Novell iChain™

1.5

www.novell.com

INSTALLATION AND SETUP

Novell

Legal Notices

Novell, Inc.
1800 South Novell Place
Provo, UT 84606
U.S.A.

www.novell.com

iChain Installation and Setup
January 2001

Online Documentation: To access the online documentation for this and other Novell products, and to get updates, see www.novell.com/documentation.

Novell Trademarks

ConsoleOne is a trademark of Novell, Inc.
DirXML is a trademark of Novell, Inc.
eDirectory is a trademark of Novell, Inc.
iChain is a trademark of Novell, Inc.
NDS is a registered trademark of Novell, Inc., in the United States and other countries.
NetWare is a registered trademark of Novell, Inc., in the United States and other countries.
NLM is a trademark of Novell, Inc.
Novell is a registered trademark of Novell, Inc., in the United States and other countries.
Novell Client is a trademark of Novell, Inc.
Novell Internet Caching System is a trademark of Novell, Inc.

Third-Party Trademarks

All third-party trademarks are the property of their respective owners.

Contents

About This Guide

Introduction

The purpose of this documentation is to help you install and configure Novell® iChain™ infrastructure.

The audience for this documentation is experienced network administrators.

1 Overview

Novell® iChain™ is a network infrastructure that provides a common security and management framework for integrating your Internet-based business applications and creating an electronic business (e-business) infrastructure.

This chapter examines the need for iChain, lists and describes the components that comprise iChain, and lists and describes the features and services provided by iChain. It contains the following sections:

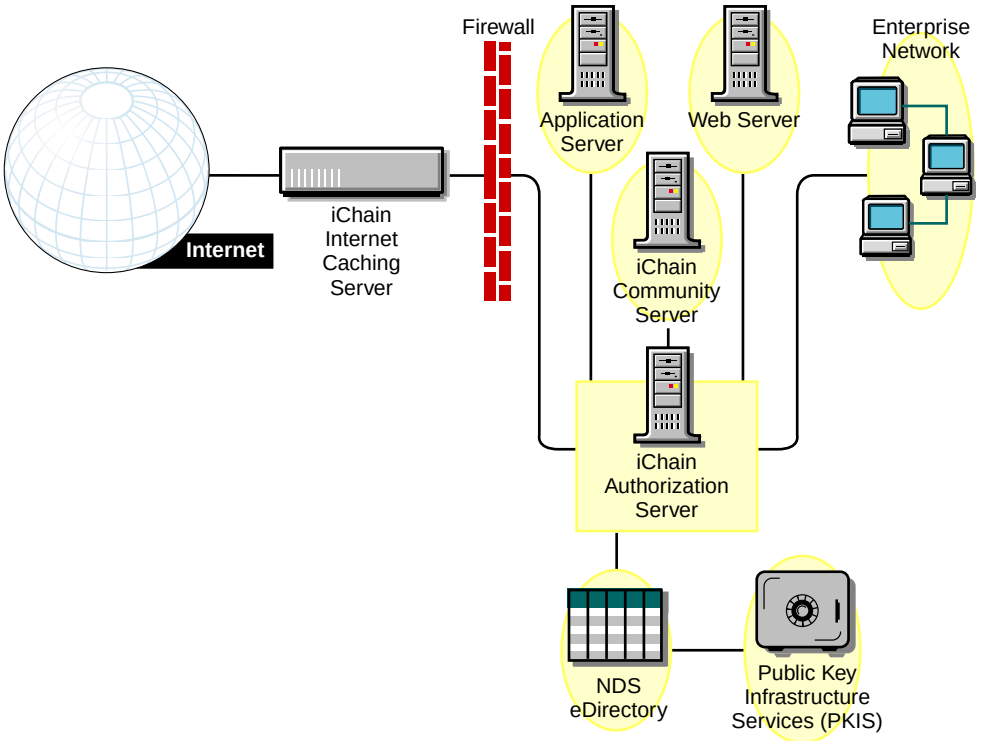
Understanding the Need for iChain

Just as the industrial age started a revolution in the way business was conducted, the Internet age is revolutionizing business again. The electronic age is affecting every aspect of business today—from the supply chain (procurement, manufacturing, inventory control, and warehousing) to the customer sale (publicity, customer assistance, order processing, and distribution).

During this electronic business revolution, many business processes and applications have been moved onto the internet. To protect access to these applications, "trusted" internal user applications have been placed behind a firewall on an intranet. Security protocols and procedures have been implemented to protect intranet applications. "Untrusted" external user applications have been placed outside the firewall on extranets on which little or no security has been implemented.

However, as the electronic business model evolves, it is becoming more difficult to categorize only internal users as trusted and all external users as untrusted. As users from inside and outside the network need access to various services on the network, additional security mechanisms for both the service and the user must be set up. Both the administrator and the user experience the complexities and difficulties of maintaining these security mechanisms as access and management become more confusing and time-consuming.

iChain enables you to simplify your network infrastructure. It allows you to integrate the users, applications, services, and processes of your electronic business into a single, cohesive structure. iChain eliminates the boundary between the intranet and the extranet and simplifies the user interface, the management experience, and the security model. With iChain, your network infrastructure might look like the following:



Understanding iChain Components

iChain is not a single application. It is a set of components for building an electronic business infrastructure. iChain is comprised of the following components:

iChain Internet Caching Server

A key component of the iChain infrastructure is the iChain Internet caching 1.5 server. This component is a specialized and customized release of the Novell Internet Caching System™ (ICS) server based on code from the ICS 1.3 and 2.0 releases.

NOTE: For details on the features specific to the iChain Internet caching 1.5 server, refer to Appendix 5, "Managing iChain," on page 49.

The iChain Internet caching server becomes the primary access point for all users into the infrastructure. To access the Internet services on your network, users must log in through the iChain Internet caching server and authenticate to the iChain authorization server. The iChain Internet caching server communicates with the iChain authorization server to determine access privileges and community membership. All iChain authentication, access privilege, and community membership information is stored in NDS®. The iChain Internet caching server uses LDAP to access this information. It then enforces the access control policies for the user and serves up a customized user interface for the user based on community membership.

iChain Authorization Server

The iChain authorization server can be installed on an NDS™ eDirectory version 8.5 server. The iChain authorization server functions as the initial authentication server in the iChain infrastructure. User authentication information is stored in an NDS or LDAP database. The iChain Internet caching server accesses this information when authenticating users.

In addition to being repository and access point for user authentication information, the iChain authorization server is also the repository and access point for access privilege and community membership information for your iChain implementation that is stored in the NDS database. The iChain authorization server accesses this data directly to determine the appropriate access for iChain users.

The iChain authorization server provides two basic functions: user authentication and access control. In the implementation described in this document, these two functions are provided by the same server. However, because these functions can be separated and implemented on different servers, other implemenations are possible. For example, you could install an iChain authorization server on the public side of the firewall in one tree to implement the iChain user authentication functions and install another iChain authorization server on the private side of the firewall in a separate tree to implement the access control functions. In this case, user authentication information would be stored in an LDAP database and special DirXML™ synchronization modules would keep the LDAP database and the iChain NDS database information in synchronization.

iChain Community Server

The iChain community server is an optional component available for your iChain infrastructure. iChain community server software is available for download from www.novell.com. iChain community server software can be installed and run on the Microsoft* Internet Information Server (IIS) 4.0 or a NetWare® 5.1 server with a Novell Enterprise Web Server. An iChain community server is a Web server with additional servlet code to enable a variety of community services provided by iChain. Using a community server in your infrastructure is optional.

iChain Communities

Communities are a special feature of iChain that enable you to take advantage of iChain community services. Using communities is optional. Communities are groups of users to which you grant access to your network. Configuration properties for a community are stored in an NDS object and are configured using the iChain ConsoleOne™ administration utility. iChain communities enable you to control access and provide customized interfaces and services to members of these communities. Because you implement communities specifically for controlling iChain infrastructure access, they can be designed to provide just the access needed and give you greater control than NDS groups over access to Internet services.

iChain Web Servers

Web servers store and serve up the data that iChain end users need to access. They also provide Web interfaces to integrated applications. The iChain infrastructure allows you to integrate Web servers from any vendor on any platform.

NDS eDirectory

NDS eDirectory version 8.5 is the underlying integration point for iChain. NDS is used to maintain and store all the configuration, access control, and relationship information for iChain. Information about the iChain Internet caching server, the iChain authorization server, the iChain community server, the accessible Web servers, community membership, access control policies, special features implemented, and parameters for building customized user interfaces are stored within NDS. This enables you to configure and manage iChain components and features from ConsoleOne, the NDS management interface.

In addition, using NDS allows easy integration with external applications by supporting Lightweight Directory Access Protocol (LDAP) and DirXML.

The NDS eDirectory natively supports LDAP version 3. Because LDAP is becoming the industry standard for application-to-directory access over the Internet, this gives you tremendous platform independence when implementing applications. Any application written to use LDAP as its directory access protocol can use the NDS directory as its data store. This means that all your LDAP-compliant applications are already NDS applications.

The NDS eDirectory also supports DirXML, giving you further platform and application integration flexibility. DirXML is quickly becoming the industry standard for directory-to-directory communication. DirXML allows you maintain the proprietary directories for applications in your network, and also allows exchange of this information between the proprietary directory and NDS. This enables you to integrate proprietary directory-based applications into your iChain infrastructure.

iChain Public Key Infrastructure Services

iChain includes Novell Public Key Infrastructure Services (PKIS 2.0). The PKIS component provides cryptography and enables certificate services in your iChain network. PKIS allows you to establish Certificate Authorities, create server and end-user certificates, and perform management activities to use public key cryptography. PKIS is a robust and secure platform built upon Novell's international standard, the Novell International Cryptographic Infrastructure (NICI). In Novell's PKIS security model, the server is believed more secure than the client, which can contain viruses, Trojan horses, and so on that could compromise the cryptographic keys. Because of this, PKI key generation is only performed within the protected domain on the server. In addition, in Novell PKI, private keys are never transmitted in the clear over a network wire and are never manipulated or managed by the client.

iChain Internet Caching Server Browser-Based Administration Utility

The iChain Internet caching server is based on the Internet Caching System and supports a browser-based administration utility. This utility enables you to manage and administer your iChain Internet caching server from a browser. To launch the utility, you simply access a special management URL on the iChain Internet caching server.

The URL must contain either the 10-net management address or the IP address you have already configured for the server, followed by :1959/appliance/config.html. For example,

http://10.1.1.1:1959/appliance/config.html

iChain ConsoleOne Utility

The ConsoleOne utility enables you to manage and administer your iChain network. ConsoleOne is installed with NDS eDirectory. During iChain installation, additional capabilities and features are provided by an iChain

snap-in module that is installed into ConsoleOne utility and allows you to manage and administer iChain. The iChain snap-in contains features for setting up and managing the NDS objects that contain your iChain configuration.

NOTE: The iChain ConsoleOne snap-in components require the latest Novell Client™ software.

Understanding iChain Services

The iChain infrastructure enables your network infrastructure to provide the following types of services:

- “Security Services” on page 15
- “Management Services” on page 16
- “Integration Services” on page 16

Security Services

Security services allow the infrastructure within iChain to provide authentication and ensure data confidentiality. iChain provides the following security-related services:

- Authentication flexibility. With NDS user ID-based authentication, you can set up contexts in which to search for users. Contextless login and single sign-on are supported as well. Once an iChain user is authenticated, that user can access data on other Web servers data without needing to reauthenticate because the user credential information is passed in the HTTP request header.
- Data confidentiality. iChain includes several mechanisms to ensure that data confidentiality is not compromised. SSL capabilities provide both server-to-server and server-to-client confidentiality. SSL is fully supported. Additionally, if you want to ensure that your data is always encrypted, you can enable the SSLizer on the link. With the SSLizer, all data between the ICS server and the end-user browser is redirected to and served to the end user via a secure connection.
- PKI services and X.509 Certificates Distribution. iChain includes NetWare PKIS 2.0 services. This service allows you to manage any X.509 certificate, such as Entrust*, Verisign*, or Baltimore Technologies* and easily distribute certificates to users.

Management Services

Management services provide the structure and tools to configure, monitor, and manage your iChain infrastructure. iChain provides the following management-related services:

- ConsoleOne configuration and administration. Configuration is stored in NDS objects. The ConsoleOne utility enables you to quickly and easily configure and administer the NDS object properties that govern your iChain infrastructure.
- Browser-based administration. A browser-based administration utility is available for configuring the iChain Internet caching server.
- Delegated administration. This feature enables you to grant third parties or business partners control over a logical structure of your network so they can add or delete users and communities.
- Text-based audit logs. Audit results are reported in two text-based logs. The audit log reports all attempts to gain access to the system through iChain, whether successful or not, and includes the date and time, user ID, user actions, and number of access attempts. You can view these logs through any browser or editor, or export them with third-party applications. Because the logs are in a standard, text-based format, third-party applications such as WebTrends* can be used to analyze logged information.

Integration Services

Integration services enable you to integrate users, applications, and management capabilities over a variety of platforms and protocols. iChain provides the following integration-related services:

- Add-on application integration. LDAP directory support within NDS facilitates application integration regardless of platform. DirXML support allows communication and integration between NDS and proprietary directories.
- Web servers and Web application integration. iChain fully integrates and supports access to Web servers and Web applications on a variety of platforms, including the Novell Enterprise Web Server, the Microsoft Internet Information Server, the Netscape* Web Server, and the Apache Web Server.

Understanding iChain Configuration

The NDS eDirectory is the underlying integration point for iChain. NDS maintains and stores most of the configuration, access control, and relationship information for iChain in its directory objects. Additional configuration information is stored in the iChain Internet caching server configuration files. The following topics provide further detail in the iChain configuration:

iChain Internet Caching Server Configuration

iChain Internet caching server configuration is administered from a browser-based administration utility. The utility allows you to define the following:

- Web Accelerator for HTTP services within iChain—Web Accelerator parameters are set from the ICS server's Cache > Web Accelerator tab. These parameters include such HTTP service attributes as the Web server address and port number. You can also enable the SSLizer feature on the port, specify the SSL listening port and key ID, set the idle time before requiring a new login, set whether to request a client certificate during authentication and whether the user is required to authenticate only when accessing restricted pages, and enable logging.
- Access Control Parameters for iChain—Access control parameters are set from the ICS server's Cache > Access Control tab. These parameters include the iChain Service Object for the iChain service and various LDAP configuration settings for the iChain LDAP access control server (a username for the LDAP guest user account, the IP address of the LDAP server, the LDAP server listening port and the LDAP server administrator username and password).
- Authentication Parameters for iChain—Authentication parameters are set from the ICS server's Cache > Authentication tab. These parameters define an iChain authentication profile. An authentication profile specifies the scheme or schemes required to authenticate and be granted access. Available authentication schemes are LDAP or Certificate.

iChain Service Object

The iChain Service object (ISO) is administered from the ConsoleOne utility. The ISO contains attributes that define an iChain "domain." ISO attributes define users, groups, communities, host servers, web servers, and applications for the domain. The iChain Service object attributes determine which users, groups, or communities access iChain infrastructure resources. The iChain Service object consists of the following property pages:

- General—Allows you to define a description of the service object, the mail server host name or IP address, the default scheme, the guest users allowed, and the administrator for the object. It also allows you to enable self-registration for the users associated with this service object.
- Communities—Allows you to list the communities associated with this service object or domain.
- Host Servers—Defines the iChain community server for the service object.
- Protected Resources—Defines the resources that are accessible from the service. Access to these resources is protected or controlled by the iChain service. Protected resources can be Web servers, Web sites or Web-based add-on applications. Protected resources are applications that require additional user information about the user to be passed into the application to protect the resource or customize the user interface. This additional information is usually stored in NDS or some other database. A special plug-in is used to access the database and retrieve the additional information.
- User Password—Allows you to manage passwords for users of community services. In addition to standard NDS password management, iChain provides other password management capabilities , such as dictionary lookups.
- NDS Rights—Defines the trustees of the service object.
- Other—Defines any special attributes of the object not defined in the other custom pages.
- Rights to Files and Folders—Defines the trustee file system rights for the object.

Community Object

The Community object contains attributes that define iChain communities. This object is similar to an OU object. It contains many of the same tab pages and properties of an OU object with additional Community tabs that enable you to set up special iChain community services.

NOTE: These additional community services are not available unless you download and install the Community Server software.

Access Control List Object

The Access Control List (ACL) object contains attributes that define iChain access control rules. iChain ACLs can be applied to users, groups, or communities. The iChain Access Rules tab allows you to create, name, and define a description for a rule, enable or disable the rule, and enable or disable authorized logging for the rule. It also allows you to specify a list of resources to which the rule controls access; specify a list of communities, groups, or users to which this rule applies and grants access to the URLs; and specify an exception list containing a subset of users, groups, or community members of the Apply To list to whose access to the URLs should be denied.

2 Installing iChain Services

This chapter provides instructions for installing Novell® iChain™ services software and contains the following sections:

Product Components

Your iChain installation includes the following components:

- iChain Internet Caching Server CD-ROM
- iChain Authorization Server CD-ROM

Installation Scenario

iChain is a flexible software solution that can be implemented in a variety of configurations depending on the needs of your network. Because the possible variations in installation scenarios are endless, procedures in this chapter describe the installation of a basic or standard infrastructure consisting of the following:

- One iChain Internet caching server, a special release of Novell Internet Caching System™ (ICS) version 1.3 server to support iChain.
- One iChain authorization server.

- An iChain extended schema on the NDS® tree where the iChain authorization server resides.
- An iChain Service object (ISO).
- An administrator workstation with ConsoleOne™ iChain snap-ins installed.

NOTE: For increased security, we recommend installing the iChain authorization server in a tree separate from your corporate file/print tree. DirXML™ can be used to synchronize user account information between trees if desired.

System Requirements

Review the following system requirements to ensure that your server and client environments meet installation prerequisites:

- "iChain Internet Caching Server Requirements" on page 22
- "iChain Authorization Server Requirements" on page 22
- "Administrator Workstation Requirements" on page 23

iChain Internet Caching Server Requirements

iChain Internet caching server code can only be installed on an Internet Caching System server appliance from an approved vendor that meets the requirements for the ICS 1.3 (with Support Pack 1) release. ICS appliances are available from a variety of ICS OEM vendors. For information on current ICS OEM vendors, refer to the ICS Partner page (http://www.novell.com/products/ics/howtobuy.html).

NOTE: ICS 1.3 Support Pack 1requires 256 MB memory. Because not all ICS 1.3 platforms meet this requirement (for example, the Compaq ProLiant* 1850R has only 128 MB), you must verify your system meets this memory requirement.

iChain Authorization Server Requirements

The iChain authorization server can be installed on the following NDS eDirectory™ version 8.5 platforms:

- NetWare® 5.1 with Support Pack 1
- Windows* NT* 4.0 Support Pack 4

For additional information on the supported platforms and system requirements for NDS eDirectory version 8.5, refer to the NDS eDirectory 8.5 Quick Start available at the Novell Documentation site (http://www.novell.com/documentation/lg/ndsedir/docui/index.html).

For NetWare installations, verify the following:

- Installation of iChain on a new, clean NetWare (NetWare 5.1 with Support Pack 1) server is recommended. Installation of the NDS eDirectory 8.5 component is required, but the following NetWare server components should NOT be installed:

  Print server
  FTP server
  Enterprise Web Server
  Web Search
  IBM* WebSphere* Application Server

- The certificate authority should be installed with the iChain authorization server, not with the iChain community server.
- The TCP/IP network interface on the server should be bound and configured.
- LDAP should be configured on the server.

NOTE: For increased security, we recommend installing the iChain authorization server in a tree separate from your corporate file/print tree. DirXML can be used to synchronize user account information between trees if desired.

We also recommend that the authorization server should be the first server in the tree and therefore contain the master replica of the tree.

Administrator Workstation Requirements

The administrator workstation requirements are as follows:

- Pentium* 233-MHz processor or higher
- Minimum 35 MB of free disk space
- Minimum 128 MB of RAM
- One LAN card
- Windows 95, Windows 98, or (preferably) Windows NT 4.0
- Current Service Pack for Windows
- Current Novell Client™

- Connectivity between the iChain authorization server and the ICS server

Installing iChain Services Software

To install a basic iChain infrastructure, complete the following procedures:

- “Installing the iChain Internet Caching Server Software” on page 24
- “Installing iChain Services Schema Extensions on the iChain Authorization Server” on page 25
- “Installing the iChain ConsoleOne Snap-ins” on page 25

Installing the iChain Internet Caching Server Software

The iChain Internet caching server is installed on an Internet Caching System version 1.3 (with Support Pack 1) hardware appliance. To install the iChain Internet caching server software to your ICS server appliance, perform the following steps:

1. Insert the *iChain 1.5 Internet Caching Server* CD in the CD drive of the ICS server appliance.
2. At the license screen, type **YES** and press Enter.

   The disk image is copied. When the copy is complete, the system reboots itself.
3. At the Internet Caching System prompt, enter the following commands to set an Ethernet port so the device can be accessed for configuration:

   ```
   set eth0 address = xx.xx.xx.xx

   apply
   ```

NOTE: After installation, your iChain Internet caching server will require some basic setup to support your iChain implementation. The basic steps are detailed in “Setting Up the iChain Internet Caching Server” on page 28. For more general information on the functionality and configuration of your ICS server, refer to the *Internet Caching System Administration Guide* available at the Novell Documentation site (http://www.novell.com/documentation/lg/ics13/docui/index.html).

Installing iChain Services Schema Extensions on the iChain Authorization Server

The iChain Authorization server is the access point that the ICS server uses to retrieve authentication, access privileges, user, group and community membership information for your iChain implementation from the NDS database. All you need to do to make your NetWare 5.1 or Windows NT 4.0 NDS eDirectory server platform into an iChain authorization server is install the iChain schema extensions onto the NDS tree from that server.

To install iChain schema extensions on the iChain authorization server, complete the following steps:

1. If you have not already done so, install NDS eDirectory 8.5 on the machine that will be your iChain authorization server.
2. At a client on the private network, log in to the iChain authorization server within the NDS tree where you want to install the schema extensions for iChain.

   NOTE: You must have administrator rights to the tree to perform schema extensions
3. Insert the iChain authorization CD in the CD drive.

   The installation program launches automatically.
4. Click Install iChain Schema.
5. Enter the administrator user name in LDAP format (for example, cn=admin, 0=novell) > click OK.
6. Enter the administrator password > click OK.
7. Enter the IP address of the server on which to extend the schema > click OK.
8. Restart the server.

Installing the iChain ConsoleOne Snap-ins

You must install the iChain ConsoleOne snap-in files in order to administer the iChain NDS objects such as the iChain Service Object. You may install the snap-in files onto the iChain authorization server, another server in the tree, or to an administrator workstation.

To install the iChain ConsoleOne snap-ins to a server or an administrator workstation, complete the following steps:

1. If the server or workstation does not already have ConsoleOne installed, install ConsoleOne.
2. Insert the iChain authorization CD in the CD drive of the server or the administrator workstation.

   The installation program launches automatically.
3. Click Install ConsoleOne Snapins for iChain.
4. From the Welcome screen, click Next.
5. Read the license agreement. If you accept the terms of the agreement, click Yes.
6. Select the target drive where you want to copy the snap-in files.
7. Click Next to start copying the files.
8. Click Finish.

After completing the full installation, you will need to use ConsoleOne to create the iChain service group object, along with the access control list (ACL) rule objects and make any other configuration adjustments. The access control lists can apply to users, groups, Organizational Units (OUs), or communities, so that they can be easily applied to many users. The access control lists also specify the URLs that users are allowed to access. Access to all other URLs is not allowed through the iChain Internet caching server. You can use a wildcard character (*) to represent any string of characters. NDS user or group objects reference the ACL rule objects to indicate the users to which they apply.

3 Setting Up a Basic Configuration

This chapter explains the tasks you need to complete to set up a basic Novell® iChain™ infrastructure. For details on the basic scenario, refer to “Installation Scenario” on page 21. This chapter contains the following sections:

Creating an iChain Service Object

The iChain Service object contains configuration information for the iChain service. An iChain service is a logical entity that defines an iChain domain and the resources of that domain. Configuration for your iChain service is contained in an iChain Service object (ISO). To set up a basic iChain implementation, you must create an ISO and set up the service parameters for the object. In addition, to set up your basic implementation, you must also set up access to Web-based application resources.

To create an iChain Service object (ISO), complete the following steps:

1. From ConsoleOne™, select an OU in which to create your ISO > select File > New > New Object.

   or

   Click the New iChain object icon (to the right of the New Object icon).

2 Select iChain Service and define a name for the service or domain (for example, Test) > click OK.

Setting Up the iChain Internet Caching Server

The iChain Internet caching server is implemented on an Internet Caching System™ (ICS) server. The iChain Internet caching server functions as the primary access point into your iChain infrastructure.

NOTE: After installation, your iChain Internet caching server will require some setup to support your iChain implementation. The basic steps are detailed in this section. For more general information on the functionality and configuration of your ICS server, refer to the *Internet Caching System Administration Guide* located at:

http://www.novell.com/documentation/lg/ics13/docui/index.html

To set up the iChain Internet caching server for an iChain implementation, complete the following:

1 Access the URL of the ICS server where you installed the iChain Internet caching server software to launch the ICS browser-based adminstration tool.

For example, http://*xx.xx.xx.xx*:1959/appliance/config.html

where *xx.xx.xx.xx* is the IP address for the ICS server. You should have configured an IP address during the installation of the iChain Internet caching server software.

NOTE: If the ICS server is located behind a firewall and you are accessing the ICS browser-based administration utility from a browser outside that firewall, you must open ports 1959, 2222, and 1100 on the firewall to administrer the ICS server.

2 Accept the default username (do not enter a password) > click OK.

3 Click System > Actions> Password > set a password for the ICS server.

4 Click Home > Introduction > verify that ICS version 1.5 is installed and running on the server.

5 Click Network > IP Addresses > configure the Ethernet ports as follows:

- Accept the Eth0 adapter existing setting.
- Set the Eth1 adapter with the private IP address for the network.
- Set the Eth2 adapter with the public IP address for the network.

6 Click Gateway-Firewall > set the ICS default gateway to the gateway necessary to access your public IP address.

7. Click Network > DNS > specify the DNS domain name (for example, www.novell.com) and the IP address of the DNS server.
8. Click System > Actions > verify the internal connection to your network by pinging a server within your internal network.

To set up access to the authorization server for access control functions, perform the following steps:

1. Click Cache > Access Control.
2. Specify the fully distinguished name of the ISO object name for the iChain service. You must use commas as delimiters.
3. Specify a fully distinguished name for the LDAP Guest User account.
4. Specify the following LDAP profile settings:
   - LDAP server address for the iChain LDAP access control server
   - LDAP server listening port on the iChain LDAP access control server
   - LDAP user name
   - LDAP password
5. Click Refresh ACLCHECK.

To set up access to the iChain authorization server for authentication functions, perform the following steps:

1. Click Cache > Authentication.
2. Insert a new profile > name the profile > select LDAP Authentication and LDAP options.
3. Set the server IP address to iChain authorization server.
4. Select port 389 for non-secured LDAP.
5. Set Use Distinguished Name > insert an LDAP context.
6. Click Insert > enter an LDAP context)for example OU=TEST O=MYCOMPANY)
7. Click OK > OK > Apply.

To set up a Web Server accelerator, complete the following steps:

1. Click Cache > Web Server Accelerator > Insert.
2. Specify a name for the accelerator, using a maximum of 8 characters.

3 Specify a DNS name for the accelerator (for example, www.novell.com).

4 In Web Server address, click Insert > specify the IP address for ICS to use to reroute to the Web Server. You can also enter the DNS name for the Web server.

5 In Accelerator IP address, check the public IP address.

6 Check Enable Authentication.

7 Click Authentication Options > select an existing profile from the list > click Add to set the profile as the Service Profile.

8 Click OK > OK > Apply.

Setting Up a Guest User Account for Public Access

If there are areas on your Web server you want to make accessible to the public, you can set those areas up for public access. To set up public access, complete the following steps:

1 Access the URL of the ICS server where you installed the iChain Internet caching server software to launch the ICS browser-based adminstration tool.

For example, http://*xx.xx.xx.xx*:1959/appliance/config.html

where *xx.xx.xx.xx* is the IP address.

2 Click Cache > Web Server Accelerator > Modify.

3 Click Authentication Options.

4 Check Authenticate Only When User Attempts to Access a Restricted Page.

5 Click OK > OK > Apply.

6 From ConsoleOne, select File > New > New Object and create a new User object in the NDS® tree.

This user will act as the guest account for unregistered users. Make sure this user does not have a password.

7 Specify the new User object to be a guest user by selecting the object from the Guest User list in the ISO object for the service.

8 Select File > New > New Object, or click the New iChain icon and create an ACL rule that defines the resources that the public will be allowed to access, as follows:

8a Define the list of URLs that the public will be allowed to access.

8b Add the root of your NDS tree to the OU/Group/User List.

8c Check the Enable Access Control check box.

Setting Up Protected Resources

To integrate and allow access to Web-based application resources, you must set the appropriate parameters in the iChain Service object. Sometimes Web-based resources, called Protected Resources in iChain, need additional information about the user to be passed into the application to protect the resource or customize the user interface. This additional information is usually stored in NDS or some other database. iChain uses special object-level access control plug-ins to access the database and retrieve the additional information. The iChain Internet caching server then passes the information to the application by adding it to the URL query string.

NOTE: For further information on setting up object-level access control, including information on the plug-ins provided with iChain refer to "Setting Up Object-Level Access Control (OLAC)" on page 41.

To set up an accessible and protected resource for an iChain service, complete the following steps:

1. From ConsoleOne, select File > New > New Object and create a new User object in the NDS tree.

   This user will act as the guest account to enable users to access applications without authenticating to NDS. Make sure this user does not have an NDS password. Any needed information to access an application, such as an application proprietary username and password, will be forwarded to access application in the authentication header.

   NOTE: The user object you create must not have a password. You may use the same guest user you created for public access in "Setting Up a Guest User Account for Public Access" on page 30.

2. Right-click the iChain Service object and select Properties.
3. Click the General tab.
4. Select the Guest User Add button and select the user object you created in Step 1 to be the guest user for the ISO object for the service.
5. Click the Protected Resources tab.
6. Click Add.

7 Specify a name for the resource and the URL for the resource.

8 Select the resource you just added > click Details.

The Details icon is left of the Add icon.

9 For each object-level access control attribute to be passed to the application, define the Name, Data Source, and Value.

NOTE: Object-level access control attributes should be created only for protected resources that are configured to process query strings. Typically, access to these protected resources is handled by CGI scripts, Web server plug-ins, dynamic pages, or similar methods. If this is not the case, users may encounter errors when attempting to access URLs within the protected resource.

The Name field contains an identifying name for the attribute that is passed to the application. The Data Source field contains the name of the database from which to retrieve the information. The Value field contains a key for retrieving the attribute value in the data source. The attribute is added to the URL query string in the format Name =Value. For example, if the application requires the last name of the user and the data is stored in an LDAP-accessible directory under Surname, the entries would be LastName, LDAP, and Surname. If the user's last name is "Smith," the attribute "LastName=Smith" would be added to the URL query string whenever the user accesses the protected resource.

For a table of appropriate values for the Data Source and Value fields, refer to "Setting Up Object-Level Access Control (OLAC)" on page 41.

10 Access the URL of the ICS server where you installed the iChain Internet caching server software to launch the ICS browser-based adminstration tool.

For example, http://*xx.xx.xx.xx*:1959/appliance/config.html

where *xx.xx.xx.xx* is the IP address.

11 Click Cache > Web Server Accelerator > Modify.

12 Click Authentication Options.

13 If you want to support passing the brower's IP address in the HTTP request to the Web server, check the Enable X-Forward-For check box.

14 Check Forward Auth to Web Server.

15 Click OK > OK > Apply.

16 Use a text editor to edit the SYS:/SYSTEM/APPSTART.NCF file on the ICS server.

17 Add the following line to the end of the file to automatically restart the Java* modules whenever the server is restarted.

```
OACJAVA
```

18 Use a text editor to edit the SYS:/ICHAIN/OAC/OAC.PROPERTIES file on the ICS server.

19 Under the [OAC] heading, edit the following:

- In the ISO Object Name field, specify the fully distinguished name for the iChain Service object for the service.
- In the Provider URL field, specify the IP address of the LDAP service provider (the server address where LDAP is running) for this service. This can be the IP address of the iChain LDAP server.
- In the Security Principal field, specify the name of the user who will have rights to read the object.
- In the Security Credentials field, specify the user's password.
- In the Refresh Time field, specify the time interval in minutes for periodically refreshing the protected resource information used by object-level access control. (After making changes, you can also refresh the protected resource information manually by entering **`oacrefresh`** at the ICS server console. The changes will take effect for future user logins. To apply the changes to currently logged-in users as well, reload the ICS server modules, including the oacint and authchk modules.)

20 Copy the Provider URL, the Security Principal, and the Security Credentials field values and settings to the [LDAP] and [iChain Processor] headings.

21 Save the OAC.PROPERTIES file.

22 From ConsoleOne, right-click the LDAP Group object associated with the LDAP service you specified in Step 19 > select Properties.

23 Check Allow Clear Text Password > click OK.

24 To apply the setup changes in the appropriate modules, enter **RESTART** at the console prompt of the ICS server associated with the service object.

Defining ACL Rules

After a user has logged in, access control list (ACL) rules control what resources the user can access. By default, the user has access to nothing. Only those resources explicitly listed in your ACL rules (specified by the URL) can be accessed by the communities, OUs, groups, or users listed in the Apply To list for the rule. Whenever possible, it is recommended that you use the highest-level object in the list of allowed users, making it easier and faster to configure an ACL rule. An exceptions list enables you to exclude certain users, communities, or group members listed in the Apply To List that you do not want have access to the specified URLs.

ACL rules are also used to specify the URLs that will be available for public access. In most cases, only one public access rule is required. Usually, one ACL rule is sufficient if it lists all of the public access URLs and specifies the root of the NDS tree in the Apply To List. When a guest user object is added at the root of the NDS tree, this rule will allow all registered and unregistered users to access the specified public access URLs.

When a guest user accesses a URL that is specified as a public access site, no login is required. However, when a user attempts to access a secured URL (any URL that is not on the list of public access sites), the user must log in to NDS and provide a password. When the user is authenticated, the ACL rules are checked to see if the user is allowed to access the site.

iChain automatically creates ACL rules when a link is added for a community. Do not alter these ACL rules. If you delete an ACL rule that has been automatically created for a community's link, the members of the community will no longer be able to access the corresponding URL. However, if you delete an automatically created ACL rule, that rule will be automatically recreated whenever any change is made to the community's list of links.

ACL rules allow the use of an asterisk (*) or question mark (?) as wildcard characters when specifying URLs. The asterisk indicates that the user can have access to the folder content and all subfolders. The question mark indicates the user can have access to the folder contents, but not the subfolders. Also, each ACL rule can be individually disabled or enabled, allowing you to turn on or off a particular rule for a time without losing its parameter settings.

ACL rules are stored in a cache that is updated periodically at a configurable interval. For performance reasons, the recommended cache refresh interval is three to six hours. If you make changes or additions to the ACL rules and want the cache to be updated immediately, use the manual Refresh option available

in the Cache > Access Control tabs of the ICS Browser-based administration tool.

When creating an entry in the URL list of an ACL rule, at least one of the two fields (Resource Name and URL) are required.

If only the URL is specified, it must be given as an absolute URL (for example, http://www.novell.com/index.html, not /index.html). The URL may contain wildcards. The ACL rule will match any request for the URL (including wildcards).

If only the Resource Name is specified, the ACL rule will match any request for the exact path of the Resource Name. For example, if the protected resource myserver has been defined as http://www.novell.com, and a URL list entry is created with myserver as the Resource Name and with no URL, then the ACL rule will apply only to the http://www.novell.com URL only. Wildcards are not allowed when defining protected resources.

If both the Resource Name and the URL are specified, the URL must be given as a relative URL (/index.html, not http://www.novell.com/index.html) and may include wildcards. The ACL rule will match requests for the combined Resource Name and URL, including wildcards. For example, if the Resource Name is myserver and the URL is /documentation/*, then the ACL rule will apply to http://www.novell.com/documentation/*.

To create a new ACL rule for iChain, complete the following steps:

1. From ConsoleOne, select File > New > New Object.

   or

   Click the New iChain Object icon.

2. Select ACL Rule > click OK

3. Define a name for the rule > click Define Additional Properties > click OK.

4. Under the list of URLs, click Add > define a name and URL for a resource that this rule will control access to.

   You can use an asterisk (*) or question mark (?) as a wildcard character when specifying URLs. The asterisk indicates that the user can have access to the folder content and all subfolders. The question mark indicates the user can have access to the folder contents, but not the subfolders.

5. Under the Apply To List, click Add to browse to and select the communities, OUs, groups, and users to which this rule applies.

   Communities, OUs, groups, and users in the Apply to List are allowed access to the listed URLs.

6. Under the Exception List, click Add to browse to and select the communities, OUs, groups, and users that are exceptions to this rule.

   The communities, OUs, groups, and users in the Exceptions List are a subset of the Apply to List and are objects to which you want to deny access to the listed URLs.

7. To enable the ACL rule, check the Enable Access Control check box.

8. To disable the ACL rule and save it for later use, uncheck the Enable Access Control check box.

4 Setting Up an Advanced Configuration

The previous chapters in this document described how to install and set up the basic implementation described in “Installation Scenario” on page 21. To meet your company’s networking needs, you may need to augment or alter this implementation and use some of Novell® iChain™ services more advanced features. This chapter describes the following Novell iChain configuration procedures:

Setting Up iChain Internet Caching Server Parameters

The iChain Internet caching server is implemented on an ICS server with special code to support iChain. The iChain Internet caching server parameters are stored in a configuration file. During the installation of the iChain Internet caching server software on the ICS server, the public and private addresses of the server were defined. However, depending on the needs of your particular implementation, you may want to alter these settings or define additional values.

To set up the public and private addresses of the server, complete the following steps:

1. Access the URL of the ICS server on which you installed the iChain Internet caching server software to launch the browser-based administration tool.

   The URL must contain either the 10-net management address or the IP address you have already configured for the server, followed by :1959/appliance/config.html. For example,

   http://10.1.1.1:1959/appliance/config.html

2. Enter the password for the ICS server.
3. Click Network > IP Addresses and configure the Ethernet adapter ports with the appropriate public and private addresses and subnet masks.

Setting Up iChain Service Parameters

iChain service parameters are stored in the iChain Service object (ISO). During the basic setup of your iChain infrastructure, an ISO was created and an association between the iChain Internet caching server and the ISO was defined. However, depending on the needs of your particular implementation, you may want to perform one or both of the following procedures to create a new object or alter its settings:

- “Creating an iChain Service Object” on page 38
- “Associating an iChain Internet Caching Server with a Service” on page 39

Creating an iChain Service Object

To create an iChain Service object, complete the following steps:

1. From ConsoleOne™, select an OU in which to create your ISO > select File > New > New Object.

   or

   Click the New iChain object icon (to the right of the New Object icon).

2. Select iChain Service and define a name for the service or domain (for example, Test) > click OK.

Associating an iChain Internet Caching Server with a Service

To associate an iChain Internet caching server with an ISO, complete the following steps:

1. Access the URL of the ICS server where you installed the iChain Internet caching server software to launch the ICS browser-based adminstration tool.

   For example, http://*xx.xx.xx.xx*:1959/appliance/config.html

   where *xx.xx.xx.xx* is the IP address.

2. Click Cache > Access Control.

3. Specify a fully distinguished name for the ISO for the service.

   You must use comma delimiters.

Setting Up User Password Management

You can set up iChain so users can manage their passwords from their browsers.

NOTE: You must download and install the community server software to implement this feature. For information on downloading community server software, refer to "iChain Community Server" on page 12.

To set up this feature, complete the following steps:

1. Access the URL of the ICS server on which you installed the iChain Internet caching server software to launch the ICS browser-based adminstration tool.

   For example, http://*xx.xx.xx.xx*:1959/appliance/config.html

   where *xx.xx.xx.xx* is the IP address.

2. Click Cache > Access Control > Password Management Servlet.

3. Enter the following URL in the Servlet field:

   `http://servername/iChain/PasswordExpired.html`

   where *servername* is the server name users use to access the community server.

Setting Up SSLizer

The SSLizer is used when a Web server does not provide Secure Sockets Layer (SSL) functionality but you still want to access Web pages securely over the Internet. If the SSLizer is enabled, then all of the HTTP requests coming to the iChain Internet caching server will be redirected to HTTPS, causing the data exchanged between the browser and the server to be encrypted using SSL.

To set up SSLizer, complete the following steps:

1. Export a base-64 trusted root file for the Web server where you want to enable secure access.
2. Access the URL of the ICS server where you installed the iChain Internet caching server software to launch the ICS browser-based adminstration tool.

   For example, http://*xx.xx.xx.xx*:1959/appliance/config.html

   where *xx.xx.xx.xx* is the IP address.
3. Click Cache > Web Server Accelerator > Modify.
4. Check Enable SSLizer.
5. Specify the port number to use SSLized traffic.
6. Specify the key ID to use to establish the SSL session.
7. Click SSLizer Options.
8. Click Enable Secure Access Between the SSLizer and Web Server.
9. Click Insert > Import Trusted Root.
10. Enter a file name with a .DER extension.
11. Paste the contents of the previously exported base-64 trusted root file in the lower text box.
12. Click OK > OK > Apply.

Setting Up Public Access

If there are areas on your Web server you want to make accessible to the public, you can set those areas up for public access. To set up public access, complete the following steps:

1. Access the URL of the ICS server where you installed the iChain Internet caching server software to launch the ICS browser-based adminstration tool.

   For example, http://*xx.xx.xx.xx*:1959/appliance/config.html

   where *xx.xx.xx.xx* is the IP address.

2. Click Cache > Web Server Accelerator > Modify.
3. Click Authentication Options.
4. Check Authenticate Only When User Attempts to Access a Restricted Page.
5. Click OK > OK > Apply.
6. From ConsoleOne, select File > New > New Object and create a new User object in the NDS® tree.

   This user will act as the guest account for unregistered users. Make sure this user does not have a password.

7. Specify the new User object to be a guest user by selecting the object from the Guest User list in the ISO object for the service.
8. Select File > New > New Object, or click the New iChain icon and create an ACL rule that defines the resources that the public will be allowed to access, as follows:

   8a. Define the list of URLs that the public will be allowed to access.

   8b. Add the root of your NDS tree to the OU/Group/User List.

   8c. Check the Enable Access Control check box.

Setting Up Object-Level Access Control (OLAC)

The iChain service enables you to integrate and allow access to Web-based applications. Sometimes these resources or objects need additional access control or application information about the user to be passed into the application. This additional information about the user can be stored in NDS or some other database. Within iChain, these resources are called Protected Resources and access to them is set up through the Protected Resources tab of the iChain Service object. Refer to "Creating an iChain Service Object" on page 27 for basic setup information.

To implement this feature, two special iChain object-level access control (OLAC) plug-ins (an LDAP plug-in and an iChain communities plug-in) are available to access the database and retrieve the additional information. By default, these plug-ins allow you to define attributes that are embedded and passed within the HTTP request header. You can assign a name as the tag to the data. Common values that are recognized by community services are ICHAIN_UID and ICHAIN_PWD.

The following table lists each plug-in and its corresponding entries for the Data Source and Value fields in ConsoleOne™.

Plug-in	Description	Data Source	Value
LDAP	Adds user attributes from a directory with LDAP support.	ldap (case-insensitive)	Any LDAP user attribute (for example, surname, givenName).
iChain Communities	Adds a list of user's communities, in LDAP naming format (for example, ou=myCommunity,ou=Accounting,o=novell).	ichain (case-insensitive)	Currently, the only valid value is “communities”.

Because the plug-ins are based on iChain APIs, you may customize iChain and create OLAC plug-ins to integrate your applications as needed. The APIs for customizing your iChain infrastructure are available in the iChain Novell Developer Kit (NDK). Novell NDKs are available for download at Novell Developer site (http://developer.novell.com/ndk/).

NOTE: Only administrators familiar with programming principles and Java* programming syntax should attempt to customize OLAC plug-ins.

The settings for the OLAC framework and plug-ins are stored in the OAC.PROPERTIES file, which is typically found in the SYS:/ICHAIN/OAC directory on the ICS server. The configuration file contains a section for the framework as well as one for each plug-in. The following table lists the valid OLAC options for each section:

Name	Description	Required?	Default Value
Object-Level Access ControlOptions [OAC] section			

Name	Description	Required?	Default Value
Initial Context Factory	The JNDI factory class that creates the context for directory lookups. Typically specified as com.sun.jndi.ldap.LdapCtxFactory.	Yes	None
ISO Object Name	The name of the iChain Service object. Must be specified in the LDAP format. For example, cn=iso,o=novell.	Yes	None
Provider URL	URL for the LDAP server used to look up the iChain Service object (ISO). Typically specified as ldap://*ip-address*:389/.	Yes	None
Security Principal	The identity to use when accessing the directory. This must be a user with sufficient rights to read the ISO, such as the admin user. Must be specified in the LDAP format. For example, cn=admin,o=novell.	No	If this option is not present, the directory will be accessed as the anonymous user.
Security Credentials	The password of the user specified as the Security Principal.	No	None
Security Authentication	The method to use when authenticating to the LDAP server. Currently, only "simple" is supported.	No	simple
Server Port	The port on which the OLAC framework will listen for lookup requests from the proxy server.	No	4444

Name	Description	Required?	Default Value
Worker Count	The number of worker threads to create.	No	10
Refresh Time	The number of minutes after which the OLAC configuration will be re-read from the ISO.	No	60
Value Delimiter	The delimiter used to separate multiple values assigned to the same name in the URL query string. For example, if Value Delimiter is specified as ";", the resulting query string might look like COLORS=blue;green;yellow;orange&SHAPES=circle;square;triangle.	No	, (comma character)
LDAP Plug-In Options [LDAP Processor] section			
Initial Context Factory	The JNDI factory class that creates the context for directory lookups. Typically specified as com.sun.jndi.ldap.LdapCtxFactory.	Yes	None
Provider URL	URL for the LDAP server used to look up user attributes. Typically specified as ldap://*ip-address*:389/.	Yes	None
Security Principal	The identity to use when accessing the directory. This must be a user with sufficient rights to read the ISO, such as the admin user. Must be specified in the LDAP format. For example, cn=admin,o=novell.	No	If this option is not present, the directory will be accessed as the anonymous user.

Name	Description	Required?	Default Value
Security Credentials	The password of the user specified as the Security Principal.	No	None
Security Authentication	The method to use when authenticating to the LDAP server. Currently, only "simple" is supported.	No	simple
Class Name	The name of the class implementing the LDAP plug-in. Must be com.novell.ichain.oac.ldap.ParamListBuilder.	Yes	None
iChain Plug-In Options [iChain Processor] section			
Initial Context Factory	The JNDI factory class that creates the context for directory lookups. Typically specified as com.sun.jndi.ldap.LdapCtxFactory.	Yes	None
Provider URL	URL for the LDAP server used to look up the user attributes and communities. Typically specified as ldap://*ip-address*:389/.	Yes	None
Security Principal	The identity to use when accessing the directory. This must be a user with sufficient rights to read the ISO, such as the admin user. Must be specified in the LDAP format. For example, cn=admin,o=novell.	No	If this option is not present, the directory will be accessed as the anonymous user.
Security Credentials	The password of the user specified as the Security Principal.	No	None

Name	Description	Required?	Default Value
Security Authentication	The method to use when authenticating to the LDAP server. Currently, only "simple" is supported.	No	simple
Class Name	The name of the class implementing the LDAP plug-in. Must be com.novell.ichain.oac.ldap.ParamListBuilder.	Yes	None

Using Third-Party Certificates

Novell iChain includes Novell Public Key Infrastructure Services (PKIS 2.0) to provide cryptography and enable certificate services in your iChain infrastructure. A Novell server certificate is installed and configured automatically when you install Novell iChain; however, you may want to use other third-party certificates, such as Baltimore* certificates, in your infrastructure. In order to use third-party certificates in your iChain infrastructure, you must request a certificate from a Certificate Authority (CA), have the CA sign the certificate, collect and export the certificate and its trusted root, and then import the certificate and its trusted root to the iChain Internet caching server. The following procedure describes the process for a Baltimore certificate.

To create a Certificate Signing Request (CSR) for a server certificate for the iChain Internet caching server, perform the following steps:

1. From ConsoleOne, access the tree containing the iChain Internet caching server.
2. From within the server's OU, click New Object > NDSPKI:KeyMaterial.
3. Specify a name for the certificate, such as ICS certificate.
4. Select Custom > click Next.
5. Select External Certificate Authority.
6. Select the key size > click Next.
7. Click Next to accept the default subject name of the certificate.
8. Click Finish.

9. Click Save to save the CSR.

To sign the CSR, perform the following steps:

1. Copy the CSR onto a diskette.
2. Insert the diskette with the CSR into the drive of the Baltimore Certificate Authority.
3. From the Registration Authority Operatory (RAO) menu of the Baltimore CA, click Face to Face Requests > Register a New User.
4. Select the Baltimore policy you have previously created for the ICS server certificate.
5. Locate the CSR file on the diskette > select the file > click Open.
6. Click Accept to process the CSR.

To collect response to the CSR and export the trusted root, perform the following steps:

1. Click Collect Reply from Last Request.
2. Click File.
3. Click DER Encoded Certificate.
4. Save the response file to the diskette as a .DER file.
5. Click OK to acknowledge.
6. Click OK on the yellow back arrow.
7. From the Certificate Authority Operator (CAO) menu, click Open/Create PKI.
8. Right-click the CA object > click Export Certificate.
9. Click DER Encoded Certificate.
10. Save the Trusted Root file to the diskette as a .DER file.
11. Select PKI > Done.

To import the new Trusted Root into the server certificate, perform the following steps:

1. From ConsoleOne, right-click on the server certificate object you created when you made the certificate request and click Properties.
2. Click Certificates > Trusted Root > Import > Read From File.
3. Select the Trusted Root file on the diskette > click Open > Next.

4. Click Read From File.
5. Select the certificate response file from the diskette > click Open.
6. Click Finish.

Enabling Debugging Messages for Access Control

If you need to run the iChain Internet caching server with the debugging option, complete the following steps:

1. On the ICS server on which you installed iChain, edit the APPSTART.NCF file.
2. Find the line containing the LOAD ACLCHECK command > add a \D4 switch to the line as follows:

   LOAD ACLCHECK /D4
3. Shut down and restart the ICS server.

5 Managing iChain

Novell® iChain™ includes a variety of logging tools that allow you to view information about access rule, news server, mail server, and Web server activity to help you manage your infrastructure. This chapter describes the following iChain management procedures to enable and view logging information:

Enabling and Viewing the ACL Rule Log File

The logs for authorized access attempts and unauthorized access attempts can be turned on or off globally. The rules for logging authorized access attempts for an individual access control list (ACL) can also be turned on or off. However, because unauthorized access attempts are usually the result of a user not being defined in any ACL rule, the logging of unauthorized access attempts cannot be turned on or off for individual ACL rules. To enable ACL rule logging, complete the following steps:

To enable or disable ACL rule logging on a global level, complete the following steps:

1. Access the URL of the ICS server on which you installed the iChain Internet caching server software to launch the ICS browser-based adminstration tool.

   For example, http://*xx.xx.xx.xx*:1959/appliance/config.html

   where *xx.xx.xx.xx* is the IP address.

2. Click Cache > Web Server Accelerator > Modify.

3. Check the Enable Logging check box.

To enable or disable ACL rule logging for an individual ACL rule, complete the following steps:

1. From ConsoleOne™, right-click an ACL Rule object.
2. Select Properties.
3. Check the Authorized Logging check box.

The ACL log files for each 24-hour period are saved to SYS:\ETC\ACLLOG*yymmdd-a*.LOG where *yymmdd* is today's date represented by two digits for the year, month and date. The default maximum size of the file is 1 MB. Each file contains the following fields:

- Date
- Time
- Source IP address
- Destination IP address
- Protocol
- Source port
- Destination port
- TCP flag
- Access (Allow=1, Deny=0)
- IP headers
- IP payload
- Username
- Destination host name
- URL (the user requested)
- Rule Object name (if access was allowed; if denied field displays a —)

Enabling and Viewing the Mail Server Log File

To enable logging for the mail server, complete the following steps:

1. Access the URL of the ICS server on which you installed the iChain upgrades to launch the ICS browser-based adminstration tool.

For example, http://*xx.xx.xx.xx*:1959/appliance/config.html

where *xx.xx.xx.xx* is the IP address.

2. Click Cache > Web Server Accelerator > Modify.
3. Select Mail Server.
4. Check the Enable Indexed Format Logging check box.

The mail server log file contains the following fields:

- Keyword—MAIL. If the Combine Log Files option was selected, the keyword is at the beginning of each mail proxy audit log line.
- Date.
- Time.
- Source IP address.
- Destination IP address.
- User—Typeless NDS® name or IP address of user.
- Protocol—Simple Mail Transfer Protocol (SMTP) or Post Office Protocol 3 (POP3).
- Status of the SMTP or POP3 request, such as Success, ACL check failure, Spool creation error, Failed connection, Spool size limitation, Protocol and transport failure, and Resource allocation failure.
- Command—SMTP or POP3 command used.
- Source domain—Domain Name System (DNS) domain name (for SMTP use only).
- Recipients—First 256 bytes of a comma-separated list in user@domain format (for SMTP use only).
- Process step—Examples of process steps, include Incoming, Spool processing, and Forwarding (for SMTP use only).

A iChain Internet Caching Server

The Novell® iChain™ Internet Caching 1.5 Server is a key component of the iChain infrastructure. The 1.5 release is a specialized and customized release of the Novell Internet Caching System™ (ICS) server based on the 1.3 and 2.0 releases.

The iChain Internet caching server is optimized to perform the functions needed for your iChain infrastructure. Because of this, the release supports some new features specific to iChain and no longer supports some features in the ICS 1.3 release.

NOTE: For more general information on the functionality and configuration of your ICS server, refer to the *Internet Caching System Administration Guide* located at:

http://www.novell.com/documentation/lg/ics13/docui/index.html

Although the majority of the ICS 1.3 release documentation is accurate and should be consulted as a resource to implement your iChain infrastructure, you should note that there are some feature differences between the two releases and how these differences are reflected in the administration interfaces of the two releases.

iChain Internet Caching Server Features

The iChain Internet Caching 1.5 server supports the following new features:

- iChain specific access control
- iChain specific authentication
- SSLizer support
- iChain access control rule hit logging
- Customized cache control header

The iChain Internet Caching 1.5 server supports the following ICS 1.3 features:

- Accelerating origin Web servers (including multi-homed servers)
- Accelerating proxy servers (both CERN and ICP hierarchies)
- Failover protection (through the use of Web server accelerator groups or appliance clusters)
- Appliance logging capabilities
- Appliance routing capabilities
- Time synchronization
- Performing appliance upgrades
- Saving and restoring configurations
- Restarting the appliance
- Restoring factory settings

The iChain Internet Caching 1.5 Server does not support the following ICS 1.3 features:

- Accelerating browsers
- Accelerating FTP requests
- Appliance-based filtering
- Configuring appliance-generated error messages for the browser

iChain Internet Caching Server Browser-Based Administration Tool

Just as with the ICS server, the iChain Internet caching server supports a browser-based administration utility allowing you to manage and administer your iChain Internet caching server from a browser. To launch the utility, you simply access a special management URL on the iChain Internet caching server. The URL must contain either the 10-net management address or the IP address you have already configured for the server, followed by :1959/appliance/config.html. For example,

http://10.1.1.1:1959/appliance/config.html

NOTE: If the ICS server is located behind a firewall and you are accessing the ICS browser-based administration utility from a browser outside that firewall, you must open ports 1959, 2222, and 1100 on the firewall to administrer the ICS server.

Because the iChain release does not support all the features of the standard ICS server, the following tabs have been removed:

- Cache > Client Accelerator
- Cache > Transparent Handling
- Cache > FTP
- Cache > Filtering
- Cache > Mini-Web

To support the various new features of iChain, some tab pages and dialog boxes of the iChain Internet caching server browser-based administration utility have been modified. Details of the changes are described in the following topics:

- "Access Control Tab" on page 55
- "Web Server Accelerator Dialog Box" on page 56
- "Authentication Dialog Box" on page 62

Access Control Tab

Path: Cache > Access Control

The Access Control tab is a new tab specific to iChain that lets you define the parameters necessary to setup iChain access control. Since iChain mechanisms control all access to the ICS appliance and iChain infrastructure, this tab replaces the System > IP Access Control tab. This tab includes the following fields:

ISO Object Name: Specifies the name of the iChain Service Object (ISO) containing parameter settings defining your iChain dormain or infrastructure.

LDAP Guest User Name: Defines the username for a guest account on the LDAP server

Password Management Servlet: Defines the URL of the password management servlet on the ICS server. This servlet enables users within your iChain infrastructure to change their passwords. The password management servlet is installed at /ichain/PasswordExpired.html on the ICS server. The

URL for this servlet should be defined as http://*x.x.x.x*/ichain/PasswordExpired.html where *x.x.x.x* is the IP address of the ICS server.

LDAP Server Address: Specifies the IP address of the iChain LDAP access control server.

LDAP Listening Port: Specifies the port on which the LDAP server will listen for access control requests. The default is 389.

LDAP User Name: Specifies the LDAP username for the ICS server to use when making requests for access control information from the iChain LDAP access control server.

LDAP Password: Specifies the LDAP password for the ICS server to use when making requests for access control information from the iChain LDAP access control server.

Enable Authorized Access Rule Hit Logging: Specifies whether to enable logging for authorized access rule hits. The default is disabled.

Enable Unauthorized Access Rule Hit Logging: Specifies whether to enable logging for unauthorized access rule hits. The default is disabled.

Refresh ACLCheck: Refreshes the ACLCheck parameter settings.

Web Server Accelerator Dialog Box

Path: Cache > Web Server Accelerator > Insert

The Web Server Accelerator dialog box lets you create Web server accelerator services for handling requests to Web servers. This dialog includes the following fields:

Enable This Accelerator: Specifies whether to enable the defined Web server accelerator after you have configured it. The default is Enabled.

Name: Each Web server accelerator service requires a name you create. For example, you can select a name that indicates which Web server is being serviced by the appliance, or alternately, a set of browsers configured to access the Web server accelerator as a proxy server. A valid name consists of a DOS-style, eight-character name with no special characters or spaces.

If logging is enabled, the appliance uses the Web server accelerator name as the directory name that the log for the Web server accelerator is kept in.

DNS Name: If you are accelerating multiple Web servers on the same IP address, you must create a Web server accelerator definition for each DNS name that is used in browser requests. This name must exactly match one of the names in the requests. If your infrastructure supports multi-homing, refer to "Appliance Multihoming Support" in the *Internet Caching System Administration Guide* for further information.

This name is also used if the appliance is part of an ICP hierarchy that needs to resolve relative URLs.

Enable Logging for This Accelerator: Causes a log file to be kept for this Web server accelerator.

Accelerator Proxy Port: The port number that the proxy server is listening on for incoming connections. The default for HTTP is 80 (1 – 65535).

Web Server Port: The port number that the origin Web server is listening on for incoming connections. The default for HTTP is 80 (1 – 65535).

Web Server Addresses: The IP address or local DNS name of each Web server from which the appliance fills the cache for this Web server accelerator. The appliance must be able to fill all requests through any of these names or addresses.

Accelerator IP Addresses: The appliance's IP addresses to which DNS resolves the Web server's DNS name and on which the Web server accelerator listens for incoming connections from the Internet.

Use Host Name Sent By Browser (Multi-homing Web Server): Checking this option preserves the host name in the HTTP header exactly as it came in the browser request.

Alternate Hostname: Checking this option causes the specified string to be substituted for the host name in the HTTP header before the request is forwarded to the Web server.

Return Error If Hostname Sent by Browser Does Not Match Accelerator DNS Host Name: Checking this option causes the system to match the host name in the DNS header that came from the browser against the DNS name specified in this accelerator definition. If the names don't match, the request is not forwarded to the Web server. Instead, the system returns an error to the requesting browser.

Enable X-Forward-For: X-Forwarded-For headers are used to pass browser ID information along with browser request packets. If the headers are

included, Web servers can determine the origin of browser requests they receive. If the headers are not included, browser requests have anonymity.

Checking the X-Forward-For option causes the appliance to either add information to an existing X-Forwarded-For or Forwarded-For header, or to create a header if one doesn't already exist.

Leaving the option unchecked causes the appliance to remove X-Forwarded-For headers from any Web accelerator requests passing through the appliance.

Deciding whether to check the option requires that you weigh the desires of browser users to remain anonymous against the desires of Web server owners (e-commerce sites, for example) to collect data about who is accessing their site.

Enable Authentication: Checking this box causes the appliance to require authentication of users wanting to use its Web server accelerator services. Clicking Authentication displays the Add Authentication Profiles dialog box. For more information, refer to "Add Authentication Profiles Dialog Box" in the *Internet Caching System Administration Guide*.

IMPORTANT: The system requires that each service (including authentication) use a unique IP address and port combination. The default authentication port is 443. Attempts to enable authentication for more than one service on the same IP address and port will result in a TCP bind error.

Enable SSLizer: Specifies whether to enable the SSLizer. The default is disabled.

SSL Port: Specifies the port that will be used for SSLized communication.

SSL Key ID: Specifies the Key ID to establish the SSL session.

Enable Path-Based Multi-Homing: This option is enabled only when you have created another accelerator definition and have not created a standard multi-homing relationship between previously defined accelerators on the ICS appliance. In other words, you don't have multiple accelerators sharing the same accelerator IP address and port.

Path-based multi-homing lets you configure the system so that a multi-homing master accelerator fills general requests from a site's main Web server and routes specific requests to specialized child accelerators that fill from other specialized Web servers. This option lets you create child accelerators for path-based multi-homing configurations.

When you enable path-based multi-homing for the accelerator you are defining, you must also click the Path Based Options button and specify a path

rule that the multi-homing master can use to route traffic to the accelerator you are defining. For more information, see “Path Rule Options Dialog Box” on page 59.

You will also notice that if you have created multiple accelerators that can function as multi-homing masters, when you select a name in the Multi-Homing Master drop-down list, the DNS Name, Accelerator Proxy Port, and Accelerator IP Addresses selections dynamically change to match the accelerator whose name you have selected.

Multi-Homing Master: This drop-down list contains the names of accelerators you have defined that can function as multi-homing masters, meaning they are not configured as child accelerators to other multi-homing masters.

Enable Custom Cache Control Header: Checking this option and clicking Header Options provides access to the Custom Cache Control Header dialog box. For more information, see “Custom Cache Control Header Options Dialog Box” on page 60.

Path Rule Options Dialog Box

Path: Cache > Web Server Accelerator > Insert > Enable Path-Based Multi-Homing > Path Rule Options

This dialog box lets you specify a string which, if present in the browser request, will cause the multi-homing master accelerator to route the request to the child accelerator being defined.

The string match can occur immediately following the DNS name (the Starts With option) or at the end of the URL (the Ends With option).

Sub-Path Match String: This is the string the multi-homing master will compare against the browser request. If the string is not found, the multi-homing master accelerator attempts to file the request through the Web server addresses in its accelerator definition. If the string is found, the multi-homing master accelerator routes the request to the accelerator with the matching string.

Starts With: Checking this option indicates to the multi-homing master that the Sub-Path Match String field contains a path that might immediately follow the DNS name in the browser request. If the string matches the path in the request, the multi-homing master forwards the request to the child accelerator being defined. The initial forward slash (/) is not required in the string because the system automatically assumes it is in the request.

Remove Sub-Path from URL: You should check this option if the path string doesn't actually appear at the root of the Web server. If this option is checked, the string is stripped from the request before the request is sent to the Web server. This probably indicates the the object is at the root of the Web server. If this option is not checked, the matched string is retained in the request sent to the Web server.

Ends With: Checking this option indicates to the multi-homing master accelerator that the Sub-Path Match String field contains a file extension, such as gif, jpg, mpg, cgi, etc. If a match is found at the end of the browser request, the multi-homing master will route the request to the child accelerator being defined.

Custom Cache Control Header Options Dialog Box

Path: Cache > Web Server Accelerator > Insert > Enable Custom Cache Control Header > Header Options

This dialog box lets you specify object headers that the appliance recognizes as overriding standard HTTP cache directives.

Only the accelerator service containing the custom header definition follows the cache policies specified in the custom headers. All other caches, including the requesting browsers and external proxy caches do not recognize the custom headers and therefore follow the cache policy specified by the standard cache control headers.

This means that you have the following options:

- You can specify that browsers and/or external caches cannot cache the objects. This lets you offload request processing from the origin Web server while still requiring that users return to the site each time they request an object.
- You can also specify separate cache times for browsers, external caches, and the accelerator you are defining.

Implementing Custom Cache Headers

To implement custom cache headers, you must do the following:

- Enter a header string in the Custom Cache Control Header dialog box—for example, MYCACHE.

- Configure the Web server to send an HTTP header containing the defined string and the time in seconds that the object should be retained in the cache—for example, MYCACHE:60.

  If the number is non-zero, the system treats the reply as if it had the following headers:

  Cache-Control:public

  Cache-Control:max-age=*number*

  If the number is zero (0), the system treats the reply as if it had the following header:

  Cache-Control:no-cache

- Ensure that the Web server continues to send standard HTTP cache-control headers so that browsers and external caches follow the caching policies you intend them to.

  For example, you could

  - Use an Expires or Cache-Control: max-age header to specify that browsers should cache an object for two minutes
  - Use a Cache-Control:private header to prevent external caches form caching the object at all
  - Use a custom cache control header such as MYCACHE: 1800 to indicate that the accelerator should cache the object for 30 minutes.

Custom Cache Control Headers override the following standard HTTP cache-control headers on the appliance but do not affect how browsers and external caches respond to them:

- Cache-Control: no-store
- Cache-Control: no-cache
- Cache-Control: max-age=*number*
- Cache-Control: private
- Cache-Control: public
- Pragma: no-cache
- Expires: *date*

For example, you do the following:

1. While configuring a Web server, accelerator service, you insert a string in the Custom Cache Control Header list with the value of FOOTTL. The appliance will now recognize FOOTTL as a custom cache control header on objects requested through the service you are configuring.
2. You then configure the accelerated Web server to send FOOTTL: 60 in the headers of objects you want to be cached at the appliance. The appliance will recognize this header as overriding the standard HTTP cache-control headers listed above when objects are requested through the accelerator service you are configuring.
3. Finally, you ensure that the Web server continues to send standard HTTP cache-control headers that prevent browsers and external caches from caching objects. This will ensure that users reture to the Web site for each object request.

When your Web server sends an object with the FOOTTL header in response to an appliance request made through the accelerator service, your appliance recognizes the header and caches the object. Requesting browsers and external caches do not recognize the header, and therefore, they do not cache the object.

Thus, the appliance offloads a processing burden from the Web server by caching the frequently-requested object for at least 60 seconds (the value you specified in Step 2). Browsers and other caches, on the other hand, must always access the appliance to get the object, which is kept fresh in the appliance's cache due to its relatively brief time-to-live value.

Authentication Dialog Box

Path: Cache > Authentication > Insert under the Authentication Profiles list.

The Authentication dialog box lets you assign an authentication profile name and specify NDS Authentication, Background SSL Mutual Authentication, LDAP Authentication, or RADIUS Authentication as the authentication source.

IMPORTANT: The system doesn't recognize case differences in profile names. MyProfile and myprofile are, effectively, the same profile name.

Also, the system partially overwrites and concatenates previously created profiles without warning if a duplicate name is used. Therefore, if you create a profile named MyProfile and later create another profile named myprofile, the system will remove the first name, concatenate parts the first profile with the second, and use the second name.

To avoid these problems, ensure that each profile has a unique name.

After selecting the authentication source, you must configure the source by clicking its respective Options button. See “LDAP Options Dialog Box” on page 63 for information on configuring LDAP authentication options.

LDAP Options Dialog Box

Path: Cache > Authentication > Insert > LDAP Authentication > LDAP Options

iChain uses LDAP authentication. Use the LDAP Options dialog box to configure the appliance for having users authenticate through an LDAP database.

LDAP Server Address: The DNS name or IP address of the LDAP server.

LDAP Server Listening Port: The port number the LDAP server is listening on for requests from LDAP clients. The default is 389 for normal access. Use 636 for secure access.

Enable Secure Access to LDAP Server: Checking this box causes the data sent between the LDAP client and the LDAP server to be sent using SSL.

LDAP Server Trusted Root File: The path on the appliance to a trusted root file that contains the Certificate Authority (CA) used by the LDAP server in the profile you are creating.

The system fills this field with information for the trusted root file you create using the Import Trusted Root button. See the instructions found in "Import Trusted Root Dialog Box" in the *Internet Caching System Administration Guide* for further information.

If the LDAP server uses a CA for which you have previously created a trusted root file, you can manually type the path and filename in this field. For example, you might be using the same LDAP server for multiple authentication profiles.

LDAP Login Name Format: Refer to “LDAP Login Name Format” on page 63 for information on the fields in this box.

Import Trusted Root: Clicking this button opens the Import Trusted Root dialog box (see "Import Trusted Root Dialog Box" in the *Internet Caching System Administration Guide* for further information).

LDAP Login Name Format

The contents of this box change depending on the option selected.

Use User's E-Mail

Select this option to have users log in using their e-mail name field in the LDAP database. You must provide one of more contexts in which the LDAP server will search for the e-mail name.

This option is somewhat redundant with Use Field Name because the e-mail name is simply an LDAP field name. E-mail is offered separately because it is used so often.

LDAP Search Base: Click Insert to enter the context of one or more LDAP containers from which the search for the e-mail name should begin.

You must also provide authentication information for the appliance to access the LDAP server using one of the following two options:

Use Anonymous Bind for LDAP Search: Select this option if the appliance can authenticate to the LDAP server using anonymous bind.

Use User Name / Password Bind for LDAP Search: Select this option if anonymous bind is not enabled on the LDAP server > enter the username and password pair through which the appliance authenticates to use the LDAP server's authentication services.

Use Distinguished Name

Select this option to allow users to authenticate using their LDAP usernames. Users can use either their fully distinguished (full LDAP contexts) LDAP usernames, or you can provide a list of LDAP contexts so users only need to type their usernames.

LDAP Contexts: This list contains specific contexts in which the LDAP server will look for usernames. This provides a shortcut to authentication of users by allowing them to type only their LDAP usernames.

The appliance searches each context until it either locates the name or exhausts the search. If duplicate names exist in different context, the appliance searches until the correct name / password match is found.

Use Field Name

Select this option to require that users enter a specific LDAP field name.

Field Name: The LDAP field name (such as User ID) through which users can authenticate.

LDAP Search Base: Click Insert to enter the context of one or more LDAP containers. The appliance will perform a sub-tree search in all containers in the list and in their sub-containers.

Use Anonymous Bind for LDAP Search: Select this option if the appliance can authenticate to the LDAP server using anonymous bind.

Use User Name / Password Bind for LDAP Search: Select this option if anonymous bind is not enabled on the LDAP server > enter the username and password pair through which the appliance authenticates to use the LDAP server's authentication services.