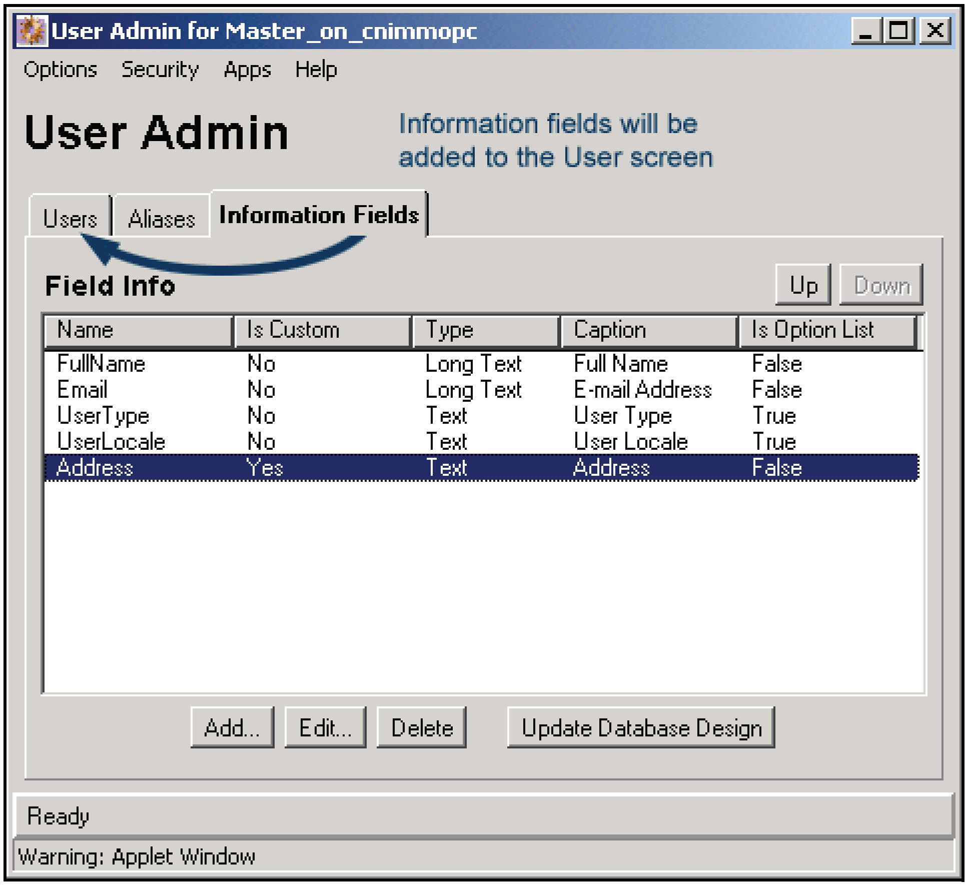
   The Add/Edit Custom Info Field Screen (page 5-34) is displayed.
2. Add, edit, or delete option values.
3. Click **OK**.

Information Fields Interface Screens

The following screens are used when defining information fields.

- User Admin Screen: Information Fields Tab (page 5-31)
- Add Metadata Name Field Screen (page 5-32)
- Add/Edit Custom Info Field Screen (page 5-34)
- Option List Screen (page 5-36)
- Update Database Design Screen (page 5-37)

User Admin Screen: Information Fields Tab



Name	Is Custom	Type	Caption	Is Option List
FullName	No	Long Text	Full Name	False
Email	No	Long Text	E-mail Address	False
UserType	No	Text	User Type	True
UserLocale	No	Text	User Locale	True
Address	Yes	Text	Address	False

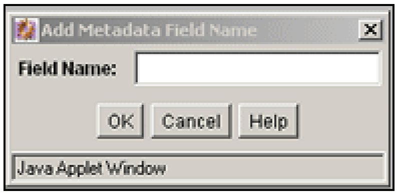
The Information Fields tab of the User Admin screen is used to add, edit, and delete user information fields. To access this tab, display the User Admin Application (page 2-20) and click **Information Fields**.

- When a field is added in the Information Fields tab, it is also added to the user information on the Users tab.
- You do not need to rebuild the search index after adding new user fields.

Feature	Description
Up button	Moves the selected user information field up in the list.
Down button	Moves the selected user information field down in the list.
Name column	The name of the user information field.

Feature	Description
Is Custom column	**No**—Indicates a system (predefined) user information field. **Yes**—Indicates a custom user information field.
Type column	The type of field: **Text**: 30 characters. **Long Text**: 100 characters. **Date**: Date format (such as dd/mm/yyyy or dd/mm/yy for the English-US locale). **Memo**: 255 characters. **Integer**: -231 to 2 31 (-2 billion to +2 billion). By definition, an integer is a natural number, so decimal values and commas are not permitted.
Caption column	The field label that appears on content server pages.
Is Option List column	**False**—The user information field does not have an option list. **True**—The user information field has an option list.
Add button	Displays the Add Metadata Name Field Screen (page 5-32), on which you can add a new field name.
Edit button	Displays the Add/Edit Custom Info Field Screen (page 5-34).
Delete button	Deletes the selected custom user information field. (System user information fields cannot be deleted.)
Update Database Design button	Displays the Update Database Design Screen (page 5-37).

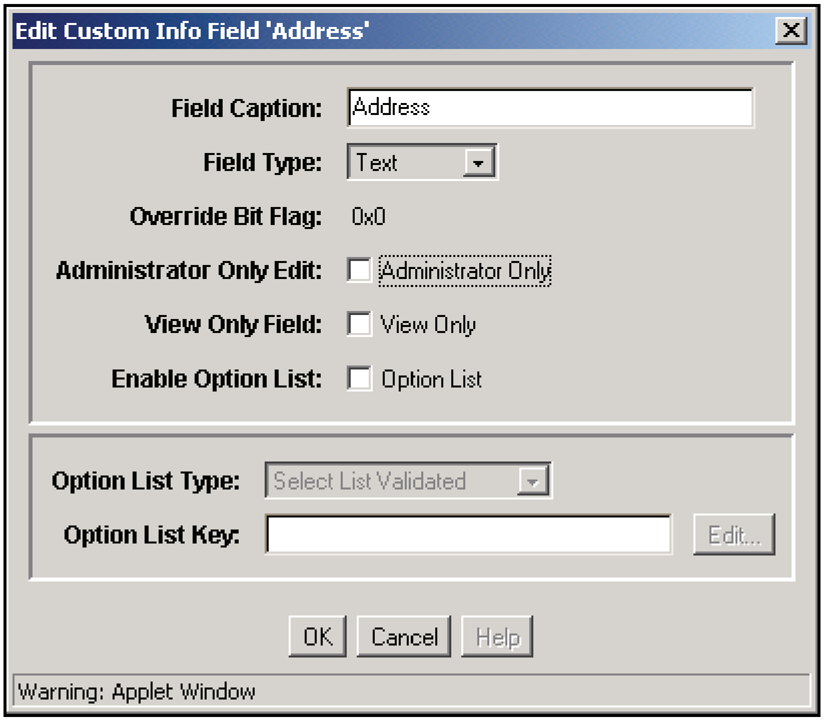
Add Metadata Name Field Screen



The Add Custom Info Field screen is used to define the name of a custom user information field. To access this screen, click **Add** on the User Admin Screen: Information Fields Tab (page 5-31).

<table>
<tr><th>Feature</th><th>Description</th></tr>
<tr><td>Field Name field</td><td>Duplicate names are not allowed. Maximum field length is 29 characters. The following are not acceptable: spaces, tabs, line feeds, carriage returns and ; ^ ? : @ & + " # % < * ~ |
Note: When you add a custom user information field, the system automatically prefixes the name with a “u” to ensure that it is unique and does not conflict with any reserved names. However, you must be careful not to inadvertently use restricted names for columns in the user logins table because they may conflict with reserved names in databases.
For example, if you try to use “ID” to name a new custom user information field, the result will be “UID” when the system adds the prefix. This causes an error because UID is a reserved database name.
Similarly, when you define a custom metadata field, the system automatically prefixes the name with an “x” to ensure that it is unique and does not conflict with any reserved names. For more information about adding custom metadata fields, see Add Metadata Name Field Screen (page 5-32).</td></tr>
<tr><td>OK button</td><td>Displays the Add/Edit Custom Info Field Screen (page 5-34).</td></tr>
</table>

Add/Edit Custom Info Field Screen



Edit Custom Info Field 'Address'

Field Caption: Address

Field Type: Text

Override Bit Flag: 0x0

Administrator Only Edit: □ Administrator Only

View Only Field: □ View Only

Enable Option List: □ Option List

Option List Type: Select List Validated

Option List Key:

Warning: Applet Window

The Add/Edit Custom Info Field screen is used to define a user information field. To access this screen, do one of the following:

- Enter a field name and click **OK** on the Add Metadata Name Field Screen (page 5-32)
- Select a user information field and click **Edit** on the User Admin Screen: Information Fields Tab (page 5-31).

Feature	Description
Field Caption field	Label for the field that is displayed on content server pages.
Field Type	**Text**: 30 characters. **Long Text**: 100 characters. **Date**: Date format (such as dd/mm/yyyy or dd/mm/yy for the English-US locale). **Memo**: 255 characters. **Integer**: -231 to 2 31 (-2 billion to +2 billion). By definition, an integer is a natural number, so decimal values and commas are not permitted.
Override Bit Flag	For internal use.
Administrator Only Edit	**Selected—**The field is not displayed on the User Profile pages. However, the field is visible to an admin user via the User Admin applet. **Clear—**The field is displayed on the User Profile page.
View Only Field	**Selected—**The field is displayed on the User Profile page, but cannot be edited by the user. **Clear—**If the Administrator Only Edit check box is clear, the field can be edited by the user on the User Profile page.
Enable Option List	The field has an option list that is defined by the Option List Type and Option List Key.

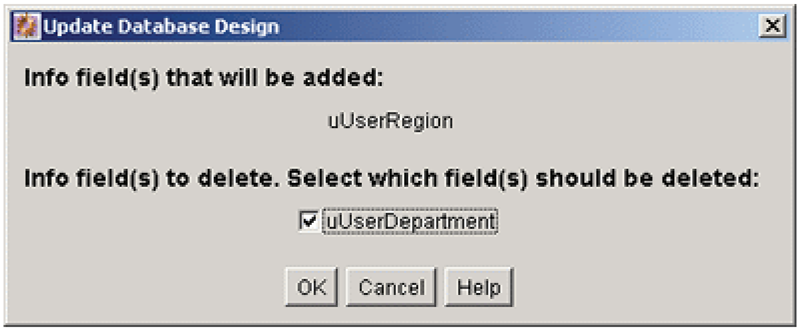
Feature	Description
Option List Type	Specifies the type of option list: • **Select List Validated:** For Batch Load and Archiver purposes, this option ensures that only users whose specified values are current options for this field are imported. • **Select List Not Validated:** For Batch Load and Archiver purposes, this option permits loading of users whose specified values are not current options for this field. • **Edit and Select List:** Provides both a text field and a combo box. Users can enter values that are not in the option list. • **Edit and Multiselect List:** Provides both a text field and a combo box. Users can enter values that are not in the option list. Additionally, they can select or enter multiple values.
Option List Key	A designation generated from the field caption for a specific list of values that are displayed in the option list. The list can be reused for more than one field using the option list key.

Option List Screen

The Option List screen is used to create an option list for a custom user information field. To access this screen, select the Enable Option List check box, and click **Edit** on the Add/Edit Custom Info Field Screen (page 5-34).

See Option List Screen (page 5-17) for more details.

Update Database Design Screen



Update Database Design

Info field(s) that will be added:

uUserRegion

Info field(s) to delete. Select which field(s) should be deleted:

☑ uUserDepartment

The Update Database Design screen is used to add or delete user information fields in the content server database. To access this screen, add or delete a user information field and click **Update Database Design** on the Add/Edit Custom Info Field Screen (page 5-34).

Feature	Description
Info field(s) that will be added	Lists the user information fields that were added since the last time the database was updated.
Info field(s) to delete check boxes	Lists the user information fields that were deleted since the last time the database was updated. **Selected**—The user information field will be deleted from the database. **Clear**—The user information field will not be deleted from the database. The field remains hidden on the User Admin screen and User Profile pages, but it still exists in the database.

SELF-REGISTRATION

This section covers these topics:

Concepts

- About Self-Registration (page 5-38)

Tasks

- Setting Up Self-Registration (page 5-38)

About Self-Registration

Self-registration enables users to create their own content server logins (user name and password).

- Self-registered users are created as global users (see Types of Users (page 2-16)).
- You can combine self-registration with other user login strategies (for example, local users using the standard Login button, and external users using an integrated Windows login).
- Enabling the self-registration function adds a Self-Registration link to the portal navigation bar.
- See the *Oracle Content Server User Guide* for instructions on how to self-register.

Setting Up Self-Registration

To set up and enable self-registration:

1. In the Additional Configuration Variables field on the General Configuration page of the Admin Server, or in the <*Install_Dir*>/config/config.cfg file, add the following lines:

   ```
   UseSelfRegistration=true
   SelfRegisteredRoles=contributor,guest
   ```

Note: The *SelfRegisteredRoles* setting defines the roles that will be assigned to all self-registered users. The value must be a comma-separated list of roles.

2. If accounts are enabled, add the following line to the Additional Configuration Variables field or to the *config.cfg* file:

   ```
   SelfRegisteredAccounts=Acme(RW),#none(RWDA),<$NewUser$>(RWDA)
   ```

Note: The *SelfRegisteredAccounts* setting defines the accounts that will be assigned to all self-registered users. The value must be a comma-separated list of accounts, with account permissions in parentheses after each account. You can specify the following special accounts:

- **#none** assigns permission to content items that do not have an account specified
- **#all** assigns permission to all accounts
- **<$NewUser$>** creates an account that is the same as the user name (such as *pkelly*).

3. In the Admin Server, click **Save**, or save and close the *config.cfg* file.
4. Restart the content server.
5. Update the portal page.

Important: Users will not see the Self-Registration link in the portal navigation bar until you update the portal page.

Chapter

6

External Security: Active Directory

Overview

This section covers these topics:

Concepts

Tasks

Interface

- Active Directory Configuration Page (page 6-17)

Active Directory Overview

This section covers these topics:

- About Active Directory (page 6-2)
- Active Directory Structure (page 6-3)
- Domains (page 6-4)

About Active Directory

Microsoft Active Directory is a directory service that is included with Windows 2000 Server and later versions. It is specifically designed to operate in a Windows networking environment. Active Directory stores information about objects on a network and makes this information available to applications, users, and network administrators. With Active Directory, authorized network users can access resources anywhere on the network using a single login process. Active Directory provides the same high-level functionality as LDAP (Lightweight Directory Access Protocol) to manage resources in a network.

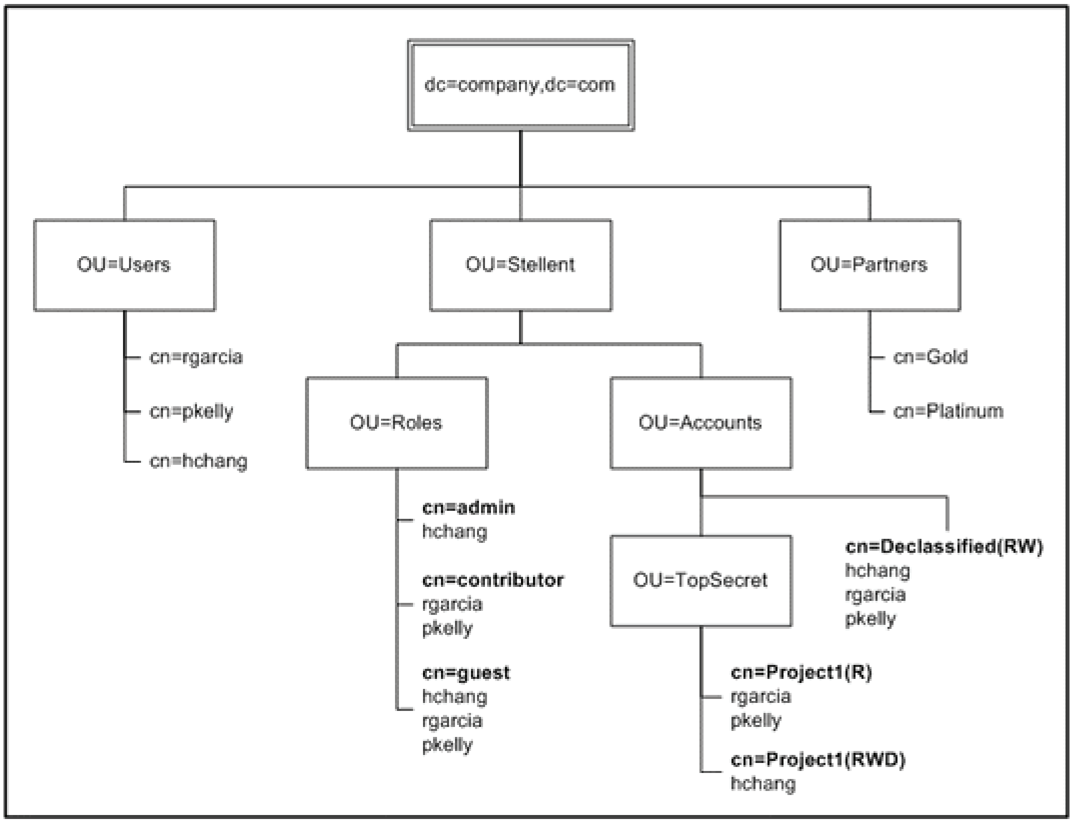
Note: Active Directory security is subject to Windows domain rules and limitations. See Domains (page 6-4) and Domains and Oracle Content Server (page 6-7) for more information.

Note: If you are planning to authenticate users against an Active Directory server using an LDAP provider instead of the direct integration described in this document, you will need to use the Active Directory LDAP component. This component is installed with Content Server.

Active Directory Structure

Active Directory organizes network objects in an LDAP-style hierarchical structure called a *directory tree*. The tree diagram in Figure 6-1 shows a typical Active Directory structure:

Figure 6-1 Example of an Active Directory Structure



In this case, a user could be assigned to a group that has a DN (distinguished name) of:

```
cn=Project1,OU=TopSecret,OU=Accounts,OU=Stellent,dc=company,dc=com
```

where the group DN follows these LDAP conventions:

LDAP Abbreviation	LDAP Designation	Description
dc	domain component	Top-level unit that specifies the naming context.
OU	organizational unit	Typically represents a division, department, or other discrete business group. In an Oracle integration, roles and accounts are typically specified as OUs.
cn	common name	A typical lowest-level attribute in a DN, which identifies the unique name. For users, this could also be a “user ID” unit, or *uid*.

Domains

In a Windows network, a *domain* is a group of computers that share a common domain name and security information. One Windows Server machine is selected as the *primary domain controller* to administer the security for the domain. A user can be either a local user, with security information defined on the user’s local computer, or a global (domain) user, with security administered by the primary controller. When a global user logs in to the domain, the primary controller is queried to get the user’s security credentials.

Tech Tip: Ensure that the latest Windows Service Pack is installed on the primary domain controller machine. Content Server security integration is ***not*** supported for a network environment that is running on Microsoft Windows 2000 with Service Pack 1 or no Service Pack.

There are three Windows network components that affect security and determine whether network users will be able to access a Content Server instance:

❖ Domain security is usually established through the use of *global* (or *domain*) *groups, local groups*, and for a native mode domain, *universal groups*. (Universal groups are not available in a mixed mode domain.)

❖ Each global or local group has a *scope*, which defines it as either a *distribution group* or a *security group*. For the purposes of integration with Content Server, only

Windows security groups can be mapped to Content Server roles and accounts; Distribution groups do not provide sufficient information for authenticating user credentials.

- Each domain is set up as either a *mixed mode domain* or a *native mode domain*, which can include only Windows 2000 Servers.

Groups can be nested, depending on the type of group and mode of the domain. For example, you can assign a user to a global group, and then assign the global group to a local group that has permission to access resources on the local computer.

The following table shows how Windows security groups can be nested:

Group Type	Domain Mode	Can Contain Local Groups?	Can Contain Global Groups?	Can Contain Universal Groups?
Local	Mixed mode	No	Yes	Not available
	Native mode	Yes	Yes	Yes
Global	Mixed mode	No	No	Not available
	Native mode	No	Yes	No
Universal (Active Directory)	Mixed mode	Not available	Not available	Not available
	Native mode	No	Yes	Yes

Trusted Domains

Although small organizations can store user accounts and resources in a single domain, large organizations typically establish multiple domains. For example, user accounts may be stored in one domain and resources in another domain. Windows Server integrates security across multiple domains through *trust relationships*, which are links that combine two domains into one administrative unit that can authorize access to resources on both domains.

There are two types of trust relationships:

- **One-way trust relationship**—One domain trusts the users in another domain to use its resources. More specifically, one domain trusts the domain controllers in the other domain to validate user accounts that reside in that other domain.
- **Two-way trust relationship**—Two one-way trusts are set up so that each domain trusts the users in the other domain. Users can log in from computers in either domain

and use resources that reside in either domain. Global groups in either domain can also grant rights and permissions to resources in either domain.

See Also

– *Domains and Oracle Content Server (page 6-7)*

INTEGRATING ACTIVE DIRECTORY SECURITY

This section covers these topics:

- Microsoft Login (page 6-6)
- Active Directory Security Restrictions (page 6-7)
- Domains and Oracle Content Server (page 6-7)
- Active Directory Authentication Process (page 6-8)
- Mapping Roles and Accounts (page 6-9)

Microsoft Login

When you integrate Active Directory security with Content Server, the user logins, passwords, and permissions are derived from Active Directory information. A Microsoft Login button on the portal navigation bar enables users to log in to Content Server without having to re-enter their user name and password. Clicking the standard Login button still prompts the user for a user name and password.

Note: If an Override check box is selected on a user's User Profile page, any user information defined in the Content Server database overrides the user information derived from Active Directory credentials.

Tech Tip: How does a browser know when to use Active Directory authentication?

- If the Microsoft Login button is clicked, user information will be retrieved from Active Directory rather than from the Content Server database.
- If Active Directory security is enabled and a user requests a secure Content Server resource without logging in, the browser knows to use Active Directory authentication because a cookie is installed the first time a user's credentials are sent to Content Server.

Tech Tip: If you are logging in as different Content Server users for testing purposes, you can force a login from the Microsoft Login button in Internet Explorer as follows:

1. In Internet Explorer, select **Tools—Internet Options**.
2. Click the Security tab.
3. Select the Internet zone (or whichever zone includes Content Server).
4. Click **Custom Level**.
5. Under **User Authentication—Logon**, select the **Prompt for user name and password** option.
6. Click **OK** twice.

Active Directory Security Restrictions

Below is a list of Active Directory security restrictions:

- **Web Server**: You must use IIS 5.0 or newer.
- **Users**: Users must be in a global or local group in a Windows domain to have secured access to Content Server. (Domain users can be granted guest access without being in a domain group.)

Note: Active Directory security is subject to Windows domain rules and limitations. See Domains (page 6-4) and Domains and Oracle Content Server (page 6-7)for more information.

Domains and Oracle Content Server

There are several things to consider when integrating a Content Server instance with Windows network domains:

- Local users cannot be granted access to Content Server roles and accounts. Because Content Server access is determined by group names, Content Server users must be assigned to a global group or local group.
- A domain user is distinguished by having a DOMAINNAME\ prefix added to their user name.
  - Users in global groups in the same domain as Content Server do not need to include the DOMAINNAME\ prefix when they log in to Content Server; the domain of the primary domain controller is assumed.

  - Users in local groups must include the DOMAINNAME\ prefix upon login for Content Server to recognize their user name.
  - Users in groups that are in a different domain than Content Server must include the DOMAINNAME\ prefix upon login for Content Server to recognize their user name.

❖ If Content Server users are members of global groups in a different domain than the Content Server domain, you will need to create two-way trust relationships between those domains.

❖ If you want users from one domain to map to roles in Content Server in another domain, the domain name must be included in the role name. For example:

  - Content Server is in the *corporate* domain.
  - Users are in the *HRStaff* group in the *HR* domain.
  - The users will have permission to a Content Server role named *HR\HRStaff*. They will not have permission to a role named *HRStaff*.

See Also

– *Domains (page 6-4)*

Active Directory Authentication Process

When Active Directory security is integrated with Content Server, IIS and the web server filter work together as follows to authenticate a user's credentials:

1. A client makes a request to the IIS web server.
2. If the web server filter requires credentials for the requested resource, the web server filter checks to see if the user is defined as an internal user in Content Server.
   - If the user is defined as a local or global Content Server user, the roles, accounts, and user attributes defined for this user in Content Server are used.
   - If the user is not defined, or is defined only as an external user in Content Server, the web server filter initiates the challenge/response sequence:
     a. The web server queries the Active Directory server for the user password.
     b. The web server filter validates the user password.
     c. When the challenge response/sequence has been completed successfully, the web server filter retrieves the groups the user belongs to from the Active Directory server.

   d. The user's Active Directory groups are mapped to Content Server roles and accounts. (See Mapping Roles and Accounts (page 6-9)).

   e. If specified on the Active Directory Configuration Page (page 6-17), the web server filter retrieves user information, such as e-mail and user type, from the Active Directory server.

3. If the request is a CGI URL, the request is forwarded to Content Server along with the user's roles and accounts.

4. If the request is a static URL, the web server filter checks the user's roles and accounts to verify that the user has sufficient permission or rights to access the requested resource.

   - If the user does not have sufficient credentials, an error page is retrieved.
   - If the user has sufficient credentials, the request is forwarded to Content Server along with the user's roles and accounts.

Mapping Roles and Accounts

When Active Directory security is integrated with Content Server, the Active Directory groups that a user belongs to are mapped to Content Server roles and accounts as follows:

- ❖ The Group Filtering (Role and Account Prefixes) (page 6-10) and Full Group Names (page 6-10) settings are used to parse group names to determine the user's roles and accounts.
- ❖ If a group name is parsed as a role and matches the name of a Content Server role, the user is granted permissions based on that role.
- ❖ If a group name is parsed as an account, the user is assigned that account; you can create new accounts from Active Directory groups in this manner.
- ❖ Parsed group names that do not match a Content Server role or account prefix will be ignored.
- ❖ Account permissions are determined either from the group name itself or the default permission setting. See Account Permissions (page 6-13).

Group Filtering (Role and Account Prefixes)

Group Filtering is used to specify which parts of an Active Directory group name map to a Content Server role or account.

- When Group Filtering is enabled on the Active Directory Configuration Page (page 6-17), the *Role Prefix* and *Account Prefix* fields are used to filter each user's group names. You can specify an unlimited number of role prefixes and account prefixes, which are substrings that are compared to each Active Directory group name. If a prefix substring is found in the group name, the rest of the group name after this prefix (at a lower level in the directory tree) will be parsed as the role or account. The resulting role and account mapping depends on whether the Full Group Names (page 6-10) setting is enabled.
- If Group Filtering is disabled, all group names are parsed as Content Server *roles*, according to the Full Group Names (page 6-10) setting.
- Each role prefix and account prefix can have a Depth (page 6-12) parameter, which specifies the maximum number of levels that can exist between the prefix and the last unit in the group name for the group to be considered a valid role or account.
- Placing an asterisk (*) in the depth parameter for a specific prefix ensures that the short name for any group mapped through the prefix is used. For example, For a group with a DN of CN=TestApp,OU=Apps,OU=Roles,[LdapSuffix]:

```
Role Prefix -> Role Name
OU=Roles[4] -> Apps/TestApp
OU=Roles[*4] -> TestApp
```

The following are typical role and account prefixes, where the directory tree includes an OU named *Stellent*, with child OUs named *Roles* and *Accounts*:

Role Prefix: `OU=Roles,OU=Stellent`

Account Prefix: `OU=Accounts,OU=Stellent`

Note: Do not include spaces before or after the commas that separate units in a prefix. Also, Content Server turns any % into a slash to make hierarchical account structures. For example, Group Name FOO%BOO%BASH is mapped to the account FOO/BOO/BASH.

Full Group Names

Full Group Names is used to include the entire tree structure of an Active Directory group name in Content Server role or account names. This setting is particularly useful for security models that use a hierarchical account structure in Oracle Content Server.

- When Full Group Names is enabled on the Active Directory Configuration Page (page 6-17), all units in the directory tree except for the naming context (such as *dc=company,dc=com*) will be included in the role or account name.

Note: Placing an asterisk (*) in the depth parameter for a specific prefix ensures that the short name for any group mapped through the prefix is used.

- When Full Group Names is disabled, only the last unit in the directory tree is mapped as the role or account name.
- Role and account mappings also depend on whether the Group Filtering (Role and Account Prefixes) (page 6-10) setting is enabled.
- The following example shows the result of enabling or disabling the Full Group Names setting:

  Group: `CN=admin,OU=Mgr,OU=Dept,OU=Roles,OU=Stellent,dc=company,dc=com`

  Account name (Full Group Names disabled): `admin`

  Account name (Full Group Names enabled): `Dept/Mgr/admin`

Mapping Examples

This section illustrates some possible combinations of the Group Filtering (Role and Account Prefixes) (page 6-10) and Full Group Names (page 6-10) settings:

Role Mapping Example

Group: `CN=admin,OU=Mgr,OU=Dept,OU=Roles,OU=Stellent,dc=company,dc=com`

Role Prefix: `OU=Roles,OU=Stellent[2]`

Group Filtering	Full Group Names	Result
Enabled	Enabled	**Role** = Dept/Mgr/admin
Enabled	Disabled	**Role** = admin
Disabled	Enabled	**Role** = Stellent/Roles/Dept/Mgr/admin
Disabled	Disabled	**Role** = admin

Account Mapping Example

Group: `CN=admin,OU=Mgr,OU=Dept,OU=Accounts,OU=Stellent,dc=company,dc=com`

Role Prefix: `OU=Accounts,OU=Stellent[2]`

Group Filtering	Full Group Names	Result
Enabled	Enabled	**Account** = Dept/Mgr/admin
Enabled	Disabled	**Account** = admin
Disabled	Enabled	**Role** = Stellent/Accounts/Dept/Mgr/admin
Disabled	Disabled	**Role** = admin

Depth

The *depth* parameter of a role prefix or account prefix specifies the maximum number of levels that can exist between the prefix and the last unit in the group name for the group to be considered a valid role or account. The depth parameter helps prevent inadvertent granting of permission to groups deep in the directory tree.

Note: Placing an asterisk (*) in the depth parameter for a specific prefix ensures that the short name for any group mapped through the prefix is used.

❖ The depth is a number specified in square brackets [ ] after the prefix definition. For example, `OU=Roles,OU=Stellent[1]`.

❖ If the number of levels between the prefix and the last unit is the same or less than the depth setting, the group will be mapped to a role or account. If a group name has more levels between the prefix and the last unit than specified by the depth parameter, the group will not be mapped to a role or account.

For example:

Role Prefix: `OU=Roles,OU=Stellent[1]`

Group 1: `CN=admin,OU=Roles,OU=Stellent,dc=company,dc=com`

Group 2: `CN=admin,OU=Mgr,OU=Roles,OU=Stellent,dc=company,dc=com`

Group 3: `CN=admin,OU=Mgr,OU=Dept,OU=Roles,OU=Stellent,dc=company,dc=com`

In this case:

- Group 1 would be considered a valid role, because there is less than one level between the prefix and the last unit.

- Group 2 would be considered a valid role, because there is exactly one level between the prefix and the last unit.
- Group 3 would **not** be considered a valid role, because there is more than one level between the prefix and the last unit.

❖ If no depth is specified for a prefix, the depth defaults to 0. This means that the lowest level in the group name must be directly after the prefix.

For example:

Role Prefix: `OU=Roles,OU=Stellent`

Group 1: `CN=admin,OU=Roles,OU=Stellent,dc=company,dc=com`

Group 2: `CN=admin,OU=Mgr,OU=Roles,OU=Stellent,dc=company,dc=com`

In this case:

- Group 1 would be considered a valid role, because there are no levels between the prefix and the last unit.
- Group 2 would **not** be considered a valid role, because there is one level between the prefix and the last unit.

Account Permissions

Account permissions can be specified in the following ways:

❖ From the Active Directory group name itself, preceded by the character specified in the Account Permissions Delimiter field (page 6-21) on the Active Directory Configuration page.

❖ From the default account settings specified in the Default Network Accounts field (page 6-23) on the Active Directory Configuration page. The default setting is *#none(RWDA)*, which means that Active Directory users have Admin permission to all content items that do not have an account assigned.

Note: The default account permissions are added to the permissions defined by an Active Directory group. For example, if the default is *#none(RW),Project(R)* and a user's group maps to *Project(RWD)* permission, the user's permissions are *#none(RW),Project(RWD).*

SETTING UP ACTIVE DIRECTORY SECURITY

This section covers these topics:

- Setting Up the Content Server for Active Directory (page 6-14)
- Enabling Active Directory Security (page 6-14)
- Configuring Active Directory Security (page 6-15)

Setting Up the Content Server for Active Directory

To integrate Active Directory security with Oracle Content Server, you must set up Content Server as follows:

1. Make sure that the IIS web server is configured for external security integration.
2. Enable Active Directory security. See Enabling Active Directory Security (page 6-14).
3. Set configuration entries to match your Active Directory setup. See Configuring Active Directory Security (page 6-15).

Note: If you are planning to authenticate users against an Active Directory server using an LDAP provider instead of the direct integration described in this document, you will need to use the Active Directory LDAP component. This component is installed with Content Server.

Enabling Active Directory Security

Use one of the following procedures to enable Active Directory security in Content Server.

- During Installation

  During installation, select the Active Directory Security option on the Security screen. This enables Active Directory security and displays the Microsoft Login button in the portal navigation bar.

- Using System Properties

  1. Access **System Properties** for your Content Server instance.
  2. Click the Internet tab.

3. Select the Use Microsoft Security check box.
4. Select the Active Directory Security (ADSI) option.
5. Click **OK**.
6. Restart Content Server.
7. Restart the IIS service.
8. Go to the Content Server Home page or Administration page, and verify that the Microsoft Login button appears in the portal navigation bar.

❖ Using the Admin Server

1. Log in to Content Server as the system administrator.
2. Click **Administration** in the portal navigation bar.
3. Click **Admin Server**.
4. Click **Master_on_*instance.***
5. Click **Internet Configuration** in the portal navigation bar.
6. Under Use Microsoft Security, select the Active Directory Security option.
7. Click **Save**.
8. Restart Content Server.
9. Restart the IIS service.
10. Go to the Content Server Home page or Administration page, and verify that the Microsoft Login button appears in the portal navigation bar.

Configuring Active Directory Security

To configure Active Directory security:

1. Enable Active Directory security. See Enabling Active Directory Security (page 6-14).
2. Log in to Content Server as the system administrator.
3. Click the Administration link in the portal navigation bar.
4. Click **Filter Administration**.

   The Configure Web Server Filter page is displayed.
5. Click **Configure** under Special Integrations.

   The Active Directory Configuration Page (page 6-17) is displayed.

6. To specify a Role Prefix and/or Account Prefix:
   a. Select the Group Filtering check box.
   b. Enter the role or account prefix in the appropriate Prefix field.
   c. Enter a number in the Depth field. (If no number is specified, the default is 0.)
   d. Click **Add**.

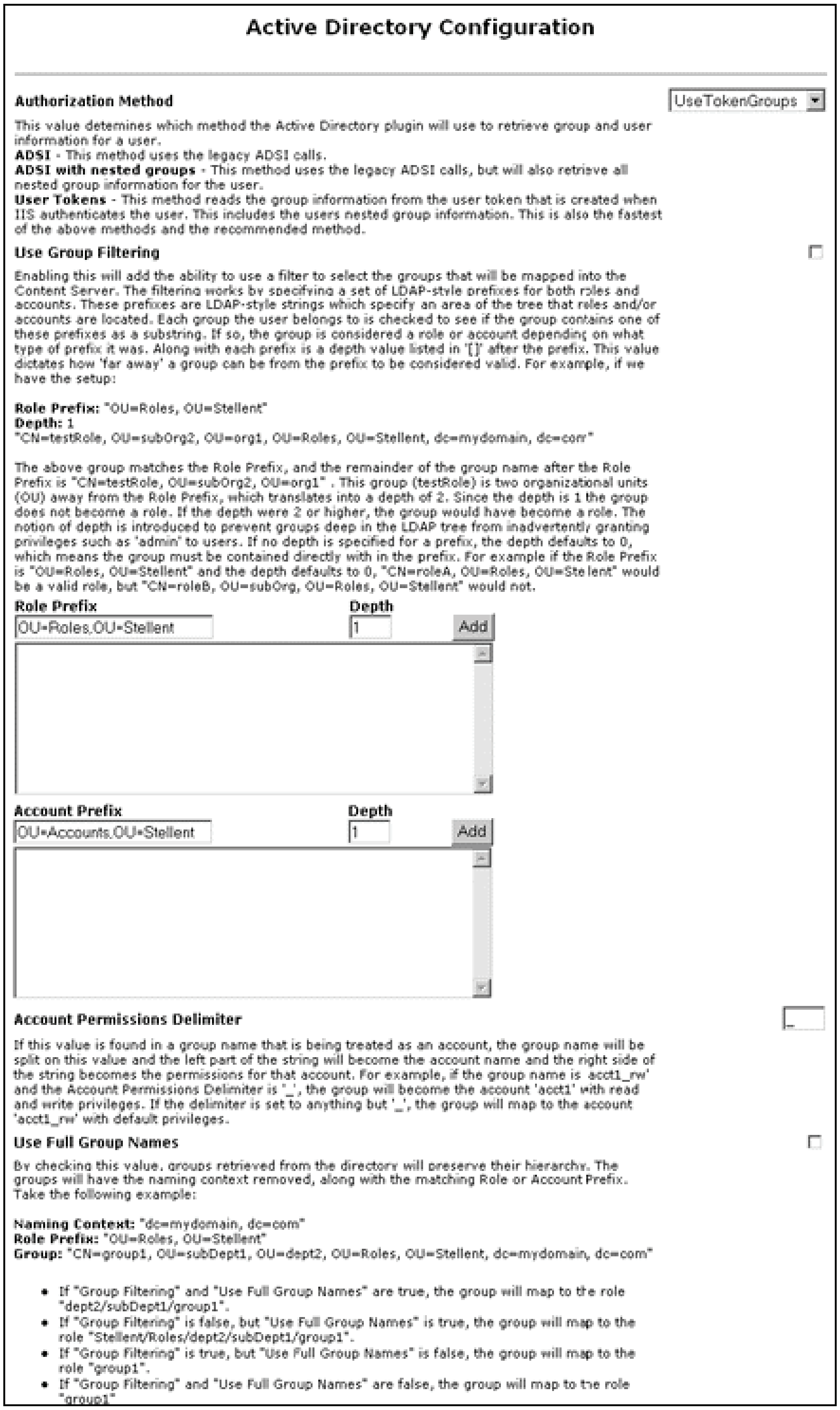
      The role or account prefix is added to the corresponding text box.
   e. If necessary, edit the prefixes directly in the text boxes.

Note: See Mapping Roles and Accounts (page 6-9) for more information.

7. To specify a user attribute map:
   a. In the Attribute Map section, enter an Active Directory user attribute in the LDAP Attribute field.
   b. Select a Content Server user information field from the User Attribute list.
   c. Click **Add**.

      The attribute map is added to the text box.
   d. If necessary, edit the attribute maps directly in the text box.
8. Change any other configuration settings as necessary.
9. Click **Update**.
10. If your changes do not appear to have been made, restart the IIS service.

ACTIVE DIRECTORY CONFIGURATION PAGE



Active Directory Configuration

Authorization Method UseTokenGroups ▾

This value detemines which method the Active Directory plugin will use to retrieve group and user information for a user.
ADSI - This method uses the legacy ADSI calls.
ADSI with nested groups - This method uses the legacy ADSI calls, but will also retrieve all nested group information for the user.
User Tokens - This method reads the group information from the user token that is created when IIS authenticates the user. This includes the users nested group information. This is also the fastest of the above methods and the recommended method.

Use Group Filtering □

Enabling this will add the ability to use a filter to select the groups that will be mapped into the Content Server. The filtering works by specifying a set of LDAP-style prefixes for both roles and accounts. These prefixes are LDAP-style strings which specify an area of the tree that roles and/or accounts are located. Each group the user belongs to is checked to see if the group contains one of these prefixes as a substring. If so, the group is considered a role or account depending on what type of prefix it was. Along with each prefix is a depth value listed in '[]' after the prefix. This value dictates how 'far away' a group can be from the prefix to be considered valid. For example, if we have the setup:

Role Prefix: "OU=Roles, OU=Stellent"
Depth: 1
"CN=testRole, OU=subOrg2, OU=org1, OU=Roles, OU=Stellent, dc=mydomain, dc=com"

The above group matches the Role Prefix, and the remainder of the group name after the Role Prefix is "CN=testRole, OU=subOrg2, OU=org1" . This group (testRole) is two organizational units (OU) away from the Role Prefix, which translates into a depth of 2. Since the depth is 1 the group does not become a role. If the depth were 2 or higher, the group would have become a role. The notion of depth is introduced to prevent groups deep in the LDAP tree from inadvertently granting privileges such as 'admin' to users. If no depth is specified for a prefix, the depth defaults to 0, which means the group must be contained directly with in the prefix. For example if the Role Prefix is "OU=Roles, OU=Stellent" and the depth defaults to 0, "CN=roleA, OU=Roles, OU=Stellent" would be a valid role, but "CN=roleB, OU=subOrg, OU=Roles, OU=Stellent" would not.

Role Prefix: OU=Roles,OU=Stellent **Depth**: 1

Account Prefix: OU=Accounts,OU=Stellent **Depth**: 1

Account Permissions Delimiter _

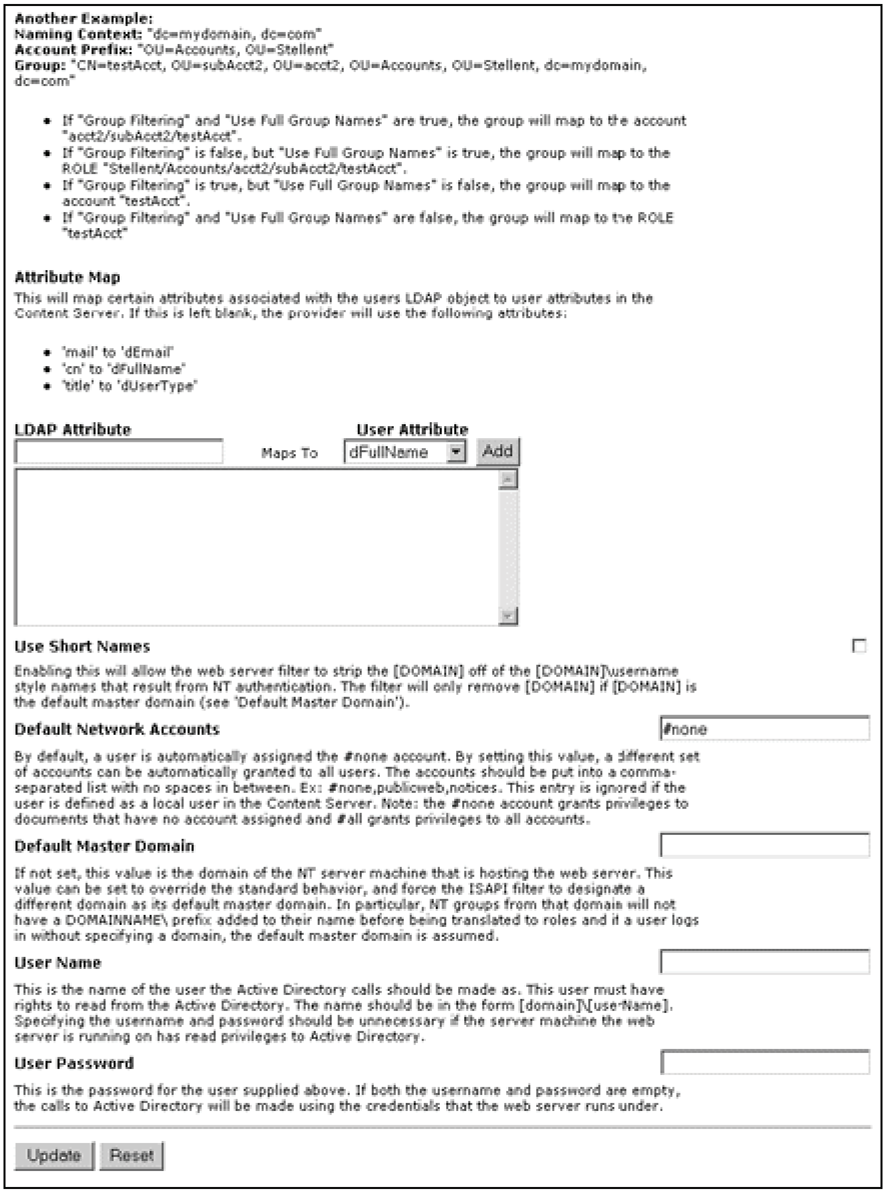
If this value is found in a group name that is being treated as an account, the group name will be split on this value and the left part of the string will become the account name and the right side of the string becomes the permissions for that account. For example, if the group name is 'acct1_rw' and the Account Permissions Delimiter is '_', the group will become the account 'acct1' with read and write privileges. If the delimiter is set to anything but '_', the group will map to the account 'acct1_rw' with default privileges.

Use Full Group Names □

By checking this value, groups retrieved from the directory will preserve their hierarchy. The groups will have the naming context removed, along with the matching Role or Account Prefix. Take the following example:

Naming Context: "dc=mydomain, dc=com"
Role Prefix: "OU=Roles, OU=Stellent"
Group: "CN=group1, OU=subDept1, OU=dept2, OU=Roles, OU=Stellent, dc=mydomain, dc=com"

- If "Group Filtering" and "Use Full Group Names" are true, the group will map to the role "dept2/subDept1/group1".
- If "Group Filtering" is false, but "Use Full Group Names" is true, the group will map to the role "Stellent/Roles/dept2/subDept1/group1".
- If "Group Filtering" is true, but "Use Full Group Names" is false, the group will map to the role "group1".
- If "Group Filtering" and "Use Full Group Names" are false, the group will map to the role "group1"



Another Example:
Naming Context: "dc=mydomain, dc=com"
Account Prefix: "OU=Accounts, OU=Stellent"
Group: "CN=testAcct, OU=subAcct2, OU=acct2, OU=Accounts, OU=Stellent, dc=mydomain, dc=com"

- If "Group Filtering" and "Use Full Group Names" are true, the group will map to the account "acct2/subAcct2/testAcct".
- If "Group Filtering" is false, but "Use Full Group Names" is true, the group will map to the ROLE "Stellent/Accounts/acct2/subAcct2/testAcct".
- If "Group Filtering" is true, but "Use Full Group Names" is false, the group will map to the account "testAcct".
- If "Group Filtering" and "Use Full Group Names" are false, the group will map to the ROLE "testAcct"

Attribute Map

This will map certain attributes associated with the users LDAP object to user attributes in the Content Server. If this is left blank, the provider will use the following attributes:

- 'mail' to 'dEmail'
- 'cn' to 'dFullName'
- 'title' to 'dUserType'

LDAP Attribute Maps To **User Attribute** dFullName

Use Short Names □

Enabling this will allow the web server filter to strip the [DOMAIN] off of the [DOMAIN]\username style names that result from NT authentication. The filter will only remove [DOMAIN] if [DOMAIN] is the default master domain (see 'Default Master Domain').

Default Network Accounts #none

By default, a user is automatically assigned the #none account. By setting this value, a different set of accounts can be automatically granted to all users. The accounts should be put into a comma-separated list with no spaces in between. Ex: #none,publicweb,notices. This entry is ignored if the user is defined as a local user in the Content Server. Note: the #none account grants privileges to documents that have no account assigned and #all grants privileges to all accounts.

Default Master Domain

If not set, this value is the domain of the NT server machine that is hosting the web server. This value can be set to override the standard behavior, and force the ISAPI filter to designate a different domain as its default master domain. In particular, NT groups from that domain will not have a DOMAINNAME\ prefix added to their name before being translated to roles and if a user logs in without specifying a domain, the default master domain is assumed.

User Name

This is the name of the user the Active Directory calls should be made as. This user must have rights to read from the Active Directory. The name should be in the form [domain]\[userName]. Specifying the username and password should be unnecessary if the server machine the web server is running on has read privileges to Active Directory.

User Password

This is the password for the user supplied above. If both the username and password are empty, the calls to Active Directory will be made using the credentials that the web server runs under.

The Active Directory Configuration page is used to configure Content Server integration with Active Directory security. To access this page, enable Active Directory security (see Enabling Active Directory Security (page 6-14)), and click **Configure** on the Configure Web Server Filter page.

In the following tables, the term in parentheses in the first column is the corresponding configuration setting in the <*Install_Dir*>/data/users/config/filter.hda file.

Feature	Description
Authorization Method Section	
Authorization Method field	Select an authorization method from the option list: • UseTokenGroups • UseNestedGroups • UseBasicGroups The Active Directory plug-in will use this value to retrieve group and user information for a user. **User Tokens**—This method reads the group information from the user token created when IIS authenticates the user, including the user’s nested group information. This is the fastest method and the recommended method. **ADSI with nested groups**—This method uses the legacy ADSI calls, but also retrieves all nested group information for the user. **ADSI**—This basic method uses the legacy ADSI calls.
Use Group Filtering Section See Mapping Roles and Accounts (page 6-9) for more information.	
Group Filtering check box (UseGroupFilter)	**Selected—**The Role Prefix and Account Prefix definitions will be used to select the Active Directory groups that will be mapped to Content Server roles and accounts. **Clear—**All Active Directory groups will be mapped to Content Server roles and accounts. This is the default.
Role Prefix field	The string that specifies where in the Active Directory group name to start matching a Content Server role name.
Role Prefix Depth field	A number that specifies how many levels the Active Directory group name can contain after the Role Prefix for the group name to be considered a valid role.
Role Prefix Add button	Adds the Role Prefix string and Depth as a clause in the Role Prefix box.

Feature	Description
Role Prefix box (RolePrefix)	Lists the Role Prefix clauses that will be used to select Active Directory groups when the Group Filtering check box is selected. This box can be edited directly. **Note:** Do not include spaces before or after the commas that separate units in a prefix.
Account Prefix field	The string that specifies where in the Active Directory group name to start matching a Content Server account name. This field appears only if accounts are enabled in Content Server.
Account Prefix Depth field	A number that specifies how many levels the Active Directory group name can contain after the Account Prefix for the group name to be considered a valid account. This field appears only if accounts are enabled in Content Server.
Account Prefix Add button	Adds the Account Prefix string and Depth as a clause in the Account Prefix box. This button appears only if accounts are enabled in Content Server.
Account Prefix box (AcctPrefix)	Lists the Account Prefix clauses that will be used to select Active Directory groups when the Group Filtering check box is selected. This box can be edited directly. **Note:** Do not include spaces before or after the commas that separate units in a prefix. This box appears only if accounts are enabled in Content Server.

Feature	Description
Account Permissions Delimiter Section See Mapping Roles and Accounts (page 6-9) for more information.	
Account Permissions Delimiter field (AcctPermDelim)	The string that separates the account name from the account permissions in an Active Directory group name. • If an Active Directory group name is mapped to an account and contains this substring, the string to the left of this substring will be the account name, and the string to the right of this substring will be the account permissions. • For example, if the delimiter is defined as a + (plus sign), the group name *Acct1+rw* would map to an account named *Acct1* with Read and Write permission. If the delimiter is defined as _ (underscore), the *Acct1+rw* group name would map to an account named *Acct1+rw*, with RWDA permission by default. • Default is _ (underscore). • This field appears only if accounts are enabled in Content Server.
Use Full Group Names Section See Mapping Roles and Accounts (page 6-9) for more information.	
Use Full Group Names check box (UseFullGroupName)	**Selected—**The entire hierarchy (up to the specified prefix or naming context) for an Active Directory group will be included in the mapping to a Content Server role or account. **Clear—**Only the lowest level unit of an Active Directory group will be mapped to a Content Server role or account. This is the default.
Attribute Map Section	
LDAP Attribute field	Enter an Active Directory user attribute to be mapped to a Content Server user information field.

Feature	Description
User Attribute field	Select a Content Server user information field to be mapped from the LDAP Attribute field. • All Content Server user information fields for which you can change the value are listed. • Standard user information fields begin with a “d”. • Custom user information fields begin with a “u”.
Add button	Adds the LDAP Attribute and User Attribute as a colon-separated clause in the Attribute Map box.

Attribute Map box
(AttributeMap)
	Lists the Attribute Map clauses that will be used to map Active Directory user attributes to Content Server information fields.
	• This box can be edited directly.
	• If this field is left blank, the default is:
	`mail:dEmail`
	`cn:dFullName`
	`title:dUserType`

Use Short Names Section	
Use Short Names check box (UseShortNamesAlways)	**Selected—**The web server filter removes the DOMAINNAME\ prefix from all Active Directory user names. **Clear—**The DOMAINNAME\ prefix is included with all Active Directory user names except the default master domain. This is the default.

Feature	Description
Default Network Accounts Section	
Default Network Accounts field (DefaultNetworkAccounts)	Defines the default account permissions for users who log in to Content Server with Active Directory credentials. • This must be a comma-separated list of accounts. **Note:** Do not include spaces before or after the commas that separate accounts. • Permissions for each account can be specified in parentheses after the account name, such as *account(RWDA)*. If no permissions are specified, RWDA permission is granted by default. • The *#none* entry grants permission to documents that have no account assigned. • The *#all* entry grants permission to all accounts. • Default is *#none(RWDA)*. • This setting does not apply to anonymous users. • This setting defines the *minimum* account permissions. Account permissions defined by the external user base are added to these permissions. For example, if the default is *#none(RW),Project(R)* and a user's group maps to *Project(RWD)* permission, the user's permissions are *#none(RW),Project(RWD)*. • This field appears only if accounts are enabled in Content Server.

Feature	Description
Default Master Domain Section	
Default Master Domain field (DefaultMasterDomain)	The Windows domain that is the default master domain for the web server filter. • Active Directory users from the specified domain will not have a DOMAINNAME\ prefix added to the user name before groups are translated to roles and accounts. • If a user logs in without specifying a domain prefix, this default master domain is assumed. • Default is the domain of the Windows Server machine that is hosting the web server.
User Name Section	
User Name field (AdsUserName)	If the web server is running on a Windows Server that does not have Read permission to Active Directory, enter a user name that has Read permission for Active Directory. • The user name must be in the form *DOMAIN_NAME\username*. • If this setting and the User Password setting are not specified, the default is the user name that the web server runs under.
User Password Section	
User Password field (AdsUserPassword)	If the web server is running on a Windows Server that does not have Read permission to Active Directory, enter the password for the User Name. • If this setting and the User Name setting are not specified, the default is the user name that the web server runs under.

Chapter

7

External Security: LDAP

Overview

This section covers these topics:

Concepts

Tasks

Interface

LDAP Overview

This section covers these topics:

❖ About LDAP (page 7-2)

❖ LDAP Directory Structure (page 7-3)

About LDAP

LDAP (Lightweight Directory Access Protocol) is a protocol that uses TCP/IP to access information stored in an LDAP directory, such as iPlanet Directory Server or Novell eDirectory. The LDAP directory stores information about objects on a network and makes this information available to applications, users, and network administrators. With LDAP, authorized network users can access resources anywhere on the network using a single login process.

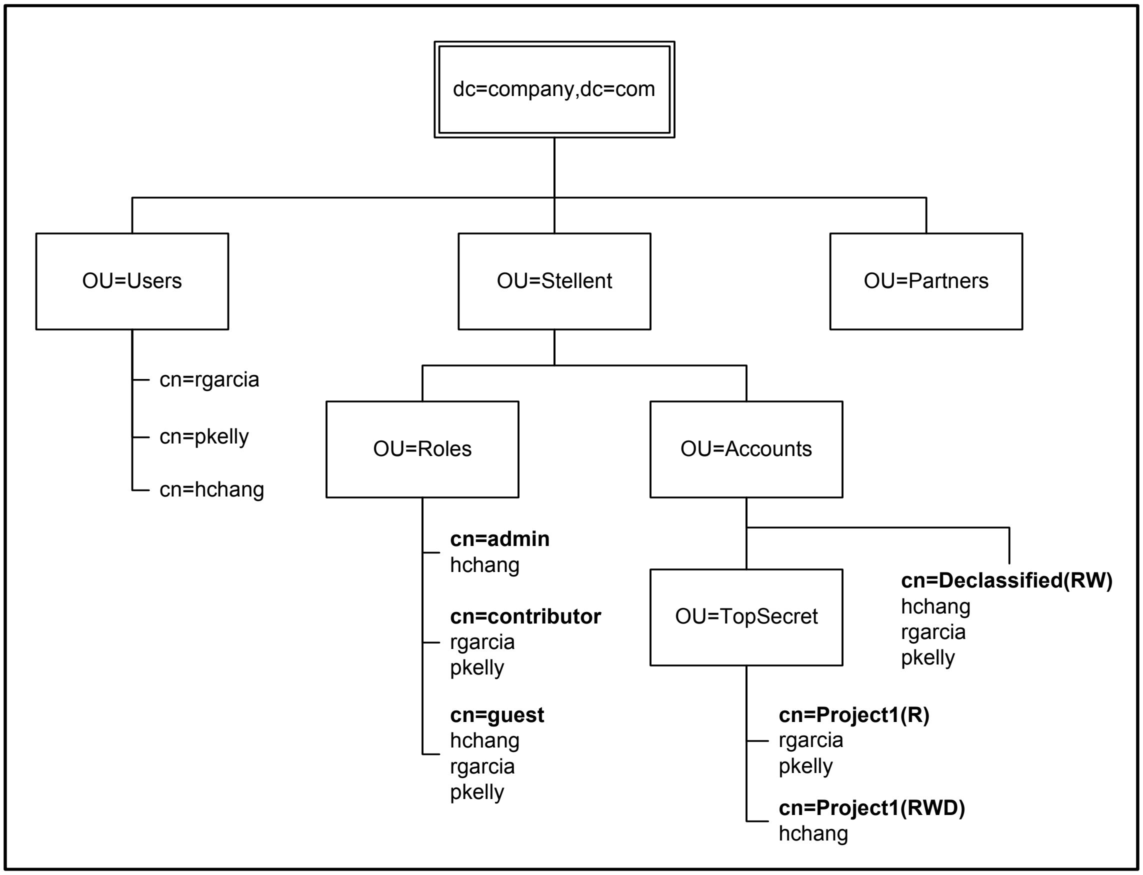
Note: Although it is not required, you are encouraged to have Consulting Services assist you with creating an LDAP security model and deploying the LDAP integration.

Note: If you are planning to authenticate users against an Active Directory server using an LDAP provider instead of the direct integration described in this document, you need to use the Active Directory LDAP component. This component is installed with Content Server.

LDAP Directory Structure

LDAP organizes network objects in a hierarchical structure called a *directory tree*. The tree diagram in Figure 7-1 shows a typical LDAP directory structure:

Figure 7-1 Example of an LDAP directory Structure



In this case, a user could be assigned to a group that has a DN (distinguished name) of:

```
cn=Project1,OU=TopSecret,OU=Accounts,OU=Stellent,dc=company,dc=com
```

where the group DN follows these LDAP conventions:

LDAP Abbreviation	LDAP Designation	Description
dc	domain component	Top-level unit that specifies the naming context. Can also be combined into one “organization” designation, such as *o=company.com*.
OU	organizational unit	Typically represents a division, department, or other discrete business group. In a Content Server integration, roles and accounts are typically specified as OUs.
cn	common name	A typical lowest-level attribute in a DN, which identifies the unique name. For users, this could also be a “user ID” unit, or *uid*.

INTEGRATING LDAP SECURITY

This section covers these topics:

- LDAP Login (page 7-4)
- LDAP Authentication Process (page 7-5)
- Mapping Roles and Accounts (page 7-5)

LDAP Login

When you integrate LDAP security with Content Server, the user logins, passwords, and permissions are derived from information in an LDAP server. Users log in to Content Server using the Login button in the portal navigation bar.

If a user name is defined both in Content Server and in the LDAP server, the Content Server user’s password will take precedence unless the domain name is included as a prefix (for example, *DOMAINNAME\user_name*).

LDAP Authentication Process

When LDAP security is integrated with Content Server, a user's credentials are authenticated as follows:

1. A client browser issues a request to Content Server for a secure resource. The request is received by the web server.
2. The web server filter sends a challenge back to the user to enter a user name and password.
3. The request is sent again to the web server, this time with the user credentials.
4. The web server filter forwards the request to Content Server.
5. If the user name is not recognized as an internal user, the credentials are passed by the LDAP provider to the LDAP directory for authentication.
6. The LDAP directory authenticates the user and sends the user's LDAP group and attribute information to Content Server.
7. Content Server maps the user's groups to Content Server roles and accounts, and maps any user attributes (such as full name and e-mail address) to Content Server user information fields.
8. If the user has permission to access the secure resource that was requested, Content Server serves the response to the client browser.

Mapping Roles and Accounts

When LDAP security is integrated with Content Server, the LDAP groups that a user belongs to are mapped to Content Server roles and accounts as follows:

- The Group Filtering (Role and Account Prefixes) (page 7-6) and Full Group Names (page 7-7) settings are used to parse group names to determine the user's roles and accounts.
- If a group name is parsed as a role and matches the name of a Content Server role, the user is assigned that role.
- If a group name is parsed as an account, the user is assigned that account; you can create new accounts from LDAP groups in this manner.

- Parsed group names that do not match a Content Server role or account prefix will be ignored.
- Account permissions are determined either from the group name itself or the default permission setting. See Account Permissions (page 7-9).
- An LDAP user with no defined role(s) is automatically assigned a guest role.

  If you do not want an LDAP user with no defined role(s) to be assigned a guest role, then configure the LDAP provider to add a role name for a role that does not exist to the Default Network Roles field. For example, you could add the role name “NoRole”.

Group Filtering (Role and Account Prefixes)

Group Filtering is used to specify which parts of an LDAP group name map to a Content Server role or account.

- When Group Filtering is enabled on the LDAP Provider Page (page 7-15), the Role Prefix and Account Prefix fields are used to filter each user’s group names. You can specify an unlimited number of role prefixes and account prefixes, which are substrings that are compared to each LDAP group name. If a prefix substring is found in the group name, the rest of the group name after this prefix (at a lower level in the directory tree) will be parsed as the role or account. The resulting role and account mapping depends on whether the Full Group Names (page 7-7) setting is enabled.
- If Group Filtering is disabled, all group names are parsed as Content Server *roles*, according to the Full Group Names (page 7-7) setting.
- Each role prefix and account prefix can have a Depth (page 7-8) parameter, which specifies the maximum number of levels that can exist between the prefix and the last unit in the group name for the group to be considered a valid role or account.
- The following are typical role and account prefixes, where the directory tree includes an OU named *Stellent*, with child OUs named *Roles* and *Accounts*:

  Role Prefix: `OU=Roles,OU=Stellent`

  Account Prefix: `OU=Accounts,OU=Stellent`

Note: Do not include spaces before or after the commas that separate units in a prefix.

Full Group Names

Full Group Names is used to include the entire tree structure of an LDAP group name in Content Server role or account names. This setting is particularly useful for security models that use a hierarchical account structure in Content Server.

- When Full Group Names is enabled on the LDAP Provider Page (page 7-15), all units in the directory tree except for the naming context (such as *dc=company,dc=com*) will be included in the role or account name.
- When Full Group Names is disabled, only the last unit in the directory tree is mapped as the role or account name.
- Role and account mappings also depend on whether the Group Filtering (Role and Account Prefixes) (page 7-6) setting is enabled.
- The following example shows the result of enabling or disabling the Full Group Names setting:

  Group: `CN=admin,OU=Mgr,OU=Dept,OU=Roles,OU=Stellent,dc=company,dc=com`

  Account name (Full Group Names disabled): `admin`

  Account name (Full Group Names enabled): `Dept/Mgr/admin`

Mapping Examples

This section illustrates some possible combinations of the Group Filtering (Role and Account Prefixes) (page 7-6) and Full Group Names (page 7-7) settings:

Role Mapping Example

Group: `CN=admin,OU=Mgr,OU=Dept,OU=Roles,OU=Stellent,dc=company,dc=com`

Role Prefix: `OU=Roles,OU=Stellent[2]`

Group Filtering	Full Group Names	Result
Enabled	Enabled	**Role** = Dept/Mgr/admin
Enabled	Disabled	**Role** = admin
Disabled	Enabled	**Role** = Stellent/Roles/Dept/Mgr/admin
Disabled	Disabled	**Role** = admin

Account Mapping Example

Group: `CN=admin,OU=Mgr,OU=Dept,OU=Accounts,OU=Stellent,dc=company,dc=com`

Role Prefix: `OU=Accounts,OU=Stellent[2]`

Group Filtering	Full Group Names	Result
Enabled	Enabled	**Account** = Dept/Mgr/admin
Enabled	Disabled	**Account** = admin
Disabled	Enabled	**Role** = Stellent/Accounts/Dept/Mgr/admin
Disabled	Disabled	**Role** = admin

Depth

The *depth* parameter of a role prefix or account prefix specifies the maximum number of levels that can exist between the prefix and the last unit in the group name for the group to be considered a valid role or account. The depth parameter helps prevent inadvertent granting of permission to groups deep in the directory tree.

- The depth is a number specified in square brackets [ ] after the prefix definition. For example, `OU=Roles,OU=Stellent[1]`.
- If the number of levels between the prefix and the last unit is the same or less than the depth setting, the group will be mapped to a role or account. If a group name has more levels between the prefix and the last unit than specified by the depth parameter, the group will not be mapped to a role or account.

  For example:

  Role Prefix: `OU=Roles,OU=Stellent[1]`

  Group 1: `CN=admin,OU=Roles,OU=Stellent,dc=company,dc=com`

  Group 2: `CN=admin,OU=Mgr,OU=Roles,OU=Stellent,dc=company,dc=com`

  Group 3: `CN=admin,OU=Mgr,OU=Dept,OU=Roles,OU=Stellent,dc=company,dc=com`

  In this case:

  - Group 1 would be considered a valid role, because there is less than one level between the prefix and the last unit.
  - Group 2 would be considered a valid role, because there is exactly one level between the prefix and the last unit.

    - Group 3 would **not** be considered a valid role, because there is more than one level between the prefix and the last unit.

- If no depth is specified for a prefix, the depth defaults to 0. This means that the lowest level in the group name must be directly after the prefix.

  For example:

  Role Prefix: `OU=Roles,OU=Stellent`

  Group 1: `CN=admin,OU=Roles,OU=Stellent,dc=company,dc=com`

  Group 2: `CN=admin,OU=Mgr,OU=Roles,OU=Stellent,dc=company,dc=com`

  In this case:

    - Group 1 would be considered a valid role, because there are no levels between the prefix and the last unit.
    - Group 2 would **not** be considered a valid role, because there is one level between the prefix and the last unit.

Account Permissions

Account permissions can be specified in the following ways:

- From the LDAP group name itself, preceded by the character specified in the Account Permissions Delimiter field on the LDAP Provider page.
- From the default account settings specified in the Default Network Accounts field on the LDAP Provider page. The default setting is *#none(RWDA)*, which means that LDAP users have Admin permission to all content items that do not have an account assigned.

SETTING UP FOR LDAP SECURITY

This section covers these topics:

Setting Up Content Server for LDAP Security

To integrate LDAP security with Content Server, you must set up Content Server as follows:

1. Make sure that the web server is configured for external security integration.
2. Create an LDAP provider. See Creating an LDAP Provider (page 7-10).
3. Set configuration entries to match your LDAP setup. See Configuring LDAP Security (page 7-12).

Note: If you are planning to authenticate users against an Active Directory server using an LDAP provider instead of the direct integration described in Chapter 6 (*External Security: Active Directory)*, you will need to use the Active Directory LDAP Component. This component is installed with Content Server.

Creating an LDAP Provider

To add an LDAP provider:

1. Log in to Content Server as sysadmin.
2. Click the Administration link in the left navigation bar.
3. Click the Providers link.

   The Providers page is displayed.
4. In the Create a New Provider table, click **Add** in the Action column for the *ldapuser* provider type.

   The LDAP Provider Page (page 7-15) is displayed.
5. Complete the following fields:

   Required fields

   - Provider Name
   - Provider Description
   - Provider Class (predefined)
   - Source Path
   - LDAP Server
   - LDAP Suffix
   - LDAP Port (default is 389, or 636 if SSL is used)

Optional fields

- Connection Class (predefined)
- Configuration Class
- Number of connections (predefined)
- Connection timeout (predefined)

6. If there is more than one LDAP provider for this Content Server instance, enter a number in the Priority field. This number indicates the order in which the providers will be checked when new users request Content Server credentials. Each LDAP provider must have a unique Priority number. (For more information see Setting Up Additional LDAP Providers (page 7-13).)
7. Select the Use Netscape SDK check box (recommended for better performance).
8. To secure communication between the LDAP server and the content server, select the Use SSL check box. (You must have the appropriate certificates installed on the LDAP server.)

Note: If you use SSL, set the Ldap Port field to **636**.

9. If necessary, change the Account Permissions Delimiter.
10. If necessary, add any Default Network Roles.
11. If necessary, change or add to the Default Network Accounts.
12. To specify a Role Prefix and/or Account Prefix:
    a. Select the Use Group Filtering check box.
    b. Enter the role or account prefix in the appropriate Prefix field.
    c. Enter a number in the Depth field. (If no number is specified, the default is 0.)
    d. Click **Add**.

       The role or account prefix is added to the corresponding text box.

    e. If necessary, edit the prefixes directly in the text boxes.

Note: See Mapping Roles and Accounts (page 7-5) for more information.

13. To specify a user attribute map:
    a. In the Attribute Map section, enter an LDAP user attribute in the LDAP Attribute field.

b. Select a Content Server user information field from the User Attribute list.

c. Click **Add**.

The attribute map is added to the text box.

d. If necessary, edit the attribute maps directly in the text box.

14. Enter a user name and password that will be used when Content Server makes calls to the LDAP server.
    - This user must have Read rights to the LDAP server.
    - If the user name is left blank, the provider will connect to the LDAP server anonymously.
    - If the provider is communicating with Active Directory, a user name and password is required, and must be a valid domain user in the format *DOMAIN\username*.

15. Click **Add.**

    The Providers page is displayed with the new provider added to the Providers table.

16. Restart Content Server.

17. Restart the web server.

Configuring LDAP Security

To configure LDAP security:

1. Create an LDAP provider. See Creating an LDAP Provider (page 7-10).
2. Log in to Content Server as sysadmin.
3. Click the Administration link in the left navigation bar.
4. Click the Providers link.

   The Providers page is displayed.

5. In the Providers table, click the Info link in the Action column for the LDAP provider.

   The LDAP Provider Information page is displayed.

6. Click **Edit**.

   The LDAP Provider Page (page 7-15) is displayed.

7. To specify a Role Prefix and/or Account Prefix:

   a. Select the Group Filtering check box.

b. Enter the role or account prefix in the appropriate Prefix field.

c. Enter a number in the Depth field. (If no number is specified, the default is 0.)

d. Click **Add**.

The role or account prefix is added to the corresponding text box.

e. If necessary, edit the prefixes directly in the text boxes.

Note: See Mapping Roles and Accounts (page 7-5) for more information.

8. To specify a user attribute map:

a. In the Attribute Map section, enter an LDAP user attribute in the LDAP Attribute field.

b. Select a Content Server user information field from the User Attribute list.

c. Click **Add**.

The attribute map is added to the text box.

d. If necessary, edit the attribute maps directly in the text box.

9. If necessary, specify the user name that will be making calls to the LDAP server in the LDAP Admin DN field.

10. If a user name was specified for calls to the LDAP server, then specify the password for the user in the LDAP Admin Password field.

11. Click **Update**.

12. Restart Content Server.

13. Restart the web server.

Setting Up Additional LDAP Providers

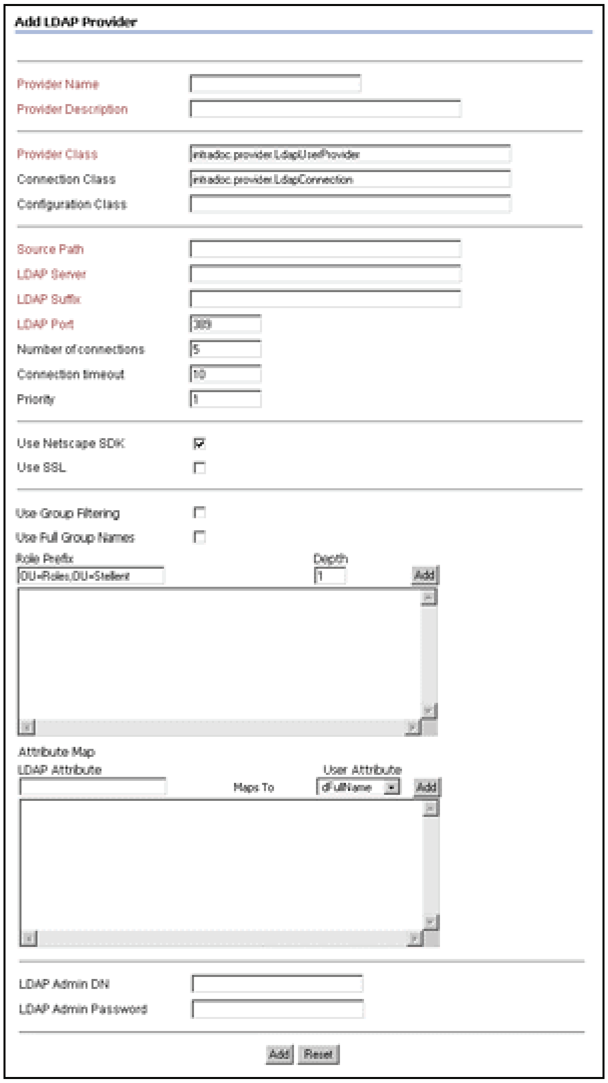
With Content Server you can create additional LDAP providers to use as a failover for another LDAP provider. This is useful if the main LDAP server goes down and the users are unable to log in to Content Server.

To set up an additional LDAP provider, follow the instructions for Creating an LDAP Provider (page 7-10), and use the Priority field that can be specified in the provider definition. The setting in the Priority field tells Content Server the order in which to try an LDAP provider when looking for user information. If the highest-order LDAP provider fails, or the user isn’t found, Content Server tries the next LDAP provider (by the priority rank).

When using multiple LDAP providers, the following items are relevant if the primary LDAP server fails:

- The first few users may notice a slight delay as the connections to the LDAP provider fail.
- When Content Server finds a user using the backup LDAP provider, it will remember that the user came from the backup LDAP provider and go there the next time it needs user information, even if the primary LDAP provider comes back online, and even across Content Server restarts. This behavior prevents Content Server from having to check a whole list of providers each time it asks for a user’s information. Instead, Content Server remembers where it successfully obtained the information the last time and uses that LDAP provider. However, if the backup LDAP provider fails, Content Server will check the primary LDAP provider for the user information, find it, and will continue to go to the primary LDAP provider from then on.

LDAP PROVIDER PAGE



Add LDAP Provider

Provider Name	
Provider Description	
Provider Class	intradoc.provider.LdapUserProvider
Connection Class	intradoc.provider.LdapConnection
Configuration Class	
Source Path	
LDAP Server	
LDAP Suffix	
LDAP Port	389
Number of connections	5
Connection timeout	10
Priority	1
Use Netscape SDK	☑
Use SSL	□
Use Group Filtering	□
Use Full Group Names	□

Role Prefix	Depth
OU=Roles,OU=Stellent	1

Attribute Map

LDAP Attribute		User Attribute
	Maps To	dFullName

LDAP Admin DN	
LDAP Admin Password	

The LDAP Provider page is used to create an LDAP provider and configure Content Server integration with LDAP security. To access this page, click **Add** or **Edit** for the *ldapuser* provider type on the Providers page.

In the following tables, the term in parentheses in the first column is the corresponding configuration setting in the <*Install_Dir*>/data/providers/*provider_name*/provider.hda file.

Feature	Description
Provider Name field*	The name of the provider, which will become a subdirectory in the *<Install_Dir>*/data/providers/ directory.
Provider Description field*	A user-friendly description of the provider.
Provider Class field* (ProviderClass)	The name of the Java class that implements the provider. • Default is *intradoc.provider.LdapUserProvider*. • If the provider is communicating with Active Directory, this class should be *ldap.ActiveDirectoryLdapProvider*.
Connection Class field (ProviderConnection)	The name of the Java class that implements the connection to the LDAP server. Default is *intradoc.provider.LdapConnection*.
Configuration Class field (ProviderConfig)	The name of a Java class that performs some extra configuration. This class is useful for database providers, where the connection classes are already providers.
Source Path field* (SourcePath)	A unique string that identifies the LDAP provider. The first time a user requests credentials through the provider, this string is stored with the user information so it can be used to match the user with the provider next time the user asks for credentials. We suggest using the name of the provider as the Source Path.
Ldap Server field* (LdapServer)	Host name of the LDAP server. If the provider is communicating with Active Directory, this should be the host name of a primary domain controller.
Ldap Suffix field* (LdapSuffix)	The root suffix (naming context) to use for all LDAP operations (such as *o=company.com* or *dc=company,dc=com*). All mapping of LDAP groups to Content Server roles and accounts will begin at this root. **Note:** Do not include spaces before or after commas.
Ldap Port field* (LdapPort)	The port the LDAP server listens on. The default is 389. If you are using SSL, you should set this to 636.

Feature	Description
Number of connections field (NumConnections)	The number of LDAP server connections the provider maintains.
Connection timeout field	The amount of time (in minutes) that a provider connection to the LDAP server is held open before the provider connection is closed and reopened. **Note:** For best results, set the amount of time to less than 15 minutes. If the amount of time is 15 minutes or greater, there could be a problem with the JNDI layer not holding the connection open.
Priority field (Priority)	Specifies the order in which LDAP providers will be checked for the user credentials. • This field is used only when a user has not previously logged into Content Server. If the user has previously requested credentials, the Source Path will be stored for that user, so the LDAP provider specified by the Source Path will be used. • Each LDAP provider in a Content Server instance must have a unique Priority number.
Using Netscape SDK check box (UseNetscape)	There are two ways to connect to the LDAP server: using the Netscape SDK or using JNDI. This check box acts as a toggle between the two options. **Selected** = Use Netscape SDK **Clear** = Use JNDI
Use SSL check box (UseSecureLdap)	If you select this check box, you must have the appropriate certificates installed on the LDAP server. Once SSL is initiated, the certificates will secure communication between the LDAP server and the content server.

Feature	Description
Use Group Filtering Section See Mapping Roles and Accounts (page 7-5) for more information.	
Use Group Filtering check box (UseGroupFilter)	**Selected—**The Role Prefix and Account Prefix definitions will be used to select the LDAP groups that will be mapped to Content Server roles and accounts. **Clear—**All LDAP groups will be mapped to Content Server roles and accounts. This is the default.
Use Full Group Names check box (UseFullGroupName)	**Selected—**The entire hierarchy (up to the specified prefix or naming context) for an LDAP group will be included in the mapping to a Content Server role or account. **Clear—**Only the lowest level unit of an LDAP group will be mapped to a Content Server role or account. This is the default.
Account Permissions Delimiter field (AcctPermDelim)	The string that separates the account names from the account permissions in an LDAP group name. • If an LDAP group name is mapped to an account and contains this substring, the string to the left of this substring will be the account name, and the string to the right of this substring will be the account permissions. • For example, if the delimiter is defined as a + (plus sign), the group name *Acct1+rw* would map to an account named *Acct1* with Read and Write permission. If the delimiter is defined as an _ (underscore), the *Acct1+rw* group name would map to an account named *Acct1+rw*, with RWDA permission by default. • The default is _ (underscore). • This field appears only if accounts are enabled in Content Server.
Default Network Roles field	The default role or roles assigned to a user who enters through this provider; for example, *contributor*.

Feature	Description
Default Network Accounts field (DefaultNetworkAccounts)	Defines the default account permissions for users who log in to Content Server with LDAP credentials. • This must be a comma-separated list of accounts. **Note:** Do not include spaces before or after the commas that separate accounts. • Permissions for each account can be specified in parentheses after the account name, such as *account(RWDA)*. If no permissions are specified, RWDA permission is granted by default. • The *#none* entry grants permission to documents that have no account assigned. • The *#all* entry grants permissions to all accounts. • The default is *#none(RWDA)*. • This setting does not apply to anonymous users. • This setting defines the minimum account permissions. Account permissions defined by the external user base are added to these permissions. For example, if the default is *#none(RW)*,*Project(R)*, and a user's group maps to *Project(RWD)* permission, the user's permissions are *#none(RW)*,*Project(RWD)*. • This field appears only if accounts are enabled in the Content Server.
Role Prefix field	The string that specifies where in the LDAP group name to start matching a Content Server role name.
Role Prefix Depth field	A number that specifies how many levels the LDAP group name can contain after the Role Prefix for the group name to be considered a valid role. Placing an asterisk (*) in the depth parameter for a specific prefix ensures that the short name for any group mapped through the prefix is used.
Role Prefix Add button	Adds the Role Prefix string and Depth as a clause in the Role Prefix box.

Feature	Description
Role Prefix box (RolePrefix)	Lists the Role Prefix clauses that will be used to select LDAP groups when the Group Filtering check box is selected. This box can be edited directly. **Note:** Do not include spaces before or after the commas that separate units in a prefix.
Account Prefix field	The string that specifies where in the LDAP group name to start matching a Content Server account name. This field appears only if accounts are enabled in Content Server.
Account Prefix Depth field	A number that specifies how many levels the LDAP group name can contain after the Account Prefix for the group name to be considered a valid account. This field appears only if accounts are enabled in the Content Server. Placing an asterisk (*) in the depth parameter for a specific prefix ensures that the short name for any group mapped through the prefix is used.
Account Prefix Add button	Adds the Account Prefix string and Depth as a clause in the Account Prefix box. This button appears only if accounts are enabled in Content Server.
Account Prefix box (AcctPrefix)	Lists the Account Prefix clauses that will be used to select LDAP groups when the Group Filtering check box is selected. This box can be edited directly. **Note:** Do not include spaces before or after the commas that separate units in a prefix. This box appears only if accounts are enabled in Content Server.
Attribute Map Section	
LDAP Attribute field	Enter an LDAP user attribute to be mapped to a Content Server user information field.

Feature	Description
User Attribute field	Select a Content Server user information field to be mapped from the LDAP Attribute field. • All Content Server user information fields for which you can change the value are listed. • Standard user information fields begin with a “d”. • Custom user information fields begin with a “u”.
User Attribute Add button	Adds the LDAP Attribute and User Attribute as a colon-separated clause in the Attribute Map box.

Attribute Map box
(AttributeMap)
	Lists the Attribute Map clauses that will be used to map LDAP user attributes to Content Server information fields.
	• This box can be edited directly.
	• If this field is left blank, the default is:
	`mail:dEmail`
	`cn:dFullName`
	`title:dUserType`

Account Permissions Delimiter Section See Mapping Roles and Accounts (page 7-5) for more information.	
Default Network Accounts Section	
LDAP Admin DN field (LdapAdminDN)	The user name that will be making calls to the LDAP server. • This user must have Read rights to the LDAP server. • If the user name is left blank, the provider will connect to the LDAP server anonymously. • If the provider is communicating with Active Directory, a user name and password is required, and must be a valid domain user in the format *DOMAIN\username*.

Feature	Description
Ldap Admin Password field (LdapAdminPassword)	The password for the user that will be making calls to the LDAP server.
Add/Update button	Saves the provider information.
Reset button	Resets the provider information to the last saved values.

Chapter

8

PROXIED CONNECTIONS

OVERVIEW

Proxied connections are an optional component which can be installed and enabled when you install Content Server. The information in this chapter is only relevant if that component has been installed and enabled.

This section covers the following topics:

Concepts

Tasks

- Creating a Credentials Map (page 8-6)
- Creating a Proxied Connection (page 8-9)
- Configuring the HTTP Provider (page 8-11)

Interface

- Credential Maps Screen (page 8-13)
- Proxied Connections Screen (page 8-14)
- Edit Outgoing Http Provider Page (page 8-16)

ABOUT PROXIED CONNECTIONS

Proxied connections provide additional levels of security for Content Server through the following functions:

- Security credentials mapping from one content server to another content server.
- Secured "named" password connections to content servers (password protected provider connections).
- HTTP protocol communication between content servers (HTTP-based proxy servers).

While it is possible to use both named password connections and HTTP-based proxy server communication, it is most likely that one type of connection will be more useful. For both types of connections, credentials mapping can provide additional security.

Typical uses of the proxied connections include the following:

- To provide the capability to perform archive replication of content items. For example, a company has acquired another company, but they do not a have common infrastructure for sharing information. Both companies have a Secure Sockets Layer (SSL) connection to the Internet. The company wishes to share content between the two sites. Proxied connections can be used to set up a secure Internet connection between the companies' servers so that content can be securely accessed from one site, replicated, and archived at the other site.
- To better restrict access to content servers by using "named" passwords to target master server proxied connections. For example, a company that uses Active Directory Service Interfaces (ADSI) for authentication on their content server wishes to apply additional security to connections coming from a partner's server to the

company’s content server. Using “named” passwords, an administrator can restrict access by incoming connections to those with preset proxied connections and named passwords.

- To support Enterprise search capabilities. For example, a company may wish certain users to be able to perform Enterprise searches on one or more content servers. The administrator can set up credentials mapping so that certain users or roles or accounts are allowed controlled access to specified content servers.

Credentials Mapping

Administrators can create multiple credentials maps for users, roles, and accounts. Credentials mapping can be useful in a master-to-master proxy scenario, for example, where credentials for users, roles, or accounts created on a content server can be mapped to the users, roles, or accounts on another content server, thus allowing users controlled access to information on a master or proxy content server for tasks such as Enterprise searching.

This section covers the following topics:

- About Credentials Mapping (page 8-3)
- Credential Values (page 8-4)
- Matching Accounts and Roles (page 8-5)
- Creating a Credentials Map (page 8-6)

About Credentials Mapping

When you create a credentials map you enter a unique identifier for the map and specific credential values for users, roles, and accounts. In a proxied connection, when user credentials match an input value, then the user is granted the credentials specified in the output value. The user credentials are evaluated in the following order:

1. All the roles.
2. All the accounts.
3. The user name.

After the translation is performed, the user only has the attribute values that were successfully mapped from input values.

When you have created credential maps, you can specify a credentials map along with a named password connection when configuring an outgoing provider. You also can specify a credentials map when configuring a user provider (such as LDAP).

Note: The default behavior for an LDAP provider is that the guest role is not automatically assigned to users. However, if a user logs on to a proxied Content Server with a master server's credentials, the user will automatically pick up the guest role.

Note: Credentials mapping implementation is duplicated in the web server plug-in and in Content Server. It is designed and implemented for optimal performance, so that any changes in the mapping are applied immediately. (This can be compared to performance in NT or ADSI user storage using the NT administrator interfaces, where changes are cached and not reflected in the content server for up to a couple of minutes.)

Credential Values

A credential input value is matched if there is an exact match in the case of a role or user name. An input account value is matched if one of the user accounts has a prefix, except for the case of a filter (see Matching Accounts and Roles (page 8-5)). For example, the following credential values reduce all users who might otherwise have the admin role to instead have the guest role:

```
admin, guest
```

The following table lists the basic syntax for credential values:

Value	Prefix or Sequence	Example
User name	&	`&name`
Role		`admin`
Account	@	`@marketing`
Empty account	@#none	`@#none`
All accounts	@#all	`@#all`
Ignore the value or “comment out” the value	#	`#comment`

You can view which credentials are applied by default if no credential map is assigned. Use the following mapping, which maps everything without change. This mapping first filters all roles, then all accounts. For additional information about mapping syntax see Matching Accounts and Roles (page 8-5).

```
|#all|,%%
@|#all|,@%%
```

Caution: If your credentials map does not at least assign the minimum set of privileges that an anonymous user gets when visiting the content server website, then logged in users may experience unusual behavior. For example, a common reaction for a browser that receives an ACCESS DENIED response is to revert back to being an anonymous user. In particular, a user may experience unpredictable moments when it is possible or not possible to access a document (depending on whether at that moment the browser chooses to send or not to send the user's authentication credentials). This is particularly true of NTLM authentication because that authentication has to be renewed periodically.

Matching Accounts and Roles

A special filter is available for matching accounts and roles. For example, the syntax for an account filter is designated by starting the account value with specifying the prefix @| and ending with a | (for example, `@|accountname|`). The pipe (|) represents a command redirection operator that processes values through the filter. For proxied connections a space-separated list of accounts is specified; each account optionally starts with a dash (-) to denote a negative value. A filter is matched if any of the specified account strings that do not start with a dash are a prefix for a user account and all of the account strings that do start with a dash are not prefixes for that user account.

Caution: The filter will not map the account `@#all`. The `all accounts` account value must be mapped explicitly by using `@#all, @#all` mapping.

Roles can be mapped (using the same rules) by removing the @ sign from the beginning of the filter. For example, the following input value passes through all roles except those that begin with the prefix `visitor`. Note that the expression `#all` matches all roles.

```
|#all -visitor|, %%
```

Reference Input Value

The special sequence %% in the output value can be used to reference the input value. For example, given the following mapping, any account that did not start with `financial` as a prefix would map to the same account but with the prefix `employee/` attached at the front:

```
@|#all -financial|, @employee/%%
```

If a user had the account `marketing`, then after the mapping the user would have the account `employee/marketing`.

Privilege Levels

A particular privilege level (read, write, delete, all) can be granted to an account in the output value by following the account specification with the letters “R”, “W”, “D”, or “A” enclosed in parentheses. For example, all the privilege levels for all the accounts could be reduced to having read privilege by the following syntax:

```
@|#all -financial|, @employee/%%(R)
```

Substitution

In certain cases it is useful to remove a prefix before the substitution %% is applied. An offset for the substitution can be specified by using the syntax %%[n] where `n` is the starting offset to use before mapping the input value into the %% expression. The offset is zero based so that %%[1] removes the first character from the input value. For example, to remove the prefix `DOMAIN1\` from all roles, the following expression can be used:

```
|domain1\|, %%[8]
```

Another use for this function might be to replace all accounts that begin with the prefix `marketing/` and replace it with the prefix `org1/mkt`. The expression for this would look like the following:

```
@|marketing/|, @org1/mkt/%%[10]
```

Special Characters

In certain cases roles will have unusual characters that may be hard to specify in the input values. The escape sequence %*xx* (where *xx* is the ASCII hex value) can be used to specify characters in the input value. For example, to pass through all roles that begin with `#,& |@` (hash, comma, ampersand, space, pipe, at) the following expression can be used:

```
|%35%2c%26%20%7c%40|, %%
```

Creating a Credentials Map

To create a credentials map, follow these steps:

1. Open a new browser window and log in to Content Server as the system administrator.
2. Select **Administration—Credential Maps**.

   The Credential Maps Screen (page 8-13) is displayed.
3. Enter the unique identifier for the credentials map you are creating.

Important: More than one named password connection can be used to connect to a content server. Each named password connection can have a different credentials map.

4. Enter values in two columns with a comma to separate the columns and a carriage return between each row of values. The first column specifies input values and the second column specifies output values.
5. Click **Update.**

Important: To apply a credentials map to roles and accounts retrieved using NT integration, set the Content Server configuration entry ExternalCredentialsMap to the name of the credentials map of your choice.

SECURED CONNECTIONS TO CONTENT SERVERS

Secured connections to content servers can be supported by creating password protection on incoming requests. A content server can act as proxy for another content server in a password protected fashion.

This section covers the following topics:

About Named Password Connections

Using the Proxied Connections page you can create “named” passwords, which are passwords that you assign to specific proxied connections by name. Each named password can be associated with a host and IP address filter on both the direct socket communication to a content server and on any communication performed through the controlling web server (the HTTP filter) for a content server. When an outside agent (such as a web server for another content server) wants to communicate with the master content server, it can use a named password connection. A named password connection also can be associated with a credentials map so that the privileges of users accessing the content server can be reduced or changed.

Proxied connections entry fields are provided in the forms for configuring outgoing socket providers and outgoing HTTP providers in which you can specify a named password connection. (To view provider selections for your instance, select **Administration—Providers**.)

Passwords are hashed (SHA1 message digest) with their allowed host and IP address wildcard filter on the client side. This means that if the copy of a stored password is exposed, it will only allow access from clients that satisfy both the host and IP address filter.

Caution: All passwords are hashed by a time-out value before being sent to a server. This means that if a password value is exposed while in communication to a server, the password will only be usable until the expiration time (approximately fifteen minutes after the time the request is issued). Also, the password will only be usable in a replay attack from the same source host and IP address, as previously described. If firewall-protected internal host and IP addresses are not being used, a very committed attacker could spoof the host and IP addresses by hijacking any of the major DNS servers—an event that has occurred in at least a couple of cases.

Important: The expiration implementation for passwords means that the various servers involved must have their clocks reasonably synchronized (within a few minutes at least).

Guidelines for Proxied Connections Data

The data you enter in the Proxied Connections page defines different passwords that can be used by external agents to connect to a content server. Instead of an external agent being forced to provide a password for each user, which may be unavailable to the client for many reasons (such as message digest algorithms that do not use clear text passwords), proxied connections enable the agent to authenticate using a single named connection password. Each named password connection can be linked to rules to restrict which hosts can connect to the master content server and to control the privileges granted to users. Each named password connection is uniquely identified, and the calling agent must supply the identifier along with the password.

The host name and IP address filters are used to determine which host names or IP addresses are allowed to use a named password connection when performing direct socket connections to a content server. The rules for defining the filters are identical to those defined in the System Properties editor (the wild card symbols * = *match 0 or many* and | = *match either or* can be used to create flexible rules). If an entry is empty then it provides no restriction on its target attribute (either the host name or IP address of the client depending on which of the following two fields is involved).

The HTTP IP address filter must be specified for a content server to proxy another content server through its web server. This filter is applied to the IP address of the client server, and if the master server is satisfied then the communication is allowed to continue.

Two options are implemented through the Providers page:

- ❖ Whenever you add an outgoing provider you have the option to use named password connections and to choose whether the provider is a proxied server (so that web access and security is controlled through a remote server).
- ❖ Whenever you add a user provider (such as LDAP) you can choose to use an available credentials map.

No credentials maps are defined in the Proxied Connections page. For information on creating a credentials map, see Credentials Mapping (page 8-3).

Creating a Proxied Connection

To create a proxied connection, follow these steps:

1. Open a new browser window and log in to Content Server as the system administrator.
2. Click **Administration**.
3. Click **Connection Passwords**.

   The Proxied Connections Screen (page 8-14) is displayed.
4. Enter information for the fields in the Proxied Connections page.

   If credentials maps exist, you can choose to use an existing credentials map, or you can create one to be used for the proxied connection.
5. Click **Update**.

Content Server Proxy Using the HTTP Protocol

Administrators can create a proxy content server using the HTTP protocol. For example, you could have two content servers where both have web servers for accessing their functionality. If you have a large number of users who wish to use browsers to access information on one of the content servers, but not all the users can access that server directly, this feature can be useful. If you set up a proxy content server, users could access the master content server and through it also access information on the proxy content server.

The HTTP protocol also can be useful for transferring archives. The HTTP provider works with Secure Sockets Layer (SSL), the HTTPS protocol, which enables secure communication between two content servers.

This section covers the following topics:

- About Using HTTP Protocol for Content Server Proxy (page 8-10)
- Configuring the HTTP Provider (page 8-11)
- Edit Outgoing Http Provider Page (page 8-16)

About Using HTTP Protocol for Content Server Proxy

Administrators can implement an *httpoutgoing* provider, configurable through the Providers page, which allows communication from one Content Server (the master) to the other Content Server (the proxy). All static URL requests to <*Weblayout_Dir*>/groups/ in the Content Server are forwarded to another web server, creating a proxy at the web server level as well as the proxy at the Content Server level.

If you choose to add an httpoutgoing HTTP provider, you have the following additional options:

- Specify a CGI URL.
- Specify a named password connection and client IP filter.
- Select whether the provider is a proxy server (so that web access and security is controlled through a remote server).

The HTTP provider used in this case is very similar to the regular outgoing provider in how you configure proxying versus proxied, and how Enterprise Search is configured.

To view the httpoutgoing HTTP provider selection, select **Administration—Providers** from the Content Server navigation panel.

Note: If you want to proxy a master content server to a master content server, then you should choose a common value for IdcRealm (formerly IntradocRealm) and put the same value into the config.cfg file of each server. The realm is the value above the user name and password when a user is challenged for a login by the browser. If you do not set up a common value for IdcRealm, then users may have to log in again when they start switching from content delivered by one master server to content delivered by another master server.

Caution: Creating proxy relationships between two content servers can take some preparation. The master server should not use the same relative web root for its weblayout directory as the proxy server. It may require some component architecture changes to provide the extra navigation links to the proxy server. If the proxy server's relative URL is *proxied1*, then the path to the dynamically rendered version of the home page (viewed from the master server) in the following example would be:

```
http://<host_name>/idcm1/idcplg/proxied1/pxs?IdcService=GET_DOC_PAGE&
Action=GetTemplatePage&Page=HOME_PAGE
```

The */proxied1/pxs* suffix construction tells the web server filter that this request is destined for the proxy server.

Note: If you set up a proxy content server with its web server using SSL and the master server's front end uses HTTP, then users who try to access the proxy server by modifying the master server's URL in a browser can get an error because of the differences between HTTPS (requiring a credential) and HTTP. To resolve this issue use the BrowserUrlPath component, available with Content Server. For more information see the *Oracle Content Server Installation Guide* for your operating system and the component readme.txt file.

Configuring the HTTP Provider

To configure the proxy HTTP provider, complete the following steps:

Note: If you are setting up a proxy HTTP provider only for transferring archives, then you do not need to create a stub weblayout directory. Start your configuration with Step 8.

1. On the master content server add an http-web-stubs directory under the Content Server install. For example, C:\stellent\http-web-stubs.

2. Create a directory beneath the http-web-stubs directory with the same name as the relative web root of the proxy content server. For example, C:\stellent\http-web-stubs\stellent.
3. Copy and paste the proxy server's weblayout directory (except for the groups subdirectory) under the C:\stellent\http-web-stubs directory on the master server.
4. In the master server's web server, add a virtual directory. The name of the virtual directory must be the same as the proxy server's relative web root. Point the virtual directory to the weblayout directory under the http-web-stubs directory.

Important: The location of this copied content can be put anywhere and does not have to go under the weblayout directory of the master server as long as the web server is set up to map the appropriate relative URL to that directory.

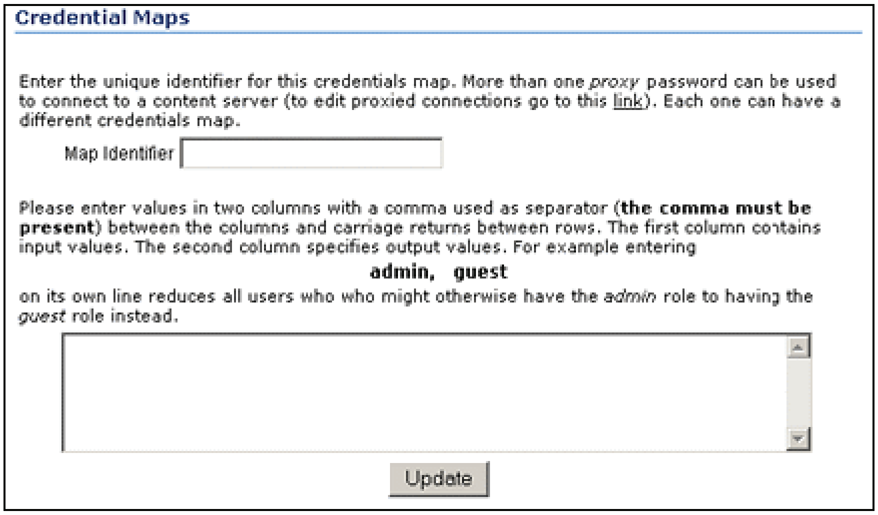
5. Restart the master Content Server.
6. Restart the master web server.
7. If you need Enterprise search capability from the proxy server to the master server, repeat steps 1 through 6 for the proxy server.
8. Add an httpoutgoing provider on the master Content Server. If you need Enterprise Search capability from the proxy server to the master server, repeat this step for the proxy server.
   a. In a browser, go to the Administration page and click **Providers**.
   b. Click **Add** next to the httpoutgoing provider type.
   c. Enter the necessary information for the httpoutgoing provider. For more information see the table in Edit Outgoing Http Provider Page (page 8-16):
      ❖ For Server Options, select **Proxied**.
      ❖ For Enterprise Search, select **Enterprise Searchable**.
9. Create a proxied connection on the proxy server that uses the named password connection and connection password that you specified in the previous step.
   a. On the proxy server select **Administration—Connection Passwords**.
   b. Fill in the information for a proxied connection. The IP address filter entry should have the IP address of the master server.

PROXIED CONNECTION INTERFACE SCREENS

The following screens are used when creating proxied connections:

- Credential Maps Screen (page 8-13)
- Proxied Connections Screen (page 8-14)
- Edit Outgoing Http Provider Page (page 8-16)

Credential Maps Screen



Credential Maps

Enter the unique identifier for this credentials map. More than one *proxy* password can be used to connect to a content server (to edit proxied connections go to this link). Each one can have a different credentials map.

Map Identifier

Please enter values in two columns with a comma used as separator (**the comma must be present**) between the columns and carriage returns between rows. The first column contains input values. The second column specifies output values. For example entering

admin, guest

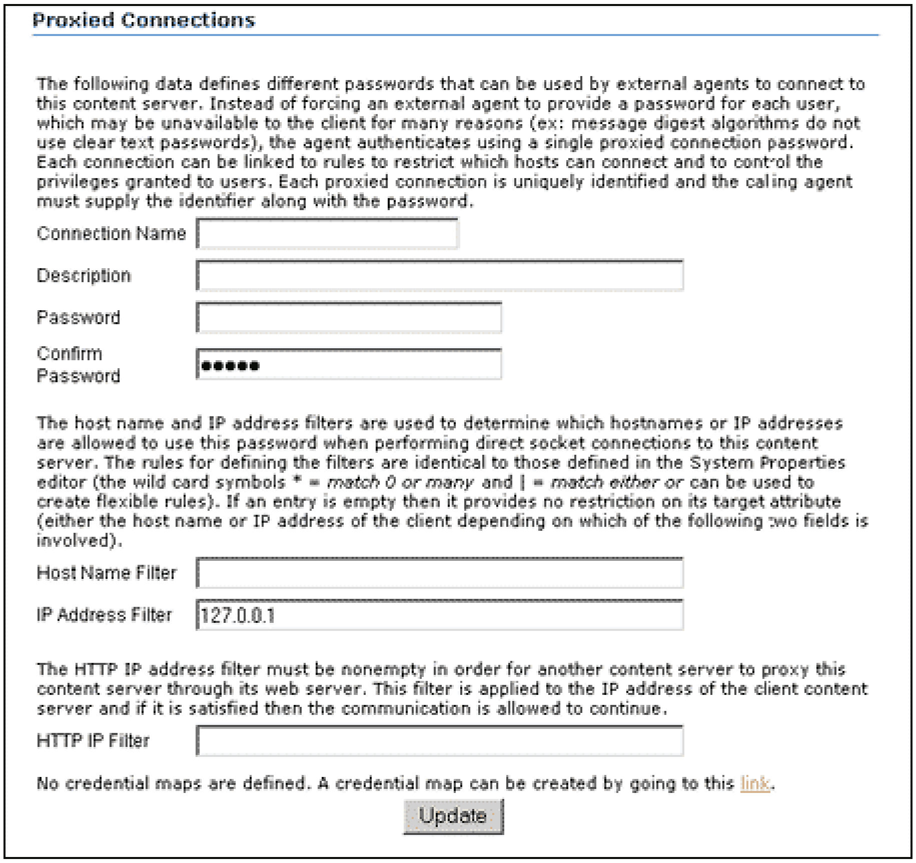
on its own line reduces all users who who might otherwise have the *admin* role to having the *guest* role instead.

This screen enables administrators to create credentials for specific users that can be mapped to allow users controlled access between a master content server and a proxy content server. To access the page select **Administration—Credential Maps** from the Content Server navigation panel.

Feature	**Description**
Map Identifier field	Enter the unique identifier for the credentials map.
Values field	Enter the credential values in two columns with a comma used as a separator between the columns, and a carriage return between rows. The first column specifies input values. The second column specifies output values.

Feature	Description
Update button	Inputs the credential values specified in the Credential Maps page.

Proxied Connections Screen



Proxied Connections

The following data defines different passwords that can be used by external agents to connect to this content server. Instead of forcing an external agent to provide a password for each user, which may be unavailable to the client for many reasons (ex: message digest algorithms do not use clear text passwords), the agent authenticates using a single proxied connection password. Each connection can be linked to rules to restrict which hosts can connect and to cont'ol the privileges granted to users. Each proxied connection is uniquely identified and the cal ing agent must supply the identifier along with the password.

Connection Name	
Description	
Password	
Confirm Password	●●●●●

The host name and IP address filters are used to determine which hostnames or IP addresses are allowed to use this password when performing direct socket connections to this content server. The rules for defining the filters are identical to those defined in the System Properties editor (the wild card symbols * = *match 0 or many* and | = *match either or* can be used to create flexible rules). If an entry is empty then it provides no restriction on its target attribute (either the host name or IP address of the client depending on which of the following :wo fields is involved).

Host Name Filter	
IP Address Filter	127.0.0.1

The HTTP IP address filter must be nonempty in order for another content server to proxy this content server through its web server. This filter is applied to the IP address of the client content server and if it is satisfied then the communication is allowed to continue.

HTTP IP Filter	

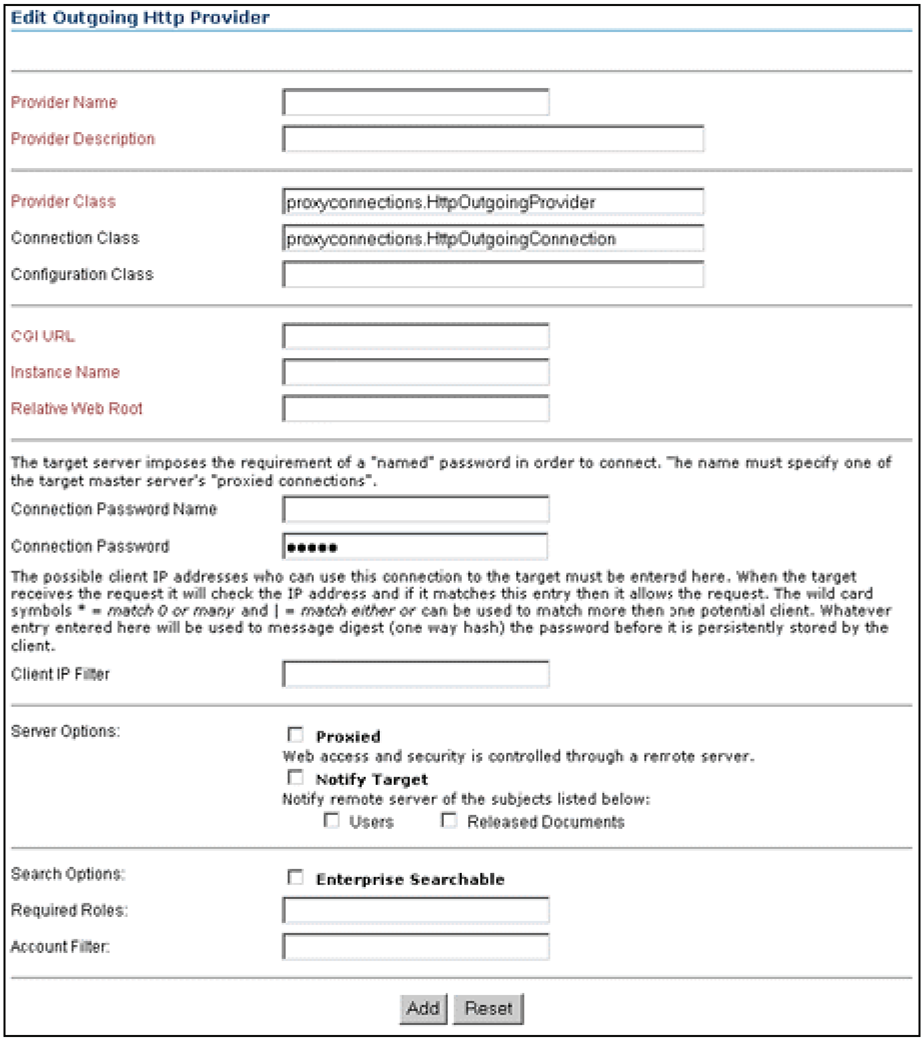
No credential maps are defined. A credential map can be created by going to this link.

This screen enables administrators to create *named passwords*, which are passwords that are assigned to specific proxied connections by name. To access the page select **Administration—Connection Passwords** from the Content Server navigation panel.

Feature	Description
Connection Name field	Name given to the proxied connection.
Description field	Brief description of the proxied connection.

Feature	Description
Password field	Password assigned to the proxied connection.
Confirm Password field	Password assigned to the proxied connection.
Host Name Filter field	Host name that can use the password when performing a direct socket connection to the master server.
IP Address Filter field	IP address number of the client content server.
HTTP IP Filter field	HTTP IP address filter, applied to the IP address of the client content server.

Edit Outgoing Http Provider Page



Edit Outgoing Http Provider

Provider Name	
Provider Description	
Provider Class	proxyconnections.HttpOutgoingProvider
Connection Class	proxyconnections.HttpOutgoingConnection
Configuration Class	
CGI URL	
Instance Name	
Relative Web Root	

The target server imposes the requirement of a "named" password in order to connect. "he name must specify one of the target master server's "proxied connections".

Connection Password Name	
Connection Password	●●●●●

The possible client IP addresses who can use this connection to the target must be entered here. When the target receives the request it will check the IP address and if it matches this entry then it allows the request. The wild card symbols * = *match 0 or many* and | = *match either or* can be used to match more then ɔne potential client. Whatever entry entered here will be used to message digest (one way hash) the password before it is persistently stored by the client.

Client IP Filter	
Server Options:	□ **Proxied** Web access and security is controlled through a rerrote server. □ **Notify Target** Notify remote server of the subjects listed below: □ Users □ Released Documents
Search Options:	□ **Enterprise Searchable**
Required Roles:	
Account Filter:	

This screen enables an administrator to add an httpoutgoing provider on the master Content Server. To access this page, follow these steps:

1. Select **Administration—Providers** from the Content Server navigation panel.

   The Providers page is displayed

2. Select **Add** in the Action column for the httpoutgoing provider under **Create a New Provider**.

Feature	Description
Provider Name field*	The name of the provider.
Provider Description field*	A user-friendly description of the provider.
Provider Class field*	The name of the Java class for the provider. For example: *proxyconnections.HttpOutgoingProvider*
Connection Class field	The name of the Java class that implements the provider connection. For example: *proxyconnections.HttpOutgoingConnection*
Configuration Class field	The name of a Java class that performs some extra configuration. Leave this blank.
CGI URL field*	The URL for the proxy server.
Instance Name field*	The instance name of the proxy content server.
Relative Web Root field*	The relative web root of the content server instance.
Connection Password Name field	The name of a password connection (this can be an existing name or a name for a password connection that you will create on the proxy server). The name must specify one of the target master server's proxied connections. The target server requires a named password.
Connection Password field	The password for the named password connection.
Client IP Filter field	The client IP address or addresses that can use this connection to the target server.
Proxied check box	Enable this option if the provider is connecting to a content server that will be controlled by the current instance.

Feature	Description
Notified Target check box	Enable this option if the provider is connecting to a content server that is acting as a controlling instance, and you want this content server to notify the controlling instance when user information and/or content item information changes.
Users check box	Enable this option if you want this content server to notify the controlling instance when user information changes.
Released Documents check box	Enable this option if you want this content server to notify the controlling instance when content item information changes.
Enterprise Searchable check box	Enable this option if you have enabled Enterprise Search and you want this content server instance to be searchable. See the *Stellent Enterprise Search Administration and User Guide* for more information.
Required Roles field	Enter roles that have permission to search this content server instance using Enterprise Search. If no roles are entered, all users will have permission.
Account Filter field	Enter accounts that have permission to search this content server instance using Enterprise Search. If no accounts are entered, all users will have permission.
Add button	Saves the provider information.
Reset button	Resets the provider information to the last saved values.

* Required metadata fields.

Appendix

A

THIRD PARTY LICENSES

OVERVIEW

This appendix includes a description of the Third Party Licenses for all the third party products included with this product.

- Apache Software License (page A-1)
- W3C® Software Notice and License (page A-2)
- Zlib License (page A-4)
- General BSD License (page A-5)
- General MIT License (page A-5)
- Unicode License (page A-6)
- Miscellaneous Attributions (page A-7)

APACHE SOFTWARE LICENSE

```
* Copyright 1999-2004 The Apache Software Foundation.

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*      http://www.apache.org/licenses/LICENSE-2.0

*
```

```
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
```

W3C® SOFTWARE NOTICE AND LICENSE

```
* Copyright © 1994-2000 World Wide Web Consortium,
* (Massachusetts Institute of Technology, Institut National de
* Recherche en Informatique et en Automatique, Keio University).
* All Rights Reserved.  http://www.w3.org/Consortium/Legal/
*
* This W3C work (including software, documents, or other related items) is
* being provided by the copyright holders under the following license. By
* obtaining, using and/or copying this work, you (the licensee) agree that
* you have read, understood, and will comply with the following terms and
* conditions:
*
* Permission to use, copy, modify, and distribute this software and its
* documentation, with or without modification, for any purpose and without
* fee or royalty is hereby granted, provided that you include the following
* on ALL copies of the software and documentation or portions thereof,
* including modifications, that you make:
*
*   1. The full text of this NOTICE in a location viewable to users of the
*      redistributed or derivative work.
*
*   2. Any pre-existing intellectual property disclaimers, notices, or terms
*      and conditions. If none exist, a short notice of the following form
*      (hypertext is preferred, text is permitted) should be used within the
*      body of any redistributed or derivative code: "Copyright ©
*      [$date-of-software] World Wide Web Consortium, (Massachusetts
*      Institute of Technology, Institut National de Recherche en
*      Informatique et en Automatique, Keio University). All Rights
*      Reserved. http://www.w3.org/Consortium/Legal/"
*
*   3. Notice of any changes or modifications to the W3C files, including the
*      date changes were made. (We recommend you provide URIs to the location
*      from which the code is derived.)
*
* THIS SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS
* MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT
* NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR
* PURPOSE OR THAT THE USE OF THE SOFTWARE OR DOCUMENTATION WILL NOT INFRINGE
* ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.
*
* COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR
* CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR
* DOCUMENTATION.
*
* The name and trademarks of copyright holders may NOT be used in advertising
* or publicity pertaining to the software without specific, written prior
* permission. Title to copyright in this software and any associated
* documentation will at all times remain with copyright holders.
*
```

ZLIB LICENSE

* zlib.h -- interface of the 'zlib' general purpose compression library version 1.2.3, July 18th, 2005

Copyright (C) 1995-2005 Jean-loup Gailly and Mark Adler

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly jloup@gzip.org

Mark Adler madler@alumni.caltech.edu

GENERAL BSD LICENSE

Copyright (c) 1998, Regents of the University of California

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

"Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

"Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

"Neither the name of the <ORGANIZATION> nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

GENERAL MIT LICENSE

Copyright (c) 1998, Regents of the Massachusetts Institute of Technology

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

UNICODE LICENSE

UNICODE, INC. LICENSE AGREEMENT - DATA FILES AND SOFTWARE

Unicode Data Files include all data files under the directories http://www.unicode.org/Public/, http://www.unicode.org/reports/, and http://www.unicode.org/cldr/data/ . Unicode Software includes any source code published in the Unicode Standard or under the directories http://www.unicode.org/Public/, http://www.unicode.org/reports/, and http://www.unicode.org/cldr/data/.

NOTICE TO USER: Carefully read the following legal agreement. BY DOWNLOADING, INSTALLING, COPYING OR OTHERWISE USING UNICODE INC.'S DATA FILES ("DATA FILES"), AND/OR SOFTWARE ("SOFTWARE"), YOU UNEQUIVOCALLY ACCEPT, AND AGREE TO BE BOUND BY, ALL OF THE TERMS AND CONDITIONS OF THIS AGREEMENT. IF YOU DO NOT AGREE, DO NOT DOWNLOAD, INSTALL, COPY, DISTRIBUTE OR USE THE DATA FILES OR SOFTWARE.

COPYRIGHT AND PERMISSION NOTICE

MISCELLANEOUS ATTRIBUTIONS

Adobe, Acrobat, and the Acrobat Logo are registered trademarks of Adobe Systems Incorporated.

FAST Instream is a trademark of Fast Search and Transfer ASA.

HP-UX is a registered trademark of Hewlett-Packard Company.

IBM, Informix, and DB2 are registered trademarks of IBM Corporation.

Jaws PDF Library is a registered trademark of Global Graphics Software Ltd.

Kofax is a registered trademark, and Ascent and Ascent Capture are trademarks of Kofax Image Products.

Linux is a registered trademark of Linus Torvalds.

Mac is a registered trademark, and Safari is a trademark of Apple Computer, Inc.

Microsoft, Windows, and Internet Explorer are registered trademarks of Microsoft Corporation.

MrSID is property of LizardTech, Inc. It is protected by U.S. Patent No. 5,710,835. Foreign Patents Pending.

Oracle is a registered trademark of Oracle Corporation.

Portions Copyright © 1994-1997 LEAD Technologies, Inc. All rights reserved.

Portions Copyright © 1990-1998 Handmade Software, Inc. All rights reserved.

Portions Copyright © 1988, 1997 Aladdin Enterprises. All rights reserved.

Index

I