

DNSDHCP Administration Guide for Linux

Novell® Open Enterprise Server

2SP1

December, 2008

www.novell.com



Legal Notices

Novell, Inc., makes no representations or warranties with respect to the contents or use of this documentation, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc., reserves the right to revise this publication and to make changes to its content, at any time, without obligation to notify any person or entity of such revisions or changes.

Further, Novell, Inc., makes no representations or warranties with respect to any software, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc., reserves the right to make changes to any and all parts of Novell software, at any time, without any obligation to notify any person or entity of such changes.

Any products or technical information provided under this Agreement may be subject to U.S. export controls and the trade laws of other countries. You agree to comply with all export control regulations and to obtain any required licenses or classification to export, re-export or import deliverables. You agree not to export or re-export to entities on the current U.S. export exclusion lists or to any embargoed or terrorist countries as specified in the U.S. export laws. You agree to not use deliverables for prohibited nuclear, missile, or chemical biological weaponry end uses. See the [Novell International Trade Services Web page \(http://www.novell.com/info/exports/\)](http://www.novell.com/info/exports/) for more information on exporting Novell software. Novell assumes no responsibility for your failure to obtain any necessary export approvals.

Copyright © 2008 Novell, Inc. All rights reserved. No part of this publication may be reproduced, photocopied, stored on a retrieval system, or transmitted without the express written consent of the publisher.

Novell, Inc., has intellectual property rights relating to technology embodied in the product that is described in this document. In particular, and without limitation, these intellectual property rights may include one or more of the U.S. patents listed on the [Novell Legal Patents Web page \(http://www.novell.com/company/legal/patents/\)](http://www.novell.com/company/legal/patents/) and one or more additional patents or pending patent applications in the U.S. and in other countries.

Novell, Inc.
404 Wyman Street, Suite 500
Waltham, MA 02451
U.S.A.
www.novell.com

Online Documentation: To access the latest online documentation for this and other Novell products, see the [Novell Documentation Web page \(http://www.novell.com/documentation/\)](http://www.novell.com/documentation/)

Novell Trademarks

For Novell trademarks, see [the Novell Trademark and Service Mark list \(http://www.novell.com/company/legal/trademarks/tmlist.html\)](http://www.novell.com/company/legal/trademarks/tmlist.html)

Third-Party Materials

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

Contents

| | |
|---|-----------|
| About This Guide | 11 |
| 1 Understanding DNS and DHCP Services | 13 |
| 1.1 DNS | 13 |
| 1.1.1 DNS Hierarchy | 14 |
| 1.1.2 DNS Name Service | 17 |
| 1.1.3 Resource Records | 20 |
| 1.1.4 DNS Structure | 21 |
| 1.2 Novell DNS Service | 24 |
| 1.2.1 Novell eDirectory and DNS | 25 |
| 1.2.2 eDirectory Schema Extensions for DNS | 27 |
| 1.2.3 Dynamic DNS | 29 |
| 1.2.4 Zone Transfer | 31 |
| 1.2.5 Dynamic Reconfiguration | 31 |
| 1.2.6 Fault Tolerance | 32 |
| 1.2.7 Cluster Support | 32 |
| 1.2.8 Notify | 32 |
| 1.2.9 Load Balancing | 33 |
| 1.2.10 Forwarding | 33 |
| 1.2.11 No-Forwarding | 34 |
| 1.2.12 Benefits of Integrating a DNS Server with eDirectory | 34 |
| 1.2.13 rndc Support | 35 |
| 1.2.14 Managing DNS Objects | 35 |
| 1.3 DHCP | 41 |
| 1.3.1 DHCP and BOOTP | 41 |
| 1.3.2 IP Address Allocation | 42 |
| 1.3.3 Virtual LAN Environments | 44 |
| 1.4 Novell DHCP Service | 44 |
| 1.4.1 DHCP Options | 45 |
| 1.4.2 eDirectory Objects for DHCP | 47 |
| 1.4.3 Managing DHCP Objects | 50 |
| 1.4.4 Java Management Console for DHCP (OES Linux) | 51 |
| 1.5 What's Next | 54 |
| 2 What's New in DNS/DHCP | 55 |
| 2.1 What's New in DHCP | 55 |
| 2.2 What's New in DNS | 55 |
| 2.3 What's Next | 55 |
| 3 Planning Your DNS/DHCP Implementation | 57 |
| 3.1 Resource Requirements | 57 |
| 3.2 eDirectory Permissions | 57 |
| 3.3 Planning a DNS Strategy | 58 |

| | | |
|----------|---|-----------|
| 3.3.1 | Planning Zones | 59 |
| 3.3.2 | Using the Novell DNS Server as a Primary Name Server | 59 |
| 3.3.3 | Using the Novell DNS Server as a Secondary Name Server | 59 |
| 3.3.4 | Configuring a DNS Server to Forward Requests | 59 |
| 3.3.5 | Setting Up the Forward Zone Type | 60 |
| 3.3.6 | Setting Up the in-addr.arpa Zone | 60 |
| 3.3.7 | Registering Your DNS Server with Root Servers | 61 |
| 3.4 | Planning a DHCP Strategy | 61 |
| 3.4.1 | Network Topology | 62 |
| 3.4.2 | eDirectory Implementation | 62 |
| 3.4.3 | Lease Considerations | 62 |
| 3.4.4 | IP Address Availability | 65 |
| 3.4.5 | Hostnames | 65 |
| 3.5 | What's Next | 66 |
| 4 | Running DNS/DHCP Services in a Virtualized Environment | 67 |
| 5 | Comparing Linux and NetWare | 69 |
| 5.1 | DNS on NetWare versus DNS on Linux | 69 |
| 5.1.1 | Management Interface | 69 |
| 5.1.2 | Commands | 69 |
| 5.1.3 | Filenames and Paths | 70 |
| 5.1.4 | Installation Difference | 71 |
| 5.1.5 | Features Not Supported | 71 |
| 5.2 | DHCP on NetWare versus DHCP on Linux | 71 |
| 5.2.1 | Management Interface | 71 |
| 5.2.2 | Commands | 71 |
| 5.2.3 | Filenames and Paths | 72 |
| 5.2.4 | Features Not Supported | 72 |
| 5.3 | Limitations | 72 |
| 5.4 | What's Next | 72 |
| 6 | Installing and Configuring DHCP | 73 |
| 6.1 | Planning Your Installation | 73 |
| 6.1.1 | Prerequisites | 73 |
| 6.1.2 | eDirectory Permissions | 73 |
| 6.1.3 | Recommendations | 74 |
| 6.2 | Installing DHCP | 75 |
| 6.3 | Verifying the Installation | 79 |
| 6.4 | Upgrading to OES 2 Linux from OES 1 Linux | 80 |
| 6.5 | What's Next | 80 |
| 7 | Administering and Managing DHCP | 81 |
| 7.1 | Using iManager to Manage DHCP | 81 |
| 7.1.1 | Scope Setting | 81 |
| 7.1.2 | Service Management | 82 |
| 7.1.3 | Server Management | 83 |
| 7.1.4 | Shared Network Management | 85 |
| 7.1.5 | Subnet Management | 86 |
| 7.1.6 | Pool Management | 88 |
| 7.1.7 | Host Management | 90 |
| 7.1.8 | Class Management | 92 |
| 7.1.9 | Zone Management | 93 |

| | | |
|-----------|--|------------|
| 7.1.10 | TSIG Key Management | 95 |
| 7.1.11 | Failover Peer Management | 96 |
| 7.1.12 | Importing and Exporting the DHCP Configuration | 98 |
| 7.2 | Using the Java Management Console for DHCP (OES Linux) | 99 |
| 7.2.1 | Service Management | 99 |
| 7.2.2 | Server Management | 100 |
| 7.2.3 | Shared Network Management | 100 |
| 7.2.4 | Subnet Management | 100 |
| 7.2.5 | Pool Management | 101 |
| 7.2.6 | Host Management | 102 |
| 7.2.7 | Class Management | 102 |
| 7.2.8 | Zone Management | 103 |
| 7.2.9 | TSIG Key Management | 103 |
| 7.2.10 | Failover Peer Management | 104 |
| 7.2.11 | Managing DHCP (OES Linux) Objects in the Java Management Console | 105 |
| 7.2.12 | Importing and Exporting the DHCP Configuration | 107 |
| 7.3 | Starting the DHCP Server | 108 |
| 7.4 | Stopping the DHCP Server | 108 |
| 7.5 | What's Next | 108 |
| 8 | Configuring DHCP with Novell Cluster Services for the NSS File System | 109 |
| 8.1 | Benefits of Configuring DHCP for High Availability | 109 |
| 8.2 | Prerequisites | 109 |
| 8.3 | Installation and Configuration | 109 |
| 8.3.1 | Prerequisites | 110 |
| 8.3.2 | Verifying the Novell Cluster Services Setup | 110 |
| 8.3.3 | Installing and Configuring a Cluster | 110 |
| 8.3.4 | DHCP Load and Unload Scripts | 112 |
| 8.4 | Loading and Unloading the DHCP Server | 113 |
| 8.4.1 | Loading the DHCP Server | 113 |
| 8.4.2 | Unloading the DHCP Server | 114 |
| 8.5 | What's Next | 114 |
| 9 | Configuring DHCP with Novell Cluster Services for the Linux File System | 115 |
| 9.1 | Benefits of Configuring DHCP for High Availability | 115 |
| 9.2 | DHCP Installation and Configuration | 115 |
| 9.2.1 | Prerequisites | 115 |
| 9.2.2 | Configuring DHCP on the Shared Disk | 116 |
| 9.2.3 | Configuring the dhcpd.conf File | 116 |
| 9.2.4 | Creating a dhcpd.leases File | 116 |
| 9.2.5 | Novell Cluster Services Configuration and Setup | 117 |
| 9.3 | What's Next | 121 |
| 10 | Security Guidelines for DHCP | 123 |
| 10.1 | Best Practices | 123 |
| 10.2 | What's Next | 123 |
| 11 | Installing and Configuring DNS | 125 |
| 11.1 | Planning the Installation | 125 |
| 11.1.1 | Prerequisites | 125 |
| 11.1.2 | eDirectory Permissions | 125 |
| 11.2 | Installing the DNS Server | 127 |

| | | |
|-----------|--|------------|
| 11.3 | Setting Runtime Credentials | 129 |
| 11.4 | Verifying Installation | 129 |
| 11.5 | Additional Information | 130 |
| 11.6 | What's Next | 130 |
| 12 | Migrating DNS/DHCP from OES 1 NetWare to OES 2 SP1 Linux | 131 |
| 13 | Administering and Managing a DNS Server | 133 |
| 13.1 | Using iManager to Manage DNS | 133 |
| 13.1.1 | Scope Settings | 133 |
| 13.1.2 | DNS Server Management | 134 |
| 13.1.3 | Zone Management | 138 |
| 13.1.4 | Resource Record Management | 146 |
| 13.1.5 | DNS Key Management | 149 |
| 13.2 | Using the Java Management Console to Manage DNS | 150 |
| 13.2.1 | Installing the Java Management Console | 151 |
| 13.2.2 | DNS Server Management | 151 |
| 13.2.3 | Zone Management | 154 |
| 13.2.4 | Resource Record Management | 160 |
| 13.2.5 | DNS Key Management | 161 |
| 13.3 | Configuring Roles for a Novell DNS Server | 163 |
| 13.3.1 | Configuring a DNS Server to Forward Queries to Root Name Servers | 164 |
| 13.3.2 | Configuring a DNS Server as a Cache-Only Server | 164 |
| 13.3.3 | Configuring Child (Sub) Zone Support | 164 |
| 13.3.4 | Configuring a Multi-Homed Server | 165 |
| 13.3.5 | Configuring Dynamic DNS | 165 |
| 13.4 | novell-named Command Line Options | 165 |
| 13.4.1 | Description of Command Line Options | 166 |
| 13.5 | Starting the DNS Server | 167 |
| 13.6 | Running DNS Server in chroot Mode | 167 |
| 13.7 | Running DNS Server as a Non-Root User | 168 |
| 13.8 | Stopping the DNS Server | 168 |
| 13.9 | What's Next | 168 |
| 14 | Configuring DNS with Novell Cluster Services | 169 |
| 14.1 | Prerequisites | 169 |
| 14.2 | Installation and Configuration | 169 |
| 14.2.1 | Verifying the Novell Cluster Services Setup | 169 |
| 14.2.2 | Installing and Configuring a Cluster | 170 |
| 14.2.3 | DNS Load and Unload Scripts | 172 |
| 14.3 | Loading and Unloading the DNS Server | 173 |
| 14.3.1 | Loading the DNS Server | 173 |
| 14.3.2 | Unloading the DNS Server | 173 |
| 14.4 | What's Next | 174 |
| 15 | Security Considerations for DNS | 175 |
| 15.1 | Logging | 175 |
| 15.2 | Commands | 175 |
| 15.3 | Cryptographic Algorithms | 175 |
| 15.4 | Best Practices | 175 |
| 15.5 | What's Next | 176 |

| | |
|--|------------|
| 16 DNS/DHCP Advanced Features | 177 |
| 16.1 Configuring the ICE Zone Handler | 177 |
| 16.1.1 Modifying the ice.conf File | 177 |
| 16.1.2 Enabling Clear-Text Passwords | 177 |
| 16.1.3 Importing Configuration and Script Files | 178 |
| 16.1.4 Exporting Configuration and Script Information | 179 |
| 16.2 What's Next | 181 |
| 17 DNS-DSfW Integration | 183 |
| 17.1 Normal eDirectory with DNS | 183 |
| 17.2 DSfW with DNS | 183 |
| 17.2.1 Changes for DNS | 184 |
| 17.2.2 Local DNS Server Installation | 184 |
| 17.3 DSfW with Remote DNS (Child Domains) | 185 |
| 17.3.1 DSfW on a Remote DNS Server | 185 |
| 17.4 Scenarios | 185 |
| 17.5 FAQs | 186 |
| 17.6 What's Next | 186 |
| 18 Troubleshooting DNS and DHCP Services | 187 |
| 18.1 DHCP | 187 |
| 18.1.1 DHCP Server fails to load and records a "Cannot find host LDAP entry DHCP" error in the log file | 187 |
| 18.1.2 Installing an OES Server inside a container with a separate partition on an existing tree that already has DHCP Server installed on it results in a constraint violation error | 188 |
| 18.1.3 The dhcpd.log file is empty | 188 |
| 18.1.4 The DHCP server failed to Start | 188 |
| 18.1.5 The DHCP Server displays "Unknown Error" on the console | 189 |
| 18.1.6 Permission denied to DHCP server | 189 |
| 18.1.7 segfault dhcpd - You get an error "dhcpd: Can't create new lease file: Permission denied" and "dhcpd[8249]: segfault at 0000000000000000 rip 00002abbf999db7f rsp 00007fffb18ea5e0 error 4" | 190 |
| 18.2 DNS | 190 |
| 18.2.1 Failed to Configure DNS Server | 190 |
| 18.2.2 Insufficient Permissions for LDAP Admin User | 190 |
| 18.2.3 Failed to create the DNS Server object for the Virtual NCP Server | 191 |
| 18.2.4 novell-named Failed to Start | 191 |
| 18.2.5 rcnovell-named and rcnamed interfere in their individual status/stop query functionality | 191 |
| 18.2.6 The DNS Server failed to load and provides critical error messages for NWCallInit/NWCLXInit/NWNetInit | 192 |
| 18.2.7 Error message when you add RootServInfo that gives an undefined attribute | 192 |
| 18.2.8 Removal of DNS schema post usage of Remove Schema option of dns-maint | 192 |
| 18.3 What's Next | 192 |
| 19 Linux Notes | 193 |
| 19.1 DNS Notes | 193 |
| 19.2 What's Next | 193 |
| A Appendix | 195 |
| A.1 Supported RFCs | 195 |

| | | |
|----------|--|------------|
| A.2 | Types of Resource Records | 196 |
| A.3 | DHCP Option Descriptions | 198 |
| A.3.1 | Assigning Options | 204 |
| A.4 | DNS Root Servers | 205 |
| A.5 | DNS Server Configuration Utility | 206 |
| | dns-inst. | 207 |
| A.6 | DNS Server Maintenance Utility. | 208 |
| A.7 | DHCP Server Maintenance Utility | 212 |
| A.8 | Post-Install Maintenance Tools | 216 |
| B | Documentation Updates | 217 |
| B.1 | November, 2008. | 217 |
| B.2 | December, 2008. | 218 |
| B.3 | October, 2008 | 218 |
| B.4 | September, 2008 | 218 |
| B.5 | August, 2008 | 219 |
| B.6 | July, 2008. | 219 |
| B.7 | June, 2008 | 219 |
| B.8 | April, 2008 | 219 |
| B.9 | March, 2008 | 219 |
| C | Glossary | 221 |

About This Guide

This document describes the concepts of the Domain Naming System (DNS) and the Dynamic Host Configuration Protocol (DHCP), the setup and configuration of these functions, and how to use Novell® DNS and DHCP Services in Open Enterprise Server 2 Linux.

This guide is divided into the following sections:

- ♦ Chapter 1, “Understanding DNS and DHCP Services,” on page 13
- ♦ Chapter 2, “What’s New in DNS/DHCP,” on page 55
- ♦ Chapter 3, “Planning Your DNS/DHCP Implementation,” on page 57
- ♦ Chapter 4, “Running DNS/DHCP Services in a Virtualized Environment,” on page 67
- ♦ Chapter 5, “Comparing Linux and NetWare,” on page 69
- ♦ Chapter 6, “Installing and Configuring DHCP,” on page 73
- ♦ Chapter 7, “Administering and Managing DHCP,” on page 81
- ♦ Chapter 8, “Configuring DHCP with Novell Cluster Services for the NSS File System,” on page 109
- ♦ Chapter 9, “Configuring DHCP with Novell Cluster Services for the Linux File System,” on page 115
- ♦ Chapter 10, “Security Guidelines for DHCP,” on page 123
- ♦ Chapter 11, “Installing and Configuring DNS,” on page 125
- ♦ Chapter 12, “Migrating DNS/DHCP from OES 1 NetWare to OES 2 SP1 Linux,” on page 131
- ♦ Chapter 13, “Administering and Managing a DNS Server,” on page 133
- ♦ Chapter 14, “Configuring DNS with Novell Cluster Services,” on page 169
- ♦ Chapter 15, “Security Considerations for DNS,” on page 175
- ♦ Chapter 16, “DNS/DHCP Advanced Features,” on page 177
- ♦ Chapter 17, “DNS-DSfW Integration,” on page 183
- ♦ Chapter 18, “Troubleshooting DNS and DHCP Services,” on page 187
- ♦ Chapter 19, “Linux Notes,” on page 193
- ♦ Appendix A, “Appendix,” on page 195
- ♦ Appendix B, “Documentation Updates,” on page 217
- ♦ Appendix C, “Glossary,” on page 221

Audience

The audience for this document is network administrators. This documentation is not intended for users of the network.

Feedback

We want to hear your comments and suggestions about this manual and the other documentation included with this product. Please use the User Comments feature at the bottom of each page of the online documentation, or go to www.novell.com/documentation/feedback.html (<http://www.novell.com/documentation/feedback.html>) and enter your comments there.

Documentation Updates

For the most recent version of this guide, see the *DNS/DHCP Services for Linux Administration Guide for OES 2.0 SP 1* (http://www.novell.com/documentation/oes2/ntwk_dnshcp_lx/data/bookinfo.html#bookinfo).

Additional Documentation

For information about Novell iManager, see the *Novell iManager 2.7 Administration Guide* (http://www.novell.com/documentation/imanager27/index.html?page=/documentation/imanager27/imanager_admin_27/data/bsxrjzp.html#bsxrjzp).

Documentation Conventions

In Novell documentation, a greater-than symbol (>) is used to separate actions within a step and items in a cross-reference path.

In this documentation, a trademark symbol (®, ™, etc.) denotes a Novell trademark. An asterisk (*) denotes a third-party trademark.

Understanding DNS and DHCP Services

1

Novell DNS/DHCP Services in Open Enterprise Server (OES) 2 Linux integrates the Domain Name System (DNS) and Dynamic Host Configuration Protocol (DHCP) services into eDirectory™. Integrating these services into eDirectory provides centralized administration and enterprise-wide management of DNS and DHCP services.

DNS and DHCP services can be managed by either iManager, which is a Web-based management console, or you can use the Java® Management Console.

NOTE: A Novell DNS server can only be managed by using the iManager or Java management Console utility. The DNS YaST plug-in does not support managing a Novell DNS server.

iManager provides a Web interface to manage the objects created to support DNS and DHCP and runs in a browser window. It does not require a Novell Client™ or any installed component as a prerequisite. It shares a common interface with other utilities that are based on the iManager framework, and is thus tightly integrated with OES 2 Linux. iManager Plugin is supported on Internet Explorer® 6, Internet Explorer® 7, and Firefox.

For more detailed overview information, see the following:

- ♦ [Section 1.1, “DNS,” on page 13](#)
- ♦ [Section 1.2, “Novell DNS Service,” on page 24](#)
- ♦ [Section 1.3, “DHCP,” on page 41](#)
- ♦ [Section 1.4, “Novell DHCP Service,” on page 44](#)
- ♦ [Section 1.5, “What’s Next,” on page 54](#)

1.1 DNS

DNS is a distributed database system that provides hostname-to-IP resource mapping (usually the IP address) and other information for computers on a network. Any computer on the Internet can use a DNS server to locate any other computer on the Internet.

DNS is made up of two distinct components: the hierarchy and the name service. The DNS hierarchy specifies the structure, naming conventions, and delegation of authority in the DNS service. The DNS name service provides the actual name-to-address mapping mechanism.

For more information, see:

- ♦ [“DNS Hierarchy” on page 14](#)
- ♦ [“DNS Name Service” on page 17](#)
- ♦ [“DNS Resolver” on page 19](#)
- ♦ [“Resource Records” on page 20](#)
- ♦ [“DNS Structure” on page 21](#)

DNS/DHCP supports the standards of the Internet Request For Comments (RFCs). For more information, see [Appendix A, “Appendix,” on page 195](#).

1.1.1 DNS Hierarchy

DNS uses a hierarchy to manage its distributed database system. The DNS hierarchy, also called the domain namespace, is an inverted tree structure, much like eDirectory. Each node in the tree has a text label, which is zero to 63 characters long. The label (zero length) is reserved and is used for the root.

The DNS tree has a single domain at the top of the structure called the root domain. A period or dot (.) is the designation for the root domain. Below the root domain are the top-level domains that divide the DNS hierarchy into segments.

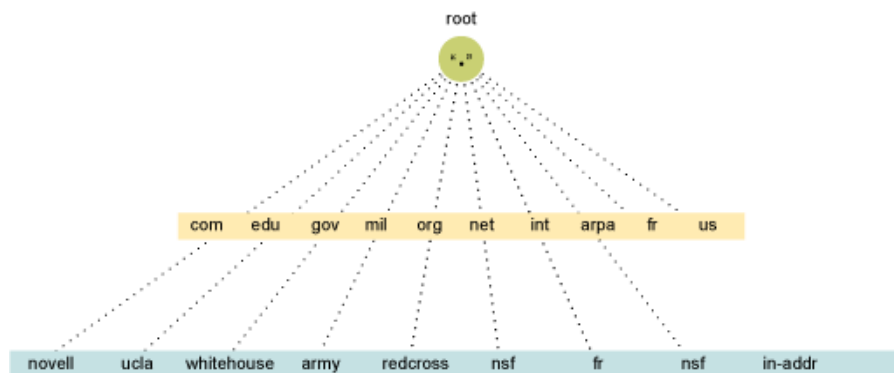
There are three types of top level domains (TLDs). Below the top-level domains, the domain namespace is further divided into subdomains representing individual organizations:

- ♦ “Generic TLD” on page 14
- ♦ “Country Code TLD” on page 15
- ♦ “Infrastructure TLD” on page 15
- ♦ “Domains and Subdomains” on page 15
- ♦ “Domain Names” on page 16
- ♦ “n-addr.arpa Domain” on page 16
- ♦ “Domain Delegation” on page 17

A list of TLDs is available at the [Internet Assigned Numbers Authority Web site \(http://www.iana.org\)](http://www.iana.org).

The DNS hierarchy is shown in the illustration below.

Figure 1-1 DNS Hierarchy



Generic TLD

The following table shows the top-level DNS domains and the organization types that use them:

Table 1-1 DNS Domains and Organization Types

| Domain | Used by |
|--------|---|
| .com | Commercial organizations, such as novell.com |
| .edu | Educational organizations, such as ucla.edu |
| .gov | Governmental agencies, such as whitehouse.gov |
| .mil | Military organizations, such as army.mil |
| .org | Nonprofit organizations, such as redcross.org |
| .net | Networking entities, such as nsf.net |
| .int | International organizations, such as nato.int |

Country Code TLD

Top-level domains organize domain namespace geographically.

Table 1-2 Country Code Domains

| Domain | Used by |
|--------|---------------|
| .fr | France |
| .in | India |
| .jp | Japan |
| .us | United States |

Infrastructure TLD

The .arpa (Address and Routing Parameter Area) TLD is used extensively for Internet infrastructure. It contains subdomains such as in-addr.arpa and ipv6.arpa.

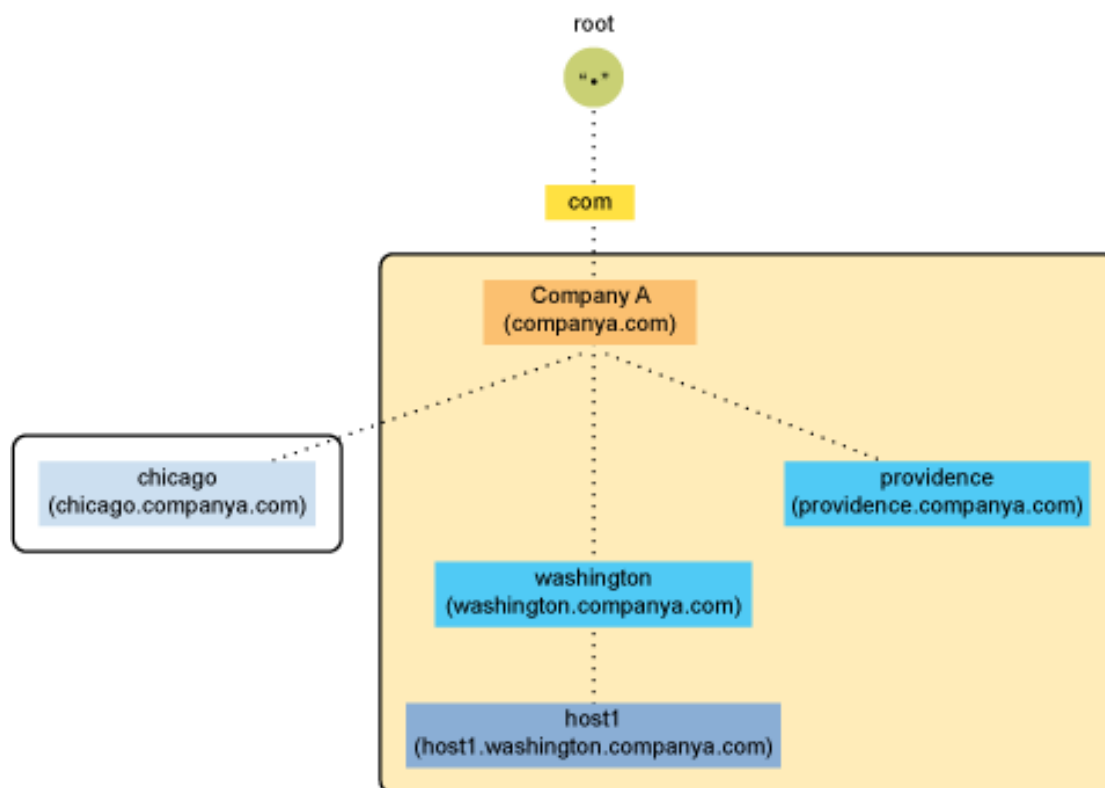
Domains and Subdomains

A domain is a subtree of the DNS tree. Each node on the DNS tree represents a domain. Domains under the top-level domains represent individual organizations or entities. These domains can be further divided into subdomains to ease administration of an organization's host computers.

For example, Company A creates a domain called companya.com under the *com* top-level domain. Company A has separate LANs for its locations in Chicago, Washington, and Providence. Therefore, the network administrator for Company A decides to create a separate subdomain for each division, as shown in [Figure 1-2, "Domains and Subdomains," on page 16](#).

Any domain in a subtree is considered part of all domains above it. Therefore, *chicago.companya.com* is part of the *companya.com* domain, and both are part of the *.com* domain.

Figure 1-2 Domains and Subdomains



Domain Names

The domain name represents the position of an entity within the structure of the DNS hierarchy. The domain name of a node is the list of the labels on the path from the node to the root of the tree. Domain names are not case sensitive and their length is limited to 255 characters. Valid characters for domain names according to RFC 1024/RFC 1035 are a-z (case insensitive), 0-9, and hyphens. Each label in the domain name is delimited by a period. For example, the domain name for the Providence domain within Company A is providence-companya-com, as shown in [Figure 1-2 on page 16](#).

NOTE: Novell DNS supports underscores using the check-names option for coexistence with Windows* DNS servers.

Each computer that uses DNS is given a DNS hostname that represents the computer's position within the DNS hierarchy. Therefore, the hostname for host1 in [Figure 1-2 on page 16](#) is host1.washington.companya.com.

The domain names in the figure end with a period, representing the root domain. Domain names that end with a period are called fully qualified domain names (FQDNs).

n-addr.arpa Domain

The in-addr.arpa domain (or zone) provides the mapping of IP addresses to names within a zone, enabling a client (or resolver) to request a hostname by providing an IP address. This function, also known as reverse lookup, is used by some security-based applications.

The file that stores the `in-addr.arpa` data contains pointer (PTR) records and additional name server records, including the Start of Authority (SOA) records, which are similar to the other DNS zone files. Within the `in-addr.arpa` zone file, IP addresses are listed in reverse order, and *in-addr.arpa* is appended to the address. A query for a host with an IP address of 10.10.11.1 requires a PTR query with the target address of 1.11.10.10.in-addr.arpa.

Domain Delegation

Domain delegation gives authority to an organization for a domain. Having authority for a domain means that the organization's network administrator is responsible for maintaining the DNS database of hostname and address information for that domain. Domain delegation helps in distributing the DNS namespace.

A Division can be made between any two adjacent nodes in the namespace. After all divisions are made, each group of connected namespace is considered as a separate zone. The zone is authoritative for all names in the connected region, and these cuts are managed by domain delegation. All the host information for a zone is maintained in a single authoritative database.

For example, in [Figure 1-2 on page 16](#), the `companya.com` domain is delegated to company A, creating the `companya.com` zone. There are three subdomains within the `companya.com` domain:

- ♦ `chicago.companya.com`
- ♦ `washington.companya.com`
- ♦ `providence.companya.com`

The company A administrator maintains all host information for the zone in a single database and also has the authority to create and delegate subdomains.

For example, if company A's Chicago location has its own network administrator, they could make a division between the `chicago.companya.com` domain and the `companya.com` domain and then delegate the `chicago.companya.com` zone. Then `companya.com` would have no authority over `chicago.companya.com`. Company A would have two domains:

- ♦ `companya.com`, which has authority over the `companya.com`, `washington.companya.com`, and `providence.companya.com` domains
- ♦ `chicago.companya.com`, which has authority over the `chicago.companya.com` domain

1.1.2 DNS Name Service

DNS uses the name service component to provide the actual name-to-IP address mapping that enables computers to locate each other on an internetwork. The name service uses a client/server mechanism in which clients query name servers for host address information.

- ♦ [“Name Servers” on page 18](#)
- ♦ [“Root Name Servers” on page 18](#)
- ♦ [“DNS Resolver” on page 19](#)
- ♦ [“Name Resolution” on page 19](#)
- ♦ [“Caching” on page 20](#)

Name Servers

Name servers are information repositories that make up the domain database. The database is divided into sections called zones, which are distributed among the name servers. The name servers answer queries by using data in their zones or caches. A DNS name server can be either a primary name server or a secondary name server.

In addition to local host information, name servers maintain information about how to contact other name servers. Name servers in an intranet are able to contact each other and retrieve host information. If a name server does not have information about a particular domain, the name server relays the request to other name servers up or down the domain hierarchy until it receives an authoritative answer for the client's query.

All name servers maintain information about contacting name servers that are available in other parts of the DNS namespace. This process of maintaining information is called linking to the existing DNS hierarchy. This is done by providing information about the root name servers. The administrator also enters information into the database about name servers in the lower-level domains. For example, when creating a subdomain, the administrator provides the name server information of the subzone.

- ♦ [“Primary Name Servers” on page 18](#)
- ♦ [“Secondary Name Servers” on page 18](#)
- ♦ [“Forward Name Servers” on page 18](#)

Primary Name Servers

One DNS name server in each administrative zone maintains the read-write copies of hostname database and address information for an entire domain. This name server is the primary name server, and the domain administrator updates it with hostnames and addresses as changes occur. Primary and secondary name servers are also called masters and slaves.

Secondary Name Servers

Secondary name servers have read-only copies of the primary name server's DNS database. Secondary name servers provide redundancy and load balancing for a domain.

Periodically, and when a secondary name server starts, the secondary name server contacts the primary name server and requests a full or incremental copy of the primary name server's DNS database. This process is called zone transfer.

If necessary, a primary name server can also function as a secondary name server for another zone.

Forward Name Servers

The Forward DNS server forwards all queries to another DNS server and caches the results. Unlike primary and secondary zones, there is no functional difference between a designated server and other servers.

Root Name Servers

Root name servers contain information for the name servers in all top-label domains. The root server plays a very significant role in resolving DNS queries. Currently, there are 13 name servers available, which contain information of the name servers for all TLDs.

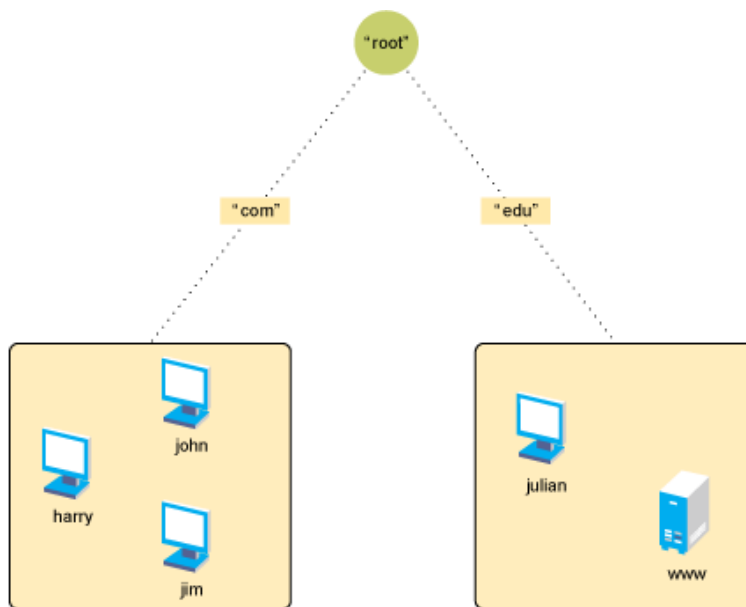
DNS Resolver

DNS resolvers are client programs. They interface user programs to domain name servers. A resolver receives a request from a user program and returns the desired information. It basically does a name-to-address, address-to-name, and general lookup.

Name Resolution

DNS is a distributed database with multiple servers that maintain different parts of the same tree. The links between the servers are through root server and domain delegation, as shown in the following figure.

Figure 1-3 DNS Namespace



DNS queries can be resolved in two ways:

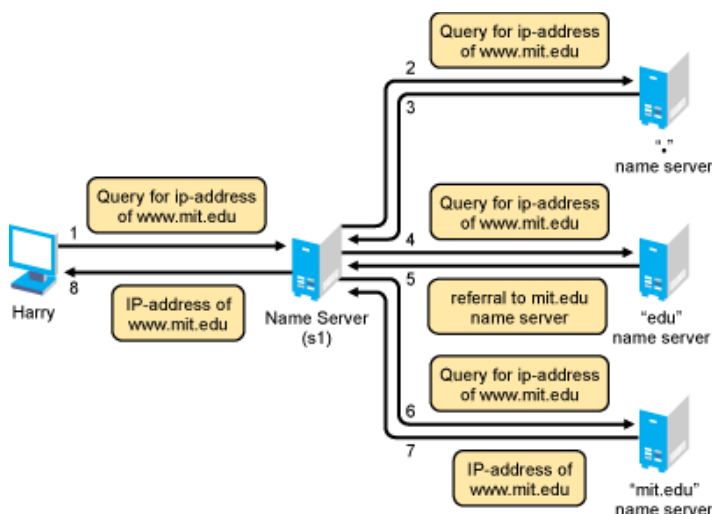
- ♦ **Iterative query:** An iterative request from a client expects the best actual answer or referral that the DNS server can immediately provide, without contacting other DNS servers.

For example, in [Figure 1-4 on page 20](#), Harry initiates an iterative query for the A record of `www.mit.edu`. After receiving this query, the name server (s1) might return the answer, if the answer is available in the cache. If the answer is not available, the server returns a referral [NS and A rrs] to the other DNS servers that are closer to the names queried by the client.

- ♦ **Recursive query:** A recursive request from a client expects the actual answer that the DNS server can provide either from its own cache or by contacting other DNS servers.

For example, in [Figure 1-3 on page 19](#) and [Figure 1-4 on page 20](#), Harry initiates a recursive query for the A record of `www.mit.edu`. After receiving this query, the name server (s1) contacts the root server to resolve this query and gets the referrals (name server info) for the `.edu` zone. Now, s1 again initiates a query for A record of `www.mit.edu` to the name server (using the referrals received) of `edu.zone` and will get the referrals for `mit.edu.zone`. Server s1 will again initiate a query for the A record of `www.mit.edu` to the name server (using the referrals received) and gets the A record. This A record is returned to Harry.

Figure 1-4 Name Resolution



Caching

Caching is a mechanism to improve the performance of query resolution. The cache memory is empty when a server first starts. This cache is built as it starts resolving queries. It caches all the answers and referrals during recursive queries, and the cached data remains in the cache memory until the Time-To-Live (TTL) expires. The TTL specifies the time interval that the entries can be cached before they are discarded.

1.1.3 Resource Records

Resource records (RRs) contain the host information maintained by the name servers and make up the DNS database. Different types of records contain different types of host information. For example, an Address A record provides the name-to-address mapping for a given host, and Start of Authority (SOA) record specifies the start of authority for a given zone.

A DNS zone must contain several types of resource records in order for DNS to function properly. Other RRs can be present, but the following records are required for standard DNS:

- ♦ **Name Server (NS):** Binds a domain name with a hostname for a specific name server.

The DNS zone must contain NS records (for itself) for each primary and secondary name server of the zone. It must also contain NS records of the lower-level zones (if any) to provide links within the DNS hierarchy.

- ♦ **Start of Authority (SOA):** Indicates the start of authority for the zone.

The name server must contain only one SOA record, specifying its zone of authority.

For example, the name server for a zone must contain the following:

- ♦ An SOA record identifying its zone of authority
- ♦ An NS record for the primary name server within the zone
- ♦ An NS record for each secondary name server within the zone
- ♦ NS records for delegated zones, if any
- ♦ A records for the NS record (if applicable)

For more information about Resource Record types and their RDATA (Resource Record data), see [Section A.2, “Types of Resource Records,”](#) on page 196.

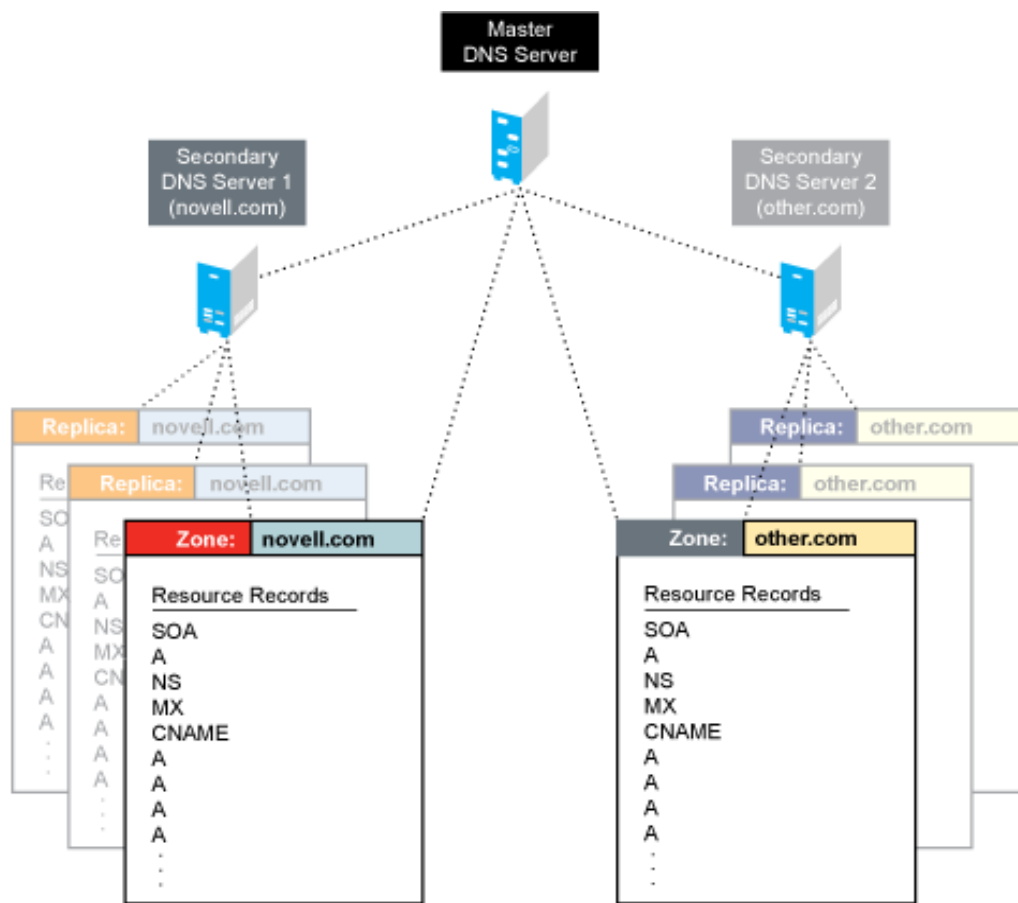
1.1.4 DNS Structure

DNS is administered by building a database of information that includes all the resource records of a zone into a text file called a master file. The administration of these files is difficult and cumbersome. Initial versions of the Novell DNS server used Btrieve* as its database. Other vendors also use large files to store information required for a DNS zone.

Figure 1-5 represents a traditional DNS strategy. A zone, such as novell.com, uses a primary DNS server to handle queries about the entities within it. A DNS server supports more than one zone, and it has at least one secondary server for backup (redundancy) or load-sharing purposes. The primary DNS server provides DNS name service for two zones: novell.com and other.com. The secondary DNS server 1 provides backup support for the novell.com zone, and the secondary DNS server 2 provides backup support for the other.com zone. When changes occur to the DNS database, the master files corresponding to that zone at the secondary server are updated by zone transfers.

The file storing the resource records for a zone might have hundreds or thousands of entries for different types of resources, such as users’ addresses, hosts, name servers, mail servers, and pointers to other resources.

Figure 1-5 DNS Structure



The next section provides details on the following:

- ♦ “DNS Master File” on page 22
- ♦ “Master File Directives” on page 23

DNS Master File

A DNS master file is a text file that contains resource records that describe a zone. When you build a zone, the DNS objects and their attributes translate into resource records for that zone.

You can import a DNS master file if it conforms to IETF RFCs 1034, 1035, and 1183 and is in BIND master file format. A sample DNS master file is shown in the following example.

```
$ORIGIN companya.com.
@ soa ns.companya.com. admin.novell.com (
  1996091454 /* SOA serial number */
  3600 /* Zone Refresh interval */
  300 /* Zone retry interval */
  604800 /* Zone Expire interval */
  86400) /* Zone Minimum TTL */
  ns ns1.companya.com.
  ns ns2.companya.com.
  mx 5 companya.com.
$ORIGIN companya.com.
ns1 a 123.45.67.89
ns2 a 123.45.68.103; End of file
```

Master File Directives

The master file directives include \$GENERATE, \$ORIGIN, \$INCLUDE, and \$TTL.

- ♦ **\$GENERATE:** Enables you to create a series of resource records that differ from each other only by an iterator.

The syntax is:

```
$GENERATE range lhs type rhs [comment]
```

Range: Can be set to start-stop or start-stop/step. All values for start, stop, and step must be positive.

lhs: The owner name of the records to be created. The \$ symbols in the lhs are replaced by the iterator value. Using \ \$ allows a \$ symbol in the output. A \$ can be optionally followed by a modifier as \${offset[,width[,base]]}.

A modifier can have an offset, a width, and a base. The offset is used to change the value of the iterator, base specifies the output format in which the values are printed, and width is used for padding. The available base values are decimal (d), octal (o), and hexadecimal (x or X). The default modifier is \${0, 0, d}. If the lhs is not absolute, the current value of \$ORIGIN is appended to the name.

Type: The resource record type. The supported types are PTR, CNAME, DNAME, A, AAAA, and NS.

rhs: The domain name. Processed similarly to lhs.

For example,

```
$ORIGIN 0.0.192.IN-ADDR.ARPA
$GENERATE 1-2 0 NS SERVER$.EXAMPLE.com
```

is equivalent to

```
0.0.0.192.IN-ADDR.ARPA NS server1.example.com
0.0.0.192.IN-ADDR.ARPA NS server2.example.com
```

- ♦ **\$ORIGIN:** Enables you to set the domain name as the origin. The origin is appended to all domain names in the zone data file that do not end with a dot.

The syntax is:

```
$ORIGIN domain-name [comment]
```

For example,

```
$ORIGIN example.com.
WWW CNAME Web server
```

is equivalent to

```
WWW.EXAMPLE.COM. CNAME webserver.example.com.
```

NOTE: If the \$ORIGIN directive is not already included, make sure you include this directive at the start of the zone file.

- ♦ **\$INCLUDE:** Enables you to include another file in the current file. The included file can be read and processed as if it were present in the current file at that point. The domain name can also be specified with the \$INCLUDE directive, to process the file included with \$ORIGIN set to that value. If the origin is not specified, the current \$ORIGIN is used.

After the included file is processed, the origin and the domain name values are reset to their previous values before processing the included file.

The syntax is:

```
$INCLUDE filename [origin] [comment]
```

NOTE: The \$INCLUDE directive is not supported through the management utilities.

- ♦ **\$TTL:** Enables you to set the default time to live for the subsequent resource records without any TTL values. The time range for TTL is from 0 to 214748367 seconds. If the \$TTL value is not present in the master file, SOA minimum TTL is used as the default.

The syntax is:

```
$TTL default-ttl [comment]
```

1.2 Novell DNS Service

- ♦ [Section 1.2.1, “Novell eDirectory and DNS,” on page 25](#)
- ♦ [Section 1.2.2, “eDirectory Schema Extensions for DNS,” on page 27](#)
- ♦ [Section 1.2.3, “Dynamic DNS,” on page 29](#)
- ♦ [Section 1.2.4, “Zone Transfer,” on page 31](#)
- ♦ [Section 1.2.5, “Dynamic Reconfiguration,” on page 31](#)
- ♦ [Section 1.2.6, “Fault Tolerance,” on page 32](#)
- ♦ [Section 1.2.7, “Cluster Support,” on page 32](#)
- ♦ [Section 1.2.8, “Notify,” on page 32](#)
- ♦ [Section 1.2.9, “Load Balancing,” on page 33](#)
- ♦ [Section 1.2.10, “Forwarding,” on page 33](#)
- ♦ [Section 1.2.11, “No-Forwarding,” on page 34](#)
- ♦ [Section 1.2.12, “Benefits of Integrating a DNS Server with eDirectory,” on page 34](#)
- ♦ [Section 1.2.13, “rndc Support,” on page 35](#)
- ♦ [Section 1.2.14, “Managing DNS Objects,” on page 35](#)

The Novell DNS service provides the following DNS features:

- ♦ All DNS configuration is stored in eDirectory, facilitating enterprise-wide management.
- ♦ A Novell DNS server can be a secondary name server to a zone (DNS data loaded into eDirectory through a zone transfer), or it can be a primary name server.
- ♦ DNS data can be imported using a BIND Master file to populate eDirectory for convenient upgrades from BIND implementations of DNS.
- ♦ DNS data can be exported from eDirectory into BIND Master file format.
- ♦ Root server information is stored in eDirectory and shared by all eDirectory-based DNS servers.

- ♦ Zone transfers are made to and from Novell servers. Full Zone Transfer-In (AXFR) and Incremental Zone Transfer-In (IXFR) are supported when the server is a designated secondary. Any type of server can perform a zone-out transfer.
- ♦ A Novell DNS server can be authoritative for multiple domains.
- ♦ Novell DNS servers maintain a cache of data from eDirectory so they can quickly respond to queries.
- ♦ A Novell DNS server can act as a caching or forwarder server. Forwarding can be configured both at server and zone level. A forwarder specifies how the behavior of queries is controlled if the server is not authoritative and the answers do not exist in the cache
- ♦ A Novell DNS server supports fault tolerance when there is an eDirectory service outage.
- ♦ Novell DNS servers support multihoming.
- ♦ Novell DNS server software supports shuffling responses to queries that have multiple resource records.
- ♦ Novell DNS servers support dynamic reconfiguration (automatic detection of the configuration and data changes).

1.2.1 Novell eDirectory and DNS

The DNS software in Open Enterprise Server 2 Linux integrates DNS information into Novell eDirectory.

Integrating DNS with eDirectory greatly simplifies network administration by enabling you to enter all configuration information into one distributed database. The DNS configuration information is replicated just like any other data in eDirectory.

By integrating DNS into eDirectory, Novell has shifted the concept of a primary or secondary away from the server to the zone itself. After you have configured the zone, the data is available to any of the Novell DNS servers you select to make authoritative for the zone. The Novell DNS server takes advantage of the peer-to-peer nature of eDirectory by replicating the DNS data.

The Novell DNS Service interoperates with other DNS servers. The Novell DNS server can act as either a master DNS server or a secondary DNS server in relation to non-Novell DNS servers. The Novell DNS server can act as the master DNS server and transfer data to non-Novell secondary servers. Alternatively, one Novell DNS server can act as a secondary DNS server and get transferred data from a non-Novell master server. All Novell DNS servers can then access the data through eDirectory replication.

Novell has integrated DNS into eDirectory by extending the eDirectory schema and creating new eDirectory objects to represent zones, resource records, and DNS name servers. Integrating these new objects into eDirectory simplifies the administration of DNS, enabling centralized administration and configuration.

A Zone object is an eDirectory container object that holds RRSets, which are leaf objects. A DNS server object is a leaf object. For detailed information about these objects, see [“Understanding DNS and DHCP Services” on page 13](#).

In traditional DNS, all data changes are made on a single primary name server. When changes are made, the secondary name servers request transfers of the changes from the primary name server. This process is called a zone transfer. The master-slave approach has several disadvantages, the most significant being that all changes must be made at the primary server.

Using the primary and secondary zone concept, the Novell approach allows changes from anywhere in the network through eDirectory, which is not dependent on one server. Zone data is stored within eDirectory and is replicated just like any other data in the eDirectory tree.

The Novell implementation of DNS supports the traditional primary-secondary DNS name server approach to moving DNS data in and out of eDirectory. Although all Novell servers can recognize DNS data after the data is placed in the directory through eDirectory replication, only one server is required for a zone transfer. The server assigned to perform this function in a secondary zone is called the Zone-in (Designated Secondary) DNS server.

In a secondary zone, the Zone-in server is responsible for requesting a zone transfer of data from the external primary name server. The Zone-in server determines which data has changed for a zone and then makes updates to eDirectory so that other servers are aware of the changes.

A Forward zone, acts as a forwarder and forwards all queries on zones to primary or secondary servers of the zone.

The Designated DNS (DDNS) server is a server identified by the network administrator to perform certain tasks for a primary zone. The DDNS server for a primary zone is the only server in that zone that receives dynamic updates from a DHCP server to perform Dynamic DNS (DDNS) updates. These updates cause additions and deletions of resource records and updates to the zone's serial number.

Figure 1-6 illustrates a Novell server as the primary and secondary DNS name server and also illustrates primary and secondary zones within eDirectory. In this example, there are two zones. Any of the Novell DNS servers assigned to a zone is able to respond authoritatively to queries for the zone. For each zone, one server is designated by the administrator to act as the DDNS server. S1 is the Designated Primary DNS server for Zone 1 and S3 is a Passive Primary server. S1 accepts the Dynamic updates from the DHCP server. S2 is the Zone In (Designated Secondary) server and S4 is the Passive Secondary server for the secondary zone zone2, called the Foreign Zone. S2 occasionally requests zone transfers from the foreign server and places the modified zone data into eDirectory, where any of the Novell servers can respond to queries for it. S3 and S4 will get the latest data through eDirectory.

Figure 1-6 Novell Server As a Primary/Secondary DNS Server

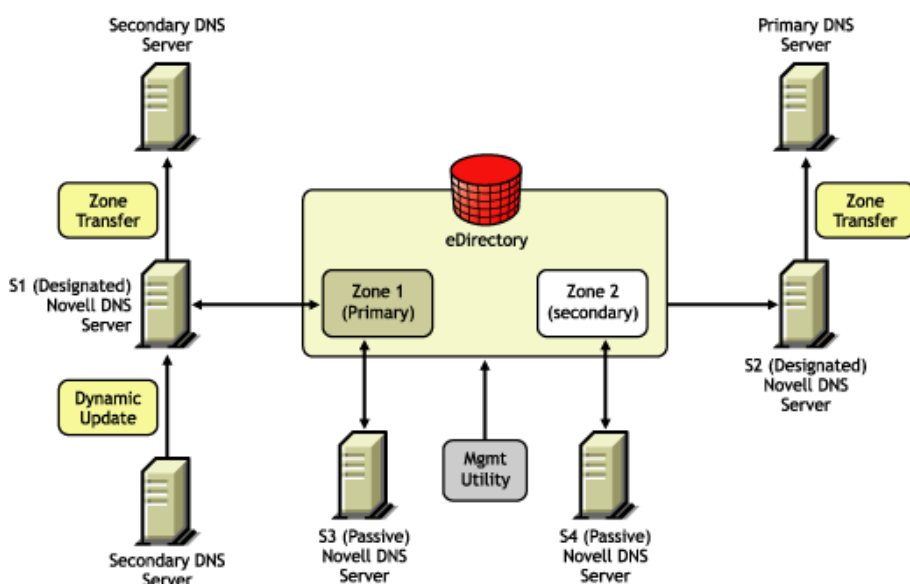
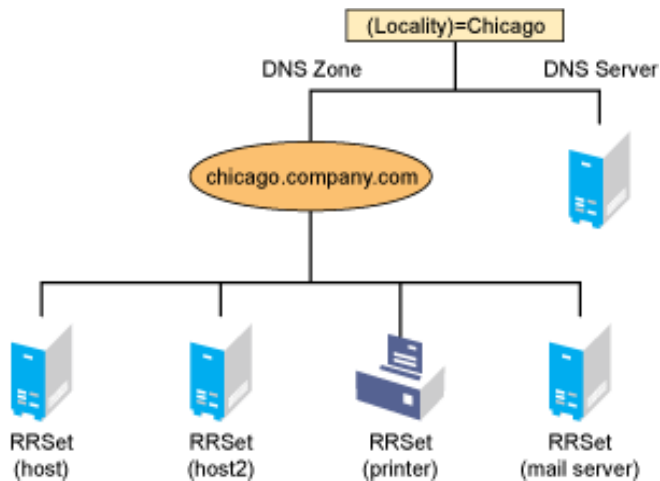


Figure 1-7 shows a representation of eDirectory objects within a DNS zone.

Figure 1-7 DNS Zone



1.2.2 eDirectory Schema Extensions for DNS

The eDirectory schema extension defines additional objects needed for DNS.

- ♦ “DNS eDirectory Objects and Attributes” on page 27

DNS eDirectory Objects and Attributes

When you select the Novell DNS Service during the OES 2 Linux installation, the eDirectory schema is extended to enable the creation of DNS objects and the following objects are created:

- ♦ DNS-DHCP (Locator object)
- ♦ DNSDHCP-Group (Group object)
- ♦ RootServInfo

Only one copy of these objects exists in an eDirectory tree. The DNS servers, DHCP servers, iManager, and Management Console must have access to these objects.

The DNSDHCP-Group object is a standard eDirectory group object. The DNS servers gain rights to DNS data within the tree through the Group object.

The DNS-DHCP Locator object is created during the OES 2 Linux installation, if the DNS option is chosen. The creator of the Locator object grants Read and Write rights to this object to the network administrators.

The DNS-DHCP Locator object contains global defaults, DNS options, and a list of DNS servers and zones in the tree. iManager and the Java Management Console use the Locator object contents instead of searching the entire tree to display these objects. The Locator object is hidden by iManager and the Java Management Console.

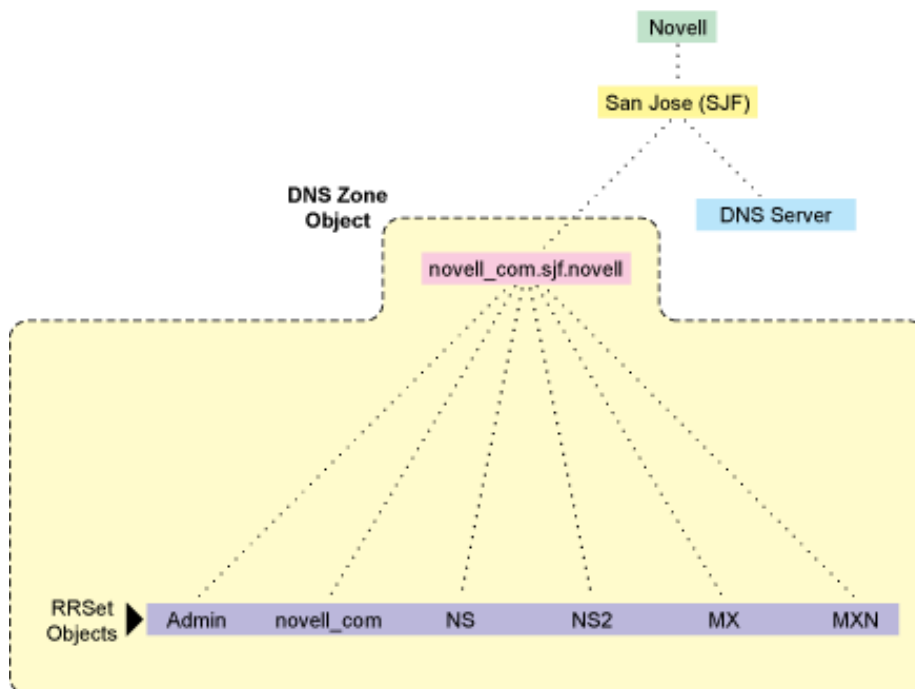
The RootServInfo is a Zone object, which is an eDirectory container object that contains resource records for the DNS root servers. The resource record sets contain Name Server records and Address records of name servers that provide pointers for DNS queries to the root servers. The RootServInfo object is the equivalent of the BIND `db.root` file.

The following new eDirectory objects are also required for DNS:

- ♦ “DNS Server Object” on page 28
- ♦ “DNS Zone Object” on page 29
- ♦ “DNS Resource Record Set Object” on page 29
- ♦ “DNS Resource Records” on page 29

Figure 1-8 shows an example of a tree with DNS objects.

Figure 1-8 eDirectory Objects for DNS



DNS Server Object

The DNS server object (or dnipDNSServer object) is created during installation. It is created in same context as the NCP server, contains DNS server configuration parameters, and includes the following:

- ♦ Zone List
- ♦ DNS Server IP Address
- ♦ Domain Name of the DNS Server
- ♦ DNS Server Options
- ♦ Forwarding List
- ♦ No-Forwarding List
- ♦ Key List
- ♦ Access Control List for zone transfer, query, recursion, notify, etc.
- ♦ Other additional advanced options to fine-tune the DNS server

DNS Zone Object

The DNS Zone object is a container object that contains all the data for a single DNS zone. A Zone object is the first level of the DNS zone description. A Zone object can be contained under an Organization (O), Organizational Unit (OU), a Country (C), or a Locality (L).

Multiple DNS zones can be represented within eDirectory by using separate, independent DNS Zone objects. A network administrator can support multiple DNS zones on a single OES 2 Linux server by creating multiple DNS Zone objects and assigning the server to serve those zones.

The DNS Zone object contains data that correlates to a DNS Start of Authority (SOA) resource record (RR) and a member list of all eDirectory-based DNS servers that serve the zone.

The DNS namespace hierarchy is not represented within the eDirectory hierarchy. A zone and its child zone might appear as peers within the eDirectory hierarchy, even though they have a parent-child relationship within the DNS hierarchy.

DNS object names are created using the DNS Zone names.

Valid characters for domain names according to RFC 1024/RFC 1035 are a-z (case insensitive), 0-9, and hyphens. For example, the name of the Zone object for newyork.companya.com zone, which exists in an eDirectory context sjf.us., shall be newyork_companya_com.sjf.us

NOTE: Novell DNS supports underscore character in domain names and Resource Records using Novell DNS server check-names statement for co-existence with Windows DNS servers.

DNS Resource Record Set Object

The DNS Resource Record Set (RRSet) object is an eDirectory leaf object contained within a DNS Zone object. An RRSet object represents an individual domain name within a DNS zone. Its required attributes are a DNS domain name and resource records (RRs).

Each domain name within a DNS zone object has an RRSet object. Each RRSet object has one or more resource records beneath it that contain additional information about the zone data.

DNS Resource Records

A DNS resource record (RR) is an attribute of an RRSet that contains the resource record type and data of a single RR. RRs are configured beneath their respective RRSet objects. Resource records describe their associated RRSet object.

The most common resource records are Address (A) records, which map a domain name to an IP address, and Pointer (PTR) records, which map an IP address to a domain name within an in-addr.arpa zone.

1.2.3 Dynamic DNS

Dynamic DNS (DDNS) provides a way to dynamically update DNS with resource records from applications such as DHCP servers, and DNS clients. DDNS eliminates the need for any additional configuration of DNS for each host address change. A Novell DNS server supports the following DDNS mechanisms:

- ♦ Novell DDNS, a mechanism by which NetWare DHCP servers update Novell DNS servers
- ♦ RFC 2136-based dynamic updates

All changes made to a zone through dynamic updates are stored in the zone's journal file. This file is automatically created by the server in a binary format when the first dynamic update takes place. The journal file name has the `.jnl` extension. This file is also used for IXFR. We recommend that you do not edit the contents of the journal file.

- ♦ “Novell DDNS” on page 30
- ♦ “2136 Dynamic Update” on page 30

Novell DDNS

The Dynamic DNS (DDNS) feature of the Novell DNS service provides a way to update DNS with accurate Address (A) records and Pointer (PTR) records for address assignments made by a DHCP server. Address (A) records map a domain name to an IP address. A Pointer (PTR) record specifies a domain name that points to some location in the domain namespace. These resource records are required for both name-to-address and address-to-name resolutions.

When DDNS is active, the DHCP server updates the DDNS server for the zone, adding or deleting the corresponding Address and Pointer records. The DHCP server also notifies the DDNS server when leases expire, causing the A and PTR records to be deleted.

When the DHCP server grants a lease to a client that is subject to DDNS updates, the DHCP server updates its IP address database and eDirectory to store the transaction. The DHCP server also contacts the DNS server and submits a request for a DNS update.

For DDNS updates, the DNS server requires the fully qualified domain name (FQDN) and the IP address of the client. The DHCP server knows the IP address, but it must assemble the FQDN from the hostname and the subnet's domain name.

The DNS server usually maintains two resource records for each client. One maps FQDNs to IP addresses using A records. The other maps the IP address to the FQDN using PTR records. When DDNS is enabled and a client receives an address from the DHCP server, the DNS server updates both of these records.

When a client loses or ends its lease and is subject to DDNS updates, the DNS server receives the DDNS update request and deletes the PTR and A records associated with the client.

NOTE: While using the Novell DHCP server, both the forward and reverse zones must be designated primary on a single server.

2136 Dynamic Update

A Novell DNS server supports dynamic updates complying the RFC 2136 standards. This support provides the ability to update various types of resource records into DNS under certain specified conditions. Dynamic update is fully described in RFC 2136.

- ♦ A 2136 dynamic update can be enabled or disabled on a zone-by-zone basis, by specifying the allow-update filter for the zone. It grants permission to the clients to update any record or name in the zone.
- ♦ DNS Key provides a means of authentication for dynamic DNS updates and for queries to a secured DNS Server. DNS Key uses shared secret keys as a cryptographically secure means of authenticating a DNS update/query.

1.2.4 Zone Transfer

Zone transfer is essential for maintaining up-to-date zone data in the server. When a Novell server is designated as primary, all the changes made by the designated primary to eDirectory are reflected in the eDirectory replicas, using the eDirectory sync property. When a Novell server is designated secondary, zone transfer is needed for receiving the most up-to-date zone data from any primary servers.

The designated secondary server sends a zone-in request after the refresh time interval or after receiving a notification from the primary server. The zone transfer-in requests are not triggered if the eDirectory services are not available.

The final step in a successful zone transfer-in is to update the SOA serial number. The passive secondary servers compare the eDirectory SOA serial number with their own copy to determine whether there is a need to synchronize the data from eDirectory.

The following types of zone transfers are supported:

- ♦ [“Full-Zone Transfer-In” on page 31](#)
- ♦ [“Incremental Zone Transfer-In” on page 31](#)

NOTE: No zone transfers-in are initiated if it fails while the zone transfer is taking place. The changed data is overwritten during the next zone transfer.

Full-Zone Transfer-In

The secondary server receives a full zone transfer-in (AXFR) into a different zone database. After the complete zone data is received, the server replaces the old database with the new one, and tries to identify the difference between the existing zone database and the new database that is received. This difference is then applied to eDirectory for better performance.

For more information on DNS AXFR, refer to RFC 1034.

Incremental Zone Transfer-In

Incremental zone transfer-in (IXFR) is considered to be a more efficient zone transfer mechanism than AXFR because it transfers only the changed data of the zone. IXFR transfers only the modified data, using the journal file maintained by the DNS server. When a server gets an IXFR request (which has the current SOA serial number of the requester), the server looks into the JNL file to identify the modified data, then sends the data to the requester. For more information on DNS IXFR, refer to RFC 1995.

1.2.5 Dynamic Reconfiguration

A Novell DNS server supports dynamic reconfiguration. The DNS server automatically detects and updates any change in the server's or zone's configuration data. This enables the server to configure itself with these changes without having the administrator intervene to stop and restart the server. Also, out-of-band data changes (creating, modifying, deleting RRs through management utility) are addressed.

The dynamic reconfiguration is also used to monitor the availability of eDirectory with respect to the individual zones, and to detect and log changes that can be used for fault tolerance.

1.2.6 Fault Tolerance

A Novell DNS server supports fault tolerance during an eDirectory service outage. The DNS server loads the configuration and zone data from eDirectory during startup. Also, the dynamic updates received for zone data are updated to eDirectory. It is essential for the DNS server to maintain backup copies of eDirectory so it can get to the zone database during eDirectory unavailability.

The DNS server supports standard DNS queries during fault tolerance mode. However, dynamic updates and zone-in transfers are not supported during this mode because eDirectory cannot be updated.

NOTE: During fault tolerance mode operation, eDirectory might not be available for all zones. Operations other than dynamic update and zone-in are supported for zones that are unavailable.

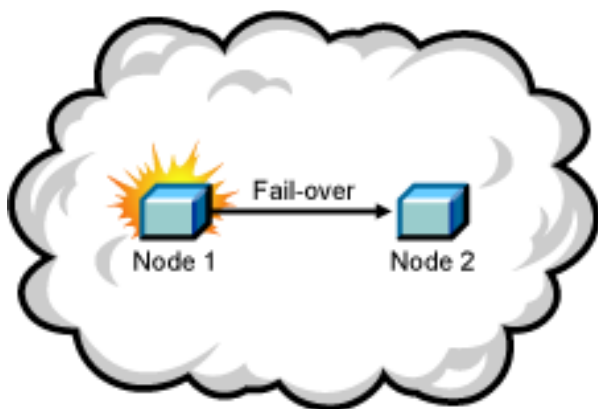
1.2.7 Cluster Support

A Novell DNS server supports cluster services with active-passive and cluster-safe modes. When a DNS server is run on any node, it uses a DNS server object in eDirectory. The cluster-enabled DNS service uses the same DNS server object for the other nodes during a node outage.

When there is a node (node1) outage, clustering enables a DNS server to automatically bring up any other node (node2), using the same server object that was used before the outage. The DNS server object contains a reference to the virtual NCP™ server, which is used to locate the DNS server object. For more information, see [Chapter 14, “Configuring DNS with Novell Cluster Services,” on page 169](#).

The Novell DNS server supports only NSS file systems.

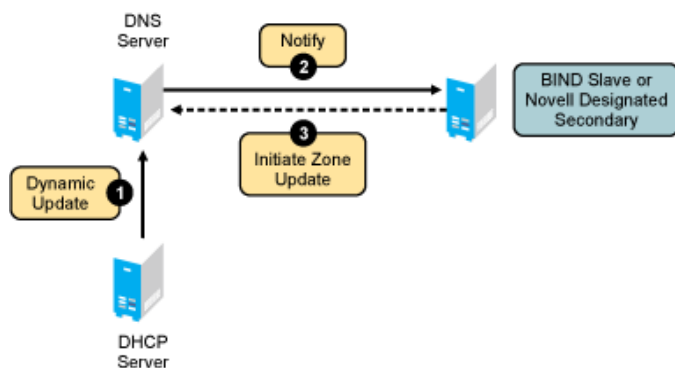
Figure 1-9 Cluster Support



1.2.8 Notify

DNS Notify is a mechanism that allows master name servers to notify their slave servers of changes to a zone's data. In response to a notification from a master server, the slave verifies which SOA serial number for the zone (sent through the notify mechanism) is the newer compared to the current SOA serial number. If the serial number is newer, a zone transfer is initiated. For Novell DNS servers, receiving Notify is valid only for designated secondary servers. Passive servers receive the latest data through eDirectory replication.

Figure 1-10 *Notify*



For more information on DNS notify, refer to RFC 1996.

1.2.9 Load Balancing

If all resolvers querying for a name get the same response, and if all of them contact the same host, that host becomes overloaded. Primitive load balancing can be achieved in DNS by using multiple records for one name. When a resolver queries for these records, the DNS server shuffles them and responds to the query with the records in a different order.

For example, suppose you have three Web servers with these three different IP addresses:

www 3600 IN A 10.10.0.1

3600 IN A 10.10.0.2

3600 IN A 10.10.0.3

The DNS server randomly shuffles the RRs so that clients randomly receive records in the order 1, 2,3; 2, 3, 1; and 3, 1, 2. Most clients use the first record returned and discard the rest.

1.2.10 Forwarding

The name server can forward some or all of the queries that it cannot satisfy from its authoritative data or cache to another name server; this is commonly referred to as a forwarder.

Forwarders are typically used when all servers at a given site should not be allowed to interact directly with the rest of the Internet servers. A typical scenario involves a number of internal DNS servers and Internet firewall servers unable to pass packets through the firewall. They forward to the server that can do it, which queries the Internet DNS servers on the internal server's behalf. An added benefit of using the forwarding feature is that the central machine develops a much more complete cache of information that all of the clients can take advantage of. Forwarding occurs only on those queries for which the server is not authoritative and does not have the answer in its cache.

A forwarding list is a list of IP addresses for the DNS servers to forward the queries to. For example, if a name server is configured to forward queries to 10.10.10.2, all queries that do not have resolutions to the name server are forwarded to 10.10.10.2. (See [“DNS Namespace” on page 19](#)). Forwarding is also possible at the zone level. When forwarders are configured at the zone level, they override the forwarders list configured at server level.

NOTE: The forwarding list syntax is different from the Bind 9.2 syntax for forwarders.

1.2.11 No-Forwarding

No-Forwarding is blocking queries to the list of DNS domains. The No-Forward list is the list of domain names whose unresolved queries are not forwarded to other DNS servers.

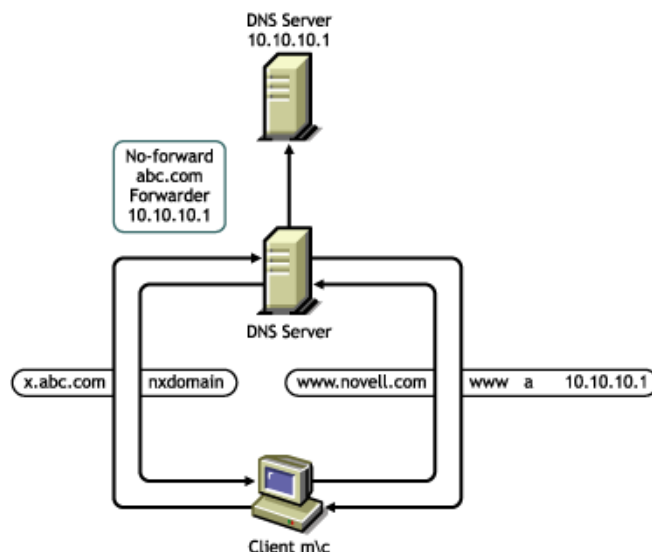
On a query from the client, the authoritative database is first checked. If the domain name is not found, the No-Forward list is checked. If the No-Forward list contains the entry, the query is not answered and the response domain does not exist (NXDOMAIN) is sent to the client. See [“Name Resolution” on page 20](#).

For example, having the domain name “abc.com” in the No-Forward list blocks the queries to “abc.com”, “support.abc.com”, or any other subdomain of abc.com.

Wildcard characters are not supported in the No-Forward list. An asterisk or a root domain in the No-Forward list cannot be used to block queries to all domain names.

For example, developer.*.com cannot be used to block the queries to developer.novell.com or developer.xyz.com, etc.

Figure 1-11 No-Forwarding



1.2.12 Benefits of Integrating a DNS Server with eDirectory

The primary benefits of integrating a DNS server with eDirectory include:

- ♦ Centralized eDirectory-based DNS configuration and management. Configuration must typically be done on a per-server basis with non-Novell DNS servers.
- ♦ DNS data is centrally managed in eDirectory, so all servers associated with a zone become primary or secondary. You get the benefit of all zones being primary, as opposed to a single zone being primary.

- ♦ DNS zone data is replicated through eDirectory replication, which eliminates the need for explicit DNS replication.
- ♦ Decentralized Administration of DNS is possible without accessing the DNS Server console or file system.

1.2.13 rndc Support

BIND includes the `rndc` utility that allows you to administer the `named` daemon, locally or remotely, with command line statements. The `rndc` program uses the `/etc/rndc.conf` file for its configuration options, which can be overridden with command line options.

For more details on the options supported by `rndc`, enter `rndc` at the command prompt.

The following `rndc` commands are not supported:

- ♦ `restart`
- ♦ `recursing`
- ♦ `flushname name [view]`
- ♦ `flush [view]`

1.2.14 Managing DNS Objects

This section provides information about iManager and the Java Management Console.

- ♦ [“iManager Management Utility” on page 35](#)
- ♦ [“Java Management Console” on page 37](#)

iManager Management Utility

iManager can be used to configure and manage eDirectory-based DNS objects and can run on any browser workstation. It does not require the Novell Client or any installed component as a prerequisite. It operates within the common iManager framework and is thus tightly integrated with OES 2 Linux.

For more information, see:

- ♦ [“Management Utility Interface” on page 35](#)
- ♦ [“Managing Roles and Tasks” on page 36](#)

Management Utility Interface

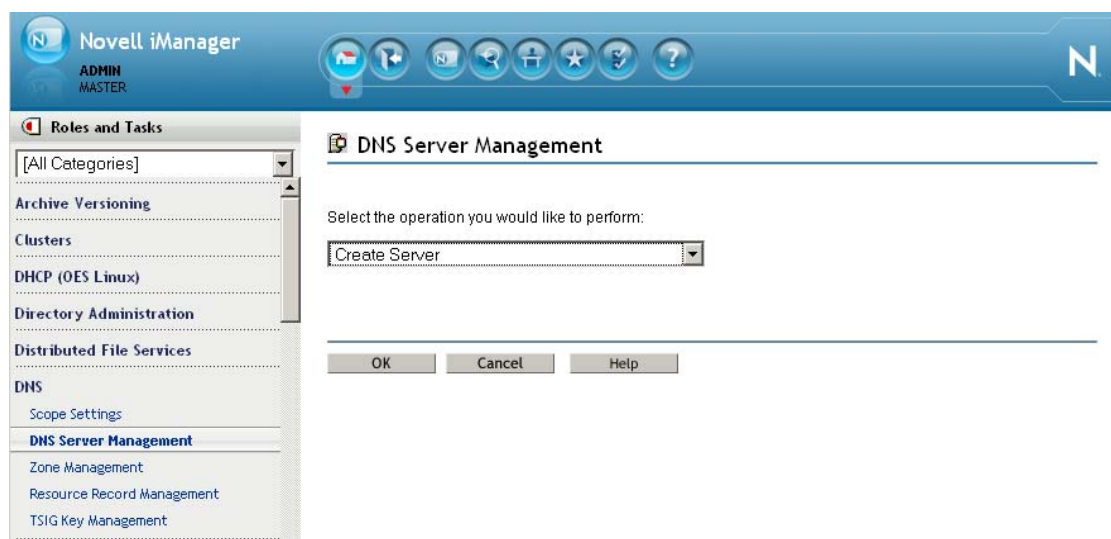
iManager manages one eDirectory tree at a time.

When iManager is started in the browser, the first screen you see is the login screen. You are prompted to provide your username, password, and the eDirectory tree whose objects you want to manage.

Administration authentication in iManager is based on the common authentication mechanism.

To manage objects in a different eDirectory tree, you must log in to the utility again, specifying the eDirectory tree you want to access. Your login identity is displayed at the top of the screen.

Figure 1-12 The DNS iManager Interface



The main screen has three parts: a taskbar at the top of the screen that displays icons for top-level management functions; a left panel that displays roles, tasks, and other administrative functions; and a main panel that allows you to manage role-based and administrative tasks.

For more information on the iManager management interface, see *Novell iManager 2.7 Administration Guide*.

Managing Roles and Tasks

The DNS services have been logically organized into roles and tasks in a way that is intuitive to network administrators. Each role consists of a set of tasks arranged in a manner that is hierarchical, top-down, and easy to administer.

To view the roles, click the *Roles and Tasks* icon on the taskbar.

The tasks under each of these roles are logically arranged in a top-down manner with the option to configure DNS scope settings at the head of each role. A role, depending on its current state, is preceded by a plus or a minus sign. An administrator can expand a role such as DNS to see the tasks it contains or collapse it for a more concise view.

The organization of roles and tasks follows the containership rules of object creation and manipulation in DNS. For example, if you expand the DNS role on the left pane, the logical tasks this role contains appear under it. At the top is the Scope Settings task. This is followed by DNS Server Management, which allows you to specify the location of the Locator object and the administrative scope for the session. At the next level is Zone Management, which manages zones handled by DNS servers. The Resource Record Management allows you to manage resource records contained within a zone. The DNS Key Management allows you to create, modify, and delete DNS Key objects. DNS Key provides a means of authentication for dynamic DNS updates and for queries to a secured DNS Server.

Each task is associated with a set of operations that appear in a drop-down menu on the main panel when you click the task.

For example, to create a new DNS zone, click *DNS > Zone Management*. This launches the Zone Management window in the main panel of the screen. Select *Create Zone* from the drop-down menu and click *OK* to open the Create New Zone window, where you can proceed with the task of creating a new zone.

You can select one object, more than one object, or all objects for deletion with the multi-select delete feature.

IMPORTANT: For improved performance, configure the DNS scope settings before you start using iManager.

Java Management Console

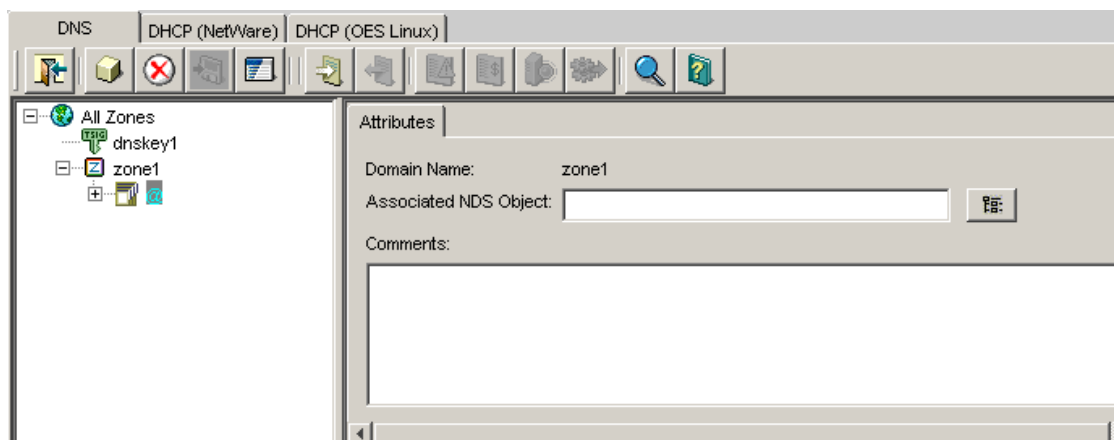
The Java Management Console can be used to configure and manage eDirectory-based DNS objects. It is supported on Windows XP and Windows Vista. It is an independent executable Java application. It can be launched through Windows by using the Programs menu. Click *Start > Programs > DNS-DHCP Management Console > DNSDHCP*. It can also be launched by double-clicking the *DNSDHCP* shortcut icon created on the desktop.

When the Java Management Console is launched, it prompts you to select a tree as the target eDirectory context.

The administrator must log in to the desired eDirectory tree before launching the Java Management Console. To manage objects in a different eDirectory tree, the administrator must exit the utility, change the context to the other eDirectory tree, and launch the utility again. The current eDirectory tree name is displayed in the utility's caption bar.

The Java Management Console manages one tree at a time. You can manage both DHCP and DNS services in the Java Management Console. **Figure 1-13** shows the main user interface window for DNS services.

Figure 1-13 DNS Java-based Management Console User Interface



For more information, see:

- ♦ “DNS Service” on page 38
- ♦ “Toolbar” on page 39
- ♦ “Status Bar” on page 39

- ♦ “Server Status” on page 40
- ♦ “Object Creation Rules” on page 40

DNS Service

DNS objects can be accessed via the *DNS Service* tab. There are three panes within each tab page. The left pane displays the managed DNS objects in tree form. The right pane displays the detailed information about the selected object in the left or bottom pane. The bottom pane lists the DNS servers configured to provide necessary services.

Resources are organized according to the object hierarchy and the implicit ordering of objects. In the DNS Services pane, all zones or resource records within a zone are listed in alphanumeric order.

All DNS objects are created as eDirectory objects and are subject to Linux Administrator conventions. Therefore, when creating a new object, you should always name the object first in each Create dialog box.

The Create dialog box of these objects has browsing capability in the eDirectory tree, so an administrator with Write or Supervisor rights can select a specific context.

A newly created object's button on the toolbar is context-sensitive in relation to the selected item in either service's left tree pane. Your rights to the DNS objects are not be verified until you perform an update, delete, or create against the target objects.

The DNS objects available in the new object dialog's creation list box depend on the selected object in the left tree pane. The following table lists the rules for each container object.

Table 1-3 *Rules for Container Objects*

| Selected Object | Objects that can be created |
|-----------------|---------------------------------------|
| All Zones | DNS Server, Zone |
| DNS Server | DNS Server, Zone |
| Zone | DNS Server, Zone, and Resource Record |
| RRSet | DNS Server, Zone, and Resource Record |
| Resource Record | DNS Server, Zone, and Resource Record |

After a new DNS object has been created, the Java Management Console grants the objects Read and Write rights to the Locator object.

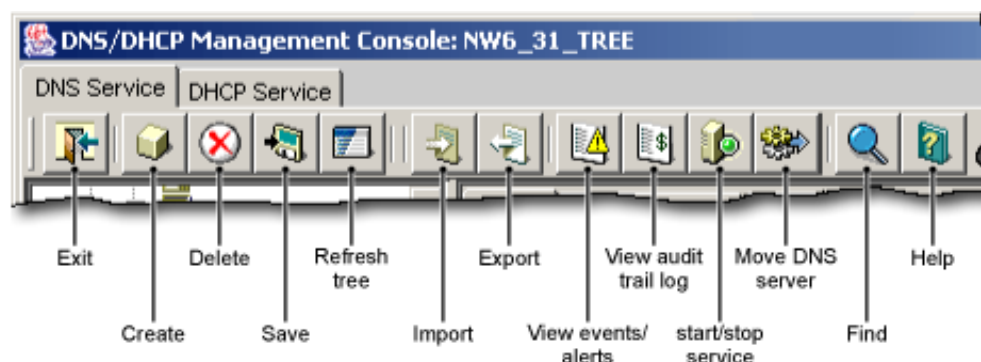
For fast and efficient searching, the distinguished names of newly created zones, DNS servers, and subnets are added to the corresponding attribute of the Locator object. Renaming or deleting these objects is automatically performed by eDirectory because of the built-in feature for eDirectory distinguished names.

After a new DNS object has been created, the Java Management Console gives you the choice of staying in its current focus or setting the focus on the newly created object. The utility also displays its detailed information page in the right pane. This feature is provided as a convenience to administrators and can be used by selecting the *Define Additional Properties* check box.

Toolbar








The Management Console offers no menu items. All functions are provided by the toolbar. The functions that are relevant for the item selected in the left tree pane or bottom server pane are highlighted to show their availability.

Figure 1-14 Toolbar



If you position the cursor over the icon, the icon's name appears. The following table lists, when the toolbar buttons are enabled in relationship to the selected object.

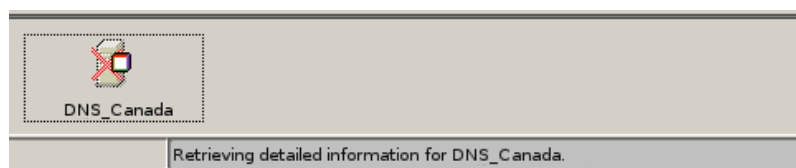
Table 1-4 Period when the Toolbar Buttons are Enabled

| Toolbar Button | Enabled |
|--|--|
| Exit  | Always enabled |
| Create  | When Our Network, Subnet, Subnet Address Range, IP Address, DHCP Server, Shared Network, All Zones, Zone, DNS Server, RRSet, or Resource Record is the selected object |
| Delete  | When Subnet, Subnet Address Range, Shared Network, Zone, RRSet, Resource Record, DHCP Server, or DNS Server is selected |
| Save  | When fields have been changed for updates or changes |
| Tree Refresh  | Always enabled |
| Global Preferences  | Enabled for DHCP Service |
| Help  | Always enabled |

Status Bar

The status bar displays two fields in the bottom pane of the Management Console. The first field shows the current database access interface in progress. The second field displays the current selected object or operation status. [Figure 1-15](#) shows the status bar and the DNS server icon. The status bar is at the bottom of the figure.

Figure 1-15 DNS Status Bar

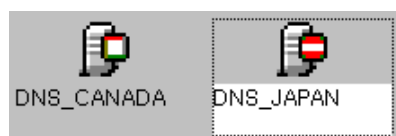


Server Status

Server icons are displayed in the lower portion of the Management Console.

Figure 1-16 shows icons representing two DNS servers. Both servers are operational, `novell-named` has been loaded and each can communicate with the Java Management Console, but the operation of the server on the right (DNS_JAPAN) has been suspended.

Figure 1-16 DNS Server Icons



Object Creation Rules

These are certain rules that govern the creation and manipulation of objects in the Linux DNS object hierarchy.

The DNS Zone object and DNS server object can be created in the context of an Organization (O), Organizational Unit (OU), Locality (L), or Country (C).

Some objects such as DNS server, DNS zone, can be created in any context. After a new DNS object has been created, iManager grants the Read and Write rights to the Locator object. For fast and efficient search operations, the distinguished names of the newly created zones and DNS servers are added to the corresponding attribute of the Locator object. Renaming or deleting these objects is automatically performed by eDirectory because of the built-in feature for eDirectory distinguished names.

iManager lets you create zone objects under Country, Locality, Organization, Organizational unit, and domain containers.

Java Console does not let you select Country and Locality domain containers. You need to specify them manually. After you specify these contexts manually, Java Console creates zone objects under these containers.

Locator object cannot be created from Java Console and iManager. However, during installation if these contexts are specified, then you can create the locator object.

Locator object can be created in the Top, treeroot, and domain containers along with existing Country, Locality, Organization, and Organizational Unit containers.

DNSServer object is always created under NCP server container. You cannot select a separate container for server object from Java Console and iManager.

1.3 DHCP

The Dynamic Host Configuration Protocol (DHCP) uses a client/server structure to provide configuration parameters to hosts. DHCP consists of a protocol for providing host-specific configuration parameters from a DHCP server (or collection of DHCP servers) to a host and a mechanism to allocate network addresses to a host.

NOTE: In this document, the term *host* represents a client in the network with statically assigned IP address. It is identified by a host name.

When the DHCP server is loaded, it reads its configuration information from eDirectory and stores the information in its cache. When the DHCP server assigns IP addresses to clients, it updates the `dhcpcd.leases` file with the addresses allocated. The network administrator can view the entries in the `dhcpcd.leases` file to see how the IP addresses have been allocated.

For more information, see:

- ♦ [Section 1.3.1, “DHCP and BOOTP,” on page 41](#)
- ♦ [Section 1.3.2, “IP Address Allocation,” on page 42](#)
- ♦ [Section 1.3.3, “Virtual LAN Environments,” on page 44](#)

DHCP provides for both static and dynamic configuration of IP clients. Static configuration enables you to assign a specific IP address and configuration to a client with a specific machine or MAC address. When DHCP assigns IP addresses dynamically, IP clients are assigned an IP address that is chosen from a range of available addresses. You can use dynamic address assignments when you are not concerned about which IP address a particular client uses. Each IP client that requests an address assignment can also use the other DHCP configuration parameters.

DHCP can limit the amount of time a DHCP client can use an IP address. This is known as the lease time. You can use the lease time to allow a large number of clients to use a limited number of IP addresses.

DHCP is based on BOOTP and maintains some backward compatibility. Novell DHCP servers can be configured to respond to requests from BOOTP clients.

For more information, see:

- ♦ [Section 1.3.1, “DHCP and BOOTP,” on page 41](#)
- ♦ [Section 1.3.2, “IP Address Allocation,” on page 42](#)
- ♦ [Section 1.3.3, “Virtual LAN Environments,” on page 44](#)

1.3.1 DHCP and BOOTP

- ♦ [“Similarities and Differences” on page 42](#)
- ♦ [“Using a BOOTP Relay Agent” on page 42](#)

Similarities and Differences

DHCP is based on the Bootstrap Protocol (BOOTP) and maintains some backward compatibility. BOOTP was designed for manual configuration of the host information in a server database. Novell has extended support for BOOTP to provide Dynamic BOOTP support. A pool of addresses can be set up for BOOTP address assignment so that each BOOTP address does not need to be configured separately.

From the clients' point of view, DHCP is an extension of BOOTP, enabling existing BOOTP clients to interoperate with DHCP servers without requiring any change to the client initialization software. Some new, additional options optimize DHCP client-server interaction.

There are two primary differences between BOOTP and DHCP. DHCP defines methods through which clients receive IP addresses for a specified period of time, enabling serial reassignment of addresses to different clients. There is no concept of a lease time in BOOTP; address assignments (even in Dynamic BOOTP) are permanent. In addition, DHCP provides a method for a client to acquire all of the IP configuration parameters it requires to operate.

If multiple servers service a single subnet, only the principal server can be designated as an automatic BOOTP server.

Another difference between the two protocols is a change in terminology to clarify the meaning of the Vendor Extension field in BOOTP messages. With DHCP, this field is called the Option field.

Using a BOOTP Relay Agent

A BOOTP relay agent (also known as a forwarder) is an Internet host that passes DHCP messages between DHCP clients and DHCP servers in a subnet environment. The forwarder usually resides on an IP router; however, any Novell server on a subnet can run the bootpfwd. The DHCP service in Novell DNS/DHCP Services provides relay agent functions as specified in the BOOTP protocol specification (Internet RFC 951).

When a client starts, it sends a UDP broadcast message, called a Discover packet, to address 0xFFFFFFFF over port 67 requesting an address.

The forwarder has an IP address on the network and acts like a DHCP server, listening for Discover packets from clients on its LAN that are meant for a DHCP server. The forwarder must be configured with the destination address of the actual DHCP server on a different LAN segment that will provide DHCP service.

The DHCP server must be configured to serve the subnet on which the forwarder is located. The DHCP server must have a subnet address range to provide service.

After receiving a Discover packet from a client, the forwarder reformats the packet and sends it to the DHCP server. The DHCP server responds to the forwarder with an Offer packet containing an address for the client.

When the forwarder receives the Offer packet from the DHCP server, the forwarder contacts the client and provides the IP address and lease information.

1.3.2 IP Address Allocation

Allocation of IP addresses, either temporary or permanent, is one of the two primary services provided by DHCP. The client requests an IP address, and the DHCP server (or collection of DHCP servers) provides an address and guarantees not to give that address to another client within a

specified time. Additionally, the server tries to return the same address to the client each time the client requests an address. The period of time over which an IP address is allocated to a client is called a lease.

DHCP supports three methods of IP address allocation:

- ♦ “Dynamic BOOTP Allocation” on page 43
- ♦ “Dynamic DHCP Allocation” on page 43
- ♦ “Manual Allocation” on page 43
- ♦ “Lease Options” on page 43

A network can use one or more of these methods. The network administrator decides which methods to use.

Dynamic BOOTP Allocation

Dynamic BOOTP enables a DHCP server to assign permanent addresses to BOOTP clients from a pool of addresses. No manual configuration of the client is required prior to address allocation.

Dynamic DHCP Allocation

Dynamic DHCP allocation is the only method enabling automatic reuse of addresses no longer required by a client. Dynamic DHCP allocation is useful for assigning an address to a client that is connected temporarily to the network or for sharing a limited number of IP addresses among a group of clients that do not require permanently assigned IP addresses.

Dynamic DHCP allocation is also useful for assigning an IP address to a new client installed on a network on which IP addresses are scarce and must be reclaimed when older hosts are removed. An additional benefit of dynamic DHCP allocation is that when a client’s lease is renewed, the DHCP server refreshes the client’s configuration.

Manual Allocation

Manual or static allocation is used to assign addresses to DHCP or BOOTP clients. A specific IP address is assigned to the client based on an identifier such as the client’s identifier or MAC address.

Manual allocation of DHCP eliminates the error-prone method of manually configuring hosts with IP addresses in networks for which IP address management without DHCP is desired. Manual allocation can be permanent or set to expire at a future time. When you manually allocate addresses, you can also create corresponding DNS Resource Records, thereby eliminating another error-prone activity.

Lease Options

A client acquires a lease for a fixed period of time. The length of the lease can be a number of hours or days, or it can be for an indefinite period.

After a lease for an IP address has been granted, a client can issue a request to extend its lease. The client can also issue a message to the server to release the address back to the server when the address is no longer required.

If a network has a limited number of IP addresses and must reassign them, the DHCP server reassigns an address when the lease has expired. The server uses configuration information to choose addresses to reuse. For example, the server might choose the least recently assigned address for reassignment. After receiving an address assignment, the host determines whether the address is in use by another host before accepting the address.

To minimize the chance of address duplication, the DHCP server can be configured to ping an address to test its validity before assigning it to a host. If the server receives a response from another device (indicating ownership of the address), the current address assignment is withdrawn so that another address can be assigned to the host.

1.3.3 Virtual LAN Environments

In environments using a virtual LAN (VLAN), multiple subnets might be defined on one physical subnet. For example, one physical subnet might contain several Class C addresses to form a larger address range than allowed for a Class C address. To accommodate a VLAN environment, a shared network object must be configured on the DHCP server to bind the multiple subnets together.

If a forwarder forwards client requests from a physical subnet with multiple subnet bindings and these subnets are bound to a single shared network, the collection of addresses available in configured subnet address ranges is available to all clients (DHCP or BOOTP) on that physical subnet. This is the primary use of the shared network object.

Clients that are on the same subnet as the DHCP server do not need to be configured for the shared network if the server is bound to all local subnet addresses, or if the server has an address on each local subnet.

1.4 Novell DHCP Service

- ♦ [Section 1.4.1, “DHCP Options,” on page 45](#)
- ♦ [Section 1.4.2, “eDirectory Objects for DHCP,” on page 47](#)
- ♦ [Section 1.4.3, “Managing DHCP Objects,” on page 50](#)
- ♦ [Section 1.4.4, “Java Management Console for DHCP \(OES Linux\),” on page 51](#)

A Novell DHCP server automatically assigns IP addresses and other configuration information to clients upon request or when the clients are restarted. Automatic assignment of configuration information reduces the amount of work required to configure and manage a large IP network.

In addition, integrating DHCP with eDirectory enables you to enter all configuration information into one distributed database. This greatly simplifies network administration and provides for the replication of DHCP configuration information.

The Novell DHCP Service provides the following features:

- ♦ All DHCP configuration is managed in eDirectory, facilitating enterprise-wide management.
- ♦ DHCP options can be set at the following levels:
 - ♦ Service level
 - ♦ Shared Network level
 - ♦ Subnet level
 - ♦ Pool level

- ♦ Host level
- ♦ Class level
- ♦ You can deny service to unwanted devices by creating Host objects identified with the MAC address of the device and setting deny booting for the host.
- ♦ The DHCP software updates `dhcpd.leases` file to record all address assignments to clients.
- ♦ You can use Dynamic DNS (DDNS) to update DNS with information about addresses assigned and rescinded.
- ♦ The DHCP software enables the server to cache addresses and other configuration information from eDirectory for quick response.
- ♦ You can configure the DHCP server to ping an address to verify that no other device is using it before assigning the address to a client.
- ♦ It provides fault tolerance as follows:
 - ♦ A server can survive a temporary local eDirectory service outage and recover automatically.
 - ♦ DHCP configuration is replicated like other eDirectory data.
- ♦ The DHCP server can work with any DNS server.
- ♦ The Failover Peer protocol support allows two DHCP servers to share a common address pool that ensures continuous availability of the network.
- ♦ The import or export DHCP service is supported to transfer the DHCP service configuration from files in Linux `dhcpd.conf` format into eDirectory or from eDirectory to a text file in a `dhcpd.conf` format.

The Novell DHCP service supports the features that were previously provided by Novell Open Enterprise Server 2 Linux and supports the standards of the following RFCs:

- ♦ RFC 2131: Dynamic Host Configuration Protocol
- ♦ RFC 2132: DHCP Options and BOOTP Vendor Extensions
- ♦ RFC 2241: DHCP Options and Novell Directory Services
- ♦ RFC 2242: NetWare/IP Domain Name and Information

Novell DHCP Services also supports the BOOTP standards of the following RFCs:

- ♦ RFC 1497: BOOTP Vendor Information Extensions
- ♦ RFC 1534: Interoperation Between BOOTP and DHCP
- ♦ RFC 1542: Clarifications and Extensions for the Bootstrap Protocol

1.4.1 DHCP Options

- ♦ “DHCP Options for eDirectory” on page 46
- ♦ “NetWare/IP Options” on page 46
- ♦ “DHCP eDirectory Objects” on page 46

The Novell DHCP service supports vendor options, DHCP options, and BOOTP parameters as defined in Internet RFC 2132 with a few exceptions. A table listing the option codes and names is found in [Section A.3, “DHCP Option Descriptions,” on page 198.](#)

DHCP Options for eDirectory

Novell has defined three DHCP options for eDirectory. These options eliminate the need to provide this information each time users log in.

Option 85 provides the IP address of one or more eDirectory servers for the client to contact for access to the eDirectory database. Option 86 provides the name of the eDirectory tree the client will be contacting. Option 87 provides the eDirectory context the client should use.

For detailed information about using these options in OES 2 Linux, refer to Internet ‘RFC 2241’, *DHCP Options for Novell Directory Services*.

NetWare/IP Options

Novell uses option codes 62 and 63 in the DHCP packet for NetWare/IP. Option 62 contains the NetWare/IP domain name.

Option 63, the IPX Compatibility option, contains general configuration information such as the primary DSS, the preferred DSS, and the nearest servers. Option 63 also provides additional information in the form of suboptions, listed in the table below.

Table 1-5 *Suboptions Codes and Meaning*

| Suboption Codes | Meaning |
|-----------------|--|
| 5 | If the value of this field is 1, the client should perform a NetWare Nearest Server Query to find out its nearest NetWare/IP server. |
| 6 | Provides a list of up to five addresses of NetWare Domain SAP/RIP servers. |
| 7 | Provides a list of up to five addresses of the nearest NetWare/IP servers. |
| 8 | Indicates the number of times a NetWare/IP client should attempt to communicate with a given DSS server at startup. |
| 9 | Indicates the amount of delay in seconds between each NetWare/IP client attempt to communicate with a given DSS server at start-up. |
| 10 | If the value is 1, the NetWare/IP client should support NetWare/IP Version 1.1 compatibility. |
| 11 | Identifies the Primary Domain SAP/RIP Service server (DSS) for this NetWare/IP domain. |

Refer to Internet ‘RFC 2242’ and NetWare IP Domain Name and Information for detailed information about using these NetWare/IP options.

DHCP eDirectory Objects

When you select Novell DHCP Services during the OES 2 Linux installation, the eDirectory schema is extended to enable the creation of DHCP objects and the following objects are created:

- ♦ dhcpLocator (Locator object)
- ♦ DHCPGroup (Group object)

Only one copy of these objects exists in an eDirectory tree. The DNS servers, DHCP servers, iManager, and Management Console must have access to these objects.

The DHCPGroup object is a standard eDirectory group object. This object is used to grant the necessary rights to the eDirectory user used by the DHCP server to access the DHCP objects

The dhcpLocator object is created during the OES 2 Linux installation and has references to dhcpServer and dhcpService objects. The creator of the Locator object grants Read and Write permissions to this object to the network administrators.

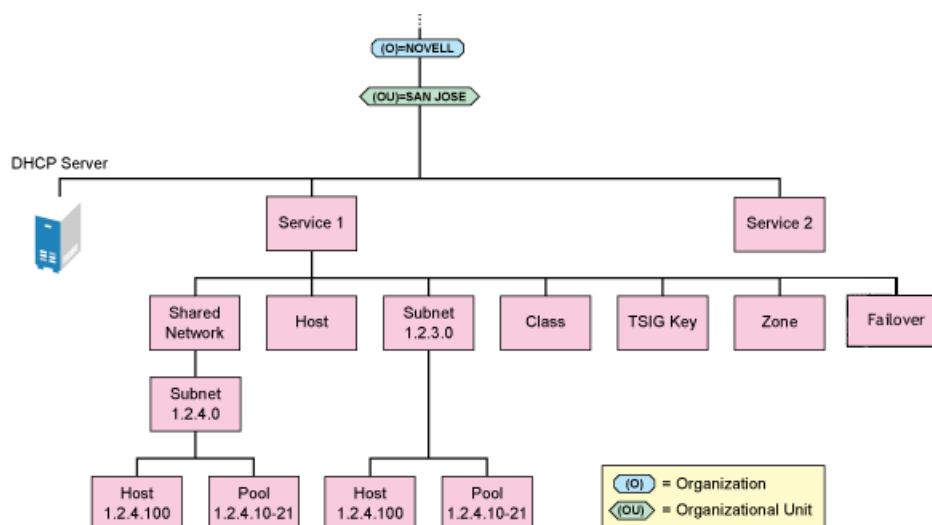
1.4.2 eDirectory Objects for DHCP

The following eDirectory objects are supported in DHCP:

- ♦ “DHCP Service” on page 48
- ♦ “DHCP Server” on page 48
- ♦ “Shared Network” on page 48
- ♦ “Subnet” on page 48
- ♦ “Pool” on page 48
- ♦ “Class” on page 49
- ♦ “Host” on page 49
- ♦ “Zone” on page 49
- ♦ “TSIG Key” on page 49
- ♦ “Failover” on page 49

Figure 1-17 shows a basic configuration of the DHCP objects. This structure might be used for a small to medium-size network.

Figure 1-17 eDirectory Objects for DHCP



DHCP Service

The DHCP Service object is a container object that contains the DHCP configuration for the entire network, a subset of the network, or a single server. It is made up of configuration details of shared networks, subnets, classes, pools and hosts. Depending on your organizational needs, there can be more than one DHCP Service object in the network.

DHCP Server

The DHCP Server object serves one or more DHCP service objects. Each DHCP server can be associated with one or more DHCP services depending on the needs of your organization. In a typical scenario, there must be one DHCP Server object configured for every system running DHCP Service.

Keeping the DHCP server and its configuration details as separate entities provides much-needed flexibility in associating and switching between DHCP servers and configurations.

Shared Network

All subnets that share the same physical network can be grouped under a Shared Network object.

Some installations have physical networks on which more than one IP subnet operates. For example, if there is a site-wide requirement that 8-bit subnet masks be used, but a department with a single physical Ethernet network expands to the point where it has more than 254 nodes, it might be necessary to run two 8-bit subnets on the same Ethernet until a new physical network can be added. In this case, the subnet declarations for these two networks must be enclosed in a shared-network declaration.

A shared network object must be created under a service object.

Subnet

The Subnet object enables you to distribute IP addresses and DHCP options to each network.

Because the DHCP Service object is the fundamental object of the hierarchy of the network, a subnet object must be created under a service element in the hierarchy. If your organizational setup requires the subnet object to be grouped under a shared network, create the subnet object under a service-shared network hierarchy.

The subnet statement is used to provide `dhcpcd` with enough information to tell whether or not an IP address is on that subnet. It can also be used to provide subnet-specific parameters and to specify what addresses can be dynamically allocated to clients booting on that subnet.

Pool

The pool declaration can be used to specify a pool of addresses that are treated differently than another pool of addresses, even on the same subnet. It is also possible to set up entirely different subnets for known and unknown clients. You can create multiple pool objects under a Subnet object. The pool object must be created under a service-shared network-subnet or service-subnet hierarchy.

You can create multiple pool objects under a Subnet object.

The pool object must be created under a service-shared network hierarchy.

Class

The Class object helps in segregating clients into classes, and these clients can be treated differently depending on the class they are in. This separation can be done either with a conditional statement, or with a match statement within the class declaration. It is also possible to create automatic subclasses based on the contents of the client packet. You can also set a limit on the number of clients within a class or subclass. A subclass is a class with the same name as a regular class, but with a specific submatch expression

To group clients into different classes based on conditional expression, you can specify a match expression within a class statement in the following manner:

```
match if substring (option dhcp-client-identifier, 1, 3) = "RAS";
```

A subclass is a class with the same name as a regular class, but with a specific submatch expression that is hashed for quick matching. To automatically create lease-limited subclasses based on client parameters, use the `spawn with` statement.

The option value sent by the client is checked with the dynamically created subclasses for the specified class and if a match is found, the client will be classified under that subclass and treated accordingly.

If no match is found, the server creates a new subclass and logs the information in the lease file, and the client is classified in this new subclass. After classification, it is processed according to the rule set for the class.

Host

The Host object represents a client in the network with statically assigned IP address. It is identified by a host name.

The DHCP Management Utility can be used to configure host objects that are manually assigned. When configuring an individual host object, you can provide specific options that override global options or those set at the subnet/service level.

Zone

The DHCP Zone object contains the references the Domain Name System (DNS).

A DHCP server uses this information to perform dynamic updates for the zone objects. A DNS server must be configured to allow updates for the zone that the DHCP server is updating.

TSIG Key

A TSIG key is used for authenticating dynamic updates to a DNS server. TSIG uses shared secret keys as a cryptographically secure means of authenticating a DNS update.

Failover

The Failover Peer protocol allows only two DHCP servers to share a common address pool. This ensures continuous availability of the network. The process defines the role of a primary server and a secondary server.

Each server has about half of the available IP addresses in the Pool at any given time for allocation. During a prolonged failure of the primary server, the secondary server recovers all the addresses that the primary server had available for allocation, and begins to reuse them.

1.4.3 Managing DHCP Objects

iManager can be used to configure and manage eDirectory-based DHCP objects and can run on any browser workstation. It does not require the Novell Client or any installed component as a prerequisite. It operates within the common iManager framework and is thus tightly integrated with OES 2 Linux.

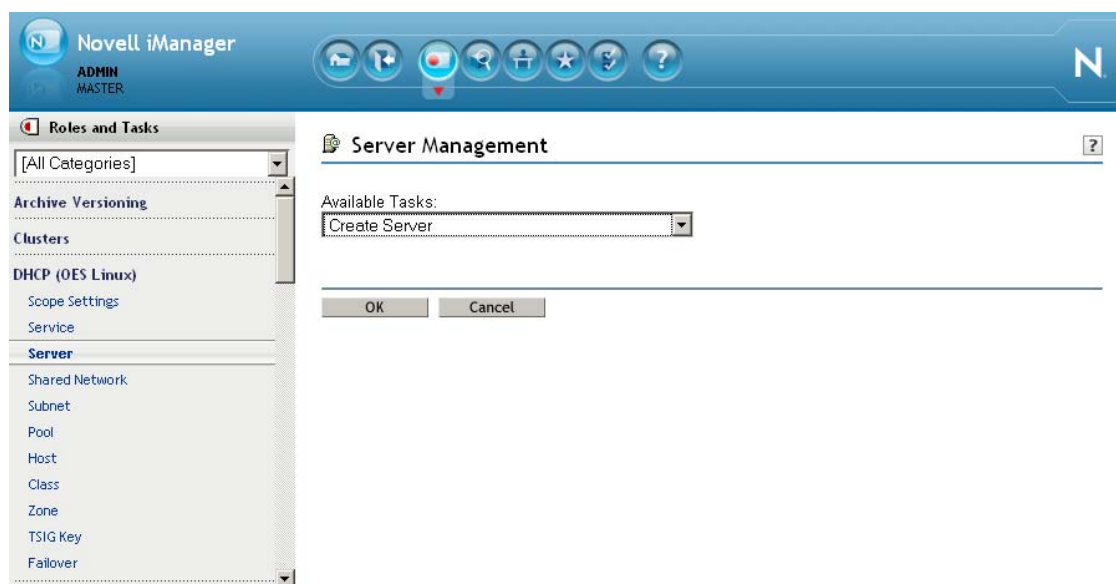
iManager manages one eDirectory tree at a time.

When iManager is started in the browser, the first screen you see is the login screen. You are prompted to provide your username, password, eDirectory context, and the eDirectory tree whose objects you want to manage.

Administration authentication in iManager is based on the common authentication mechanism.

To manage objects in a different eDirectory tree, you must log in to the utility again, specifying the eDirectory tree you want to access. Your login identity is displayed at the top of the screen.

Figure 1-18 The DHCP iManager Interface



The main screen has three parts: a taskbar at the top of the screen that displays icons for top-level management functions; a left panel that displays roles, tasks, and other administrative functions; and a main panel that allows you to manage role-based and administrative tasks.

For more information on the iManager management interface, see the *Novell iManager 2.7 Administration Guide*.

The DHCP services have been logically organized into roles and tasks in a way that is intuitive to network administrators. Each role consists of a set of tasks arranged in a manner that is hierarchical, top-down, and easy to administer.

To view the roles, click the *Roles and Tasks* icon on the taskbar.

The tasks under each of these roles are logically arranged in a top-down manner with the option to configure DHCP scope settings at the head of each role. A role, depending on its current state, is preceded by a plus or a minus sign. An administrator can expand a role such as DHCP (OES Linux) to see the tasks it contains or collapse it for a more concise view.

The organization of roles and tasks follows the containership rules of object creation and manipulation in DHCP. At the top is the DHCP Scope Settings task. At the top is the DHCP Scope Settings task, which allows you to specify the location of the Locator object and the administrative scope for the session.

IMPORTANT: For improved performance, configure the DHCP scope settings before you start using iManager.

1.4.4 Java Management Console for DHCP (OES Linux)

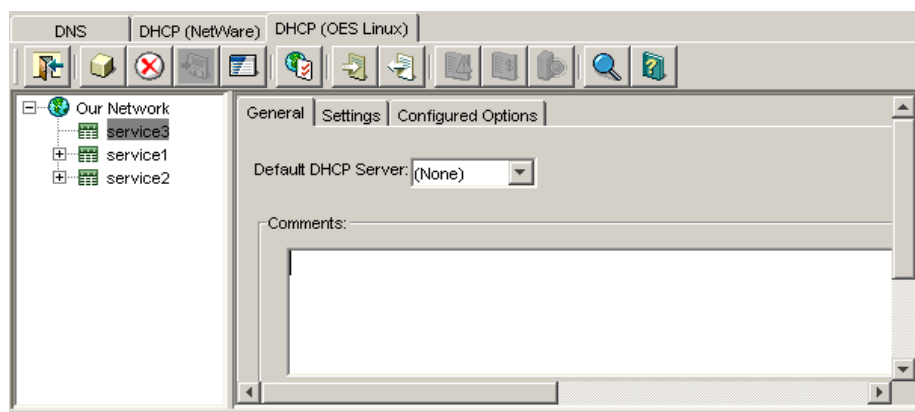
The Java Management Console can be used to configure and manage DHCP (OES Linux) objects based on eDirectory. It is an independent executable Java application. It can be launched through Windows Programs menu by clicking *Start > Programs > DNS-DHCP Management Console > DNSDHCP*. It can also be launched by double-clicking the DNSDHCP shortcut icon created on the desktop.

When the Java Management Console is launched, it prompts you to select a tree as the target eDirectory context.

You must log in to the desired eDirectory tree before launching the Java Management Console. To manage objects in a different eDirectory tree, you must exit the utility, change the context to the other eDirectory tree, and launch the utility again. The current eDirectory tree name is displayed in the utility's title bar.

The Java Management Console manages one tree at a time. **Figure 1-19** shows the main user interface window for DHCP (OES Linux) Services.

Figure 1-19 DHCP (OES Linux) Java-based Management Console User Interface



For more information, see:

- ♦ “DHCP (OES Linux)” on page 52

- ♦ “Toolbar” on page 52
- ♦ “Status Bar” on page 53
- ♦ “Server Status” on page 53
- ♦ “Object Creation Rules” on page 53

DHCP (OES Linux)

DHCP objects can be accessed via the *DHCP (OES Linux)* tab. There are three panes within each tab page. The left pane displays the managed DHCP objects in tree form. The right pane displays detailed information about the object that is selected in the left or bottom pane. The bottom pane lists the Linux DHCP servers configured to provide necessary services.

Resources are organized according to the object hierarchy and the implicit ordering of objects. In the left pane, all the DHCP Service objects are listed in alphanumeric order.

All of the objects are created as eDirectory objects and are subject to Linux Administrator conventions. Therefore, when creating a new object, you should always name the object first in each Create dialog box.

The Create dialog box for these objects has browsing capability in the eDirectory tree, so an administrator with Write or Supervisor rights can select a specific context.

A newly created object's button on the toolbar is context-sensitive in relation to the selected item in either service's left tree pane. Your rights to the objects are not verified until you perform an update, delete, or create against the target objects.

The DHCP objects available in the new object dialog's creation list box depend on the selected object in the left tree pane.

After a new DHCP object has been created, the Java Management Console grants the objects Read and Write rights to the dhcpLocator object.

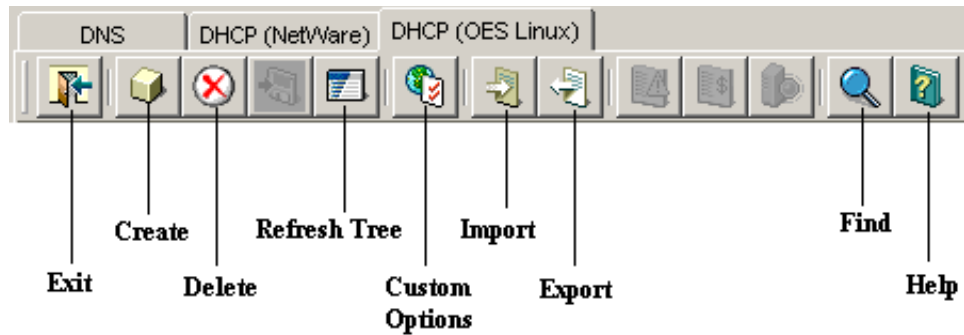
For fast and efficient searching, the distinguished names of newly created Service objects and Server objects are added to the corresponding attribute of the dhcpLocator object. Renaming or deleting these objects is automatically performed by eDirectory because of the built-in feature for eDirectory distinguished names.

After a new DHCP object has been created, the Java Management Console gives you the choice of staying in its current focus or setting the focus on the newly created object. The utility also displays its detailed information in the right pane. This feature is provided as a convenience to administrators and can be used by selecting the *Define Additional Properties* check box.

Toolbar

The Management Console offers no menu items. All functions are provided by the toolbar. The functions that are relevant for the item selected in the left tree pane or bottom server pane are highlighted to show their availability.

Figure 1-20 DHCP (OES Linux) Toolbar

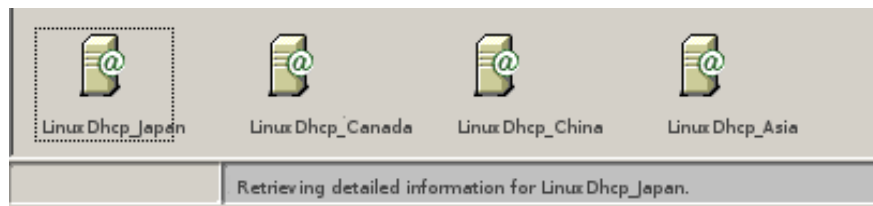


If you position the cursor over the icon, the icon's name appears. The following table lists when each toolbar button is enabled in relationship to the selected object. <Include the same table and change Configured options to Custom options.>

Status Bar

The status bar displays in the bottom pane of the Management Console. When an object is selected, the status of that object is displayed in the status bar. **Figure 1-21** displays the status bar for the Server LinuxDHCP_Japan server.

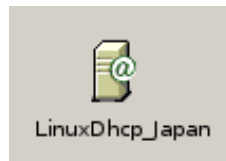
Figure 1-21 DHCP (OES Linux) Status Bar



Server Status

The icon for the Linux DHCP Server does not change as it does for DNS. The icon remains the same no matter what state the server is in. **Figure 1-22** shows the DHCP (OES Linux) server icon.

Figure 1-22 DHCP (OES Linux) Server icons



Object Creation Rules

These are certain rules that govern the creation and manipulation of objects in the Linux DHCP object hierarchy.

1.5 What's Next

The next section covers the enhancements in DNS/DHCP Services on OES 2 Linux:

- ♦ [Chapter 2, “What’s New in DNS/DHCP,” on page 55](#)

What's New in DNS/DHCP

2

Novell® DNS/DHCP Services for OES 2 Linux provides the following enhancements over the previous versions:

- ♦ [Section 2.1, “What’s New in DHCP,” on page 55](#)
- ♦ [Section 2.2, “What’s New in DNS,” on page 55](#)
- ♦ [Section 2.3, “What’s Next,” on page 55](#)

2.1 What's New in DHCP

- ♦ **eDirectory Integration Support:** To simplify the management of IP addresses, DHCP is now integrated tightly with eDirectory™ or OpenLDAP. This gives value to the customers in addition to the file-based implementation available from ISC. Centralized management for DHCP Services extends the LDAP schema and enables you to configure, administer, and manage IP addresses host names through LDAP in an enterprise-wide network.
- ♦ **Migration:** Support for migration from NetWare® to Linux.
- ♦ **New Management Plug-In for DHCP:** A new version of the iManager plug-in provides support for the new directory objects.
- ♦ **NCS Support:** Configuring DHCP with Novell Cluster Services™ ensures that the IP address range required by users to connect to the network is highly available. This is possible because the DHCP server is automatically started, stopped, and restarted on different servers in the cluster by Novell Cluster Services.
- ♦ **Java Console:** A new version of the Java Management Console to manage DHCP objects on Linux is introduced.

2.2 What's New in DNS

- ♦ **rndc Support:** DNS can now be managed through an open source tool called rndc.
- ♦ **Migration:** You can now migrate from NetWare to Linux with the aid of tools.
- ♦ **NCS Support:** Configuring DNS with Novell Cluster Services ensures that the IP address range required by users to connect to the network is highly available. This is possible because the DNS server is automatically started, stopped, and restarted on different servers in the cluster by Novell Cluster Services.
- ♦ **DNS Install:** Novell DNS can now be installed with the DSfW (Domain Services for Windows) pattern.

2.3 What's Next

The next section provides information to plan and design your implementation of DNS and DHCP services in order to maximize capabilities of the Novell DNS/DHCP Services software.

- ♦ [Chapter 3, “Planning Your DNS/DHCP Implementation,” on page 57](#)

Planning Your DNS/DHCP Implementation

3

This section provides information to plan and design your implementation of DNS and DHCP in order to maximize capabilities of these services on OES 2 Linux.

- ♦ [Section 3.1, “Resource Requirements,” on page 57](#)
- ♦ [Section 3.2, “eDirectory Permissions,” on page 57](#)
- ♦ [Section 3.3, “Planning a DNS Strategy,” on page 58](#)
- ♦ [Section 3.4, “Planning a DHCP Strategy,” on page 61](#)

3.1 Resource Requirements

The CPU requirements for DNS on an OES 2 Linux server vary depending on whether you are serving a few static zones without caching or whether you are serving an enterprise DNS.

The memory of the server must be sufficient to accommodate the cache and zones loaded off disk. OES 2 Linux requires at least 512 MB RAM. The `max-cache-size` option can be used to limit the amount of memory used by the cache, at the expense of reducing cache hit rates and creating more DNS traffic. It is still a good practice to have enough memory to load all zones and cache data into the memory. However, the best way to determine this for a given installation is to watch the name server in operation. After a few weeks, the server process should reach a relatively stable size where entries are expiring from the cache as fast as they are being inserted. Ideally, the resource limits should be set higher than this stable size.

3.2 eDirectory Permissions

Consider the following Novell eDirectory™ guidelines to maintain optimal performance when providing DNS and DHCP services on your OES 2 Linux network:

- ♦ Where to locate the DNS and DHCP Group and Locator objects. See [“Understanding DNS and DHCP Services” on page 13](#) for more information about these objects.
- ♦ Where to locate the DNS and DHCP servers
- ♦ What replication strategy to employ
- ♦ How to provide fault tolerance

We recommend the following:

- ♦ Place the DNS and DHCP objects in a separate partition that is accessible from and replicated to all points of the network where OES 2 Linux DNS and DHCP servers are located. For details on the objects in DNS and DHCP, see [“DNS eDirectory Objects and Attributes” on page 27](#) and [“DHCP eDirectory Objects” on page 46](#).

This is required because all OES 2 Linux DNS and DHCP servers, iManager, and the Java Management Console require access to these objects.

- ♦ Plan to create an Organizational Unit (OU) container object near the top of your eDirectory tree. The location of this container object should be easily and widely accessible. Locate the DNS and DHCP Group and Locator objects under the container object.
- ♦ Plan to create an Administrator Group object under the container object near the top of the eDirectory tree. An administrator group should have Read and Write permissions to both DNS and DHCP Locator object attributes except the global data and options fields. Members of this group can use iManager to modify DHCP objects, and can use both iManager and the Java Management Console to manage DNS objects.

IMPORTANT: A network administrator can access only his or her administrative domain, which might not include the DNS and DHCP Locator objects. By creating an administrative group, you enable administrators who are group members to use iManager or the Java Management Console.

- ♦ Plan to locate your DNS and DHCP servers at locations where they are geographically close to the hosts that require their services. Plan to also have one DHCP server in each partition of your network to minimize any WAN communications problems caused by normal load, configuration changes, or replication.
- ♦ Replicate the partition containing the DNS and DHCP Group and Locator objects to all parts of the network that use DNS and DHCP services. This ensures access in the event of system unavailability or hardware problems.

When planning your DNS replication strategy, consider that replication is employed for load balancing when you provide multiple name servers within the DNS zone.

Well-planned replication is the best way to provide fault tolerance for DNS and DHCP services.

3.3 Planning a DNS Strategy

To provide load balancing, fault tolerance, and robustness to your DNS implementation, install and operate a primary name server and at least one secondary name server.

When you configure your zone, the primary name server contains the most up-to-date information about the zone and all the hosts within it.

A secondary name server receives its zone data from the primary name server. When it starts and at periodic intervals, the secondary name server queries the primary name server to determine whether the information it contains has been changed. If the zone information in the secondary name server is older than the zone information in the primary name server, a zone transfer occurs and the secondary name server receives the zone information from the primary name server.

For more information, see the following:

- ♦ [“Planning Zones” on page 59](#)
- ♦ [“Using the Novell DNS Server as a Primary Name Server” on page 59](#)
- ♦ [“Using the Novell DNS Server as a Secondary Name Server” on page 59](#)
- ♦ [“Configuring a DNS Server to Forward Requests” on page 59](#)
- ♦ [“Setting Up the in-addr.arpa Zone” on page 60](#)
- ♦ [“Registering Your DNS Server with Root Servers” on page 61](#)

3.3.1 Planning Zones

If you are running a primary name server and providing DNS service for a zone, the size or geography of your network might require that you create subzones within the zone. Although there is no limitation on the size of a zone when you configure DNS, we recommend that you do not create very large zones. A Novell DNS server can support very large zones, but the higher the number of resource records in a zone, the greater the impact on DNS query resolution for that zone. Managing small zones is simpler and more efficient. You can divide your zones into smaller subzones based on the geographic locations or organizational structures.

Keep the zone data as a separate partition and replicate the partition to all places on your network where you have a name server for the zone. Doing so enables independent replication of the zone data and also provides a degree of fault tolerance in case of server down time.

3.3.2 Using the Novell DNS Server as a Primary Name Server

You must install the Novell DNS server as a primary name server to have authoritative control over your zone and to take advantage of Dynamic DNS (DDNS), which is the dynamic updating of DNS by DHCP.

When operating a Novell DNS server as a primary name server, you can use iManager or the Java Management Console to make configuration changes. When you operate a primary name server, the zone data can receive dynamic updates from DHCP servers. Non-Novell secondary name servers can obtain data from the Novell primary name server.

3.3.3 Using the Novell DNS Server as a Secondary Name Server

If you plan to operate secondary DNS servers that use Novell DNS and DHCP Services software and that connect to a non-Novell master name server, one Novell secondary name server must be specified as the Designated Secondary or zone in server. The Designated Secondary server receives zone transfer information from the non-Novell master server and provides updates to eDirectory. Other Novell secondary name servers can then access the information within eDirectory.

You might connect a Novell secondary name server to a non-Novell master name server for the following reasons:

- ♦ You are using a master DNS server and do not want to designate it as a primary name server because of the responsibility it entails.
- ♦ This approach is easy to implement in your existing DNS model.
- ♦ You want to install more secondary name servers to provide better load balancing.
- ♦ You want to gradually make the transition to operating a primary name server.

3.3.4 Configuring a DNS Server to Forward Requests

If a name server cannot answer a query, it must query a remote server. You can configure primary or secondary name servers to act as forwarders. When you designate a server to be a forwarder, all off-site queries are first sent to the forwarder.

Forwarders that handle the off-site queries develop a robust cache of information. The forwarder probably can answer any given query with information from its cache, eliminating the need to make an outside query to a remote server.

When you decide to make a server a forwarder, configure the other servers in your zone to direct their queries through the forwarder. When a forwarder receives a query, it checks its cache for the information. If the information is unavailable, the forwarder issues a query to the root server.

For more information, see:

- ♦ “Forwarding Requests” on page 60
- ♦ “Restricting Forwarding (No-Forwarding)” on page 60

Forwarding Requests

When you configure your name servers, you must provide information about where to forward requests that the servers cannot answer.

Even if you are using forwarders, a name server that does not receive a timely response from its forwarder eventually attempts to query a root server directly.

Restricting Forwarding (No-Forwarding)

If you have a primary name server with subdomains below it and the primary name server is not aware of the subdomains, the name server sends queries to external name servers.

You can configure your primary name server not to forward queries for specified internal subdomains to external name servers. Instead, the primary name server sends a negative response to any queries for the internal subdomains.

If you want to restrict some external domains, you can use No-Forwarding. You can configure your servers not to forward queries to the specified external domains and the server sends a negative response to queries for those external domain.

3.3.5 Setting Up the Forward Zone Type

If the name server is configured to serve forward zones, all queries for these zones are forwarded to the IP address configured in the Forward list of the zone. For example, if example.com is configured as a Forward Zone and is configured to forward queries to 10.10.10.3, all queries for example.com are forwarded to 10.10.10.3

3.3.6 Setting Up the in-addr.arpa Zone

Just as the data in your name server provides mapping of names to Internet addresses, the in-addr.arpa zone provides mapping of addresses to names. However, in the structure of the in-addr.arpa zone, the IP address appears in reverse. For example, an IP address of 100.20.30.4 in the san-jose.novell.com domain is *4.30.20.100.in-addr.arpa* in the in-addr.arpa subdomain.

3.3.7 Registering Your DNS Server with Root Servers

If you plan to operate a primary DNS name server, you must register your name server with your parent domain. Not all name servers need to be registered, but we recommend registering one-third to one-half of your name servers (up to a maximum of 10) with the parent domain. These servers are queried by servers outside your domain. The remaining name servers are queried only by hosts within your domain that are configured to query them.

If you provide DNS service for other domains and provide an authoritative name server for those domains, you must also register those domains.

To register a domain or subdomain, you must contact the network administrators of the parent domain (com, for example) and the in-addr.arpa domain. Provide the administrators with the name of the domain name server and the name of the domain and any subdomains for which it is authoritative. If you are setting up a new domain, you also need to provide the IP address of any server you want to register.

InterNIC is the organization that registers domain names for the root, com, org, net, edu, and gov domains. To obtain the form for domain registration from InterNIC, contact them at [InterNIC \(http://www.internic.net/\)](http://www.internic.net/). You can also obtain the form for in-addr.arpa domain registration from the same location.

Detailed information about the registration process is available from the InterNIC Web site. You can also use the InterNIC Web site to research domain names to ensure that the name you want is not already registered and to obtain additional information and help.

3.4 Planning a DHCP Strategy

When planning your implementation of DHCP, consider the following:

- ♦ Your existing network topology (how you set up your routers and subnets) provides a basic configuration for the distribution DHCP resources such as Shared Network, Subnet, Pool, Host and Class objects.
- ♦ Your existing eDirectory implementation should be incorporated into your planning. Place the Locator object near the top of your eDirectory tree where it can be easily accessed by all servers.
- ♦ The length of time you set for your leases affects traffic on the network.

For more information, see the following:

- ♦ “Network Topology” on page 62
- ♦ “eDirectory Implementation” on page 62
- ♦ “Lease Considerations” on page 62
- ♦ “IP Address Availability” on page 65
- ♦ “Hostnames” on page 65

3.4.1 Network Topology

Your existing network topology provides a basic configuration for the distribution of DHCP resources. There are two paths. However it depends on whether you are migrating from an existing DHCP solution or you are installing and configuring DHCP for the first time. See the following section:

- ♦ [“Installing the DHCP Service for the First Time” on page 62](#)

Installing the DHCP Service for the First Time

If you are planning to use DHCP for the first time, you must gather a significant amount of information. You need to make a list of all hosts to be served by the DHCP server. You must include all devices that use network addresses in every segment of your network. You must also compile lists of IP address assignments.

Organize your lists of hosts and IP addresses by geographic location. For example, if your network is spread over a WAN, make a list for each location to help you organize the distribution of DHCP resources.

You must have a list of all permanently assigned network addresses. You might also want to make a list of devices that are to be denied IP addresses and those hosts that are to receive strict limitations on leases.

After you gather the necessary information, you need to create the necessary objects to represent this information. This is done by creating pools for contiguous network addresses and other, more specific information. You will probably have a separate pool for each LAN segment of your network. You will also create objects of subnets and DHCP servers. Although there is no limitation on the size or number of subnets when you configure DHCP, we recommend that the IP address in each subnet is not more than 2048. A Novell DHCP server can support several large subnets in a DHCP configuration. However, the greater the number of IP addresses, the greater the impact on DHCP run-time performance.

3.4.2 eDirectory Implementation

Plan to create an Organizational Unit (OU), Country (C), or Locality (L) container object near the top of your eDirectory™ tree.

Plan to locate the Locator objects under the container object. The Locator objects must be easily accessible to all the servers on the network.

Create Subnet objects to represent each LAN segment. Then create one or more pool objects to represent all of your contiguous strings of IP addresses.

For fast access and availability, a DHCP server should be on the same LAN as or geographically close to the writable partitions the DHCP server uses.

3.4.3 Lease Considerations

- ♦ [“Lease Length” on page 63](#)
- ♦ [“Controlling Client Access to Leases” on page 65](#)

In deciding how long to set your client leases, consider the following factors:

- ♦ Your site's and clients' usage patterns
- ♦ Your network's goals
- ♦ Availability of servers
- ♦ Availability of network (IP) addresses

Another important consideration is that clients attempt to renew their leases halfway through the lease duration. The longer the lease, the longer it takes for client configuration changes to be registered with the DHCP server. It also takes longer for the server to realize that a previously assigned address is no longer in use.

Another issue to consider concerns outages and access to the DHCP server. If a client loses access to its DHCP server before renewing its lease, it must stop using the network after the lease expires. If a client is turned on and connected to the network at the time of the outage, however, the lease does not expire.

The longest lease provided by a DHCP server determines the length of time you might have to wait before configuration changes can be propagated within a network. This length of time could mean manually restarting every client or waiting the amount of time required for all leases to be renewed before the changes take effect. If your site policy is to turn off workstation power at the end of the day, clients could acquire configuration changes at least once per day.

NOTE: All lease considerations refer to DHCP clients or devices only. For clients or devices that use BOOTP, you must bring down the device and restart it to acquire any new configuration changes.

Lease Length

When considering the length of leases, ask these questions:

- ♦ Will the default of three days work well in your environment?

The default of three days provides a good balance between a long-lease and a short-lease duration.

- ♦ Do you have more clients than IP addresses?

If you have more clients than IP addresses, keep leases short to allow access to more users. A short lease could be two to four hours, or even a matter of days.

If your site's usage pattern shows that all clients request an address every day and you have half as many addresses as users, lease times in hours or minutes would provide access to more users.

- ♦ Do you provide support for remote access?

If your site has mobile users or provides remote access to clients, plan to provide service for these clients on a specific subnet. Providing support, including special options the clients might require, makes network administration of the clients easier.

- ♦ Do you support a minimum lease time?

If your site's usage pattern indicates that your users typically use an address for only one or two hours, that should be your minimum lease time.

- ♦ How many clients do you plan to support?

Shorter leases support more clients, but shorter leases also increase the load on the DHCP server and network bandwidth. A lease of two hours is long enough to serve most users, and the network load should be negligible. A lease of one hour or less might increase network load to a point that requires attention.

- ♦ How fast are your communications connections between your clients and the DHCP server?

By locating a DHCP server in close proximity to its users, the network load should be negligible over LAN connections. If a DHCP server must communicate over WAN links to provide service to clients, slowdowns and time-outs might occur.

- ♦ How long does your typical server outage last?

If your typical server outage lasts two hours, a lease of four hours would avoid loss of lease to clients that were active at the time of the server outage.

We recommend setting your lease times to twice the length of a typical server outage.

The same recommendation applies to communications line outages. If a communications line is down long enough that leases expire, you might see a significant network load when the service is restored.

- ♦ How long can your clients operate without access to the DHCP server?

If you have users who require a lease for important job functions, consider lease times for them that are twice the length of a maximum server outage. For example, if your DHCP server were to go down on Friday evening and require the entire workday Monday to be restored, that would be an outage of three days. In this case, a six-day lease might be appropriate.

- ♦ Do you have users who advertise their IP addresses for services they render?

If you have users who are setting up Web pages or archiving data for others to access, they want addresses that do not change. You might want to assign permanent addresses for these users instead of assigning long lease times.

The relevant length of time is the maximum amount of time you want to allow a client to keep an address, even if the host computer is turned off. For example, if an employee takes a four-week vacation and you want the employee to keep his or her address, a lease of eight weeks or longer is required.

The following table lists examples of lease times and the reasons these times were chosen.

Table 3-1 *Examples of Lease Times*

| Lease Time | Reason |
|------------|--|
| 15 minutes | Keeps the maximum number of addresses free when there are more users than available addresses, but results in significant traffic and frequent updates to eDirectory |
| 6 hours | Covers a DHCP server outage of 6 hours |
| 12 hours | Ensures that retraction of an address assignment takes less than one day |
| 3 days | Used by many sites simply because of software defaults |
| 6 days | Allows for a weekend server outage without losing leases |
| 4 months | Enables students to keep their address over a summer vacation, for example |

Controlling Client Access to Leases

There is usually a trade-off when you attempt to control specific client access to leases. Typically, you manually configure each client and dedicate an IP address permanently to each client. However, Novell's DHCP server provides control based on the client's hardware address.

3.4.4 IP Address Availability

- ♦ "Identifying Your Addresses" on page 65
- ♦ "Subnetting Your Addresses" on page 65
- ♦ "Assigning Addresses Manually" on page 65

Identifying Your Addresses

If you have been using a previous version of Novell DHCP, another vendor's product, or another method of tracking your IP address information, information about your addresses should be close at hand. To prevent communication problems, we recommend verifying the accuracy of your IP address records by performing a site audit.

If you are unsure of the range of your IP addresses, contact your Internet Service Provider (ISP) or check other records you have on file.

Subnetting Your Addresses

One of the more difficult configuration tasks is configuring your routers if you have multiple subnets. Each might require one or more subnets, depending on your router configuration. Create a Subnet object for each LAN segment that requires dynamic IP address assignment.

Assigning Addresses Manually

Your site might have devices, such as servers and printers, that have addresses assigned by means other than DHCP. Assign addresses to these devices manually.

You must also provide these devices with any specific configuration information they might require. If you want to provide configuration using DHCP, the device must be capable of acting as a DHCP client. You can assign a static address to a device and still provide configuration information through DHCP.

To ensure that the assigned addresses are not used by DHCP, use iManager to exclude the addresses from assignment. You can use the utility to exclude single addresses or entire ranges from address assignment.

3.4.5 Hostnames

Every host on your network that uses the Internet or that can be reached from the Internet should have a name. Each resource record has a hostname field.

The following simple rules are used for hostnames to conform to accepted Internet standards:

- ♦ Hostnames are called labels and can have alphabetic and numeric characters.
- ♦ A hyphen is allowed if it separates two character strings.

- ♦ Labels might not be all numbers, but they can have a leading digit.
- ♦ Labels must begin and end only with a letter or digit.

3.5 What's Next

The next section provides information on the differences between the DNS/DHCP solution on NetWare and the new solution on Novell Open Enterprise Server 2.

- ♦ [Chapter 4, “Running DNS/DHCP Services in a Virtualized Environment,” on page 67](#)

Running DNS/DHCP Services in a Virtualized Environment

4

DNS/DHCP Services runs in a virtualized environment just as it does on a physical Netware® server, or on a physical server running on OES 2 Linux, and requires no special configuration or other changes.

To get started with virtualization, see “[Introduction to Xen Virtualization](#)” in the *Virtualization: Getting Started* guide.

For more information on setting up virtualized NetWare, see “[Setting Up Virtual Machines](#)” in the *Virtualization: Getting Started* guide and “[NetWare Virtual Machines](#)” in the *Virtualization: Guest Operating System Guide*.

For more information on setting up virtualized OES 2 Linux, see “[Setting Up Virtual Machines](#)” in the *Virtualization: Getting Started* guide and “[OES Linux Virtual Machines](#)” in the *Virtualization: Guest Operating System Guide*.

Comparing Linux and NetWare

5

This section describes differences between DNS and DHCP Services on Netware[®] versus DNS/DHCP on Linux. The audience for this section is mainly administrators who are familiar with DHCP on NetWare and intend to move to the new solution available with OES 2 Linux.

- ♦ [Section 5.1, “DNS on NetWare versus DNS on Linux,” on page 69](#)
- ♦ [Section 5.2, “DHCP on NetWare versus DHCP on Linux,” on page 71](#)
- ♦ [Section 5.3, “Limitations,” on page 72](#)
- ♦ [Section 5.4, “What’s Next,” on page 72](#)

5.1 DNS on NetWare versus DNS on Linux

- ♦ [Section 5.1.1, “Management Interface,” on page 69](#)
- ♦ [Section 5.1.2, “Commands,” on page 69](#)
- ♦ [Section 5.1.3, “Filenames and Paths,” on page 70](#)
- ♦ [Section 5.1.4, “Installation Difference,” on page 71](#)
- ♦ [Section 5.1.5, “Features Not Supported,” on page 71](#)

5.1.1 Management Interface

- ♦ [“iManager” on page 69](#)
- ♦ [“Java Console” on page 69](#)

iManager

iManager can be used to configure and manage eDirectory[™] based on DNS and can run on any browser workstation. It does not require the Novell Client or any installed component as a prerequisite.

Java Console

The Management Console is a Java application that provides a graphical user interface to manage the objects created to support DNS and DHCP. The Management Console functions as a standalone utility.

For customers who are moving from Netware to Linux, the management interfaces remain the same as before.

5.1.2 Commands

The following table provides details on difference between commands used to administer DNS Server.

Table 5-1 *Commands to Administer DNS Server*

| Feature | NetWare | Linux |
|---------------------------------------|--------------|--|
| Starting the DNS Server | named | rcnovell-named start or /etc/init.d/novell-named start |
| Stopping the DNS Server | unload named | rcnovell-named stop or /etc/init.d/novell-named stop |
| Checking the status of the DNS Server | m named | rcnovell-named status or /etc/init.d/novell-named status |
| Journal log size | jsize | Specify using the iManager plug-in > max-journal-size field. |

Unsupported Command Line Parameters

On NetWare, the following command line parameters can be used to administer the DNS Server. In the new solution on OES 2 Linux, the parameters are not supported:

- ♦ [-dc categories]
- ♦ [-mstats]
- ♦ [-n number_of_cpus]

For additional details on using novell-named, see [Section 13.4, “novell-named Command Line Options,” on page 165](#)

5.1.3 Filenames and Paths

In OES 2 Linux, the binary names and paths of the files have changed. The following table explains the differences:

Table 5-2 *Difference Between NetWare and Linux Binary Names and Paths*

| Files | On NetWare | On Linux |
|-----------------|--------------------------|---------------------------------------|
| Server Binaries | sys:/system/named.nlm | opt/novell/named/bin/novell-named |
| *.db file | sys:/etc/named/*.db | /etc/opt/novell/named/*.db |
| *.jnl file | sys:/etc/named/*.jnl | /etc/opt/novell/named/*.jnl |
| stat files | sys:/etc/named/named.sta | /var/opt/novell/log/named/named.stats |

| Files | On NetWare | On Linux |
|----------|----------------|---|
| Log file | sys:/named.run | /var/opt/novell/log/named/ named.run |

5.1.4 Installation Difference

- ♦ “Installing on NetWare” on page 71
- ♦ “Installing on Linux” on page 71

Installing on NetWare

On NetWare, installation was managed using the `dnipinst.nlm` utility. This utility also provides the capability to perform upgrade.

Installing on Linux

On OES 2.0 Linux, DNS and DHCP services can be installed through YaST. This method also installs the dependencies.

5.1.5 Features Not Supported

- ♦ Auditing
- ♦ Monitoring
- ♦ SNMP
- ♦ Screen Logging
- ♦ Zone Information

5.2 DHCP on NetWare versus DHCP on Linux

This section addresses the differences between Linux and NetWare. The audience for this section is mainly administrators who are familiar with DHCP on NetWare and intend to move to the new solution on OES 2 Linux.

- ♦ Section 5.2.1, “Management Interface,” on page 71
- ♦ Section 5.2.2, “Commands,” on page 71
- ♦ Section 5.2.3, “Filenames and Paths,” on page 72
- ♦ Section 5.2.4, “Features Not Supported,” on page 72

5.2.1 Management Interface

In OES 2.0 Linux, there is a new iManager plug-in to manage DHCP objects. This plug-in lets you manage the new and existing eDirectory objects with ease.

5.2.2 Commands

The following table provides details on command used to administer DHCP Server.

Table 5-3 *Commands to Administer DHCP Server*

| Command | On NetWare | On Linux |
|--|------------------------------|------------------------------------|
| Starting the DHCP Server | <code>load dhcpsrvr</code> | <code>rcnovell-dhcpd start</code> |
| Stopping the DHCP Server | <code>unload dhcpsrvr</code> | <code>rcnovell-dhcpd stop</code> |
| Checking the status of the DHCP Server | <code>m dhcpsrvr</code> | <code>rcnovell-dhcpd status</code> |

5.2.3 Filenames and Paths

Table 5-4 *Filenames and Paths*

| Files | On NetWare | On Linux |
|--------------------|------------|---|
| Configuration file | NA | <code>/etc/dhcpd.conf</code> |
| Startup Log | NA | <code>/var/log/dhcp-ldap-startup.log</code> This is the dump of DHCP configurations read from eDirectory, when the DHCP Server comes up |
| Server Log | NA | <code>/var/log/dhcpd.log</code> |

5.2.4 Features Not Supported

- ♦ SNMP Support
- ♦ Auditing

5.3 Limitations

- ♦ The YaST plug-in supports only SUSE® Linux Enterprise Server (SLES) DNS servers. If you want manage a Novell DNS server, use iManager or the Java Management Console.

5.4 What's Next

The next section describes methods to install and configure DHCP.

- ♦ [Chapter 6, “Installing and Configuring DHCP,” on page 73](#)

Installing and Configuring DHCP

6

This section describes how to install and configure the Novell® DHCP service on Novell Open Enterprise Server 2 SP1.

- ♦ [Section 6.1, “Planning Your Installation,” on page 73](#)
- ♦ [Section 6.2, “Installing DHCP,” on page 75](#)
- ♦ [Section 6.3, “Verifying the Installation,” on page 79](#)
- ♦ [Section 6.4, “Upgrading to OES 2 Linux from OES 1 Linux,” on page 80](#)
- ♦ [Section 6.5, “What’s Next,” on page 80](#)

6.1 Planning Your Installation

Before you start the installation process for DHCP, review the following:

- ♦ [Section 6.1.1, “Prerequisites,” on page 73](#)
- ♦ [Section 6.1.2, “eDirectory Permissions,” on page 73](#)
- ♦ [Section 6.1.3, “Recommendations,” on page 74](#)

6.1.1 Prerequisites

- ♦ iManager or the Java Management Console is required for creation of the DHCP Service object. For details see [Section 1.4.2, “eDirectory Objects for DHCP,” on page 47](#).

IMPORTANT: Do not install any of the following service combinations on the same server as Novell DHCP. Although not all of the combinations cause pattern conflict warnings, Novell does not support any of the combinations shown.

- ♦ Xen™ Virtual Machine Host Server
-

6.1.2 eDirectory Permissions

- ♦ [“First-Time Installation on an eDirectory Tree” on page 73](#)
- ♦ [“Installing on an eDirectory Tree Where a DHCP Server Already Exists \(Separate Container\)” on page 74](#)

First-Time Installation on an eDirectory Tree

If you are installing OES 2 SP1 Linux DHCP Server on an eDirectory tree for the first time, you require the following rights on the Server, Locator, and Group containers at the eDirectory™ level before you start the DHCP installation.

- ♦ Create permission at the entry level of the container for the following:
 - ♦ DHCP Server Object Container
 - ♦ Locator Object Container (dhcpLocator)

- ♦ Group Object Container (DHCPGroup)
- ♦ DHCP Proxy User Object Container

Installing on an eDirectory Tree Where a DHCP Server Already Exists (Separate Container)

If you are installing OES 2 SP1 Linux DHCP Server on an eDirectory tree, make sure that you have the following permissions:

- ♦ Create permissions at the entry level of the container for the following objects:
 - ♦ DHCP Server Object Container
 - ♦ DHCP Proxy User Object Container
- ♦ Read and Write permissions for the existing objects at attribute level of the objects.
 - ♦ Locator Object Container (dhcpLocator)
 - ♦ Group Object Container (DHCPGroup)
- ♦ To retrieve the existing Locator and Group objects context, specify the following rights:
Read and compare rights at the attribute level and browse rights at the entry level of the container where DHCP Locator and Group objects are present.

Table 6-1 eDirectory Object and Attribute Permissions

| DHCP Objects and Attributes | Permissions |
|-----------------------------|--|
| dhcpLocator | Read, Write |
| DHCPGroup | Read, Write |
| DHCP Server object | Create |
| Proxy User | Create rights at the entry level of the proxy user container |

6.1.3 Recommendations

- ♦ Create the dhcpLocator object and the DHCPGroup object at the top of the eDirectory tree.
- ♦ Replicate the eDirectory tree on the server running DHCP so that the search for the objects is fast.

NOTE: In the eDirectory tree, only a single copy of dhcpLocator object and the DHCPGroup object must exist. If the dhcpLocator and DHCPGroup objects already exist, then Read permissions for the DHCPGroup object and Write permissions for the dhcpLocator object are required.

6.2 Installing DHCP

YaST Install

There is a predefined system of installing components along with the associated dependencies. For a service to function properly, all the dependent products must be installed. Pattern deployment provides patterns for different services. Selecting a pattern automatically selects and installs its dependencies:

- 1 In the YaST install, on the *Installation Settings* page, click Software to go to the Software Selections and System Tasks page.
- 2 Under the OES Services option, select *Novell DHCP*.
- 3 Click *Accept*.

After the installation is done, the system reboots for the changes to take effect.

- 4 Follow the on-screen instructions, specifying configuration parameters in each of the pages shown below. Refer [Table 6-2](#) for details on the fields.

Table 6-2 Fields - Configuration Parameters and Details

| Configuration Parameter | Details |
|-------------------------|---|
| DHCP Server Context | Specify a context for the DHCP Server object. For example: o=novell |
| DHCP Server Object Name | <p>Specify the name of the Server object that these DHCP services will be running on.</p> <p>For example: DHCP_servername.</p> <p>This is the DHCP server object that contains a list of DHCP Services (configuration) served by the DHCP Server.</p> |
| DHCP Locator Context | <p>Specify the context for the DHCP Locator object. For example: o=novell.</p> <p>The DHCP Locator object has references to dhcpServer and dhcpService objects.</p> |
| Group Context | <p>Specify the context for the DHCP Group object. For example: o=novell.</p> <p>This object is used to grant the necessary rights to the eDirectory user used by the DHCP server to access the DHCP objects.</p> |
| Log File Location | <p>Specify the path and filename for the DHCP server to dump the configurations it reads from eDirectory. Specify the path manually or click <i>Browse</i> to locate the log.</p> <p>The default path is /var/log/dhcp-ldap-startup.log.</p> |

| Configuration Parameter | Details |
|--|--|
| LDAP Method | <p>Select <i>Static</i> if you do not want the DHCP server to query the LDAP server for host details.</p> <p>Select <i>Dynamic</i> if you want the DHCP server to query for host details from the LDAP server for every request.</p> <p>Selecting the dynamic LDAP method ensures that the responses you receive to queries are accurate, but the server takes a longer time to respond.</p> |
| Referrals | <p>Select <i>Chase Referral</i> if you want the DHCP server to follow referrals. Otherwise, select the <i>Do Not Chase Referral</i> option.</p> <p>A referral is a message that the LDAP server sends to the LDAP client informing it that the server cannot provide complete results and that more data might be on another LDAP server.</p> |
| eDirectory Server address or host name | <p>The IP address shown is the default LDAP server for this service. If you do not want to use the default, select a different LDAP server in the list.</p> <p>If you are installing into an existing tree, ensure that the server you select has a master replica or read/write replica of eDirectory.</p> <p>If you need to add another LDAP server to the list, add it by using the LDAP Configuration for Open Enterprise Services dialog box.</p> |
| Use secure channel for configuration | <p>This option is selected by default. When you are configuring DHCP services, it ensures that all configuration is transferred over a secure channel.</p> <p>Deselecting this option lets a user with fewer privileges configure LDAP services.</p> |
| LDAP User Name with Context | <p>Specify a distinguished name and context for an LDAP user. For example: cn=joe, o=novell.</p> <p>This user should be an eDirectory user that can access the DHCP server.</p> |
| LDAP User Password | Type a password for the LDAP user. |
| LDAP port for DHCP Server | Select a port for the LDAP operations to use. |
| Secure Channel for DHCP Server | <p>This option is selected by default to ensure that the data transferred between DHCP server and LDAP server is secure and private.</p> <p>If you deselect this option, the data transferred is in clear text format.</p> |

| Configuration Parameter | Details |
|--------------------------------------|--|
| Request Certificate | <p>Specifies the checks to perform on the server certificate in a SSL/TLS session. Select one of the following options:</p> <p>Never: The server does not ask the client for a certificate.</p> <p>Allow: The server requests for a client certificate but if a certificate is not provided or a wrong certificate is provided, the session still proceeds normally.</p> <p>Try: The server requests the certificate. If none is provided, the session proceeds normally. If a certificate is provided and it cannot be verified, the session is immediately terminated.</p> <p>Hard: The server requests a CA certificate and a valid certificate must be provided, or the session is immediately terminated.</p> |
| LDAP CA File | The LDAP CA file contains CA certificates |
| LDAP client certificate | The LDAP client certificate contains the client certificate. The client is a user, service, or any client. |
| LDAP client Key file | The LDAP client key file contains the key file for the client certificate. |
| Network Cards for Novell DHCP Server | The network cards for novell dhcp server contains the network interfaces to which the DHCP server should listen from the available interfaces. |

Novell DHCP Services Configuration

Use this dialog to specify options for configuring a DHCP server that is integrated with eDirectory on this server.

DHCP Server Context
Specify a context for the DHCP Server object.
For example: o=novell

DHCP Server Object Name
Specify the name of the Server object that these DHCP services will be running on.
For example: DHCP_servicename

This is the DHCP server object that contains a list of DHCP Services (configuration) served by the DHCP Server.

DHCP Locator Context
Specify the context for the DHCP Locator object.
For example: o=novell

The DHCP Locator object has references to dhcpServer and dhcpService objects.

Group Context
Specify the context for the DHCP Group object.
For example: o=novell

This object is used to grant the necessary rights to the eDirectory user used by DHCP Server to access the DHCP objects.

DHCP Server context (e.g. o=novell)
ou=novell,ou=vix-child,ou=rc11,ou=novell

DHCP Server object name
DHCP_server

Common DHCP Configuration Object Contexts

Locator context (e.g. o=novell)
o=novell Retrieve

Group context (e.g. o=novell)
o=novell

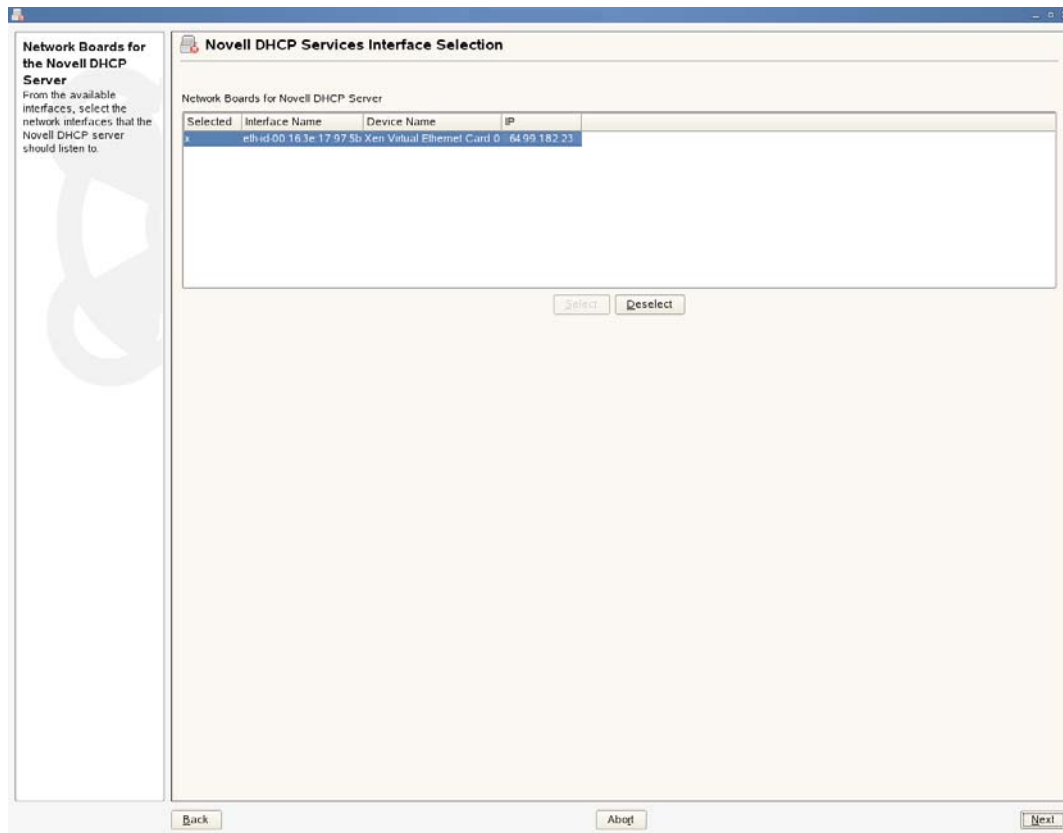
Log file location
/var/log/dhcp-ldap-startup.log Browse

LDAP method
☒ Static
☐ Dynamic

Referrals
☒ Chase referral
☐ Do not chase referral

Back Abort Next

78 OES 2 SP1: Novell DNS/DHCP Administration Guide for Linux



- 5 Click *Next* to complete the installation process.

6.3 Verifying the Installation

After the installation is done, you can verify the installation by using the following procedure:

- 1 Use iManager to check for the presence of the following objects in the eDirectory tree:
 - ♦ **dhcpLocator:** This object is in the Locator context specified during installation.
 - ♦ **DHCPGroup:** This object is in the Group context specified during installation.
 - ♦ **DHCP_<hostname>:** This object is in the DHCP server context specified during installation.

The schema is extended only during the first installation. For subsequent installs, you only need to verify DHCP Server object creation.

- 2 Check the following parameters in the `/etc/dhcpd.conf` file:

```
ldap-server
ldap-port
ldap-ssl
ldap-base-dn
ldap-method
ldap-debug-file
ldap-dhcp-server-cn
ldap-referrals
```

- 3 (Conditional) If you selected *Use Secure Channel for DHCP Server* option and *Request Certificate* as *Never*, check the `ldap_tls_reqert` parameter included in the `dhcpd.conf` file.
- 4 (Conditional) If you selected *Use Secure Channel for DHCP Server* option and *Request Certificate* as *Try/Allow/Hard*, check the following parameters and the availability of the following certificate files:

```
ldap-tls-reqert  
ldap--tls-ca-file  
ldap-tls-cert  
ldap-tls-key
```

6.4 Upgrading to OES 2 Linux from OES 1 Linux

During an upgrade from OES 1 Linux (file-based) to OES 2 (eDirectory based) the following steps are performed:

- ♦ The existing `/etc/dhcpd.conf` file is backed up as `/etc/dhcpd.conf.bak`
- ♦ The existing `/etc/dhcpd.conf` file is overwritten with new values for the following parameters:

```
ldap-server  
ldap-port  
ldap-ssl  
ldap-base-dn  
ldap-method  
ldap-debug-file  
ldap-dhcp-server-cn  
ldap-referrals  
ldap certificate details
```

To verify if the upgrade was successful, start the DHCP server by using the `rcnovell-dhcpd start` command.

6.5 What's Next

The next section describes methods to administer the DHCP server and manage it through iManager.

- ♦ [Chapter 7, “Administering and Managing DHCP,” on page 81](#)

Administering and Managing DHCP

7

The following sections indicate the steps that need to be executed to configure and manage a DHCP server:

- ♦ [Section 7.1, “Using iManager to Manage DHCP,” on page 81](#)
- ♦ [Section 7.2, “Using the Java Management Console for DHCP \(OES Linux\),” on page 99](#)
- ♦ [Section 7.3, “Starting the DHCP Server,” on page 108](#)
- ♦ [Section 7.4, “Stopping the DHCP Server,” on page 108](#)
- ♦ [Section 7.5, “What’s Next,” on page 108](#)

7.1 Using iManager to Manage DHCP

This section provides details on using iManager to manage objects in eDirectory™.

- ♦ [Section 7.1.1, “Scope Setting,” on page 81](#)
- ♦ [Section 7.1.2, “Service Management,” on page 82](#)
- ♦ [Section 7.1.3, “Server Management,” on page 83](#)
- ♦ [Section 7.1.4, “Shared Network Management,” on page 85](#)
- ♦ [Section 7.1.5, “Subnet Management,” on page 86](#)
- ♦ [Section 7.1.6, “Pool Management,” on page 88](#)
- ♦ [Section 7.1.7, “Host Management,” on page 90](#)
- ♦ [Section 7.1.8, “Class Management,” on page 92](#)
- ♦ [Section 7.1.9, “Zone Management,” on page 93](#)
- ♦ [Section 7.1.10, “TSIG Key Management,” on page 95](#)
- ♦ [Section 7.1.11, “Failover Peer Management,” on page 96](#)
- ♦ [Section 7.1.12, “Importing and Exporting the DHCP Configuration,” on page 98](#)

7.1.1 Scope Setting

Configuring the scope settings for a session significantly improves the session's performance. If you do not configure the scope settings for the session, you receive a warning before every task you attempt to perform. However, you can still proceed with the task.

Setting the scope of the DHCP services requires two specifications for the session: the Novell® eDirectory context of the Locator object and the administrative scope of the session. Specifying the eDirectory context of the Locator object at the start of the session significantly improves performance because it eliminates the need to search for the Locator object. Specifying the administrative scope of the session also improves performance significantly because it restricts the retrieval of DHCP objects for viewing to the scope you specify.

When you configure the DHCP scope settings for a session, they only last as long as the session lasts. If you start a new session, you must configure the DHCP Scope Settings again.

To configure DHCP scope settings:

- 1 In iManager, click *DHCP (OES Linux)* > *Scope Settings* to open the Scope Settings window.
- 2 Specify the eDirectory context of the DHCP Locator object or browse to select it.
- 3 Specify the eDirectory context of the container object or browse to select it. This provides the administrative scope of the current session.

If you specify only the eDirectory context of the DHCP Locator object and not the administrative scope of the current session, you can proceed with administrative tasks without receiving a warning message. However, performance is further optimized if you also define the administrative scope.

- 4 Click *OK to configure the scope settings*.

A message indicates that the scope setting request was successful.

- 5 Click *OK* to complete the process.

Or

- 6 Click *Repeat Task* to configure the scope settings again.

7.1.2 Service Management

The Service object acts as a container object for subnet, shared network, host, zone, class, TSIG key, and failover.

The Service Management role consists of the following tasks:

- ♦ “Creating a Service” on page 82
- ♦ “Viewing or Modifying a Service” on page 82
- ♦ “Deleting a Service” on page 83

Creating a Service

- 1 In iManager, click *DHCP (OES Linux)* > *Service* to open the Service Management window in the main panel.
- 2 From the drop-down menu, select *Create Service*, then click *OK* to open the Create Service window.
- 3 Specify the name of the service.
- 4 Specify the name of the eDirectory context or browse to select it.
- 5 Select a server from the drop-down list.
- 6 Click *Create*.

A message indicates that the new DHCP service object has been created.

Viewing or Modifying a Service

- 1 In iManager, click *DHCP (OES Linux)* > *Service* to open the Service Management window in the main panel.

- 2 From the drop-down menu, select *View/Modify Service*, then click *OK* to open the View/Modify Service window.
- 3 Select the DHCP service from the drop-down menu, then click *OK*.
- 4 Modify the attributes of the service.

| Attributes | Function |
|--------------------|--|
| General | You can use this task to assign DHCP servers for a service. You can also record comments against a particular service. |
| Available Options | Displays the options that the server can use to communicate with the client. These are the predefined options in the system and it is not possible to modify them. In addition to the existing options, you can declare custom options. |
| Settings | These settings are used to define configuration settings for a service. |
| Configured Options | <p>You can define values for the predefined options that are declared in the <i>Available Options</i> list.</p> <p>For example, <i>Time Offset</i> is a predefined option in the <i>Available Options</i>. Use the Configured Options task to set a value for the <i>Time Offset</i> option.</p> |

Deleting a Service

- 1 In iManager, click *DHCP (OES Linux) > Service* to open the Service Management window in the main panel.
- 2 From the drop-down menu, select *Delete Service*, then click *OK* to open the Delete Service window.
 - ♦ To remove all the services in the list, click the top-level check box, then click *Delete*.
 - or
 - ♦ To remove one or more services, click the check box next to the service, then click *Delete*.

7.1.3 Server Management

DHCP uses the client-server structure to allocate network addresses to a host.

The DHCP server reads its configuration information from eDirectory and stores the information in its cache. The Server Management role consists of the following tasks:

- ♦ “Creating a Server” on page 83
- ♦ “Viewing or Modifying a Server” on page 84
- ♦ “Deleting a Server” on page 84
- ♦ “Loading or Unloading a Server” on page 84

Creating a Server

- 1 In iManager, click *DHCP (OES Linux) > Server* to open the Server Management window in the main panel.

- 2 From the drop-down menu, select *Create Server*, then click *OK* to open the Create Server window.
- 3 Specify the name of the server.
- 4 Specify the eDirectory context or browse to select it.
- 5 Click *Create*.

A message indicates that the new server object has been created.

Viewing or Modifying a Server

- 1 In iManager, click *DHCP (OES Linux) > Server* to open the Server Management window in the main panel.
- 2 From the drop-down menu, select *View/Modify Server*, then click *OK* to open the View/ Modify Server window.
- 3 Select the server object from the drop-down menu.
- 4 Click *OK*.
- 5 Modify the attributes of the server.

| Attribute | Function |
|-----------|--|
| Settings | Indicates all of the configuration parameters that can be defined for the server |
| General | lists the services that are assigned to this server. You can provide comments for every service that is defined. |

Deleting a Server

- 1 In iManager, click *DHCP (OES Linux) > Server* to open the Server Management window in the main panel.
- 2 From the drop-down menu, select *Delete Server*, then click *OK* to open the Delete Server window.
 - ♦ To remove all servers in the list, click the top-level check box, then click *Delete*.
 - or
 - ♦ To remove one or more DHCP servers, click the check box next to it, then click *Delete*.

Loading or Unloading a Server

IMPORTANT: To load/unload a server, *owcimomd* daemon must be running on the remote server. The user on the remote server must be LUM enabled and should have supervisor rights at the entry level to the UNIX workstation object.

- 1 Click *DHCP (OES Linux) > Server Management* to open the Server Management window in the main panel.
- 2 From the drop-down menu, select *Load/Unload Server*, then click *OK* to open the Load/Unload Server window.
- 3 Specify the name or IP address of the server you want to load/unload.
- 4 Click *OK*.

The server name and server status is displayed. Depending on the server status, you can choose to load or unload the DHCP server.

- 5 To load a server, select the *Load Server* option.

or

To unload a server, select the *Unload Server* option.

- 6 Click *OK* to confirm the action.

For the Load or Unload task to work, the `owcimond` daemon should be running on the remote server and the remote server must be LUM-enabled.

7.1.4 Shared Network Management

All subnets that share the same physical network can be grouped under a Shared Network object.

The Shared Network Management role consists of the following tasks:

- ♦ “Creating a Shared Network” on page 85
- ♦ “Viewing or Modifying a Shared Network” on page 85
- ♦ “Deleting a Shared Network Object” on page 86

Creating a Shared Network

- 1 In iManager, click *DHCP (OES Linux) > Shared Network* to open the Shared Network Management window in the main panel.
- 2 From the drop-down menu, select *Create Shared Network*, then click *OK* to open the Create DHCP Shared Network window.
- 3 Specify the name of the shared network.
- 4 Select a service from the drop-down list.
- 5 Click *Create*.

A message indicates that the new Shared Network has been created.

Viewing or Modifying a Shared Network

- 1 In iManager, click *DHCP (OES Linux) > Shared Network* to open the Shared Network Management window in the main panel.
- 2 From the drop-down menu, select *View/Modify Shared Network*, then click *OK* to open the View/Modify Shared Network window.
- 3 Select a service from the list.
- 4 All of the shared networks created under the service are listed in the drop-down list. Select the shared network you want to modify, then click *OK*.
- 5 Modify the attributes of the shared network.

| Attribute | Function |
|-----------|--|
| General | Lists the subnets that are serviced by this shared network. You can provide comments for every subnet that is defined. |

| Attribute | Function |
|--------------------|---|
| Settings | These settings are used to define configuration settings for a shared network |
| Configured Options | <p>You can define values for the predefined options that are declared in the Available Option table.</p> <p>For example, Time Offset is a predefined option in the Available Option list. You use the Configured Options task to set a value for the <i>Time Offset</i> option.</p> |

Deleting a Shared Network Object

- 1 In iManager, click *DHCP (OES Linux) > Shared Network* to open the Shared Network Management window in the main panel.
- 2 From the drop-down menu, select *Delete Shared Network*, then click *OK* to open the Delete Shared Network window.
- 3 Select a service from the list. All the Shared Networks created under the service are listed below.
 - ♦ To remove all the Shared Network objects in the list, click the top-level check box, then click *Delete*.

or

 - ♦ To remove one or more than one Shared Network object, click the check box next to the object, then click *Delete*.

7.1.5 Subnet Management

The Subnet object is the most fundamental DHCP object. It enables you to distribute IP addresses and DHCP options to each network.

The Subnet object contains host definitions for fixed address allocation and pools with address range.

A Subnet object's specific DHCP options and configuration parameters apply to the entire subnet and override the options set at the shared network and service levels.

The Subnet Management role consists of the following tasks:

- ♦ [“Creating a Subnet Object” on page 86](#)
- ♦ [“Viewing or Modifying a Subnet Object” on page 87](#)
- ♦ [“Deleting a Subnet Object” on page 88](#)

Creating a Subnet Object

- 1 In iManager, click *DHCP (OES Linux) > Subnet* to open the Subnet Management window in the main panel.
- 2 From the drop-down menu, select *Create Subnet*, then click *OK* to open the Create Subnet window.

- 3 Select a service from the drop-down list. It is mandatory to select a service, because a Subnet can only be created under a service or a Shared Network.
- 4 If you want the Subnet object to be created under service and shared network hierarchy, select a Shared Network object from the drop-down list.
- 5 Specify a subnet address and a subnet mask in the fields provided.
- 6 Click *Create*.

A message indicates that the new subnet has been created.

Viewing or Modifying a Subnet Object

- 1 In iManager, click *DHCP (OES Linux) > Subnet* to open the Subnet Management window in the main panel.
- 2 From the drop-down menu, select *View/Modify Subnet*, then click *OK* to open the View/Modify Subnet window.
- 3 Select the Service from the drop-down list.
If you have created Subnets under a Service and Shared Network hierarchy, the associated Shared Networks and Subnets are displayed in the drop-down list.
- 4 Modify the attributes of the subnet.

| Attribute | Function |
|------------------------|--|
| General | Select a zone from the drop-down list. The specified zone is updated when an IP address from this subnet is leased or when the lease expires. You can record comments about the Subnet object you are modifying. |
| Lease | |
| Lease Time | Specifies the length of time for an address assignment. A lease time can be either permanent or timed. Permanent leases never expire; the client is assigned an IP address for an indefinite period. Timed leases are defined in days, hours, or minutes. Timed leases expire, unless the client renews the lease. |
| Boot Parameter Options | Select this option to specify the server address, server name, and boot file name for the BOOTP service. This is the address and name of a server the client can contact for a boot image, as well as a boot file name. |
| Settings | These settings are used to define configuration settings for a subnet: |
| Configured Options: | You can define values for the predefined options that are declared in the Available Option table. For example, Time Offset is a predefined option in the Available Option list. You use the Configured Options task to set a value for the <i>Time Offset</i> option. |

Deleting a Subnet Object

- 1 In iManager, click *DHCP (OES Linux) > Subnet* to open the Subnet Management window in the main panel.
- 2 From the drop-down menu, select *Delete Subnet*, then click *OK* to open the Delete Subnet window.
- 3 Select a Service from the list. All the Subnets created directly under service are listed below.
The default value for a shared network is None. To list the Subnets created under a Service and Shared Network object hierarchy, select the associated Shared Network object from the drop-down list.
- 4 To remove all of the Subnet objects in the list, click the top-level check box, then click *Delete*
or
To remove one or more Subnet object, click the check boxes next to the object, then click *Delete*.

7.1.6 Pool Management

The Pool object represents a range of addresses for dynamic address assignment or for exclusion from the address assignment.

The Pool Management role consists of the following tasks:

- ♦ “Creating a Pool Object” on page 88
- ♦ “Viewing or Modifying a Pool Object” on page 88
- ♦ “Deleting a Pool Object” on page 89

Creating a Pool Object

- 1 In iManager, click *DHCP (OES Linux) > Pool* to open the Pool Management window in the main panel.
- 2 From the drop-down menu, select *Create Pool*, then click *OK* to open the Create Pool window.
- 3 Select a service from the drop-down list. All of the subnets created under the service and the service or shared network hierarchy are displayed.
- 4 Select a subnet from the drop-down list.
- 5 In the *Pool Name* field, type the name of the pool.
- 6 Specify the lower and upper limits of the address range in the *Start Address* and *End Address* fields respectively.
- 7 Click *Create*.

A message indicates that the new pool has been created.

Viewing or Modifying a Pool Object

- 1 In iManager, click *DHCP (OES Linux) > Pool* to open the Pool Management window in the main panel.
- 2 From the drop-down menu, select *View/Modify Pool*, then click *OK* to open the View/Modify Pool window.

- 3 Select a service from the list.
- 4 If there is more than one subnet associated with a service, select the subnet from the list.
- 5 All of the pools associated with the selected subnet are displayed. Select the one you want to modify from the drop-down list, then click *OK*.
- 6 Modify the attributes of the pool.

| Attribute | Function |
|------------------------------------|--|
| General Tab | |
| Range Type drop-down | |
| DHCP | <p>A range of addresses used by the DHCP server to assign addresses to clients making only DHCP requests.</p> <p>You can enable the DNS Update Option parameter by assigning the Dynamic DHCP range type. If <i>Always Update</i> is selected, the DHCP server updates DNS as dynamic addresses are assigned and released.</p> |
| BOOTP and DHCP | <p>A range of addresses used by the DHCP server to assign addresses to clients that send a DHCP or BOOTP request.</p> <p>You can enable the DNS Update Option parameter by assigning the Dynamic BOOTP and DHCP range type. If <i>Always Update</i> is selected, the DHCP server updates DNS as dynamic addresses are assigned and released.</p> |
| Excluded | The range of addresses to be excluded by the DHCP server while assigning IP addresses. |
| Select the Failover Peer drop-down | Specify the Failover Peer from the drop-down list. |
| DNS Update drop-down | If you choose to do a Dynamic update, then select <i>Always Update</i> from the DNS Update drop-down. |
| Comments | You can type your comments about the Pool in this box. This is optional. |
| Settings | These settings are used to define configuration settings for a pool. |
| Configured Option | <p>You can define values for the predefined options that are declared in the <i>Available Option</i> list.</p> <p>For example, Time Offset is a predefined option in the <i>Available Option</i> list. You use the Configured Options task to set a value for the <i>Time Offset</i> option.</p> |

Deleting a Pool Object

- 1 In iManager, click *DHCP (OES Linux) > Pool* to open the Pool Management window in the main panel.
- 2 From the drop-down menu, select *Delete Pool*, then click *OK* to open the Delete Pool window. Select the service and subnet that contains the address pool(s) to delete.
 - ♦ To delete all the pools, click the top-level check box, then click *Delete*.

or

- ♦ To delete one or more pool objects, click the check box next to the object, then click *Delete*.

7.1.7 Host Management

The Host object represents a client in the network with statically assigned IP address and is identified by a hostname.

You can use iManager to configure Host objects that are manually assigned. For dynamically or automatically assigned client addresses, DHCP updates the `dhcpd.leases` file in the `/var/lib/dhcp/db` directory.

A Host object cannot be defined within a pool defined for dynamic allocation. If you must define the object, then the subnet should be divided with all static allocations going into one address range and all other addresses allocated for dynamic allocation.

When configuring an individual Host object, you can provide specific options that override global options or those set at the subnet/service level.

The Host Management role consists of the following tasks:

- ♦ “Creating a Host” on page 90
- ♦ “Viewing or Modifying a Host” on page 91
- ♦ “Deleting a Host Object” on page 91

Creating a Host

- 1 In iManager, click *DHCP (OES Linux) > Host* to open the Host Management window in the main panel.
- 2 From the drop-down menu, select *Create Host*, then click *OK* to open the Create Host window.
- 3 Specify the hostname.
- 4 Select a Service under which the Host object has to be created. It is mandatory to select a service.

The default value of the Subnet object is None. If you want the Host object to be created under service-subnet network hierarchy, select a Subnet object from the drop-down list.

- 5 Specify the name by which you want to identify the host.
- 6 Specify the IP address of the host.
- 7 Specify the client identifier. This uniquely identifies the client.
- 8 Select the MAC type from the drop-down list.
- 9 Specify the hardware address of the NIC (Network Interface Card) in the *MAC Address* field.
- 10 Click *Create*.

A message indicates that the new host has been created.

NOTE: While creating a Host object, it is important to specify either the client identifier or the MAC address.

Viewing or Modifying a Host

- 1 In iManager, click *DHCP (OES Linux)* > *Host* to open the Host Management window in the main panel.
- 2 From the drop-down menu, select *View/Modify Host*, then click *OK* to open the View/Modify Host window.
- 3 Select a service from the list.
- 4 If you have created Host objects under a service-subnet network hierarchy, select the associated subnet from the drop-down list. The associated Host object is displayed in the drop-down list.
- 5 Click *OK* to modify the Host object.
- 6 Modify the attributes of the host.

| Attribute | Function |
|-------------------------|---|
| General Tab | |
| Host IP address | IP address of the host |
| Client Identifier: | Unique identifier of the client |
| MAC Type | The MAC address type |
| MAC Address | The hardware address of the NIC (Network Interface Card) |
| Comments | Specify comments about the Host objects you modify. |
| Lease | |
| Lease Type | <p>Specifies the length of time for an address assignment. A lease type can be either permanent or timed.</p> <p>Permanent leases never expire; the client is assigned an IP address for an indefinite period.</p> <p>Timed leases are defined in days, hours, or minutes. Timed leases expire, unless the client renews the lease.</p> |
| Boot Parameter Options: | <p>Select this option to specify the server address, server name, and boot file name for the BOOTP service.</p> <p>This information is provided at boot time. If the <i>Boot Parameter</i> option is selected, you need to specify at least one boot parameter.</p> |
| Settings | These settings are used to define configuration settings for a host. |
| Configured Options | <p>You can define values for the predefined options that are declared in the <i>Available Option</i> list for the service under which you have created this host.</p> <p>For example, <i>Time Offset</i> is a predefined option in the Available Option list. You use the Configured Options task to set a value for the <i>Time Offset</i> option.</p> |

Deleting a Host Object

- 1 In iManager, click *DHCP (OES Linux)* > *Host* to open the Host Management window in the main panel.

- 2 From the drop-down menu, select *Delete Host*, then click *OK* to open the Delete Class window.
- 3 Select a service from the list. All the subnets created directly under service are listed below.
 - ♦ The default value of Subnet is None. To list the Host objects created under a service-subnet hierarchy, select the associated Subnet object from the drop-down list.
 - ♦ To remove all the Host objects in the list, click the top-level check box, then click *Delete*.
or
To remove one or more than one Host object, click the check box next to the object, then click *Delete*.

7.1.8 Class Management

The Class object helps in segregating clients into classes. These clients are treated differently depending on the class they are in.

The Class Management role consists of the following tasks:

- ♦ “Creating a Class” on page 92
- ♦ “Viewing or Modifying a Class” on page 92
- ♦ “Deleting a Class Object” on page 93

Creating a Class

- 1 In iManager, click *DHCP (OES Linux) > Class* to open the Class Management window in the main panel.
- 2 From the drop-down menu, select *Create Class*, then click *OK* to open the Create Class window.
- 3 Enter the class name.
- 4 Select a service from the drop-down list.
- 5 Click *Create*.
A message indicates that the new pool has been created.

Viewing or Modifying a Class

- 1 Click *DHCP (OES Linux) > Class* to open the Class Management window in the main panel.
- 2 From the drop-down menu, select *View/Modify Class*, then click *OK* to open the View/Modify Class window.
- 3 Select a service from the list.
- 4 All the class objects created under the service is listed in the drop-down list. Select the class you want to modify and click *OK*.
- 5 Modify the attributes of the class.

| Attribute | Function |
|-------------|----------|
| General Tab | |

| Attribute | Function |
|------------------------|--|
| Conditional Expression | <p>Conditional Expression: A conditional expression is defined to segregate clients into different classes. This separation can be achieved with a conditional statement or with a match statement within the class declaration. Specify the conditional expression you want to define in the field.</p> <p>To group clients into different classes based on conditional expression, you can specify a match expression within a class statement in the following manner:</p> <pre>match if substring (option dhcp-client- identifier, 1, 3) = "RAS";</pre> |
| Spawn Statement | <p>A spawning class is used to automatically create lease-limited subclasses based on the client parameters. Select an option from the drop-down list.</p> <p>The option value sent by the client is checked against the dynamically created subclasses for the specified class. If a match is found, the client is classified under that subclass and treated accordingly. If no match is found, the server creates a new subclass and logs the information in the lease file, and the client is classified in this new subclass.</p> <p>After classification, the class is processed according to the rule set for the class</p> |
| Comments | Specify comments in the <i>Comments</i> field. This is optional. |
| Settings | These settings are used to define configuration settings for a class. |
| Configured Options | <p>You can define values for the predefined options that are declared in the <i>Available Option</i> list.</p> <p>For example, <i>Time Offset</i> is a predefined option in the <i>Available Option</i> list. You use the Configured Options task to set a value for the <i>Time Offset</i> option.</p> |

Deleting a Class Object

- 1 In iManager, click *DHCP (OES Linux) > Class* to open the Class Management window in the main panel.
- 2 From the drop-down menu, select *Delete Class*, then click *OK* to open the Delete Class window.
- 3 Select a service from the list. All the class objects created under the service are listed below.
 - ♦ To remove all the Class objects in the list, click the top-level check box, then click *Delete*.
or
 - ♦ To remove one or more than one class object, click the check box next to the object, then click *Delete*.

7.1.9 Zone Management

The DHCP Zone object contains the references the Domain Name System (DNS).

A DHCP server uses this information to perform dynamic updates for the zone objects. A DNS server must be configured to allow updates for the zone that the DHCP server is updating.

The Zone Management role consists of the following tasks:

- ♦ “Creating a Zone” on page 94
- ♦ “Modifying a Zone” on page 94
- ♦ “Deleting a Zone” on page 94

Creating a Zone

- 1 In iManager, click *DHCP (OES Linux) > Zone* to open the Zone Management window in the main panel.
- 2 From the drop-down menu, select *Create Zone*, then click *OK* to open the Create Zone window.
- 3 Specify the name by which you want to identify the zone.
- 4 Specify the IP address of the DNS server that will receive updates from an authorized DHCP server.
- 5 Select a service from the drop-down list.
- 6 Click *Create*.

A message indicates that the new zone has been created.

Modifying a Zone

- 1 In iManager, click *DHCP (OES Linux) > Zone* to open the Zone Management window in the main panel.
- 2 From the drop-down menu, select *Modify Zone*, then click *OK* to open the Modify Zone window.
- 3 Select a service from the list.
- 4 All of the zones associated with the selected service are displayed. Select the one you want to modify from the drop-down list, then click *OK*.
- 5 Modify the attributes as desired.

| Attribute | Function |
|------------|--|
| Addressing | If you want to modify the primary DNS server address of the zone that receives dynamic updates, specify the new IP address in the <i>DNS Server</i> IP address field |

- 6 Select a TSIG key from the drop-down list for secure DDNS update.
- 7 Specify comments in the *Comments* field.

Deleting a Zone

- 1 In iManager, click *DHCP (OES Linux) > Zone* to open the Zone Management window in the main panel.
- 2 From the drop-down menu, select *Delete Zone*, then click *OK* to open the Delete Zone window.

- 3 Select the service that contains the zones to delete.
- 4 To delete all the zones, click the top-level check box, then click *Delete*.
or
To delete one or more zone objects, click the check boxes next to them, then click *Delete*.

7.1.10 TSIG Key Management

A TSIG key is used for authenticating dynamic updates to a DNS server. TSIG uses shared secret keys as a cryptographically secure means of authenticating a DNS update.

The TSIG Key management role consists of the following tasks:

- ♦ “Creating a TSIG key” on page 95
- ♦ “Modifying a TSIG key” on page 95
- ♦ “Deleting a TSIG Key” on page 96

Creating a TSIG key

- 1 In iManager, click *DHCP (OES Linux) > TSIG Key* to open the TSIG Key Management window in the main panel.
- 2 From the drop-down menu, select *Create TSIG Key*, then click *OK* to open the Create TSIG Key window.
- 3 Specify a name to identify the TSIG key.
- 4 Specify the name of the algorithm. This algorithm is used to generate a TSIG key.
- 5 Specify a secret key that is used to decrypt the TSIG key.
- 6 Select a service from the list. The TSIG key is created under the specified service.
- 7 Click *Create*.

A message indicates that the new TSIG key has been created.

Modifying a TSIG key

- 1 In iManager, click *DHCP (OES Linux) > TSIG Key* to open the TSIG Key Management window in the main panel.
- 2 From the drop-down menu, select *View/Modify TSIG Key*, then click *OK* to open the View/Modify TSIG Key window.
- 3 Select the service from the drop-down list. All the TSIG keys created under this service are listed.
- 4 Select a TSIG key from the list.
- 5 Modify the key attributes as desired.

| Attribute | Function |
|----------------|---|
| General | |
| Algorithm Name | Specify an algorithm name. This algorithm is used to generate TSIG keys instead of the one specified earlier. |

| Attribute | Function |
|------------|--------------------------|
| Secret Key | Specify a new secret key |
| Comments | Specify comments |

Deleting a TSIG Key

- 1 In iManager, click *DHCP (OES Linux) > TSIG Key* to open the TSIG Key Management window in the main panel.
- 2 From the drop-down menu, select *Delete TSIG Key*, then click *OK* to open the Delete TSIG Key window.
- 3 Select a service from the list. All the TSIG keys created under the service are listed below.
- 4 To remove all the TSIG keys in the list, click the top-level check box, then click *Delete*.
or
To remove one or more than one TSIG keys, click the check box next to the object, then click *Delete*.

7.1.11 Failover Peer Management

The Failover Peer protocol allows only two DHCP servers to share a common address pool. This ensures continuous availability of the network. The process defines the role of a primary server and a secondary server.

Each server has about half of the available IP addresses in the Pool at any given time for allocation. During a prolonged failure of the primary server, the secondary server recovers all the addresses that the primary server had available for allocation, and begins to reuse them.

- ♦ “Creating a Failover Peer” on page 96
- ♦ “Viewing/Modifying a Failover Peer” on page 97
- ♦ “Deleting a Failover Peer” on page 97

Creating a Failover Peer

- 1 In iManager, click *DHCP (OES Linux) > Failover* to open the Failover Management window in the main panel.
- 2 From the drop-down menu, select *Create Failover*, then click *OK* to open the Create Failover window.
- 3 Specify the name of the Failover in the *Failover Name* text box.
- 4 Specify the IP address of the *Primary Server*.
- 5 Specify the port number of the primary server in the *Primary Port* text box.
- 6 Specify the IP address of the *Secondary Server*.
- 7 Specify the port number of the Secondary Server in the *Secondary Port* text box.
- 8 Specify the Failover Split. This specifies the split between the Primary and Secondary for the purpose of load balancing. Specify on the Primary, may not be specified on the Secondary.

- 9 Specify the Max Client Lead Time. This is the length of time for which a lease can be renewed by either server without contacting the other. Specify it on the Primary; it cannot be specified on the Secondary.
- 10 Select a Service from the drop-down list.
- 11 Click *Create*.
A message indicates that the new Failover has been created.

Viewing/Modifying a Failover Peer

- 1 In iManager, click *DHCP (OES Linux) > Failover* to open the Failover Management window in the main pane.
- 2 From the drop-down menu, select *View/Modify Failover*, then click *OK* to open the View/Modify Failover window.
- 3 Select a Service from the drop-down list.
- 4 Select the appropriate Failover from the drop-down list.
- 5 Click *OK* to modify the Failover.
- 6 Modify the attributes of the Failover.

The attributes, such as Primary Server, Primary Port, Secondary Server, Secondary Port, Failover Split, and Max Client Lead Time, which were defined during creating the Failover, can be modified. Additionally, the following attributes can also be modified:

| Attribute | Function |
|-------------------|---|
| Load Balance Time | The time (seconds) set so that if one of the failover peers is in a state where it responds to the failover messages but does not respond to some client requests, the other failover peer takes over its client load automatically as the clients retry. |
| Response Delay | This amount of information that needs to be processed. |
| Unacked Updates | Notifies the Primary Server of the number of messages it can send before it receives from the Failover Secondary Server. |
| Comments | Specify any comments about the Failover. |

Deleting a Failover Peer

- 1 In iManager, click *DHCP (OES Linux) > Failover* to open the Failover Management window in the main pane.
- 2 From the drop-down menu, select *Delete Failover*, then click *OK* to open the Delete Failover window.
- 3 Select a Service from the drop-down list. All the Failover instances created under the service are listed below.
- 4 To remove all the Failover instances in the list, click the top-level check box, then click *Delete*.
or
To remove one or more than one Failover instance, click the check box next to the instance, then click *Delete*.

7.1.12 Importing and Exporting the DHCP Configuration

The import or export operation is used to transfer the DHCP service configuration from files into eDirectory or from eDirectory to a text file in a `dhcpd.conf` format.

NOTE: Only Linux DHCP configuration files should be used to import or export the DHCP configuration.

Exporting the DHCP Configuration

The Export DHCP database operation transfers the DHCP configuration from eDirectory to a text file in a `dhcpd.conf` format. This file can be imported anywhere. You can also import the file back into eDirectory using the DHCP plug-in.

- 1 In iManager, click *DHCP (OES Linux) > Service* to open the Service Management window in the main pane.
- 2 From the drop-down menu, select *Export Service*, then click *OK* to open the Export Service window.
- 3 Select the service from the drop-down list that you want to export, then click *OK*.
- 4 Click *Click here to download* to start the export. The File Download dialog box opens to save the file. Click *OK* to save the file locally.
- 5 Click *Done* to continue. A confirmation message displays to confirm the export.
- 6 Click *OK* to complete the export.

Importing the DHCP Configuration

The Import DHCP database operation is used to transfer the DHCP service configuration from files into eDirectory database. The configuration files should be in `dhcpd.conf` format.

NOTE: Importing the Linux DHCP configuration file will overwrite the associated DNS server's settings.

- 1 Click *DHCP (OES Linux) > Service* to open the Service Management window in the main pane.
- 2 From the drop-down menu, select *Import Service*, then click *OK* to open the Import Service window.
- 3 Specify the name of the service in the *Service Name* text box.
- 4 Specify the name of the eDirectory context or browse to select it.
- 5 (Optional) Select the DHCP server associated with the service.
- 6 Specify the *DHCP Conf File Location* or browse to select it.
- 7 Click *Create*. An information message displays the status of the import.
- 8 Click *Done* to continue. A confirmation message displays to allow you to confirm the import.
- 9 Click *OK* to complete the import.

7.2 Using the Java Management Console for DHCP (OES Linux)


This section provides details on using the Java Management Console for DHCP (OES Linux) to manage objects in eDirectory.

- ♦ [Section 7.2.1, “Service Management,” on page 99](#)
- ♦ [Section 7.2.2, “Server Management,” on page 100](#)
- ♦ [Section 7.2.3, “Shared Network Management,” on page 100](#)
- ♦ [Section 7.2.4, “Subnet Management,” on page 100](#)
- ♦ [Section 7.2.5, “Pool Management,” on page 101](#)
- ♦ [Section 7.2.6, “Host Management,” on page 102](#)
- ♦ [Section 7.2.7, “Class Management,” on page 102](#)
- ♦ [Section 7.2.8, “Zone Management,” on page 103](#)
- ♦ [Section 7.2.9, “TSIG Key Management,” on page 103](#)
- ♦ [Section 7.2.10, “Failover Peer Management,” on page 104](#)
- ♦ [Section 7.2.11, “Managing DHCP \(OES Linux\) Objects in the Java Management Console,” on page 105](#)
- ♦ [Section 7.2.12, “Importing and Exporting the DHCP Configuration,” on page 107](#)

7.2.1 Service Management

The Service object acts as a container object for subnet, shared network, host, zone, class, and TSIG key.

To create a service

- 1 Click the *DHCP (OES Linux)* tab of Java Management Console window main panel.
- 2 Click *Create*  on the toolbar to open the Create New DHCP Record window.
- 3 Select *Service* in the Create New DHCP Record window, then click *OK*. The Create Service window opens.
- 4 Specify the *service name*.
- 5 Specify the eDirectory context or use the browse button to select the context.
- 6 Select the default DHCP server from the drop-down list.
- 7 Click *Create*.
The DHCP (OES Linux) Service object is created on the left pane of Java Management Console.


You can manage the DHCP (OES Linux) Service on the right pane of the Java Management Console by using the *General*, *Settings*, and *Configured Options* tabs.

For more information on managing objects, see [Section 7.2.11, “Managing DHCP \(OES Linux\) Objects in the Java Management Console,” on page 105](#).

7.2.2 Server Management

The Dynamic Host Configuration Protocol (DHCP) uses the client/server structure to allocate network addresses to a host. The DHCP server reads its configuration information from eDirectory and stores the information in its cache.

To create a server

- 1 Click the *DHCP (OES Linux)* tab of the Java Management Console.
- 2 Click *Create*  on the toolbar to open the Create New DHCP Record window.
- 3 Select *DHCP Server* in the Create New DHCP Record window, then click *OK*. The Create Server window opens.
- 4 Specify the server name.
- 5 Specify the eDirectory context or browse to select the context.
- 6 Click *Create*.

The server is now created and added to the *Default DHCP Server* list.


You can manage the DHCP (OES Linux) server on the right pane of the Java Management Console by using the *General* and *Settings* tabs.

For more information on managing objects, see [Section 7.2.11, “Managing DHCP \(OES Linux\) Objects in the Java Management Console,”](#) on page 105.

7.2.3 Shared Network Management

All subnets that share the same physical network can be grouped under a Shared Network object.

To create a shared network object:

- 1 Click the *DHCP (OES Linux)* tab of the Java Management Console.
- 2 Select the service for which the shared network needs to be created.
- 3 Click *Create*  on the toolbar to open the Create New DHCP Record window.
- 4 Select *shared network* from the Create New DHCP Record window, then click *OK*.
The Create Shared Network window opens.
- 5 Specify the name of the Shared Network object.
- 6 Click *Create*.

The Shared Network is now created.

You can manage the DHCP (OES Linux) Shared Network on the right pane of the Java Management Console by using the *General*, *Settings*, and *Configured Options* tabs.

For more information on managing objects, see [Section 7.2.11, “Managing DHCP \(OES Linux\) Objects in the Java Management Console,”](#) on page 105.


7.2.4 Subnet Management

The Subnet object is the most fundamental DHCP object. It enables you to distribute IP addresses and DHCP options to each network.

The Subnet object acts as a container object for Host and Pool objects.

A Subnet object's specific DHCP options and configuration parameters apply to the entire subnet and override global options.

To create a Subnet

- 1 Click the *DHCP (OES Linux)* tab of the Java Management Console.
- 2 Select the shared network or the service for which the subnet needs to be created.
- 3 Click *Create*  on the toolbar to open the Create New DHCP Record window.
- 4 Select *Subnet* from the Create New DHCP Record window, then click OK.
- 5 Specify a unique *Subnet Address* and *Subnet Mask* in the fields provided.
- 6 Click *Create*.

The Subnet is created.

You can manage the DHCP (OES Linux) Subnet on the right pane of the Java Management Console by using the *General*, *Lease*, *Settings*, and *Configured Options* tabs.


For more information on managing objects, see [Section 7.2.11, “Managing DHCP \(OES Linux\) Objects in the Java Management Console,”](#) on page 105.

7.2.5 Pool Management

A Pool is a designated range of IP addresses used to specify a range type assignment.

The Pool object represents a range of addresses for dynamic address assignment or for exclusion from the address assignment.

To create a Pool

- 1 Click the *DHCP (OES Linux)* tab of the Java Management Console.
- 2 Select a service or a shared network from the list where you want to create a pool. All of the subnets created under the service and the service or shared network hierarchy are displayed. Select the subnet under which you want to create the pool.
- 3 Click *Create*  on the toolbar to open the Create New DHCP Record window.
- 4 Select pool from the Create New DHCP Record window, then click *OK*. The Create Pool window opens.
- 5 Specify the name of the pool.
- 6 Specify the *Start Address*.
- 7 Specify the *End Address*.
- 8 Click *Create*.

The pool is created.

You can manage the DHCP (OES Linux) Pool on the right pane of the Java Management Console by using the *General*, *Settings*, and *Configured Options* tabs.


For more information on managing objects, see [Section 7.2.11, “Managing DHCP \(OES Linux\) Objects in the Java Management Console,”](#) on page 105.

7.2.6 Host Management

Hosts are a system of domain names in the network. They are used to identify DHCP clients. Host objects have an associated IP address.

For clients with statically assigned addresses or for installation where only known clients are served, each client must have a host.

Creating a Host

- 1 Click the *DHCP (OES Linux)* tab of the Java Management Console.
- 2 Select the service or the subnet where you want to create the host.
- 3 Click *Create*  on the toolbar to open the Create New DHCP Record window.
- 4 Select host from the Create New DHCP Record window, then click *OK*. The Create Host window opens.
- 5 Specify the name by which you want to identify the host in the *Host Name* field.
- 6 Specify the IP address of the host in the *Host IP Address* field.
- 7 Specify the client identifier. This uniquely identifies the client.
- 8 Select the MAC type from the drop-down list.
- 9 Specify the hardware address of the NIC (Network Interface Card) in the *MAC Address* field.
- 10 Click *Create*.
The host is now created below the required service.


You can manage the DHCP (OES Linux) host on the right pane of the Java Management Console by using the *General*, *Lease*, *Settings*, and *Configured Options* tabs.

For more information on managing objects, see [Section 7.2.11, “Managing DHCP \(OES Linux\) Objects in the Java Management Console,”](#) on page 105.

7.2.7 Class Management

The Class object helps in segregating clients into classes. These clients are treated differently depending on the class they are in.

To create a Class

- 1 Click the *DHCP (OES Linux)* tab of the Java Management Console.
- 2 Select the service for which the class needs to be created.
- 3 Click *Create*  on the toolbar to open the Create New DHCP Record window.
- 4 Select *Class* from the Create New DHCP Record window, then click *OK*. The Create Class window opens.
- 5 Specify the name of the class in the *Class Name* field.
- 6 Click *Create*.
The Class is created for the specified Service.

You can manage the DHCP (OES Linux) Class on the right pane of the Java Management Console by using the *General*, *Settings*, and *Configured Options* tabs.


For more information on managing objects, see [Section 7.2.11, “Managing DHCP \(OES Linux\) Objects in the Java Management Console,”](#) on page 105.

7.2.8 Zone Management

The DHCP Zone object defines the Domain Name System (DNS).

A DHCP server uses this information to perform dynamic updates for the zone objects. A DNS server must be configured to allow updates for the zone that the DHCP server is updating.

To create a Zone

- 1 Click the *DHCP (OES Linux)* tab of the Java Management Console.
- 2 Select the service for which the Zone needs to be created.
- 3 Click *Create*  on the toolbar.
The Create New DHCP Record window opens.
- 4 Select the zone from the Create New DHCP Record window, then click *OK*. The Create Zone window opens.
- 5 Specify the name of the zone in the *Zone Name* field.
- 6 In the *DNS Server IP Address* field, specify the IP address of the DNS server that will receive updates from an authorized DHCP server.
- 7 Click *Create*.

The zone is created for the specified service.


You can manage the DHCP (OES Linux) zone on the right pane of the Java Management Console by using the *General* tab.

For more information on managing objects, see [Section 7.2.11, “Managing DHCP \(OES Linux\) Objects in the Java Management Console,”](#) on page 105.

7.2.9 TSIG Key Management

A TSIG key provides a means of authenticating updates to a Dynamic DNS database by using shared secret keys as a cryptographically secure means of authenticating a DNS update.

Creating a TSIG key

- 1 Click the *DHCP (OES Linux)* tab of the Java Management Console.
- 2 Select the service for which the TSIG key needs to be created.
- 3 Click *Create*  on the toolbar. The Create New DHCP Record window opens.
- 4 Select *TSIG* from the Create New DHCP Record window, then click *OK*. The Create TSIG Key window opens.
- 5 Specify the name of the TSIG key in the *TSIG Key Name* field.
- 6 Specify the name of the algorithm in the *Algorithm* field.
- 7 Specify the *Secret key*.
- 8 Click *Create*.

The TSIG key is now created below the specified service.

You can manage the DHCP (OES Linux) TSIG key on the right pane of the Java Management Console by using the *General* tab.

For more information on managing objects, see [Section 7.2.11, “Managing DHCP \(OES Linux\) Objects in the Java Management Console,”](#) on page 105.


7.2.10 Failover Peer Management

The Failover Peer protocol allows only two DHCP servers to share a common address pool. This ensures continuous availability. The process defines a role of a Primary Server and a Secondary Server.

Each server has about half of the available IP addresses in the Pool at any given time for allocation. During a prolonged failure of the primary server, the secondary server recovers all the addresses that the primary server had available for allocation, and begins to reuse them.

- ♦ [“Creating a Failover Peer”](#) on page 104
- ♦ [“Managing a Failover Peer”](#) on page 104

Creating a Failover Peer

- 1 Click the *DHCP (OES LINUX)* tab of the Java Management Console.
- 2 On the left pane of the Java Management Console, select the Service for which you want to create the Failover.
- 3 Click  *Create* on the toolbar. The Create New DHCP Record window opens.
- 4 Select *Failover Peer*, then click *OK*.
- 5 Specify a name to identify the Failover.
- 6 Specify the *Primary Server IP Address*.
- 7 Specify the port number of the primary server in the *Primary Port* field.
- 8 Specify the *Secondary Server IP Address*.
- 9 Specify the port number of the secondary server in the *Secondary Port* field.
- 10 Specify the *Max Client Lead Time*. This is the length of time for which a lease may be renewed by either server without contacting the other. (Specify this on the Primary Server; it cannot be specified on the Secondary Server).
- 11 Specify the Failover Split. This specifies the split between the Primary and Secondary Servers for the purpose of load balancing.
- 12 Click *Create*. The Failover Peer is now created.

Managing a Failover Peer

You can manage the Failover Peer by using the *General* tab in the right pane of the Java Management Console:

- ♦ **Primary Server IP Address:** Modify the primary server IP Address appropriately if required.
- ♦ **Primary Port:** Modify the port number of the primary Server if required.
- ♦ **Secondary Server IP Address:** Modify the secondary server IP Address appropriately if required.

- ♦ **Secondary Port:** Modify the port number of the secondary server if required.
- ♦ **Failover Split:** Specify the split between the primary and secondary for the purpose of load balancing.
- ♦ **Max Client Lead Time:** The length of time for which a lease can be renewed by either server without contacting the other. (Specify this on the primary; and it cannot be specified on the secondary).
- ♦ **Unacked Updates:** Notifies the primary server of the number of messages it can send before it receives from the Failover secondary server.
- ♦ **Response Delay:** Refers to the amount of information that needs to be processed.
- ♦ **Load Balance Time:** Refers to the time (seconds) set so that if one of the failover peers is in a state where it responds to the failover messages but does not respond to some client requests, the other failover peer takes over its client load automatically as the clients retry.
- ♦ **Comments:** Specify comments if any.

7.2.11 Managing DHCP (OES Linux) Objects in the Java Management Console

You can manage the DHCP (OES Linux) objects by using the tabs on the right pane of the Java Management Console. The following description provide more information about the tabs:

- ♦ “General Tab” on page 105
- ♦ “Settings Tab” on page 105
- ♦ “Configured Options Tab” on page 106
- ♦ “Lease Type Tab” on page 106

General Tab

Use the *General* tab to select the preferred Server object and add comments for the DHCP (OES Linux) object. This tab must be used to manage a Service.

- 1 Select a server from the default *DHCP* drop-down list. The selected server is now associated with the service.
- 2 Type your comments about the service in the *Comments* field.
- 3 Click *OK* to save the settings.

Settings Tab

These settings are used to define the configuration for an object.

To add or modify a setting:

- 1 Click *Modify*.
The Modify DHCP Settings dialog box opens.
- 2 Select a setting from the *Settings Name* list.
- 3 Click *Add*.
To add the entire list, Click *Add All*.
- 4 Click *OK* to add the settings.

To remove setting name:

- 1 Select the setting from the *Settings Name* list, then click *Remove*.

or

To remove all the settings, click *Remove All*.

To delete a setting:

- 1 Select the setting from the list, then click *Delete*.

or

To remove multiple settings, use Ctrl+Shift to select the settings you want to remove, then click *Delete*.

Configured Options Tab

You can define values for the predefined options that are declared in the *Available Option* list.

For example, *Time Offset* is a predefined option in the *Available DHCP Option* list. Use the *Configured Options* task to set a value for the *Time Offset* option.

To add a configured option:

- 1 Click *Modify*.
The Modify DHCP Options dialog box is displayed.
- 2 Select an option from the *Option Name* list.
- 3 Click *Add*.
To add the entire list, Click *Add All*.
- 4 Click *OK*.

To remove a setting:

- 1 Select the option from the *Option Name* list, then click *Remove*.
or
To remove all the options, click *Remove All*.

To delete a setting:

- 1 To delete an option, select the option from the list, then click *Delete*.
or
To remove multiple options, use Ctrl+Shift to select the options you want to remove, then click *Delete*.

Lease Type Tab

The Lease Type specifies the length of time for an address assignment. A lease type can be either permanent or timed. This tab must be used to manage the subnet and host objects.

Permanent leases never expire; the client is assigned an IP address for an indefinite period.

Timed leases are defined in days, hours, or minutes. Timed leases expire, unless the client renews the lease.

Set Boot Parameter Option

Select this option to specify the *Server Address*, *Server Name*, and *Boot File Name* for the BOOTP service. This information is provided at boot time. If the *Boot Parameter* option is selected, you need to specify the entire boot sequence.

7.2.12 Importing and Exporting the DHCP Configuration

The import or export operation is used to transfer the DHCP service configuration from files into eDirectory or from eDirectory to a text file in a `dhcpd.conf` format respectively. Only Linux DHCP configuration files should be used to import or export the DHCP configuration.


NOTE: Before importing a DHCP configuration file, check the syntax of the file with the `rcnovell-dhcpd check-syntax` command. The command reads `/etc/dhcpd.conf` and checks the syntax.

- ♦ “Importing the DHCP Configuration” on page 107
- ♦ “Exporting the DHCP Configuration” on page 107

Importing the DHCP Configuration

The configuration file to import should be in DHCP V3 format. Importing the Linux DHCP configuration file overwrites the associated DNS server's settings.


To import the DHCP files:

- 1 Click the *DHCP (OES Linux)* tab of the Java Management Console.
- 2 Click  *Import DHCP Database* on the toolbar.
- 3 Click *Browse* to select or specify the path for the DHCP database file.
- 4 Click *Next* to open the Import - File Input window.
- 5 Specify the service name in the *Service Name* text box.
- 6 In the Select NDS Context text box, browse to select or enter specify the context where the service is to be created.
- 7 (Optional) Select a *Default DHCP Server* from the drop-down list.
- 8 Click *Import*.
- 9 Click *Finish* to complete the import operation.

If the import program encounters any error, the *Details* button is enabled in the error window. Click *Details* to view the error details.

Exporting the DHCP Configuration

The file is exported in a `dhcpd.conf` format. These files can be imported anywhere and can also be imported back to eDirectory by using the DNS/DHCP Java-based Management Console Utility.

- 1 Click the *DHCP (OES Linux)* tab of the Java Management Console.
- 2 Click  *Export DHCP Database* on the toolbar to open the *Export - DHCP* window.
- 3 Specify the name of a destination file or browse to select a filename from the dialog box, then click *Next*.

- 4 Select the services by using the Export DHCP - Service List window.
- 5 Click *Export* to store your information in a file.
- 6 Click *Finish* to complete the export.

If the export program encounters any error, the *Details* button is enabled in the error window. Click *Details* to view the error details.

7.3 Starting the DHCP Server

- 1 Start the server with the `rcnovell-dhcpd start` command.
- 2 Check the status of the server by using the `rcnovell-dhcpd status` command.

7.4 Stopping the DHCP Server

To stop the DHCP server, use the `rcnovell-dhcpd stop` command.

7.5 What's Next

The next section provides information on migrating DHCP services from a NetWare® system to a Linux system.

- ♦ Chapter 8, “Configuring DHCP with Novell Cluster Services for the NSS File System,” on page 109.

Configuring DHCP with Novell Cluster Services for the NSS File System

8

- [Section 8.1, “Benefits of Configuring DHCP for High Availability,” on page 109](#)
- [Section 8.2, “Prerequisites,” on page 109](#)
- [Section 8.3, “Installation and Configuration,” on page 109](#)
- [Section 8.4, “Loading and Unloading the DHCP Server,” on page 113](#)
- [Section 8.5, “What’s Next,” on page 114](#)

8.1 Benefits of Configuring DHCP for High Availability

A Novell® Open Enterprise Server 2 Linux DHCP server automatically assigns IP addresses and other configuration information to clients upon request or when the clients are restarted. If for some reason the Novell OES 2 Linux DHCP server is not accessible, clients lose their ability to connect to the network because they cannot obtain an IP address.

You can help prevent this problem by configuring DHCP with Novell Cluster Services™, which ensures that the IP address range required by users to connect to the network is highly available. This is possible because the DHCP server is automatically started, stopped, and restarted on different servers in the cluster by Novell Cluster Services.

Before you attempt to implement this solution, familiarize yourself with how Cluster Services works. For information, see the *OES 2 SP1: Novell Cluster Services 1.8.5 for Linux Administration Guide*.

8.2 Prerequisites

The following OES 2 Linux services are required to cluster DHCP services:

- *Novell eDirectory 8.8 Troubleshooting Guide*
- *OES 2 SP1: Novell Cluster Services 1.8.5 for Linux Administration Guide*
- *OES 2 SP1: NSS File System Administration Guide*
- *OES 2 SP1: NCP Server for Linux Administration Guide*

8.3 Installation and Configuration

- [Section 8.3.1, “Prerequisites,” on page 110](#)
- [Section 8.3.2, “Verifying the Novell Cluster Services Setup,” on page 110](#)
- [Section 8.3.3, “Installing and Configuring a Cluster,” on page 110](#)
- [Section 8.3.4, “DHCP Load and Unload Scripts,” on page 112](#)

8.3.1 Prerequisites

- ♦ The DHCP server should be installed on all the nodes in cluster or on the nodes identified for running DHCP.
- ♦ Create a shared NSS pool and volume for the DHCP server and cluster-enable the shared pool. You will configure the cluster resource later for DHCP services. For information, see “[Configuring Cluster Resources for Shared NSS Pools and Volumes](#)” in the *OES 2 SP1: Novell Cluster Services 1.8.5 for Linux Administration Guide*. You need a unique, static IP address in the same IP subnet as the cluster to assign as the IP address for this DHCP cluster resource.

8.3.2 Verifying the Novell Cluster Services Setup

To ensure that the Novell Cluster Services is set up properly:

- 1 In iManager, select the *Cluster > Cluster Options* task.
Click the *Search* icon to open the eDirectory Object Selector. Browse to locate and select the Cluster object for the DHCP cluster. The *Cluster Objects* list is displayed.
- 2 Select the check box next to the Cluster Resource object that you created for the shared NSS pool, then click the *Details* link.
This opens the Cluster Resource Properties page to the *Policies* tab.
- 3 Click the *Preferred Nodes* tab to view a list of the nodes that are assigned as the preferred nodes for failover and migration.

After executing these steps, you can mount the shared volume on the preferred nodes by using the Novell Client™. The shared volume is mounted on the preferred node so that the directories and lease files are created. This process also assigns rights to the shared volume.

8.3.3 Installing and Configuring a Cluster

- 1 Ensure that association between the DHCP Server object and the DHCP Service object is set by using iManager. For details, see “[Viewing or Modifying a Service](#)” on page 82.
- 2 Use iManager to create a DHCP Subnet and a DHCP Pool object. For details, see “[Creating a Subnet Object](#)” on page 86 and “[Creating a Pool Object](#)” on page 88.
- 3 The DHCP server by default uses the `dhcpd` user that is created in the local system during installation process. If you want to use another user, create the user by using the *Security and Users > User Management* option in YaST.
After creating the user, update `/etc/sysconfig/dhcpd` file, then set the value of the variable `DHCPD_RUN_AS` to the new user.
- 4 Click the *Users > Create User* task in iManager to open the Create User window. Specify the details and click *OK* to create a new user in eDirectory.
- 5 The user created in [Step 4](#) needs to be LUM-enabled. To do this, click the *Linux User Management > Enable Users for Linux* task. This opens the Enable Users for Linux window. Search for and select the user created in [Step 4](#), then click *OK* to select the user.
 - 5a Make sure that every user belongs to a primary group. To add a user to a group, search for an *Existing eDirectory Group object*.
 - 5b Select the `DHCPGroup` object from the list.

5c Select the workstations to which the Linux-enabled user should have access.

5d Click *Next* to confirm the selection.

The user is now Linux-enabled, included in the DHCP Group, and granted access to cluster nodes.

6 Mount the shared volume on one of the nodes in the cluster.

7 Execute the following command at the command prompt:

```
/opt/novell/dhcp/bin/ncs_dir.sh <MountPath> <FQDN of Username  
with tree-name>
```

The MountPath parameter indicates the target directory in the volume where DHCP-specific directories are created.

For example, `/opt/novell/dhcp/bin/ncs_dir.sh /media/nss/DHCPVOL/
cn=dhcpd.o=novell.T=MyTree;`

When the script is executed, it creates the following folders:

- ♦ /media/nss/DHCPVOL/etc
- ♦ /media/nss/DHCPVOL/var/lib/dhcp/db

The script also takes care of assigning permissions for these directories.

8 Copy the `/etc/dhcpd.conf` file to `/media/nss/DHCPVOL/etc` directory and modify the LDAP attributes as required.

For example, `ldap-server "192.168.0.1"; ldap-dhcp-server-cn
"DHCP_acme";`

9 Open a terminal on the node where the shared volume is mounted and execute the following command at the prompt:

```
dhcpd -cf /media/nss/DHCPVOL/etc/dhcpd.conf -lf /media/nss/  
DHCPVOL/var/lib/dhcp/db/dhcpd.leases
```

This step ensures that the DHCP server can work on a cluster setup with shared volumes.

Stop the server by executing the following command at the prompt:

```
killproc -p /var/run/dhcpd.pid -TERM /usr/sbin/dhcpd
```

10 Click *Cluster > Cluster Options* task in iManager. The Cluster objects are displayed.

Select the DHCP Cluster resource that was created as part of “Prerequisites” on page 109 and click *Details*. The Cluster Pool Properties are displayed. Click the *Scripts* tab. You can now view or edit the load or unload scripts.

10a Click *Load Script*.

10b Ensure that the DHCP load script is same as specified in “DHCP Load Script” on page 112.

10c Click *Unload Script*.

10d Ensure that the DHCP unload script is same as specified in “DHCP Unload Script” on page 113.

10e Click *OK* to save the changes.

11 The NSS volume should be upgraded to support hard links. To do this, open a terminal and enter `nsscon` at the prompt.

12 Execute the following commands at the prompt:

```
/ZLSSUpgradeCurrentVolumeMediaFormat=VolName  
/hardlinks=VolName
```

- 13** To ensure that hard links are enabled, execute the following commands in the shared volume:

```
touch testfile.txt  
ln testfile.txt testlink.txt  
unlink testlink.txt  
rm testfile.txt
```

If the hard link was successfully enabled, these commands execute without errors.

- 14** Set the DHCP resource online by using the *Clusters > Cluster Manager* task in iManager.

8.3.4 DHCP Load and Unload Scripts

- ♦ “DHCP Load Script” on page 112
- ♦ “DHCP Unload Script” on page 113

DHCP Load Script

The load script contains commands to start the DHCP service. The load script appears similar to the following example:

```
#!/bin/bash  
. /opt/novell/ncs/lib/ncsfuns  
exit_on_error nss /poolact=DHCPPPOOL  
exit_on_error ncpcon mount DHCPVOL=254  
exit_on_error add_secondary_ipaddress 10.10.2.1  
exit_on_error ncpcon bind --ncpservname=DHCPCLUSTER_DHCPPPOOL_SERVER --  
ipaddress=10.10.2.1  
exit 0
```

Configuring the DHCP Load Script

- 1** Add the following lines to the script before `exit 0` to load DHCP:

```
# get the arguments for dhcp  
DHCPD_ARGS=""  
if [ -s /etc/sysconfig/dhcpd ]; then  
    . /etc/sysconfig/dhcpd  
  
    if [ -n "$DHCPD_RUN_AS" ]; then  
        DHCPD_RUN_AS_GROUP="$(getent group $(getent passwd $DHCPD_RUN_AS | cut -d:  
-f4) | cut -d: -f1)"  
        DHCPD_ARGS="$DHCPD_ARGS -user $DHCPD_RUN_AS -group $DHCPD_RUN_AS_GROUP"  
    fi  
  
    if [ -n "$DHCPD_INTERFACE" ]; then  
        unset interfaces  
        if [ -x /sbin/getcfg-interface ]; then  
            for i in $DHCPD_INTERFACE; do  
                interfaces="$interfaces$(/sbin/getcfg interface $i) "  
            done  
        fi  
        DHCPD_ARGS="$DHCPD_ARGS $interfaces"  
    fi  
fi
```



```

MOUNT_POINT=""
# start dhcpd
exit_on_error /usr/sbin/dhcpd -cf $MOUNT_POINT/etc/dhcpd.conf -lf
$MOUNT_POINT/var/lib/dhcp/db/dhcpd.leases $DHCPD_ARGS

# return status

```

- 2 Edit the following line to assign a secondary server IP address for DHCP server:

```

exit_on_error add_secondary_ipaddress IP address of the secondary server

```

Replace *IP address of the secondary server* with the address of the secondary server.

- 3 Set MOUNT_POINT to the location where the shared volume is mounted. For example:

```

MOUNT_POINT= "/media/nss/DHCPVOL"

```

- 4 Click *Next* and continue with the unload script configuration.

DHCP Unload Script

The unload script contains commands to stop the DHCP service. The unload script appears similar to the following example:

```

#!/bin/bash
. /opt/novell/ncs/lib/ncsfuns
ignore_error ncpcon unbind --ncpservname=DHCPCLUSTER_DHCPPPOOL_SERVER --
ipaddress=10.10.2.1
ignore_error del_secondary_ipaddress 10.10.2.1
ignore_error nss /pooldeact=DHCPPPOOL
exit 0

```

Configuring the DHCP Unload Script

- 1 Add the following line after the `. /opt/novell/ncs/lib/ncsfuns` statement:

```

ignore_error killproc -p /var/run/dhcpd.pid -TERM /usr/sbin/dhcpd

```

- 2 Edit the following line to assign a secondary server IP address for DHCP server:

```

ncpservname=DHCPCLUSTER_DHCPPPOOL_SERVER --ipaddress <IP address of the
secondary server>
ignore_error del_secondary_ipaddress <IP address of the secondary server>

```

Replace the IP address of the DHCP server with the address of the secondary server.

8.4 Loading and Unloading the DHCP Server

After updating the load and unload scripts of the virtual NCP server, DHCP server is now loaded and unloaded along with the virtual NCP Server.

8.4.1 Loading the DHCP Server

- 1 Click the *Cluster > Cluster Manager* task in iManager
- 2 Click the *Search* icon to open the eDirectory Object Selector. Browse or search the list to locate and select the cluster object. The cluster objects are displayed.

- 3 Select the check box next to the DHCP Cluster and select *Online*.
- 4 Select the cluster node where you want the resource to load, then click *OK* to load the resource on that node. After a few seconds you see the server status as *Online*, indicating that the server has been loaded.

8.4.2 Unloading the DHCP Server

- 1 Click the *Cluster > Cluster Manager* task in iManager
- 2 Click the *Search* icon to open the eDirectory Object Selector. Browse or search the list to locate and select the cluster object. The Cluster objects are displayed.
- 3 Select the check box next to the DHCP cluster and select *Offline*. After a few seconds you see the server status as *Offline*, indicating that the server has been unloaded.

8.5 What's Next

The next section provides information on configuring DHCP with Novell Cluster Services for the Linux file system.

- ♦ [Chapter 9, “Configuring DHCP with Novell Cluster Services for the Linux File System,” on page 115](#)

Configuring DHCP with Novell Cluster Services for the Linux File System

9

- Section 9.1, “Benefits of Configuring DHCP for High Availability,” on page 115
- Section 9.2, “DHCP Installation and Configuration,” on page 115
- Section 9.3, “What’s Next,” on page 121

9.1 Benefits of Configuring DHCP for High Availability

A Novell® Open Enterprise Server 2 Linux DHCP server automatically assigns IP addresses and other configuration information to clients upon request or when the clients are restarted. If for some reason the Novell OES 2 Linux DHCP server is not accessible, clients lose their ability to connect to the network because they cannot obtain an IP address. Configuring DHCP with Novell Cluster Services™ helps ensure the IP address range required by users to connect to the network is highly available. This is possible because the DHCP server is automatically started, stopped, and restarted on different servers in the cluster by Novell Cluster Services.

Before you attempt to implement this solution, familiarize yourself with how Cluster Services works. For information, see the [OES 2 SP1: Novell Cluster Services 1.8.4 Administration Guide for Linux](http://www.novell.com/documentation/oes2/clus_admin_lx/data/h4hgu4hs.html) (http://www.novell.com/documentation/oes2/clus_admin_lx/data/h4hgu4hs.html).

9.2 DHCP Installation and Configuration

- Section 9.2.1, “Prerequisites,” on page 115
- Section 9.2.2, “Configuring DHCP on the Shared Disk,” on page 116
- Section 9.2.3, “Configuring the dhcpd.conf File,” on page 116
- Section 9.2.4, “Creating a dhcpd.leases File,” on page 116
- Section 9.2.5, “Novell Cluster Services Configuration and Setup,” on page 117

9.2.1 Prerequisites

- Novell Cluster Services 1.8.4
- DHCP must be installed on every server that will run it.

Novell Cluster Services 1.8 provides a DHCP resource template, which facilitates configuring DHCP with a shared Linux POSIX* volume in a cluster environment. Use the instructions in “Creating Linux POSIX Volumes on Shared Disks” in the *OES 2 SP1: Novell Cluster Services 1.8.5 for Linux Administration Guide*. Afterwards, cluster-enable the shared volume by using the DHCP template.

For details on installing the DHCP server, see [Chapter 6, “Installing and Configuring DHCP,” on page 73](#).

For details on Cluster Services, see *OES 2 SP1: Novell Cluster Services 1.8.5 for Linux Administration Guide*.

9.2.2 Configuring DHCP on the Shared Disk

To configure DHCP to use shared storage, you need to create a shared directory (file system or disk) on the shared disk system and create mount points to that shared file system on each cluster server that will run DHCP.

- 1 Use EVMS to create a file system on the shared disk system.

Enter `evmsgui` at the Linux server to start the EVMSGUI utility.

For instructions, see “[Configuring Cluster Resources for Shared Linux POSIX Volumes](#)” in the *OES 2 SP1: Novell Cluster Services 1.8.5 for Linux Administration Guide*.

- 2 On each cluster node that runs DHCP, create the directory path that is used as the mount point. At a terminal console prompt, log in as the `root` user, then enter `mkdir /mnt/dhcp`.

- 3 Log in as the `root` user and mount the shared disk (file system) that was created in [Step 1](#).

For example, depending on the mount point and directory names, you could enter a command similar to the following to mount the shared disk:

```
mount /dev/evms/dhcp /mnt/dhcp
```

- 4 At the root of the shared disk you just created (`/mnt/dhcp`), enter the following commands to create the directories specified:

```
mkdir etc
```

```
mkdir -p var/lib/dhcp/db
```

The `db` directory must be owned by the user that is used in `/etc/sysconfig/dhcpd` in the parameter `DHCPD_RUN_AS="dhcpd"`. Also, all the four directories must have permissions of `drwxr-xr-x`.

9.2.3 Configuring the dhcpd.conf File

- 1 Copy the `dhcpd.conf` file from the `/etc` directory on one of the OES 2 Linux cluster servers to the `etc` directory you created on the shared disk in [Step 4 on page 116](#).

This would be the `/mnt/dhcp/etc` directory if you used the same directory names as those given in the example above.

- 2 Modify the LDAP attributes as required. For example, `ldap-server "192.168.0.1"; ldap-dhcp-server-cn "DHCP_acme";`

9.2.4 Creating a dhcpd.leases File

A `dhcpd.leases` file is necessary for DHCP to function. The DHCP daemon requires this file before it starts. The file can be empty, and it must reside in the `var/lib/dhcp/db` directory you created in [Step 4 on page 116](#).

One way to create the empty file is to use the `touch` command. For example, if you used the directory names listed in the example above, you could enter the following to create an empty `dhcpd.leases` file:

```
touch /mnt/dhcp/var/lib/dhcp/db/dhcpd.leases
```

9.2.5 Novell Cluster Services Configuration and Setup

After DHCP is properly installed and configured, you must create and configure a DHCP resource in Novell Cluster Services. This includes configuring DHCP load and unload scripts, setting DHCP start, failover, and failback modes, and assigning the DHCP resource to specific servers in your cluster.

- ♦ “Creating a DHCP Cluster Resource” on page 117
- ♦ “Configuring DHCP Load and Unload Scripts” on page 118
- ♦ “Setting DHCP Start, Failover, and Failback Modes” on page 121
- ♦ “View or Edit DHCP Resource Server Assignments” on page 121

Creating a DHCP Cluster Resource

Novell Cluster Services includes a DHCP resource template, which greatly simplifies the process for creating a DHCP cluster resource. Much of the DHCP cluster resource configuration is performed automatically by the DHCP resource template.

To create a DHCP cluster resource:

- 1 Ensure that the shared disk (file system) you created and mounted in [Step 2 on page 116](#) is unmounted.
If you used the directory names specified in the example, you can enter `umount /mnt/dhcp` to unmount the shared disk.
- 2 Open your Internet browser and enter the URL for iManager.
The URL is `http://server_ip_address/nps/imanager.html`. Replace *server_ip_address* with the IP address or DNS name of a server in the cluster or with the IP address for Apache-based services.
- 3 Enter your username and password.
- 4 In the left column, locate *Clusters* and then click the *Cluster Options* link.
iManager displays three links under *Clusters* that you can use to configure and manage your cluster.
- 5 Specify the cluster name, or browse and select it, then click the *New* link.
- 6 Specify Resource as the resource type you want to create by clicking the *Resource* radio button, then click *Next*.
- 7 Specify a name for the DHCP resource.
Do not use periods in cluster resource names. Novell clients interpret periods as delimiters. If you use a space in a cluster resource name, that space is converted to an underscore.

8 Type the DHCP template name in the *Inherit From Template* field, or browse and select it from the list.

9 Select the *Define Additional Properties* check box, click *Next*, then continue with “**Configuring DHCP Load and Unload Scripts**” on page 118.

The DHCP resource template configures the DHCP resource by automatically creating DHCP load and unload scripts, setting failover and failback modes, and assigning DHCP as a resource to all nodes in the cluster.

Configuring DHCP Load and Unload Scripts

Table 9-1 Sample Values for DHCP Load and Unload Scripts

| Variable | Template Value | Description |
|----------------|-------------------------------------|--|
| Resource_IP | a.b.c.d | IP address of the virtual cluster server for this cluster resource. |
| MOUNT_FS | reiserfs | The file system type you made on the EVMS volume. |
| container_name | name | The name you gave to the cluster segment manager. |
| MOUNT_POINT | /mnt/dhcp | The mount location for the EVMS volume you created. This example shows a mount location with a directory named the same as the EVMS volume name. You can mount the EVMS volume anywhere. |
| MOUNT_DEV | /dev/evms/ \$container_name/dhcp | The Linux path for the EVMS volume you created. |

- ♦ “Load Script Configuration” on page 118
- ♦ “Unload Script Configuration” on page 120

Load Script Configuration

The DHCP load script page should already be displayed. The load script contains commands to start the DHCP service. You must customize some commands for your specific DHCP configuration.

NOTE: The scripts in this section are based on the template values in [Table 9-1](#). Make sure to substitute the sample values with the ones you used in your solution.

The load script appears similar to the following example:

```
#!/bin/bash
. /opt/novell/ncs/lib/ncsfuns

# Template for clustering the ISC DHCP daemon.
# You should not configure failover within the DHCP server itself.
# Instead, let NCS manage the failover.

# define the IP address
RESOURCE_IP=a.b.c.d
# define the file system type
MOUNT_FS=reiserfs
#define the container name
```

```

container_name=name
# define the device
MOUNT_DEV=/dev/evms/$container_name/dhcp
# define the mount point
MOUNT_POINT=/mnt/dhcp

#activate the container
exit_on_error activate_evms_container $container_name $MOUNT_DEV
$NCS_TIMEOUT

# mount the file system
exit_on_error mount_fs $MOUNT_DEV $MOUNT_POINT $MOUNT_FS

# add the IP address
exit_on_error add_secondary_ipaddress $RESOURCE_IP

# get the arguments for dhcp
DHCPD_ARGS=""
if [ -s /etc/sysconfig/dhcpd ]; then
    . /etc/sysconfig/dhcpd

    if [ -n "$DHCPD_RUN_AS" ]; then
        DHCPD_RUN_AS_GROUP="$(getent group $(getent passwd $DHCPD_RUN_AS | cut -d: -f4)
| cut -d: -f1)"
        DHCPD_ARGS="$DHCPD_ARGS -user $DHCPD_RUN_AS -group $DHCPD_RUN_AS_GROUP"
    fi

    if [ -n "$DHCPD_INTERFACE" ]; then
        unset interfaces
        if [ -x /sbin/getcfg-interface ]; then
            for i in $DHCPD_INTERFACE; do
                interfaces="$interfaces$(/sbin/getcfg interface $i) "
            done
        fi
        DHCPD_ARGS="$DHCPD_ARGS $interfaces"
    fi
fi

# start dhcpd
exit_on_error /usr/sbin/dhcpd -cf $MOUNT_POINT/etc/dhcpd.conf -lf
$MOUNT_POINT/var/lib/dhcp/db/dhcpd.leases $DHCPD_ARGS

# return status
exit 0

```

To customize the DHCP load script for your specific configuration:

- 1 View and, if necessary, edit the following lines for your specific container name, device, and mount point:

```

#define the container name
container_name=name
# define the device
MOUNT_DEV=/dev/evms/$container_name/dhcp
# define the mount point
MOUNT_POINT=/mnt/dhcp

```

- 2 Edit the following line to assign a unique IP address to the DHCP cluster resource:

```
RESOURCE_IP=a.b.c.d
```

Replace *<a.b.c.d>* with the IP address you want to assign to the DHCP cluster resource.

The IP address for the DHCP cluster resource allows clients to reconnect to that address regardless of which server is hosting it.

- 3 Click *Next* and continue with the **Unload Script Configuration**.

Unload Script Configuration

The DHCP unload script page should now be displayed. The unload script contains commands to stop the DHCP service. You must customize some commands for your specific DHCP configuration.

NOTE: The scripts in this section are based on the template values in [Table 9-1](#). Make sure to substitute the sample values with the ones you used in your solution.

The unload script appears similar to the following example:

```
#!/bin/bash
. /opt/novell/ncs/lib/ncsfncs

# define the IP address
RESOURCE_IP=a.b.c.d
# define the file system type
MOUNT_FS=reiserfs
# define the container name
container_name=name
# define the device
MOUNT_DEV=/dev/evms/$container_name/dhcp
# define the mount point
MOUNT_POINT=/mnt/dhcp

# request dhcpd stop
ignore_error killproc -p /var/run/dhcpd.pid -TERM /usr/sbin/dhcpd

# del the IP address
ignore_error del_secondary_ipaddress $RESOURCE_IP

# umount the file system
sleep 10 # if not using SMS for backup, please comment out this line
exit_on_error umount_fs $MOUNT_DEV $MOUNT_POINT $MOUNT_FS

#deactivate the container
exit_on_error deactivate_evms_container $container_name $NCS_TIMEOUT

# return status
exit 0
```

- 1 View and, if necessary edit the container name, device, and mount point in the applicable lines of the unload script.
- 2 Replace *<a.b.c.d>* with the same IP address you specified in the load script, then click *Next*.
The page to set Start, Failover, and Failback modes is now displayed
- 3 Continue with **“Setting DHCP Start, Failover, and Failback Modes” on page 121**

Setting DHCP Start, Failover, and Failback Modes

- 1 The DHCP resource template sets the DHCP resource Start mode and Failover mode to Auto and the Failback Mode to Disable.
 - ♦ If the DHCP resource Start mode is set to Auto, the DHCP server automatically loads on a designated server when the cluster is first brought up. If the DHCP Start mode is set to Manual, you can manually start DHCP on a specific server when you want, instead of having it automatically start when servers in the cluster are brought up.
 - ♦ If the DHCP Failover mode is set to Auto, the DHCP server automatically moves to the next server in the Assigned Nodes list in the event of a hardware or software failure. If the DHCP Failover mode is set to Manual, you can intervene after a failure occurs and before the DHCP server is started on another node.
 - ♦ If the DHCP Failback mode is set to Disable, the DHCP server continues running on the node it has failed to. If the DHCP Failback Mode is set to Auto, the DHCP server automatically moves back to its preferred node when the preferred node is brought back online. Set the DHCP Failback mode to Manual to prevent the DHCP server from moving back to its preferred node when that node is brought back online, until you are ready to allow it to happen.x
- 2 View or change the DHCP resource Start, Failover, and Failback modes, then click *Next* and continue with **“View or Edit DHCP Resource Server Assignments” on page 121**.

View or Edit DHCP Resource Server Assignments

The page to view or change DHCP resource server assignments should now be displayed. The DHCP resource template automatically assigns the DHCP resource to all nodes in the cluster. The order of assignment is the order the nodes appear in the resource list.

To view or edit DHCP resource node assignments or change the server failover order:

- 1 From the list of unassigned nodes, select the server you want the resource assigned to, then click the right-arrow button to move the selected server to the *Assigned Nodes* list.
Repeat this step for all servers you want assigned to the resource. You can also use the left-arrow button to unassign servers from the resource.
- 2 Click the up-arrow and down-arrow buttons to change the failover order of the servers assigned to the resource or volume.
- 3 Click *Apply* or *Finish* to save node assignment changes.

9.3 What's Next

The next section describes security issues and recommendations for DHCP on a Novell Open Enterprise Server 2 Linux server.

- ♦ **Chapter 10, “Security Guidelines for DHCP,” on page 123**

This section describes security issues and recommendations for DHCP on a Novell® Open Enterprise Server 2 Linux server. It is intended for security administrators or anyone who is using DHCP for Linux and is responsible for the security of the system. It requires a basic understanding of the DHCP protocol. It also requires the organizational authorization and the administrative rights to effect the configuration recommendations.

- ♦ [Section 10.1, “Best Practices,” on page 123](#)
- ♦ [Section 10.2, “What’s Next,” on page 123](#)

10.1 Best Practices

- ♦ During the DHCP Server installation, ensure that the *Use secure channel for configuration* option is selected. This ensures that the authentication mechanism is secured.
- ♦ You should not run the DHCP server as a `root` user. Instead, use the `-user` and `-group` command line option.
- ♦ To run DHCP server in a more secure environment, use the `-chroot` command line option.
- ♦ To restrict access, use the apparmor profile in `/etc/apparmor.d/usr.sbin.dhcpd`, which restricts access to directories depending on user permissions.
- ♦ You should store user credentials like usernames and passwords in CASA instead of the `/etc/dhcpd.conf` file.
- ♦ If you are setting up DHCP servers, you should include the authoritative statement at the top of the configuration file. This ensures that the DHCP server sends DHCPNAK messages to misconfigured clients.
- ♦ Decide the lease time to be allocated based on your environment. Allocate larger lease time for known-clients and a shorter lease time for unknown clients.

10.2 What’s Next

The next section describes how to install and configure the Novell DNS service on Novell Open Enterprise Server 2 Linux.

- ♦ [Chapter 11, “Installing and Configuring DNS,” on page 125](#)

Installing and Configuring DNS

11

This section describes how to install and configure the Novell® DNS service on Novell Open Enterprise Server 2 SP1 Linux.

- ♦ [Section 11.1, “Planning the Installation,” on page 125](#)
- ♦ [Section 11.2, “Installing the DNS Server,” on page 127](#)
- ♦ [Section 11.3, “Setting Runtime Credentials,” on page 129](#)
- ♦ [Section 11.4, “Verifying Installation,” on page 129](#)
- ♦ [Section 11.5, “Additional Information,” on page 130](#)
- ♦ [Section 11.6, “What’s Next,” on page 130](#)

11.1 Planning the Installation

Before you install the Novell DNS service, review the following:

- ♦ [Section 11.1.1, “Prerequisites,” on page 125](#)
- ♦ [Section 11.1.2, “eDirectory Permissions,” on page 125](#)

11.1.1 Prerequisites

- ♦ iManager 2.7
- ♦ The DNS schema is extended for the specified LDAP host.

IMPORTANT: Do not install any of the following service combinations on the same server as Novell DNS. Although not all of the combinations cause pattern conflict warnings, Novell does not support any of the combinations shown.

- ♦ DHCP and DNS server
 - ♦ Xen Virtual Machine Host Server
-

11.1.2 eDirectory Permissions

- ♦ [“First-Time Installation on an eDirectory Tree” on page 125](#)
- ♦ [“Installing on an eDirectory Tree where DNS Server already exists \(Separate Container\)” on page 126](#)

First-Time Installation on an eDirectory Tree

If you are installing OES 2 SP1 Linux DNS Server on an eDirectory™ tree for the first time, you need to have the following create permissions for the Server, Locator, Group, RootServerInfo, and NCP Server containers at the eDirectory level before you start the DNS installation:

- ♦ Create permission at the entry level of the container for the following:
 - ♦ DNS Server Object Container

- ♦ Locator Object Container (DNS-DHCP)
- ♦ Group Object Container (DNSDHCP-Group)
- ♦ DNS Proxy User Object Container
- ♦ Read and Write permissions for the NCP Server Objects for DNS Server object creation.
- ♦ Write permission at the root of the tree for extending the DNS schema.

For example: If the container for Server object is ou=Server,o=acme, the container for the Locator object, Group object, Proxy User, RootServerInfo object and for the NCP Server object is o=acme1, then the user should have create permissions at the entry level on o=acme and o=acme1.

Installing on an eDirectory Tree where DNS Server already exists (Separate Container)

If you are installing OES 2 SP1 Linux DNS Server on an eDirectory tree make sure that you have the following permissions:

- ♦ Create permission at the entry level of the container for the following objects:
 - ♦ DNS Server Object Container
 - ♦ Locator Object Container (DNS-DHCP)
 - ♦ Group Object Container (DNSDHCP-Group)
 - ♦ Proxy User Object Container
- ♦ Read and Write permissions for the existing objects at the attribute level of the objects.
 - ♦ Group Object
 - ♦ Locator Object
 - ♦ NCP Server Object
- ♦ Create permission at the entry level of the RootServerInfo objects.
- ♦ To retrieve the existing Locator and RootServer Info object context, specify the following rights:

Read and compare rights at the attribute level and browse rights at the entry level of the container where NCP server, DNS Locator and Group, and RootServer Info objects are present.

Table 11-1 eDirectory Object and Attribute Permissions

| DNS Objects and Attributes | Permissions |
|------------------------------|--|
| DNS-DHCP (Locator object) | Read, Write |
| DNSDHCP-Group (Group object) | Read, Write |
| RootServerInfo object | Create |
| NCP Server object | Read, Write |
| DNS Server object | Create |
| Proxy User | Create rights at the entry level of the proxy user container |

11.2 Installing the DNS Server

DNS can be installed on your system by using a YaST install:

YaST Install

This YaST Install is a predefined system of installing components along with the associated dependencies. For a service to function properly, all the dependent products must be installed. Pattern deployment provides patterns for different services. Selecting a pattern automatically selects and installs its dependencies

- 1 In the YaST install, on the *Installation Settings* page, click *Software* to go to the *Software Selections and System Tasks* page.
- 2 Under the OES Services option, select *Novell DNS*. Click *Accept*.
- 3 After the installation is done, the system reboots for the changes to take effect. Provide the details to configure the DNS Server. Refer to the description of the fields in the left pane for more information.

- 4 In the next screen, you can specify configuration parameters for the DNS Server. Refer to the following table for details on the screen fields.

| Configuration Parameter | Details |
|---|---|
| Directory server address | <p>The IP address shown is the default LDAP server for this service. If you do not want to use the default, select a different LDAP server in the list.</p> <p>If you are installing into an existing tree, ensure that the server you select has a master replica or read/write replica of eDirectory. If you need to add another LDAP server to the list, add it by using the LDAP Configuration for Open Enterprise Services dialog box.</p> |
| Get Context and Proxy User Information from Existing DNS Server | <p>If you are configuring DNS in an existing tree where DNS is already configured and you want to use the existing Locator, Root Server Info, Group, and Proxy User contexts, you can select the 'Get Context Information from existing DNS Server' check box and provide the IP of an NCP server hosting the existing DNS server and click 'Retrieve'. This will fetch the contexts of the Locator, Root Server Info, Group, and Proxy User Contexts. Make sure the NCP server hosting the existing DNS Server is running before hitting <i>Retrieve</i>.</p> <p>If you do not wish to use these contexts, you can provide those manually.</p> |
| Novell DNS Services Locator Object Context | <p>Specify the context for the DNS Locator object. For example: o=novell</p> <p>The Locator object contains global defaults, DHCP options, and a list of all DNS and DHCP servers, subnets, and zones in the tree.</p> |
| Novell DNS Services Root Server Info Context | <p>Specify the context for the DNS Services RootServerInfo object. For example: o=novell</p> <p>The RootServerInfo Zone is an eDirectory container object that contains resource records for the DNS root servers.</p> |
| Novell DNS Services Group Object Context | <p>Specify the context for the DNS Group object. For example: o=novell</p> <p>This object is used to grant DNS servers the necessary rights to other data within the eDirectory tree.</p> |
| Novell NCP Server Context | <p>Specify a context for the local NCP Server object. The DNS Server reference is stored in this object. For example: o=novell</p> |
| Use Secure LDAP Port | <p>This option is selected by default to ensure that the data transferred by this service is secure and private. If you deselect this option, the data transferred is in clear text format.</p> |
| Runtime Admin DN | <p>This user is used to authenticate to eDirectory to access information for DNS during runtime. The user must have eDirectory read, write, and browse rights under the specified context.</p> |
| Password | <p>Type the password for the eDirectory user that you specified for accessing DNS.</p> |
| Create DNS Server Object | <p>This option is used to create the DNS Server object in the eDirectory tree associated with the NCP server.</p> |
| Host Name | <p>Specify the unique hostname for the DNS Server object.</p> |

| Configuration Parameter | Details |
|----------------------------|--|
| Domain Name for DNS Server | Specify the domain name for the Server object. |

5 Click *Next* to complete the installation process.

11.3 Setting Runtime Credentials

The Runtime Admin name and Runtime Admin password are stored in the CASA store, which is later used by the DNS Server for eDirectory authentication. The Runtime Admin credentials are stored with root (install user) and named user permissions. There can only be one Runtime User credential for DNS stored in CASA.

NOTE: In cluster setup, Runtime Credentials must be set on all the nodes in the DNS cluster.

If you want to run `novell-named` as any non-root user other than `named`, you need to set the credentials of Runtime Admin for that non-root user because CASA store allows you to read the credentials only as the user who stored it (the non-root user must be an eDirectory user and a member of the `dhcp` Group object) by executing the following commands:

```
su <non-root user>
```

```
KEYVALUE=cn=runtimeuser,o=novell CASACli -s -n dns-ldap -k CN
```

```
KEYVALUE=password CASACli -s -n dns-ldap -k Password
```

This sets the user name and password in CASA store for DNS Server.

Replace `<user>` with a user name of your choice. Replace the `KEYVALUE` with a valid password.

Copy the `/etc/opt/novell/named/.named.cred` to same directory under the `chroot -t dir` and change the user and group ownership from `named:named` to the `-u` user and its group.

11.4 Verifying Installation

After the installation is done, you can verify the installation. Verify that CASA is updated with the new run time admin credentials using `CASACli` option.

- 1 Use iManager to check for the presence of the following objects in the eDirectory™ tree and verify whether these objects are created in the following contexts:
 - ♦ **DNS-DHCP:** This object will be in the Locator Object Context specified during installation.
 - ♦ **DNSDHCP-GROUP:** This object will be in the Group Object Context specified during installation.
 - ♦ **RootServerInfo:** This object will be in the Root Server Info context specified during installation.
 - ♦ **dnipDNSServerVersion:** A version attribute with the value of Novell DNS Server 6.0.0 is added to the NCP Server object.

NOTE: There are multiple instances of DNS-DHCP, DNSDHCPGROUP, RootServerInfo and dnipDNS Server object in the tree.

11.5 Additional Information

For details on how to install, upgrade, and update Novell® Open Enterprise Server (OES) 2 for Linux, see the *OES2 SP1: Linux Installation Guide*.

11.6 What's Next

The next section describes methods to administer the DNS server and manage it by using iManager and the Java Console.

- ♦ [Chapter 13, “Administering and Managing a DNS Server,” on page 133](#)

Migrating DNS/DHCP from OES 1 NetWare to OES 2 SP1 Linux

12

The OES 2 Migration Tool has a plug-in architecture and is made up of Linux command line utilities with a GUI wrapper. You can migrate DHCP to an OES 2 SP1 Linux server through either the GUI Migration Tool or through the command line utilities.

To get started with migration, see “[Planning for Migration](#)” in the *OES 2 SP1: Migration Tool Administration Guide*.

- ♦ For more information on migrating DHCP, see “[Migrating DHCP from NetWare to OES 2 Linux SP1](#)” in the *OES 2 SP1: Migration Tool Administration Guide*.
- ♦ For DNS migration, see “[Migrating DNS from NetWare to OES 2 Linux SP1](#)” in the *OES 2 SP1: Migration Tool Administration Guide*.

Administering and Managing a DNS Server

13

- ♦ Section 13.1, “Using iManager to Manage DNS,” on page 133
- ♦ Section 13.2, “Using the Java Management Console to Manage DNS,” on page 150
- ♦ Section 13.3, “Configuring Roles for a Novell DNS Server,” on page 163
- ♦ Section 13.4, “novell-named Command Line Options,” on page 165
- ♦ Section 13.5, “Starting the DNS Server,” on page 167
- ♦ Section 13.6, “Running DNS Server in chroot Mode,” on page 167
- ♦ Section 13.7, “Running DNS Server as a Non-Root User,” on page 168
- ♦ Section 13.8, “Stopping the DNS Server,” on page 168
- ♦ Section 13.9, “What’s Next,” on page 168

13.1 Using iManager to Manage DNS

- ♦ Section 13.1.1, “Scope Settings,” on page 133
- ♦ Section 13.1.2, “DNS Server Management,” on page 134
- ♦ Section 13.1.3, “Zone Management,” on page 138
- ♦ Section 13.1.4, “Resource Record Management,” on page 146
- ♦ Section 13.1.5, “DNS Key Management,” on page 149

13.1.1 Scope Settings

Configuring the scope settings for a session significantly improves the session's performance. If you do not configure the scope settings for the session, you receive a warning before every task you attempt to perform. However, you can still proceed with the task.

Setting the scope of the DNS services requires two specifications for the session: the Novell® eDirectory™ context of the Locator object and the administrative scope of the session. Specifying the eDirectory context of the Locator object at the start of the session significantly improves performance because it eliminates the need to search for the Locator object. Specifying the administrative scope of the session also improves performance significantly because it restricts the retrieval of DNS objects for viewing to the scope you specify.

When you configure the DNS scope settings for a session, they only last as long as the session lasts. If you start a new session, you must configure the DNS scope settings again.

To configure DNS scope settings:

- 1 Click *DNS > Scope Settings* to open the DNS Scope Settings window.
- 2 Specify the eDirectory context of the DNS Locator object or browse to select it.
- 3 Specify the eDirectory context of the container object that will provide the administrative scope of the current session.

If you specify only the eDirectory context of the DNS Locator object and not the administrative scope of the current session, you can proceed with administrative tasks without receiving a warning message. However, performance is further optimized if you also define the administrative scope.

- 4 Click *OK* to configure the scope settings.

A message indicates that the scope setting request was successful.

- 5 Click *OK* to complete the process.

Or

- 6 Click *Repeat Task* to configure the scope settings again.

13.1.2 DNS Server Management

The DNS Server Management role consists of the following tasks:

- ♦ “Creating a Server” on page 134
- ♦ “Viewing or Modifying a Server” on page 134
- ♦ “Deleting a DNS Server” on page 137
- ♦ “Loading or Unloading a DNS Server” on page 137
- ♦ “Moving a DNS Server” on page 137

Creating a Server

Use iManager to create and set up a server object for each DNS server you plan to operate.

- 1 In iManager, click *DNS > DNS Server Management* to open the DNS Server Management window in the main panel.
- 2 From the drop-down menu, select *Create Server* and click *OK* to open the Create DNS Server window.
- 3 Type the NCP™ server name or browse to select an NCP server from the eDirectory tree.
- 4 Specify a unique hostname for the DNS server object.
- 5 Specify a domain name for the server object.
- 6 Click *Create*.

A message indicates that the new DNS server was created.

NOTE: To configure DNS for an existing NetWare® 6.5 server, create the DNS server to use the iManager plug-in for DNS. If server is created, then NetWare 6.5 NCP Server should have a DNIP:LocatorPtr attribute pointing to the DNS Locator object.

Viewing or Modifying a Server

After you create a DNS server object, you can modify its configuration parameters.

- 1 In iManager, click *DNS > DNS Server Management* to open the DNS Server Management window in the main panel.
- 2 From the drop-down menu, select *View/Modify Server* and click *OK* to open the View/Modify Server window.

- 3 Select the DNS server from the drop-down list, then click *OK*.
- 4 Follow the on-screen instructions to view and modify the following DNS server configuration parameters:

List of Zones: The names of the zones that the server manages along with the roles of this server for each of the zones. This field cannot be edited.

DNS Server IP Address: The IP addresses on which the DNS server listens for queries. This field cannot be edited.

DNS TSIG Key Information: Information on available and associated keys.

- ♦ **Available DNS TSIG Keys:** A list of DNS TSIG keys that are available in the eDirectory tree. These keys can be associated with the DNS server.
- ♦ **Associated DNS TSIG Keys:** A list of DNS TSIG keys that are associated with the DNS server.

To add a DNS TSIG key, select the key, then click *Add*.

To remove a DNS TSIG key, select the key, then click *Remove*.

To add all the keys, click *Add All*.

To remove all the keys, click *Remove All*.

To add or remove multiple keys, use the Ctrl key to select the keys, then click *Add* or *Remove*.

Domain name: The domain name of the DNS server.

Comments: Add your comments about the DNS server. This parameter is optional.

Forward List: A list of IP addresses of DNS servers to which unresolved queries will be forwarded.

To add servers to the Forward List, click *Add*, specify the IP address of the server, then click *Add* again.

To remove servers from the Forward List, select the IP address of the server from the Forward List, then click *Delete*.


You can also use this list to control the behavior of queries for which the server is not authoritative and the answers do not exist in the cache. Values can be either First or Only. If you specify the value as First, which is the default, the server queries the list of forwarders first. If no answer is found, the server searches for the answer. If you specify the value as Only, the server queries only the forwarders list.

No-Forward List: A list of domain names whose unresolved queries will not be forwarded to other DNS servers.


To add domain names to the list, click *Add*, specify the domain name of the server, then click *OK*.

To remove domain names from the list, select the domain name, then click *Delete*.


Allow Recursion: A list of IP addresses or networks that can submit recursive DNS queries. If you want to disable recursion, specify a value of None.

To add the address match list element, click , then specify the IP address and the mask length. The network number is optional.


To add a generic option, select the Predefined match list, select from the available options in the drop-down list, then click *OK*.

To delete the address match list element, select the item to be deleted, then click .


Query Filter: A list of IP addresses or networks that are authorized to query the DNS server. If no IP address is specified, queries are allowed from all hosts.

To add the IP address, click , then specify the IP address and the mask length. The network number is optional.


To add a generic option, select the Predefined match list, select from the available options in the drop-down list, then click *OK*.

To delete the IP address, select the item to be deleted, then click .

Zone Out Filter: A list of IP addresses or networks that are authorized to perform zone transfer from the DNS server.


To add the address match list element, click , then specify the IP address and the mask length. The network number is optional.

If you want to add a generic option, select the Predefined match list, select from the available options in the drop-down list, then click *OK*.

To delete the address match list element, select the item to be deleted, then click .


Also Notify: A list of IP addresses of name servers that receive Notify messages, when a fresh copy of the zone is loaded.

To add the IP address, Click , specify the IP address, then click *OK*.

To delete the IP address, select the IP address you want to delete, then click .

Blacklist Server: Specifies a list of IP addresses of servers that are not approved. The DNS server does not answer queries from or forward queries to the servers listed.

To add the IP address, click , specify the IP address, then click *OK*.

To delete the IP address, select the IP address you want to delete, then click .

Maximum Cache Size: The maximum amount of memory in kilobytes that the server can use as cache.

Maximum Recursion Lookups: The maximum number of simultaneous recursive lookups the server performs on behalf of the clients.

Current Set of Additional Options: The additional global server and zone options. To view the options, click *Modify* to open the View/Modify Server window.

To add an available additional option, select the option and click *Add*.

To add all available additional options, click *Add All*.

To remove an available additional option, select the option and click *Remove*.

To remove all available additional options, click *Remove All*.

To delete all option names in the list, click the top-level check box, then click *Delete*.

To remove one or more option names, click the check box next to the option, then click *Delete*.

Check Names: Restrictions on the character set and syntax of certain domain names in the master zone and the DNS response received from the network.

For masters, slave zones, and network responses, the default is to ignore them. This parameter applies to the owner names of A, AAA, and MX records. It also applies to domain names in the RDATA of NS, SOA, and MX records, and to the RDATA of PTR records where the owner name indicates that it is a reverse lookup of a hostname.

Deleting a DNS Server

- 1 Click *DNS > DNS Server Management* to open the DNS Server Management window in the main panel.
- 2 From the drop-down menu, select *Delete Server* and click *OK* to open the Delete DNS Server window.
 - ♦ To remove all DNS servers in the list, click the top-level check box and click *Delete*.
or
 - ♦ To remove one or more DNS servers, click the check box next to it and click *Delete*.

Loading or Unloading a DNS Server

- 1 In iManager, click *DNS > DNS Server Management* to open the DNS Server Management window in the main panel.
- 2 From the drop-down menu, select *Load/Unload Server* and click *OK* to open the Load/Unload DNS server window.
- 3 Select the DNS server and specify the port number on which the DNS server is configured.
This port is required to check whether the DNS server is running or not. By default, port 53 is used if no other port number is specified.
- 4 Click *OK*.
- 5 Depending on the state and the version of the DNS server, one of the following happens:
 - ♦ If novell-named is not loaded on the machine, you are prompted to load novell-named.
 - ♦ If novell-named is already loaded on the machine, you are prompted to unload it. To unload novell-named, click *Unload*.

Currently, it is not possible to load novell-named using command line options from DNS iManager on Linux.

- 6 Click *OK* to complete the task.

Moving a DNS Server

This task enables you to move DNS Services from one NCP server to another NCP server. You can also convert a DNS server into a cluster-enabled DNS server by moving it to a virtual NCP server.

- 1 In iManager, click *DNS > DNS Server Management* to open the DNS Server Management window in the main panel.
- 2 From the drop-down menu, select *Move DNS Server* and click *OK* to open the Move DNS Server window.
- 3 Select the DNS server name from the drop-down list. Only OES 2 Linux servers are displayed in this list.
- 4 Specify the name of the NCP Server that the DNS Services will be moved to, or use the Object Selector icon to browse and select it.
- 5 Click *Move*.

13.1.3 Zone Management

The DNS Zone object is an eDirectory™ container object that is made up of Resource Record Set (RRSet) objects and resource records.

- ♦ “Creating a Zone” on page 138
- ♦ “Viewing or Modifying a Zone Object” on page 141
- ♦ “Associating a Zone to Specific DNS Servers” on page 145
- ♦ “Deleting a Zone Object” on page 145
- ♦ “Importing a Zone Object” on page 146
- ♦ “Exporting a Zone Object” on page 146

Creating a Zone

- ♦ “Creating a Primary Zone” on page 138
- ♦ “Creating a Forward Zone” on page 138
- ♦ “Creating a Secondary Zone” on page 139
- ♦ “Creating a Primary IN-ADDR.ARPA Zone” on page 139
- ♦ “Creating a Forward IN-ADDR.ARPA Zone” on page 140
- ♦ “Creating a Secondary IN-ADDR.ARPA Zone” on page 140

Creating a Primary Zone

- 1 In iManager, click *DNS > Zone Management* to open the Zone Management window in the main panel.
- 2 From the drop-down menu, select *Create Zone* and click *OK* to open the Create DNS Zone window.
- 3 Select *Create New Zone*.
- 4 Specify the eDirectory context for the zone or browse to select it.
- 5 Specify a name for the zone.
- 6 Select *Primary (default)* as the Zone Type.
- 7 Select a DNS server from the *Assigned Authoritative DNS Server* drop-down menu.
or
Specify a unique hostname in the *Name Server Host Name* box and select a domain by clicking the *Add* button, then click *OK*.
- 8 Click *Create*.
A message indicates that the new primary zone has been created.

Creating a Forward Zone

- 1 In iManager, click *DNS > Zone Management* to open the Zone Management window in the main panel.
- 2 From the drop-down menu, select *Create Zone*, then click *OK* to open the Create DNS Zone window.
- 3 Select *Create New Zone*.

- 4 Specify the eDirectory context for the zone or browse to select it.
- 5 Specify a name for the zone.
- 6 Select Forward as the *Zone Type*.
- 7 Select a DNS server from the *Assigned Authoritative DNS Server* drop-down menu. This parameter is optional.
or
Specify a unique hostname in the *Name Server Host Name* box and select a domain by clicking the *Add* button, then click *OK*.
- 8 Click *Create*.
A message indicates that the new forward zone has been created.

Creating a Secondary Zone

- 1 In iManager, click *DNS > Zone Management* to open the Zone Management window in the main panel.
- 2 From the drop-down menu, select *Create Zone*, then click *OK* to open the Create DNS Zone window.
- 3 Select *Create New Zone*.
- 4 Specify the eDirectory context for the zone or browse to select it.
- 5 Specify a name for the zone.
- 6 Select *Secondary* as the *Zone Type*.
- 7 Specify the IP address of the DNS server that will provide zone out transfers for this secondary zone.
- 8 Select a DNS server from the *Assigned Authoritative DNS Server* drop-down menu. This parameter is optional.
or
Specify a unique hostname in the *Name Server Host Name* box and select a domain by clicking the *Add* button, then click *OK*.
- 9 Click *Create*.
A message indicates that the new secondary zone has been created.

Creating a Primary IN-ADDR.ARPA Zone

- 1 In iManager, click *DNS > Zone Management* to open the Zone Management window in the main panel.
- 2 From the drop-down menu, select *Create Zone* and click *OK* to open the Create DNS Zone window.
- 3 Select *Create IN-ADDR.ARPA*.
- 4 Specify the eDirectory context for the zone or browse to select it.
- 5 Specify the network address of the zone in the *Network Address* field.
For example, specify 143.72.155 for 155.72.143.IN-ADDR.ARPA.
The IN-ADDR.ARPA zone name is displayed in the *Zone Domain Name* field.
- 6 Select the *Zone Type* as Primary (default).

- 7 Select a DNS server from the *Assigned Authoritative DNS Server* drop-down menu. This parameter is optional.
or
Specify a unique hostname in the *Name Server Host Name* box and select a domain by clicking the *Add* button, then click *OK*.
- 8 Click *Create*.
A message indicates that the new Primary IN-ADDR.ARPA Zone object has been created.

Creating a Forward IN-ADDR.ARPA Zone

- 1 In iManager, click *DNS > Zone Management* to open the Zone Management window in the main panel.
- 2 From the drop-down menu, select *Create Zone*, then click *OK* to open the Create DNS Zone window.
- 3 Select *Create IN-ADDR.ARPA*.
- 4 Specify the eDirectory context for the zone or browse to select it.
- 5 Specify the network address in the *Network Address* field.
The IN-ADDR.ARPA zone name is displayed in the *Zone Domain Name* field.
- 6 Select the *Zone Type*, then select *Forward*.
- 7 Select a DNS server from the *Assigned Authoritative DNS Server* drop-down menu. This parameter is optional.
or
Specify a unique hostname in the *Name Server Host Name* box and select a domain by clicking the *Add* button, then click *OK*.
- 8 Click *Create*.
A message indicates that the new Forward IN-ADDR.ARPA Zone object has been created.

Creating a Secondary IN-ADDR.ARPA Zone

- 1 In iManager, click *DNS > Zone Management* to open the Zone Management window in the main panel.
- 2 From the drop-down menu, select *Create Zone*, then click *OK* to open the Create DNS Zone window.
- 3 Select *Create IN-ADDR.ARPA*.
- 4 Specify the eDirectory context for the zone or browse to select it.
- 5 Specify the network address in the *Network Address* field.
The IN-ADDR.ARPA zone name is displayed in the *Zone Domain Name* field.
- 6 Under the *Zone Type*, select *Secondary*.
- 7 Select a DNS server from the *Assigned Authoritative DNS Server* drop-down menu.
or
Specify a unique hostname in the *Name Server Host Name* box and, optionally, specify a domain name or select it from the *Domain* drop-down menu.

- 8 Specify the IP address of the DNS server that will provide zone-out transfers for this secondary zone.
- 9 Click *Create*.

A message indicates that the new Secondary IN-ADDR.ARPA Zone object has been created.

Viewing or Modifying a Zone Object

After you have created a Zone object, you can modify it and provide more detailed configuration information.

- 1 In iManager, click *DNS > Zone Management* to open the Zone Management window in the main panel.
- 2 From the drop-down menu, select *View/Modify Zone* and click *OK* to open the View/Modify Zone window.
- 3 Select the DNS Zone object from the drop-down menu.
- 4 Click *OK*.
- 5 Modify the following DNS Zone configuration parameters.

- ♦ Zone Type: Specifies whether the zone is a primary or secondary zone.

To configure a server as a passive primary for a zone, specify the server name in the *Authoritative DNS servers* field of that zone. Make sure this server name is not in the *Designated Primary DNS Server* field.

To configure a server as designated primary DNS server, specify the server name in the *Authoritative DNS servers* field of that zone and select that server name from the *Designated Primary DNS Server* field.

To configure a server as a passive secondary for a zone, specify the server name in the *Authoritative DNS servers* field of that zone. Make sure this server name is not in the *Designated Secondary DNS Server* field.

To configure a server as designated secondary server, specify the server name in the *Authoritative DNS servers* field and select that server name in the *Designated Secondary DNS Server* field.


NOTE: It is not possible to change the zone type from primary/secondary to forward and vice versa.


- ♦ Zone Master IP Address: If the zone type is secondary, specify the IP address of the master server for this zone.
- ♦ Available DNS Servers: Lists the available DNS Servers that are not assigned to this zone.
- ♦ Authoritative DNS Servers: Lists all authoritative servers for this zone.
 - ♦ Click *Add All* to assign all available DNS servers to a zone.
 - ♦ Click *Remove All* to remove all authoritative DNS servers from a zone.
- ♦ Designated DNS Server: The DNS server selected in this field will act as a designated primary or designated secondary server depending on whether the zone type is primary or secondary.
- ♦ Comments: You can provide information about the zone in this field. This parameter is optional.

- ♦ **Forward List:** Specifies a list of DNS servers to which unresolved queries are sent.
- ♦ **Forwarder:** Controls the behavior of queries for which the server is not authoritative and the answers do not exist in the cache. Values can be either First or Only. The default is First. If you specify the value as First, the server will query the forwarders list first and, if the answer is not found, the server will search for the answer. If you specify the value as Only, the server will query only the forwarders list.
- ♦ **Modify Zone Out Filter:** Specifies a list of IP addresses or networks authorized to perform zone transfers for this zone from the DNS server managing it.
- ♦ **Zone Master:** Specifies the domain name of the master DNS server.
- ♦ **E-mail Address:** Specifies the e-mail address (with @ replaced by a dot) of the person responsible for this zone.
- ♦ **Serial Number:** Use this field to set a version number for the Start of Authority.
- ♦ **Interval values:** Select from the following values:
 - ♦ **Refresh:** Specifies the time interval before the secondary name server transfers a copy of the zone data to the primary name server. The default is 180 minutes.
 - ♦ **Retry:** Specifies the time that a secondary name server waits after a transfer has failed and before it tries to download the zone database again. The default is 60 minutes.
 - ♦ **Expire:** Specifies the time after which a secondary name server will be unable to download a zone database. The default is 168 hours.
 - ♦ **Minimal TTL:** Specifies the minimum TTL for a resource record. This parameter determines the period for which a DNS server retains an address mapping in the cache. The default is 24 hours.
- ♦ Select the *Define advanced properties* check box, then click *Next* to modify the following values:

- ♦ **Query Filter:** Specifies a list of IP addresses or networks that are authorized to query the DNS server for this zone. This list overrides the query filter specified at the server for this zone.


To add the IP address:

- ♦ Click 
- ♦ Specify the IP address and the mask length.
The network number is optional.
- ♦ If you want to add a generic option, check the Predefined match-list to select from the available options in the drop-down list.
- ♦ Check the Key option and specify the DNS key from the drop-down list.
- ♦ Click OK.

To delete the IP address, select the item to be deleted, then click .


- ♦ **Also Notify:** Specifies a list of IP addresses of name servers that receive Notify messages, when a fresh copy of the zone is loaded.


To add the IP address, Click , specify the IP address, then click OK.

To delete the IP address, select the IP address you want to delete, then click .

- ♦ **Allow Update:** Specifies a list of IP addresses or network addresses that are authorized to send updates for this zone. If this option is not configured, the default value of none is used, where no host is authorized to send updates.

To add the address match list element:


- ♦ Click .
- ♦ Specify the IP address and the mask length.
The network number is optional.
- ♦ If you want to add a generic option, check the Predefined match-list to select from the available options in the drop-down list.
- ♦ Select the TSIG key option from the drop-down list. If not specified, then the default value is none.
- ♦ Click OK.

To delete the address match list element, select the item to be deleted, then click .

- ♦ **Update Policy:** Specifies the policy to update the measure to implement security for a zone object. This is implemented by the default DNS server administering the zone. The update policy is a five-token string where each token has a definite function to perform. The syntax for update policy is:

Permission Identity MatchType TName RR

To add the update policy:

- ♦ Click .
- ♦ Select the Permission from the drop-down list. The permission can either be a grant or a deny.
- ♦ Specify the Identity that refers to the name of the key used to sign the update.
- ♦ Specify the MatchType from the drop-down list.
 - ♦ **name:** Matches when the domain name being updated is the same as the name in the name field.
 - ♦ **subdomain:** Matches when the domain name being updated is a subdomain of (that is, ends in) the name in the name field. The domain name must still be in the zone.
 - ♦ **wildcard:** Matches when the domain name being updated matches the wildcard expression in the name field.
 - ♦ **self:** Matches when the domain name being updated is the same as the name in the identity field, not the name field that is when the domain name being updated is the same as the name of the key used to sign the update. If nametype is self, then the name field is ignored; however, you should include the name field when using a nametype of self.
- ♦ *TName:* Specify the *TName*, which is the domain name appropriate to the *MatchType* specified.
- ♦ (Optional): Specify the *RR* (Resource Record), which can contain any valid record type.

NOTE: The Allow Update with keys option and the Update Policy options are supported for Linux DNS only.

- ♦ Click Next to associate the DNS TSIG keys with the Zone.

NOTE: In earlier versions, key association was a must before updating a policy. Now, it is not required for SAM because the keys are negotiated at run time. Because of this, no checking is done to validate the identity field for SAM-based updates.

Available DNS TSIG Keys: Displays a list of DNS TSIG keys that are available in the eDirectory tree. These keys can be associated with the Zone.

Associated DNS TSIG Keys: Displays a list of DNS TSIG Keys that are associated with the Zone.

- ◆ To add the DNS TSIG Key, then select the key > click *Add*.
- ◆ To remove the DNS TSIG key, then select the key > click *Remove*.
- ◆ To add all the keys, click *Add All*.
- ◆ To remove all the keys, click *Remove All*.

NOTE: To add or remove multiple keys, use the Ctrl key to select the keys, then click *Add* or *Remove*.

- ◆ Click *Next* to specify the current set of additional options. To modify the options, click *Modify*. Select the appropriate option from the *Available Additional Option(s)* list. The following are the additional options for the zone:
 - ◆ **allow-notify:** Specifies the list of hosts that are allowed to notify the slaves of zone changes in addition to the zone masters. You can configure this option only for a secondary zone.
 - ◆ To add the address list:
 1. Click *Add*.
 2. Specify the IP address and the mask length. The mask length is optional. OR Check *Predefined match-list* to select from the available options from the drop-down list. If you select *None*, the server will reject notifies sent by any other server.
 3. Click *OK*.
 - ◆ To delete the address list:
 1. Select the IP address to be deleted.
 2. Click *Remove*.

Allow-notify specified at the server level is overridden by the settings of this zone.

- ◆ **max-journal-size:** Sets a maximum size in bytes for the journal file. This should be configured only for a Linux zone.

NOTE: All changes made to a zone by using dynamic update are written to the zone's journal file. The server periodically flushes the complete contents of the updated zone to its zone file. This happens approximately every 15 minutes. When a server is restarted after a shutdown, it replays the journal file to incorporate into the zone any updates that took place after the last zone file update. The dynamic reconfig interval setting is immaterial for a max-journal-size event triggering.

- ♦ **notify:** Specifies if the notification of any zone data changes has to be sent to a slave server. You can select from the following options:
 - ♦ Yes: A notification is sent to all the name servers of the zone when the zone data changes.
 - ♦ Explicit: A notification is sent explicitly to the servers specified in the also-notify list when the zone data changes.
 - ♦ No: A notification is not sent.

A notification specified at the server level is overridden by the settings of this zone.

- ♦ **notify-source:** Specifies the local source address. You also have the option to specify the UDP port that is used to send notify messages. The local source address must appear in the masters list of the slave server or in the allow-notify list. The slave should also be configured to receive notify messages from this address.

Notify-source specified at the server level is overridden by the settings of this zone.

- ♦ **transfer-source:** Specifies the local addresses that are bound to the IPv4 TCP connections used by the zones that are transferred inbound by the server. It also specifies the source IPv4 address and, optionally, the UDP port. The UDP port is used to refresh queries and forward any dynamic updates.

If you have not set a value, this option defaults to a system-controlled value usually the address of the interface closest to the remote end.

Transfer-source specified at the server level is overridden by the settings of this zone.

- ♦ **zone-statistics:** Specifies the statistical information that is dumped to the statistics-file for all zones in the server. Values can be either Yes or No.

If you set the value to Yes, the server collects statistical data on all zones in the server.

Zone-statistics specified at the server level is overridden by the settings of this zone.

Click *Done* after the additional option(s) are selected.

- ♦ Click *Done* to complete the modify process. A confirmation message displays that the modify process succeeded.

Associating a Zone to Specific DNS Servers

A DNS server can be configured to serve only the queries by specifying the role of a zone as passive, secondary, or passive secondary.

To associate the existing DNS zone to a specific DNS server and specify the role of the zone:

- 1 In the iManager DNS role, select the Zone Management task.
- 2 From the list of operations, select *View Modify Zone*.
- 3 Select the zone you want to modify.
- 4 Specify the Authoritative DNS server for this zone, which is the zone for the specific DNS server.
- 5 Click *Save*.

Deleting a Zone Object

- 1 In iManager, click *DNS > Zone Management* to open the Zone Management window in the main panel.

- 2 From the drop-down menu, select *Delete Zone* and click *OK* to open the Delete DNS Zone window.
- 3 Select the DNS zones that are to be deleted.
To delete all the Zone objects in the list, click the top-level check box.
- 4 Click *Next*.
- 5 Select the zones whose sub-zones are to be deleted.
To delete all the sub-zone objects in the list, click the top-level check box.
- 6 Click *Delete*.

Importing a Zone Object

- 1 In iManager, click *DNS > Zone Management* to open the Zone Management window in the main panel.
- 2 From the drop-down menu, select *Import Zone* and click *OK* to open the Import DNS Zone window.
- 3 Specify or browse the eDirectory context where the zone is to be created.
- 4 Select a designated DNS server distinguished name.
This server will subsequently manage the zone data.
- 5 Select the *Zone Type* as Primary (default) or Secondary. If you select Secondary as the Zone type, specify the IP address of the zone.
- 6 Specify or browse to select the DNS Bind File location.
- 7 Click *OK*.

If the import operation encounters any problems, you can view the error details by downloading the log file. Also, if any of the resource records are ignored because of this problem, you can create them again by using the task in [“Creating Resource Records” on page 147](#).

Exporting a Zone Object

- 1 In iManager, click *DNS > Zone Management* to open the Zone Management window in the main panel.
- 2 From the drop-down menu, select *Export Zone* and click *OK* to open the Export DNS Zone window.
- 3 From the drop-down menu, select the DNS Zone to which the file will be exported.
- 4 Click *OK*.
- 5 Click the *Click here to download link* to open the File Download dialog box.
- 6 Select *Save to disk*, then click *OK*.
- 7 Specify the name of the file or browse to select it, then click *Save*.
- 8 Click *Done*.

13.1.4 Resource Record Management

An RRSset object represents an individual domain name within a DNS zone. Each RRSset object has one or more resource records beneath it that contain additional information about the domain.

The most common resource records are Address (A) records, which map a domain name to an IP address, and Pointer (PTR) records, which map an IP address to a domain name within an IN-ADDR.ARPA zone.

Creation, Modification, or Updating the resource records is not supported for a Forward Zone.

The Resource Record Management role consists of the following tasks:

- ♦ “Creating Resource Records” on page 147
- ♦ “Viewing or Modifying Resource Records” on page 147
- ♦ “Deleting Resource Records” on page 148

Creating Resource Records

A resource record is a piece of information about a domain name. Each resource record contains information about a particular piece of data within the domain.

To create a new resource record:

- 1 In iManager, click *DNS > Resource Record Management* to open the Resource Record Management window in the main panel.
- 2 From the drop-down menu, select *Create Resource Record* and click *OK* to open the Create Resource Record window.
- 3 From the drop-down menu, select the domain name where the resource record is to be created, then click *Create*.
Resource records cannot be created in a secondary zone.
- 4 Specify the owner name under which you want to create the resource record or click the Object Selector icon to search for an existing owner name.
If the owner name is not specified, the resource record is created under @.
- 5 Select the *RR Type*.
 - ♦ **A:** Maps a domain name to an IP address. If you select this option, you must specify the 32-bit IPv4 address that will map to the associated domainA.
 - ♦ **CNAME:** Specifies the canonical or primary name for the owner. Because the owner name is an alias, you must specify the domain name of the aliased host if you select this option.
 - ♦ **Others:** From the Others drop-down menu, select the resource record type (RR Type) and specify the appropriate resource record data corresponding to the type chosen.
- 6 Click *Create*.
- 7 Click *OK* after the resource record is created.
- 8 Click *Done* to automatically increment the serial number and complete the task.

For more information on resource record types, see [Section A.2, “Types of Resource Records,” on page 196](#).

Viewing or Modifying Resource Records

- 1 In iManager, click *DNS > Resource Record Management* to open the Resource Record Management window in the main panel.

- 2 Select *View/Modify Resource Record* from the drop-down menu and click *OK* to open the *Modify RRSset - Resource Record* window.
- 3 From the *Select Domain* drop-down menu, select the domain that contains the host or RRSset.
Resource records cannot be created in a secondary zone.
- 4 Specify or search to select the values for the: *Host Name*, *Resource Record*, and *Resource Record Type* fields.
To use the search feature to select these values:
 - ♦ Click the *Object Selector* icon to open the *Object Selector* window.
 - ♦ Select the *Resource Record* type from the drop-down menu, select the number of search results to be displayed per page, click *Search*, then click the hostname.
This automatically fills in the *Host Name*, *Resource Record*, and *Resource Record Type* fields.
- 5 Click *Modify* to modify the resource record data.
- 6 Modify the resource record data for all but the following types of resource records:
 - ♦ A (or IPv4)AAAA (or IPv6)A6
 - ♦ PTR
- 7 Specify new comments or modify existing comments for the resource record data.
- 8 Click *Done* to save the changes.

Deleting Resource Records

You can delete one, more than one, or all resource records and RRSets, using the multi-select deletion feature in iManager.

- 1 In iManager, click *DNS > Resource Record Management* to open the *Resource Record Management* window in the main panel.
- 2 From the drop-down menu, select *Delete Resource Record* and click *OK* to open the *Delete RRSset - Resource Record* window.
- 3 From the *Select Domain* drop-down menu, select the domain that contains the host or RRSset.
- 4 Delete one or more RRSets:
 - 4a To search for RRSets by owner name, specify the name of the RRSset owner.
 - 4b Select the RRSset option from the *Search Type* drop-down menu.
 - 4c Click *Search* to list the available RRSets that match the specified owner name.
 - 4d To delete all RRSets listed, click the top-level check box and click *Delete*. To delete one or more RRSets, click the corresponding check boxes and click *Delete*.
- 5 Delete one or more resource records:
 - 5a To search for a resource record by owner name, specify the name of the Resource Record owner.
 - 5b Select Resource record from the *Search Type* drop-down menu.
 - 5c Select the resource record type from the *RR Type* drop-down menu.
 - 5d Click *Search* to list the available resource records that match the specified owner name. To delete all resource records listed, click the top-level check box and click *Delete*. To delete one or more resource records, click the corresponding check boxes and click *Delete*.

NOTE: When the A and PTR type resource records are deleted, the corresponding PTR and A resource records also deleted.

13.1.5 DNS Key Management

The DNS Key Management role consists of tasks that allow you to create, modify, and delete DNS Key objects.

A DNS Key provides a means of authentication for dynamic DNS updates and for queries to a secured DNS Server. A DNS Key uses shared secret keys as a cryptographically secure means of authenticating a DNS update or query. Only HMAC-MD5 algorithm is supported for DNS Key management. HMAC-MD5 keys must be between 1 and 512 bits. For more information, see the `dnssec-keygen` manpage.

NOTE: The DNS key option is supported for Linux DNS only.

Unsupported `dnssec-keygen` features

- ♦ **-a:** RSA, RSAMD5, DH, DSA, RSASHA1 are not supported by novell-named.
- ♦ **-n:** ZONE nametype.
- ♦ **-f:** setting the flag in DNSKEY record.
- ♦ **-p:** protocol support is not affirmed as it is used in conjunction with DNSKEY for DNSSEC.

Example:
`dnssec-keygen -v`
Usage:
`dnssec-keygen -a -HMAC-MD5 -b 218 -n HOST mykey`
Version: 9.3.4
Required options:
`-a` algorithm: RSA | RSAMD5 | DH | DSA | RSASHA1 | HMAC-MD5
`-b` key size, in bits:
RSAMD5: [512..4096]
RSASHA1: [512..4096]
DH: [128..4096]
DSA: [512..1024] and divisible by 64
HMAC-MD5: [1..512]
`-n` nametype: ZONE | HOST | ENTITY | USER | OTHER
name: owner of the key

The following sections provides information on DNS Key management:

- ♦ [“Creating a DNS Key” on page 149](#)
- ♦ [“Viewing or Modifying a DNS Key” on page 150](#)
- ♦ [“Deleting a DNS Key” on page 150](#)

Creating a DNS Key

- 1 In iManager, click *DNS > DNS Key* to open the *DNS Key Management* window in the main panel.
- 2 From the drop-down menu, select *Create DNS Key* and click *OK* to open the *Create DNS Key* window.

- 3 Specify a name to identify the DNS key in the *DNS Key Name* field.
- 4 Specify the name of the *Algorithm*. The HMAC-MD5 algorithm is the only supported algorithm for a DNS key.
- 5 Specify the secret key used by the DNS server to encrypt/decrypt the hashed data. Secret-456errt4545= is the secret key generated by dnssec-keygen.
The secret key provided must be Base64 encoded, else the DNS server fails to start.
- 6 Specify or browse to select the *NDS context*.
- 7 Click *Create*. The DNS key is now created.
Example: DNS KeyName-Key1,Alorithm-HMAC-MD5,Key Secret-456errt4545=

Viewing or Modifying a DNS Key

- 1 In iManager, click DNS > DNS Key to open the *DNS Key Management* window in the main panel.
- 2 From the drop-down menu, select *View/Modify DNS Key* and click *OK* to open the View/Modify DNS Key window.
- 3 From the drop-down menu, select the DNS Key that you want to view/modify, then click *OK* to open the *Modify DNS Key* window.
- 4 Modify the attributes such as *Secret Key*, and the associated comments, then click *OK*.

Deleting a DNS Key

- 1 In iManager, click DNS > DNS Key to open the DNS Key Management window in the main panel.
- 2 From the drop-down menu, select *Delete DNS Key* and click *OK* to open the Delete DNS Key window.
- 3 Select the DNS key that is to be deleted. Click *Delete*. The DNS key is now deleted.
To delete multiple DNS keys, click the top-level folder. Click *Delete*.

NOTE: Deleting DNS key objects, deletes the references to key objects (if any) in Zone and DNS server objects.

13.2 Using the Java Management Console to Manage DNS

This section provides information about configuring DNS by using the Java-based Management Console.

- ♦ [Section 13.2.1, “Installing the Java Management Console,” on page 151](#)
- ♦ [Section 13.2.2, “DNS Server Management,” on page 151](#)
- ♦ [Section 13.2.3, “Zone Management,” on page 154](#)
- ♦ [Section 13.2.4, “Resource Record Management,” on page 160](#)
- ♦ [Section 13.2.5, “DNS Key Management,” on page 161](#)

13.2.1 Installing the Java Management Console

- ♦ Install the Java Management Console on client computers that will be used to administer DNS and DHCP.
- ♦ Install the Novell Client™ software on client computers that will be used to administer DNS and DHCP.

The following are the command line options that can be specified while launching the Java Management console:

Table 13-1 *Command Line Options*

| Option | Use |
|--------|--|
| -c | Specifies the context in which the DNS locator object is present. When you use this option, you can eliminate the search for the DNS Locator object and obtain a quicker startup for the DNS Java Management Console. |
| -s | Limits the administrative scope of the DNS Java Management console. If you manage only objects under the ctp.novell context, you can set the option as <code>-s ctp.novell</code> and launch the management Console. With this option set, you can view only those DNS objects that are under the ctp.novell eDirectory context. Using this option might improve the server performance because not all DNS objects are read. If you do not set this option, all the DNS objects in the tree are displayed. |
| -mx | Specifies the maximum heap size to be used by the DNS Java management console. The default heap size is 64 MB. If you have a large number of DNS objects to be displayed, you can increase the maximum heap size with this option. To specify 100 MB as the heap size, you can set this option as <code>-mx 100m</code> . |

You can edit the target of the Java management console shortcut to permanently set these options instead of specifying them every time you launch the management console. For example, you can set the options by editing the target as shown below:


```
C:\program files\novell\dnshcp\dnshcp.exe" -c dnshcp.novell -p 1000 -s ctp.novell -mx 100m
```

13.2.2 DNS Server Management

DNS server management involves the following tasks:

- ♦ [“Creating a DNS Server Object” on page 151](#)
- ♦ [“Viewing or Modifying a DNS Name Server Object” on page 152](#)
- ♦ [“Deleting a DNS Server” on page 153](#)
- ♦ [“Starting or Stopping a DNS Server” on page 153](#)
- ♦ [“Moving a DNS Server” on page 154](#)

Creating a DNS Server Object

- 1 Click the *DNS Service* tab of the Management Console.
- 2 Click *Create*  on the toolbar.

- 3 Select *DNS Server* in the Create New DNS Object dialog box, then click *OK*.

The Create New DNS Server dialog box is displayed, prompting you to select an NCP Server object.

- 4 Specify the desired server's name or use the browse button to select the server.
- 5 Specify the server's domain name.
- 6 Click the *Define Additional Properties* check box to view the newly created server property pages.
- 7 Click *Create*.

The DNS Server object is created and displayed in the lower pane of the Management Console.

NOTE: To configure DNS for an existing NetWare 6.5 server, create the DNS Server to use the iManager plug-in for DNS. If server is created, then the NetWare 6.5 NCP server should have DNIP:LocatorPtr attribute pointing to the DNS Locator object.

Viewing or Modifying a DNS Name Server Object

To modify an existing DNS Name Server object, click the object's icon in the lower pane of the DNS Service window to display detailed information in the right pane. A DNS Name Server object's detailed information window displays four tab pages:

- ♦ **Zones:** On this page, the zone list contains a list of all zones and the role each zone serves for the selected DNS Name Server object.

To change the zone information, you must modify the specific Zone object. This information cannot be modified from the server page.

The *DNS Server IP Address* field is read-only and is received from the DNS server.

- ♦ **Forwarding List:** This page displays a list of all forwarding IP addresses.
 - ♦ To add an address to the list, click *Add*. Specify the IP address in the Add Forward IP Address field, then click *OK*.
 - ♦ To delete an address from the list, select an IP address and click *Delete*.
- ♦ **No-Forward List** This page displays a list of all domain names to which queries are not sent.
 - ♦ To add a domain name to the No-Forward List, click *Add*. Specify the domain name in the *No-Forward Name* field, then click *OK*.
 - ♦ To delete a domain name from the list, select the domain name from the list and click *Delete*.
- ♦ **Options** This page allows you to configure maximum cache size and max recursion for a new DNS server.
- ♦ **Key List**

Available DNS Keys: Displays a list of DNS keys that are available in the eDirectory tree. These keys can be associated with the DNS server.

Selected DNS Keys: Displays a list of DNS keys that are associated with the DNS server.

- ♦ To add the DNS Key, select the key, then click *Add*.
- ♦ To remove the DNS key, select the key, then click *Remove*.

- ♦ To add all the keys, click *Add All*.
- ♦ To remove all the keys, click *Remove All*.

NOTE: To add or remove multiple keys, use the Ctrl key to select the keys. Then click *Add* or *Remove*.

- ♦ **Control Lists**

This page displays various lists that can be configured to control the behavior of the DNS server. You can configure the zone out filter, allow recursion, query filter as address match lists. You can also configure the also notify and black listed servers as a list of IP addresses.

- ♦ To add an element to the address match list, click *Add*. Specify the element to be added and click *OK*.

To delete elements from the list, select the element to be deleted and click *Delete*.

- ♦ To add an address into the list, click *Add*. Specify the IP address and click *OK*.

To delete an address from the list, select the address to be deleted and click *Delete*.


- ♦ **Advanced** This page displays all advanced configuration options. It displays the configured values and the default values for each option. The default value that is displayed is the value that the server assumes if it is not configured.

- ♦ To modify the options, click *Modify* and specify the new value, then click *OK*.
- ♦ To clear the configured values, select the option, then click *Clear*.

The allow-notify and listen-on options are multi-valued. You can also specify a port value, which is optional for listen-on.

- ♦ To add an element to the list, specify the address, then click *Add*. This populates the list with the new entry.
- ♦ To delete elements from the list, select the elements to be deleted, then click *Delete*.
- ♦ Click *Modify* to modify the configured elements.
- ♦ Click *OK* to populate the *Configured Value* column with the elements.

Deleting a DNS Server

- 1 Select the DNS server from the lower pane of the Management Console.
- 2 Click *Delete*  on the toolbar and confirm the deletion.


Starting or Stopping a DNS Server

The DNS server (*novell-named*) must be loaded before you can start or stop the server activity.

The Start/Stop service can be used to load zone data along with the modified configuration without unloading and reloading the DNS server. When you stop the DNS server by using this option, it is still loaded in the memory. However, no services are provided. You can use iManager Management utility or the Java Management Console to update the zone data. When you restart the DNS server by using this option, the server is reconfigured with the new configuration settings and the zone data is also reloaded.


This option can also be used to remotely start and stop the DNS server.

- 1 Select the DNS server from the lower pane of the Management Console.

- 2 Click Start/Stop Service  on the toolbar.
- 3 Depending on the state of the DNS Server module, one of the following operations occurs:
 - ♦ Start action: If the DNS Server module is loaded but is in Stop mode, it is started.
 - ♦ Stop action: If the DNS Server module is loaded and is in Start mode, it is stopped.

Moving a DNS Server

This task enables you to move the DNS Services from one NCP server to another NCP server. You can also convert a DNS server to a cluster-enabled DNS server by moving it to a virtual NCP server.

- 1 Select the DNS server name from the bottom panel of the Management Console.
- 2 Click the *Move DNS Server*  icon on the toolbar.
- 3 In the Move DNS Server dialog box, select the NCP server that the DNS services will be moved to, then click *Move*.

13.2.3 Zone Management


The following sections give details on zone management information.

- ♦ [“Creating a Zone Object” on page 154](#)
- ♦ [“Creating an IN-ADDR.ARPA Object” on page 155](#)
- ♦ [“Viewing or Modifying a Zone Object” on page 155](#)
- ♦ [“Associating a Zone to Specific DNS Servers” on page 158](#)
- ♦ [“Deleting a Zone Object” on page 159](#)
- ♦ [“Importing a Zone Object” on page 159](#)
- ♦ [“Exporting a Zone Object” on page 160](#)

Creating a Zone Object

The DNS Zone object is an eDirectory container object that is made up of Resource Record Set (RRSet) objects and resource records.

To create a zone object:

- 1 Click the *DNS Service* tab of the Management Console.
- 2 Click *Create*  on the toolbar, select *Zone*, then click *OK*.
- 3 Click *Create New Zone* to create a forward zone.
- 4 Use the browse button to select the eDirectory context for the zone.
- 5 Specify a name for the Zone object in the *Zone Domain Name* field.
- 6 Select the zone type.


Novell DNS servers act as primary or secondary depending on the zone type that you select.
- 7 If you select the zone type as secondary, specify the IP address of the master DNS server that will provide zone out transfers for this secondary zone.

Select a DNS server to act as an authoritative DNS server for this zone.
- 8 Click *Create*.

A message is displayed indicating that the new zone has been created. If you have created a primary zone, you are reminded to create the Address record for the host server domain name and corresponding Pointer record in the IN-ADDR.ARPA zone (if you have not already done so).

Creating an IN-ADDR.ARPA Object

After you create a DNS server object, you can use the Management Console to create and set up an IN-ADDR.ARPA Zone object.

- 1 Click the *DNS Service* tab of the Management Console.
- 2 Click *Create*  on the toolbar, select *Zone*, then click *OK*.
The Create Zone dialog box is displayed. The default setting is to create a new primary zone.
- 3 Select *Create IN-ADDR.ARPA*.
- 4 Use the browse button to select the eDirectory context for the zone.
- 5 Specify the network address in the *Network Address* field.
For example, specify 143.72.155 only for 155.72.143.IN-ADDR.ARPA.
After you specify the IP address, it is reversed and prepended to .INADDR. ARPA and reflected in the *Zone Domain Name* field.
- 6 Under the Zone Type, select *Primary* or *Secondary*.
If you select *Secondary*, you must specify the IP address of the DNS Name server that will provide zone out transfers to this zone.
- 7 In the *Assign Authoritative DNS Server* field, select a DNS server.
After you have selected an authoritative DNS server, the *Name Server Host Name* field is filled with the name of the authoritative DNS server.
- 8 Click *Create*.

Viewing or Modifying a Zone Object

To modify an existing Zone object, click the Zone object to be modified in the left pane of the DNS Service window. A Zone object's detailed information window displays the following tab pages:

- ♦ Attributes

This page allows you to configure the zone type and zone servers.

- ♦ To change a primary zone to a secondary zone, click the *secondary zone* box and specify the IP address of the primary DNS server in the *Zone Master IP Address* field.
- ♦ To assign a server to the zone, select the server to which the zone should be assigned from the Available DNS Servers and click *Add*. The server is then displayed in the *Authoritative DNS Servers* field. To delete a DNS server assignment to a zone, select the server to be removed from the *Authoritative DNS Servers* field, then click *Remove*.
- ♦ To configure one of the DNS servers as the designated server for the zone, select the server from the *Designated Primary* field in the case of a primary zone. This server is responsible for DHCP updates for the zone.
For a secondary zone, select the server from the *Designated Secondary* field. This server is responsible for receiving the zone-in transfers.
- ♦ You can specify new comments or modify existing comments for the zone.

- ♦ **Zone Out Filter**

This page allows you configure the zone out filters for the zone.

- ♦ To add an entry into the list, click *Add*.

Specify the *subnet address* and the *subnet mask* for the network, then click *OK*.

- ♦ To delete the elements in the list, select the elements to be deleted, then click *Delete*.

- ♦ **SOA Information**

This page allows you to configure the zone master, e-mail address, serial number, refresh, retry, expire, and minimum TTL values.

- ♦ **Key List:** This page allows you to associate the DNS TSIG keys with the Zone.

NOTE: In earlier versions, key association was a must before updating a policy. Now, it is not required for SAM because the keys are negotiated at run time. Because of this, no checking is done to validate the identity field for SAM-based updates.

Available DNS Keys: Displays a list of DNS TSIG keys that are available in the eDirectory tree. These keys can be associated with the Zone.

Selected DNS Keys: Displays a list of DNS TSIG keys that are associated with the Zone.

- ♦ To add the DNS TSIG key, select the key, then click *Add*.
- ♦ To remove the DNS TSIG key, select the key, then click *Remove*.
- ♦ To add all the keys, click *Add All*.
- ♦ To remove all the keys, click *Remove All*.

NOTE: To add or remove multiple keys, use the Ctrl key to select the keys. Then click *Add* or *Remove*.

- ♦ **Control Lists:** This page displays various lists that can be configured for the zone. You can configure the query filter, also notify, and allow update options.

The query filter and allow update options can be configured as address match lists.

- ♦ To add an element, click *Add*. Specify the element to be added, then click *OK*.
- ♦ To delete elements from the list, select the element to be deleted, then click *Delete*.

The also notify option can be configured as a list of IP addresses.

- ♦ To add an address to the list, click *Add*. Specify the IP address, then click *OK*.
- ♦ To delete an address from the list, select the address to be deleted, then click *Delete*.

The update policy option specifies the policy to update the measure to implement security for a zone object. This is implemented by the default DNS server administering the zone. Addition of TSIG Key at server level and zone level for Secured updates to DNS Zones and servers. The keys are added to the KeyList for DNS Zones and DNS servers by the user for associating with the ACLs. The update policy is a five-token string where each token has a definite function to perform. It can be configured by specifying the following syntax:

```
Permission Identity MatchType TName RR
```

- ♦ To add an update policy, click *Add*. Specify the following values:
 - ♦ *Permission:* Refers to a grant or deny option.

- ♦ *Identity*: Refers to the name of the key used to sign the update. Identity field may have Wildcard characters. Only "*" is the allowed wildcard character. As a valid entry for Identity field, only valid keyCN is allowed, "*", or "*" followed by "." and a character string, matching atleast one of the associated Keys for the DNS zone. Any invalid value entry will throw an error.

- ♦ *MatchType*:

The *MatchType* can be one of the following:

- ♦ *name*: Matches when the domain name being updated is the same as the name in the name field.
- ♦ *subdomain*: Matches when the domain name being updated is a subdomain of the name in the name field (The domain name must still be in the zone.)
- ♦ *wildcard*: Matches when the domain name being updated matches the wildcard expression in the name field.
- ♦ *self*: Matches when the domain name being updated is the same as the name in the identity (not name) field; that is when the domain name being updated is the same as the name of the key used to sign the update. If nametype is self, then the name field is ignored; however you must include the name field when using a nametype of self.
- ♦ *TName*: Specify the *TName*, which is the domain name appropriate to the MatchType specified. For Update Policy entries with the *MatchType* field mentioned as *wildcard*, only wildcard entries are allowed for the *Tname* field. Otherwise character strings are not allowed.
- ♦ (Optional): Specify the *RR* (Resource Record) which can contain any valid record type.

NOTE: Allow Update with keys and Update Policy options are supported for Linux DNS only. Creation of keys with same CN is not allowed in the same Linux tree.

- ♦ **Advanced**

This page displays all advanced configuration options for the zone. It displays the configured values for each option. If any option is not configured at the zone level, the default behavior is server-specific. The value configured for the zone overrides the server value. If no value is configured at the server, the default value specified for the server is used.

The following are the advanced options for the zone:

- ♦ **allow-notify**: Specifies the list of hosts that are allowed to notify the slaves of zone changes in addition to the zone masters. You can configure this option only for a secondary zone.

Allow-notify specified at the server level is overridden by the settings of this zone.

- ♦ **forward**: Specifies the forwarder address. This option can be configured only if the Forwarding list is not empty. A value of first, which is the default, causes the server to query the forwarders first, and if that does not answer the query, the server then looks for the answer in itself. If only is specified, the server queries only the forwarders.
- ♦ **max-size-journal**: Sets a maximum size in bytes for the journal file. This should be configured only for a Linux zone.

NOTE: All changes made to a zone by using dynamic update are written to the zone's journal file. The server periodically flushes the complete contents of the updated zone to its zone file approximately every 15 minutes. When a server is restarted after a shutdown, it replays the journal file to incorporate into the zone any updates that took place after the last zone file update. The dynamic reconfig interval settings are immaterial for a max-journal-size event triggering.

- ♦ **notify:** Specifies if the notification of any zone data changes must be sent to a slave server. You can select from the following options:
 - ♦ Yes: Notification is sent to all the name servers of the zone when the zone data changes.
 - ♦ Explicit: Notification is sent explicitly to the servers specified in the also-notify list when the zone data changes.
 - ♦ No: Notification is not sent.

Notify specified at the server level will be overridden by the settings of this zone.

- ♦ **notify-source:** Specifies the local source address. You also have the option to specify the UDP ports that are used to send notify messages. The local source address must appear in the masters list of the slave server or in the allow-notify list. The slave should also be configured to receive notify messages from this address.

Notify-source specified at the server level is overridden by the settings of this zone.

- ♦ **transfer-source:** Specifies the local addresses that are bound to the IPv4 TCP connections used by the zones that are transferred inbound by the server. It also specifies the source IPv4 address and optionally, the UDP port. The UDP port is used to refresh queries and forward any dynamic updates.

If you have not set a value, it defaults to a system-controlled value, usually the address of the interface closest to the remote end.

Transfer-source specified at the server level is overridden by the settings of this zone

- ♦ **zone-statistics:** Specifies the statistical information that is dumped to the statistics-file for all zones in the server. Values can be either Yes or No.

If you set the value to Yes, the server collects statistical data on all zones in the server.

Zone-statistics specified at the server level is overridden by the settings of this zone.

Modifying Advanced Zone Options

- ♦ To modify the option, click *Modify*, specify the value, then click *OK*.
- ♦ To add an element, specify the address, then click *Add*. This populates the new entry into the list.
- ♦ To delete elements from the list, select the elements to be deleted, then click *Delete*. Click *OK* to populate the *Configured Value* column with the elements.
- ♦ To clear the configured values for the options, select the option, then click *Clear*.

Associating a Zone to Specific DNS Servers

A DNS server can be configured to serve only the queries by specifying the role of a zone as secondary or passive secondary.

To associate the existing DNS zone to a specific DNS server and specify the role of the zone by using the Java Management Console:

- ♦ In the Java Management Console, select the zone that you want to configure for a specific DNS server.
- ♦ In the Attributes page of this zone, select the *Authoritative DNS Server* for this zone as the specific DNS server that will serve this zone.
- ♦ Click *Save*.

Deleting a Zone Object

1 Select the Zone object you want to delete.

2 Click *Delete*  on the toolbar.

A warning message is displayed to confirm the zone deletion. You can also delete subzones by selecting the option from the message window.

NOTE: Creation, modification or deletion of a forward zone is not supported.

Importing a Zone Object

Use the Import dialog box to convert BIND-formatted DNS files and transfer them into the eDirectory database.

To import a Zone object:

1 Click the *DNS Service* tab of the Management Console.

2 Click *Import DNS Database*  on the toolbar.

3 Specify the DNS BIND formatted filename in the field provided. You can browse to select filenames from the File Selection dialog box.

4 Click *Next* to select the context where the zone object should be created.

5 Click *Next* to select the server name that manages the zone.

You can select an existing DNS server or an NCP server where the DNS server object will be created. The selected DNS server must have DNS/DHCP services installed on it. If you select this zone type as *primary*, this DNS server acts as a designated primary; or if you select zone type as *secondary*, it acts as a designated secondary.

If you do not want to assign a DNS server for this zone at this point, leave this field blank.


6 Click *Next* to specify this zone type.

If you select the zone type as *primary*, Novell DNS servers act as primary servers for this zone; if you select *secondary*, they act as secondary DNS servers.

7 Click *Next* to view the configuration that you have selected.

8 Click *Import* to start the import operation.


If the import operation encounters any errors while transferring data, the *Details* button is enabled. Click *Details* to view the errors.

If some resource records are not transferred because of incorrect data, you can create them by clicking *Create*  on the toolbar.

9 Click *Finish* to complete the import operation.

Exporting a Zone Object

Use the Export dialog box to copy the eDirectory database to a text file. The text file enables you to save the DNS zone data to BIND master file format files. These files can be imported to other applications, including BIND servers, or they can be imported back into the eDirectory database by using the Management Console.

- 1 Click the *DNS Service* tab of the Management Console.
- 2 In the DNS Service window, select the zone you want to export and click *Export Database*  on the toolbar.
- 3 In the Export - DNS window, specify the name of the destination file or browse to select a filename from the dialog box.
- 4 Click *Export* to export the database into a file.

NOTE: Importing or exporting of forward zone is not supported.

13.2.4 Resource Record Management

- ♦ “Creating Resource Records” on page 160
- ♦ “Viewing or Modifying Resource Records” on page 161
- ♦ “Deleting Resource Records” on page 161


Creating Resource Records

A resource record is a piece of information about a domain name that contains information about a particular piece of data within the domain.

Every domain name in the zone has a corresponding RRset object under that zone container object. An RRset is not created directly. Initially, when a resource record is created and is assigned a unique domain name within a zone, the corresponding RRset is created first; then, the RR is associated with the RRset.

If you select an existing RRset and click *Create* on the toolbar to create a new RR, the Management Console sets the new RR domain name to read-only and assigns the newly created resource record to the selected RRset. Resource records cannot be created in a secondary zone. All changes to the resource record data should be done at the master server; the secondary servers receive the changes through zone transfers.

To create resource records:

- 1 In the DNS Service window, select the zone in which the resource record will be created. If you want to add another resource record to an already existing RRset, select that RRset.
- 2 Click *Create*  on the toolbar.
- 3 In the Create New DNS Object window, select the resource record, then click *OK*.
- 4 Provide information in the fields:
 - If you have selected an RRset, the owner name field is filled with the RRset name. This field does not need to be edited.
 - If you have selected a zone and want to create a new RRset, specify the domain name of that resource record in the owner name field.

The zone name part of the domain name already filled. Only the remaining portion needs to be filled.

If you are creating a resource record to zone domain name, the owner name field does not need to be filled because the zone domain name is already present.

- 5 In the Create Resource Record window, select the RR type to be created.
- 6 Specify the required data for the selected resource record, then click *Create*.

NOTE: Start of Authority (SOA) is defined as part of a Zone object attribute. A Pointer (PTR) record is created automatically when any new A resource record is created and if a primary INADDR.ARPA zone exists to which the IP address belongs. Similarly, an A type resource is created when any new PTR record is created and if a primary zone exists to which the domain name pointed by PTR record belongs.

Several resource record types correspond with a variety of data stored in the domain namespace. For a list and description of resource record types, see [Section A.2, “Types of Resource Records,” on page 196](#).

Viewing or Modifying Resource Records

When you select an existing resource record in the left pane of the DNS Service window, the detailed information for the object is displayed in the right pane. You can modify the resource record data and save changes by clicking *Save* on the toolbar.

You can modify resource record data and the associated comments for all resource records except the AAA, A6, SRV, LOC, and HINFO records.

Deleting Resource Records

You can delete one, more than one, or all resource records and RRsets, using the multi-select deletion feature in the Management Console. RRsets and resource records in a secondary zone cannot be deleted. They should be deleted from a primary server.

- 1 Click the *DNS Service* tab of the Management Console.
- 2 From *All Zones*, select the domain that contains the host or RRSet.
- 3 Select the item to be deleted.

You can delete either the entire RRSet or one or more resource records in the RRSet.

To delete one or more objects:

- ♦ Press the Shift key and select the objects.
- ♦ Click *Delete*.

NOTE: When the A and PTR type resource records are deleted, the corresponding PTR and A resource records also deleted.

13.2.5 DNS Key Management

A DNS server supports secure updates and secure queries by using the TSIG-key mechanism. The DNS Key Management role consists of tasks that allow you to create, modify, and delete DNS Key objects.

A DNS key provides a means of authentication for dynamic DNS updates and for queries to a secured DNS server. A DNS key uses shared secret keys as a cryptographically secure means of authenticating a DNS update/query.

NOTE: The DNS key option is supported for Linux DNS only. DNS keys can now be created with '.' and '_' in their names.

Unsupported dnssec-keygen features

- ♦ **-a:** RSA, RSAMD5, DH, DSA, RSASHA1 are not supported by novell-named.
- ♦ **-n:** ZONE nametype.
- ♦ **-f:** setting the flag in DNSKEY record.
- ♦ **-p:** protocol support is not affirmed as it is used in conjunction with DNSKEY for DNSSEC.

Example:

```
dnssec-keygen -v
```

Usage:

```
dnssec-keygen -a -HMAC-MD5 -b 218 -n HOST mykey
```

Version: 9.3.4

Required options:

```
-a algorithm: RSA | RSAMD5 | DH | DSA | RSASHA1 | HMAC-MD5
```

```
-b key size, in bits:
```

```
RSAMD5: [512..4096]
```

```
RSASHA1: [512..4096]
```

```
DH: [128..4096]
```

```
DSA: [512..1024] and divisible by 64
```


```
HMAC-MD5: [1..512]
```

```
-n nametype: ZONE | HOST | ENTITY | USER | OTHER
```

```
name: owner of the key
```

The following sections give details on DNS key management:

Creating a DNS Key

- 1 Click Create  on the toolbar.
- 2 In the *Create New DNS Record* window, select the DNS Key, then click *OK*.
- 3 In the *Create DNS Key* window, specify a name to identify the DNS key in the *DNS key Name* field.
- 4 Specify the Algorithm used to hash the DNS data. The HMAC-MD5 algorithm is the only supported algorithm for the DNS key.
- 5 Specify the Secret Key generated by the dnssec-keygen. This is used by the DNS server to encrypt/decrypt the hashed data. Secret-456errt4545= is the secret key generated by dnssec-keygen.

The secret key provided must be Base64 encoded, or the DNS server fails to start.

- 6 Specify or browse to select the *NDS context*.
- 7 Click *Create*. The DNS key is now created.

Example: DNS KeyName-Key1,Alorithm-HMAC-MD5,Key Secret-456errt4545=

Modifying a DNS Key

When you select an existing DNS key in the left pane of the DNS Service window, the detailed information for the object is displayed in the right pane. You can modify the DNS key data and save changes by clicking *Save* on the toolbar.


You can modify DNS key data such as secret key, and the associated Comments.

Deleting a DNS Key


You can delete one, more than one, or all DNS keys, using the multi-select deletion feature in the Management Console.

NOTE: Deleting DNS key objects, deletes the references to key objects (if any) in Zone and DNS server objects.

To delete one key:

- 1 Click the *DNS Service* tab of the Management Console.
- 2 Select the DNS key to be deleted.
- 3 Click  on the toolbar.
- 4 Click *Yes* to confirm the deletion in the *Delete Record* window.

To delete more than one DNS key:

- 1 Click the *DNS Service* tab of the Management Console.
- 2 Select the DNS key to be deleted.
- 3 Press the Shift key and select the Keys.
- 4 Click  on toolbar. Click *Yes* to confirm the deletion in the *Delete Record* window.

NOTE: For further details, please refer to the *dnssec-keygen* man page.

13.3 Configuring Roles for a Novell DNS Server

- ♦ [Section 13.3.1, “Configuring a DNS Server to Forward Queries to Root Name Servers,” on page 164](#)
- ♦ [Section 13.3.2, “Configuring a DNS Server as a Cache-Only Server,” on page 164](#)
- ♦ [Section 13.3.3, “Configuring Child \(Sub\) Zone Support,” on page 164](#)
- ♦ [Section 13.3.4, “Configuring a Multi-Homed Server,” on page 165](#)
- ♦ [Section 13.3.5, “Configuring Dynamic DNS,” on page 165](#)

Novell DNS servers act in the following roles for a zone:

- ♦ **Designated Primary server/Passive Primary server:** The role played by the server for a zone depends on the zone type. If the zone type is primary, the server acts as a designated primary server or a passive primary server. All servers that are managing a primary zone act as primary servers for that zone, and among all the primary servers, one server can be assigned as a designated primary server for that zone. All other servers are called passive primary

servers. The designated primary server accepts dynamic updates for that zone. All primary servers respond to queries for this zone and notify slave servers of this zone about changes in data that can occur due to dynamic update or changes by users.

- ♦ **Designated Secondary server/Passive Secondary server:** If the zone type is secondary, the server acts either as a designated secondary or a passive secondary. All servers that are managing a secondary zone act as secondary (or slave) servers for the zone, and among all of the secondary servers, one server can be assigned as a designated secondary server. All other secondary servers are called passive secondary servers. The designated secondary server is the one that does zone-in transfer for the zone from the master server and writes the data into eDirectory.

Designated secondary server/Passive secondary server

For details on setting zone types, see [“Viewing or Modifying a Zone Object” on page 141](#).

13.3.1 Configuring a DNS Server to Forward Queries to Root Name Servers

When you install OES 2 Linux, the root server information is automatically loaded into your system. No additional steps are required to configure your system to forward queries to the root name servers.

13.3.2 Configuring a DNS Server as a Cache-Only Server

A cache-only name server is a domain name system (DNS) server that is not authoritative for any particular domain. Its only function is to look up names for clients and cache them.

A cache-only server should be located between the clients that require address resolution and any DNS name servers that communicate over the Internet. Configure DNS clients to forward their queries to the cache-only server, and configure the cache-only server to forward its queries to a DNS server (or servers) attached directly to the Internet.

To configure a server to function as a cache-only server, follow the instructions to create a DNS server in [“Creating a Zone Object” on page 154](#). After you create the DNS server object, do not assign it to any zone. Configure this server to forward its queries to a DNS server connected to the Internet. You can do this by specifying the DNS server IP address in the Forwarders option.

13.3.3 Configuring Child (Sub) Zone Support

If you create a child zone, you must configure the glue records to associate the child zones with the parent zone.

The parent zone should contain an NS record for the child zone domain name. An NS resource record specifies a domain name for an authoritative name server for the specified class and domain. If the child zone name server domain name belongs to the parent zone or the child zone, the parent zone should have an A record for that name server domain name. For details on Resource Records, see [Section A.2, “Types of Resource Records,” on page 196](#)

When configured as described above, queries to the parent zone name server for names within the child zone are returned with the child zone’s referral records. The requester can then query the child zone’s name server directly.

13.3.4 Configuring a Multi-Homed Server

A multi-homed server is a server with more than one IP address. In an Internet environment, a multi-homed server is a single server connected to multiple data links, which might be on different networks.

If you have a DNS server with more than one IP address, and if you have specified one of the IP addresses in the listen-on option of the server, make sure the same IP address is used in the A record for the DNS server domain name.

13.3.5 Configuring Dynamic DNS

Novell Dynamic DNS (DDNS), is a mechanism by which NetWare DHCP servers update Novell DNS servers with address and pointer records for addresses and hostnames that are assigned using the DDNS feature. To use DDNS, the following configuration must already exist:

- ♦ The DNS Zone object to receive DHCP updates must be created. For all networks that are served by the DNS server, the DNS zones must have reverse zones configured. For more information on configuring the reverse zones by using iManager, see [“Creating an IN-ADDR.ARPA Object” on page 155](#) or see [“Creating a Primary IN-ADDR.ARPA Zone” on page 139](#) to configure zones by using Java Management Console.
- ♦ Subnet Address Range objects that use the DDNS must be set to range type Dynamic BOOTP and DHCP or Dynamic DHCP.

To activate the DDNS feature:

- 1 Select the Subnet object of the Subnet Address Range on which you want to activate DDNS, then specify a zone in the DNS Zone for Dynamic Update.
- 2 Select the desired Subnet Address Range and ensure that the range type is set to *Dynamic BOOTP and DHCP* or *Dynamic DHCP*.
- 3 Set the DNS update option to *Always Update*.
- 4 Click *Save*.

13.4 novell-named Command Line Options

All command line options for DNS server are optional.

If the DNS server is loaded without any options, default values for all of the options, wherever applicable, are used.

To start a DNS server, enter the following command at the server console prompt:

```
rcnovell-named start
```

Command line options can be specified in two different scenarios:

- ♦ Load: This is the first time the DNS server is loaded.
- ♦ Suspend state: The server can be suspended using the management utilities. The server remains loaded, but it supports only a limited set of services. It does not support any updates and dynamic reconfiguration.

Table 13-2 *Command Line Options*

| Command Line Options | Syntax | Default Value | Load Support |
|-------------------------|----------------------|---------------|--------------|
| Usage Display | ? | NA | Yes |
| Cluster Enabling | -V mountpoint | | Yes |
| Debug | -d <i>level</i> | | Yes |
| DNS Port | -p <i>portnumber</i> | 53 | Yes |
| Dynamic Reconfiguration | -r on off | On | Yes |
| Fault Tolerance | -F on off | On | Yes |
| Replace Characters | -R <i>character</i> | NA | Yes |
| Number of Log versions | -L | 1 | Yes |

Command line options can be specified only at load time. These options control the behavior that can be set only once for a particular running session of the DNS server. If you specify an invalid value for such an option at load time, the server exits.

Syntax: novell-named - u user [?] [-V mountpoint] [-d level] [-p portnumber] [-r on|off] [-F on|off] [-R character] [-L number_of_log_versions]

13.4.1 Description of Command Line Options

Usage Syntax: -?

Cluster Enabling Syntax: -V *mount_point*

This option enables clustering, by providing a volume other than /opt/novell/named/bin/. The volume name specified as the argument should exist and be mounted on the Linux server.

Example: Load novell-named -v *new_volume*

If new volume exists and is mounted on the Linux server, the DNS server stores all files to this volume.

Debugging Level Syntax: -d *level*

This option sets the level of information to be logged. If -d is specified, all the debug messages of type information /notice/warning/error/critical are logged for all categories.

Setting the log level to higher number captures all the logging details over the preceding levels.

All the messages are logged in the /var/opt/novell/named/named.run file. In a cluster setup, the messages are logged in the same file but in the directory structure on the mount point specified with -V option

DNS Port Syntax: -p *port_number*

The port specified in this option is used by the DNS server to listen for queries. The values for this option can be in the range 1-65535. The default port number is 53.

Dynamic Reconfiguration Syntax: `-r on|off`

If dynamic reconfiguration is enabled, the DNS server periodically checks the configuration data for the server and zones. As part of this activity, it automatically detects added, deleted, and modified zones. This option has no effect on periodically checking the directory for changes in the zone data. Even if the dynamic reconfiguration is set to off, periodic detection of zone data occurs. The default period for dynamic reconfiguration is 15 minutes.

Fault Tolerance Syntax: `-F on|off`

When this option is set to on, the DNS server can start using the backup files if eDirectory is inaccessible. If off is specified for a new load, the DNS server does not service the zones for which eDirectory is not available.

Replace Characters Syntax: `-R characters`

A set of characters that are not allowed in the hostnames. The current list is `~!@#%&^&*+=?`";'<>\()[]{}|`. This option can be used to add characters to this list. If these characters are found in the hostnames, the DNS server replaces these characters with a dash (-) before storing them in eDirectory. This option is included for backward compatibility and only allows adding one more character to the existing list.

13.5 Starting the DNS Server

- 1 Ensure that the DNS Server object is created.

For details on creating a server object by using iManager, see [“Creating a Server” on page 134](#), or for details on creating a server object using Java Management Console, see [“Creating a DNS Server Object” on page 151](#).

- 2 Before starting `novell-named`, check to see if the following daemons are active:

```
/etc/init.d/ndsd
/etc/init.d/novell-xregd
/etc/init.d/micasad
```

- 3 Load `novell-named` by using the following script or command:

```
rcnovell-named start
or
/etc/init.d/novell-named start
```

- 4 To check the status of the server, use the following command/script:

```
rcnovell-named status
or
/etc/init.d/novell-named status
```

13.6 Running DNS Server in chroot Mode

To load `novell-named` with `-t` (chroot) option, make sure that the following directories are created under the chroot directory with permissions to user specified with `-u` option:

- ♦ The configuration file directory - `/etc/opt/novell/named`
- ♦ The log file directory - `/var/opt/novell/log/named`

- ♦ The pid directory - `/var/opt/novell/run/named`
- ♦ `/etc/rndc.key`

In addition to this, edit the load command with the chroot path in the `/etc/init.d/novell-named` script.

13.7 Running DNS Server as a Non-Root User

The named user is created during installation of BIND. When you load `novell-named` with `rcnovell-named start` or `/etc/init.d/novell-named start`, DNS server reuses the named user to run the DNS service daemon.

If you want to run the DNS service daemon with any other non-root user:

- 1 In YaST, create a user using the *Security and Users > User Management* option in YaST.
- 2 Set Runtime credentials for the user created in [Step 1, Section 11.3, “Setting Runtime Credentials,” on page 129](#)
- 3 To load `novell-named` with `-t` (chroot) option, make sure that the following directories are created under the chroot directory with permissions to user specified with `-u` option:
 - ♦ The configuration file directory `/etc/opt/novell/named`
 - ♦ The log file directory `/var/opt/novell/log/named`
 - ♦ The pid directory `/var/opt/novell/run/named`
 - ♦ `/etc/rndc.key`

NOTE: In cluster setup, Runtime Credentials must be set on all the nodes in the DNS cluster.

- 4 Edit `/etc/init.d/novell-named` and modify the `checkAndCopyConfigFiles` function and make relevant changes for user name and group.
- 5 Start the DNS server using `rcnovell-named start` or `/etc/init.d/novell-named start` command.

13.8 Stopping the DNS Server

To stop the DNS server, use the following command:

```
rcnovell-named stop
```

You can also use the methods in [“Loading or Unloading a DNS Server” on page 137](#).

13.9 What's Next

The next section describes steps to cluster a DNS server.

- ♦ [Chapter 14, “Configuring DNS with Novell Cluster Services,” on page 169](#)

Configuring DNS with Novell Cluster Services

14

- ♦ [Section 14.1, “Prerequisites,” on page 169](#)
- ♦ [Section 14.2, “Installation and Configuration,” on page 169](#)
- ♦ [Section 14.3, “Loading and Unloading the DNS Server,” on page 173](#)
- ♦ [Section 14.4, “What’s Next,” on page 174](#)

Novell® OES Linux DNS server is a distributed database system that provides hostname-to-IP resource mapping (usually the IP address) and other information for computers on a network. If for some reason the Novell OES 2 Linux DNS server is not accessible, clients lose their ability to connect to the network because they cannot obtain an IP address. This is possible because the DNS server is automatically started, stopped, and restarted on different servers in the cluster by Novell Cluster Services™. You can prevent this problem by configuring DNS with Novell Cluster Services, which ensures that the IP address range required by users to connect to the network is highly available.

Before you attempt to implement this solution, familiarize yourself with how Cluster Services works. For information, see the *OES 2 SP1: Novell Cluster Services 1.8.5 for Linux Administration Guide*.

14.1 Prerequisites

- ♦ eDirectory™
- ♦ Novell Cluster Services
- ♦ Novell Storage Services (NSS)
- ♦ The DNS server should be installed on all the nodes in cluster or on the nodes identified for running DNS
- ♦ Create a NSS Shared Disk partition and cluster-enable it. For more information, refer to “Configuring Cluster Resources for Shared NSS Pools and Volumes” in the *OES 2 SP1: Novell Cluster Services 1.8.5 for Linux Administration Guide*.

14.2 Installation and Configuration

- ♦ [Section 14.2.1, “Verifying the Novell Cluster Services Setup,” on page 169](#)
- ♦ [Section 14.2.2, “Installing and Configuring a Cluster,” on page 170](#)
- ♦ [Section 14.2.3, “DNS Load and Unload Scripts,” on page 172](#)

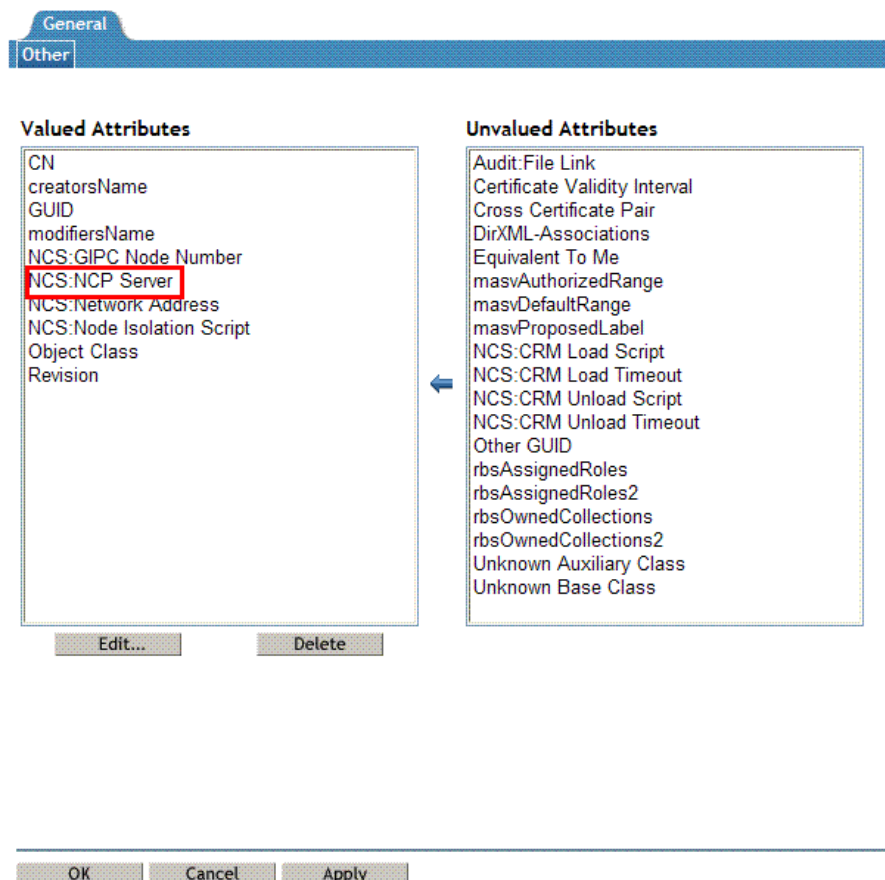
14.2.1 Verifying the Novell Cluster Services Setup

To ensure that Novell Cluster Services is set up properly:

- 1 In iManager, click the *View Objects* button.

This view contains browsing and searching functionality to find objects, including a Tree View feature similar to that used in ConsoleOne®.

- 2 Select the cluster node for which you want to set the server attribute, then click *Edit*. This opens the Modify Object window.
- 3 Ensure that the NCS: NCP Server attribute is included in the *Valued Attribute* list. If the NCS:NCP Server attribute is in the *Unvalued Attribute* list, use the ← button to move the attribute to the *Valued Attribute* list as illustrated in the following figure.



- 4 Select the NCS:NCP Server attribute from the *Valued Attributes* list. Click *Edit*.
- 5 The Edit Attribute window is displayed. Ensure that the value of the NCS:NCP attribute is set as the name of the Physical NCP Server.
- 6 Click *Apply* to save and exit.

After executing these steps, you can mount the volume by using the Novell Client™.

14.2.2 Installing and Configuring a Cluster

- 1 A DNS server by default uses the `named` user that is created in the local system during the installation process. If you want to use another user, create the user using the *Security and Users > User Management* option in YaST.

- 2 In iManager, click the *Users > Create User* task to open the Create User window. Specify the details and click *OK* to create a new user.
 - 3 The user created in **Step 2** needs to be LUM-enabled. To do this, click the *Linux User Management > Enable Users for Linux* task. This opens the Enable Users for Linux window. Search for and select the user created in **Step 2**, then click *OK* to select the user.
 - 3a Every user must belong to a primary group. To add a user to a group, search for an *Existing eDirectory Group object*.
 - 3b Select the DNSDHCP Group object from the list.
 - 3c Select the workstations to which the Linux-enabled user should have access.
 - 3d Click *Next* to confirm the selection.
The user is now Linux-enabled, included in the DNSDHCP-GROUP, and granted access to cluster nodes.
 - 3e Update the UID to Linux profile that the user created. The UID for a default named user is 44.
 - 4 Mount the shared volume on one of the nodes in the cluster.
 - 5 Execute the following command at the command prompt:

```
/opt/novell/named/bin/ncs_dir.sh MountPath [Username]
```

 It creates the following directory:

```
<mountpath>/etc/opt/novell/named
```

 The MountPath parameter indicates the target directory in the volume where named specific directories are created.
 For example, `/opt/novell/named/bin/ncs_dir /media/nss/DNSVOL/cn=named.o=novell.T=MyTree`
 When the script is executed, it creates the following directories:
 - ♦ `/media/nss/DNSVOL/etc/opt/novell/named`
 The script also assigns rights and ownership to these directories to the named user.

```
rights -f /media/nss/<Volumename>/etc/opt/novell/named -r rwfcem
```

```
trustee username.context.treename
```

 In cluster environment, the login directs to `/var/opt/novell/log/named/named.run` on the local volume where DNS server is running.
-
- NOTE:** By default the `ncs_dir.sh` script assigns permissions to the `named` user. If you want to use another user instead of `named`, modify the `ncs_dir.sh` script. Follow **Step 2** and **Step 3** for `named` user.
-
- 6 Create the DNS server on a virtual NCP server.
 - 7 Run the DNS Server by using the following command:
 - ♦ `/opt/novell/named/bin/novell-named -u <username> -d <debug log level> -V <shared volume>`
 This step ensures that DNS server can work on a cluster that is set up with shared volumes.
 - 8 Click the *Cluster > Cluster Options* task. The Cluster objects are displayed.

9 Select the Cluster resource and click *Details*. The Cluster Pool Properties are displayed. Click the *Scripts* tab. You can now view or edit the Load or Unload scripts.

9a Click *Load Script*.

9b Ensure that the DNS load script is same as the script specified in “DNS Load Script” on page 172.

9c Click *Unload Script*.

9d Ensure that the DNS unload script is same as the script specified in “DNS Unload Script” on page 172 and paste it in the Unload Script editor just before the `exit 0` statement

9e Click *OK* to save the changes.

10 Set the DNS resource online by using the *Clusters > Cluster Manager* task in iManager

11 Restart cluster services on all the nodes for changes to take effect:

```
rcnovell-ncs stop
rcnovell-ncs start
```

14.2.3 DNS Load and Unload Scripts

- ♦ “DNS Load Script” on page 172
- ♦ “DNS Unload Script” on page 172

DNS Load Script

The load script contains commands to start the DNS service. The load script appears similar to the following example:

```
#!/bin/bash
. /opt/novell/ncs/lib/ncsfuns
exit_on_error nss /poolact=DNSPOOL
exit_on_error ncpcon mount DNSVOL=254
exit_on_error add_secondary_ipaddress IP address of the secondary
server
exit_on_error ncpcon bind --ncpservname=Virtual NCP server name
--ipaddress=secondary server IP address
exit_on_error /opt/novell/named/bin/novell-named -u named -V Volume Name
exit 0
```

Configuring DNS Load Script

To customize the DNS load script for your specific configuration:

1 Edit the following line to assign a volume name:

```
exit_on_error /opt/novell/named/bin/novell-named -u named -V
volume name
```

Replace *volume name* with the name of the volume and *secondary server IP address* with the IP address of the secondary server.

DNS Unload Script

The unload script contains commands to stop the DNS service. The unload script appears similar to the following example:

```
#!/bin/bash
. /opt/novell/ncs/lib/ncsfuns
killproc -p /var/opt/novell/run/named/named.pid -TERM
/opt/novell/named/bin/novell-named
ignore_error ncpcon unbind --ncpservname=NCP server name
--ipaddress=IP address of the secondary server
ignore_error del_secondary_ipaddress IP address of the secondary server
ignore_error nss /pooldeact=DNSPOOL
exit 0
```

Configuring DNS Unload Script

To configure DNS unload script, add the following command:

```
killproc -p /var/opt/novell/run/named/named.pid - TERM
/opt/novell/named/bin/novell-named
```

14.3 Loading and Unloading the DNS Server

After updating the load and unload scripts of the virtual NCP Server, the DNS server is now loaded and unloaded along with the virtual NCP Server.

- [Section 14.3.1, “Loading the DNS Server,” on page 173](#)
- [Section 14.3.2, “Unloading the DNS Server,” on page 173](#)

14.3.1 Loading the DNS Server

You can load the virtual NCP server and DNS server by using iManager.

- 1 Click *Cluster > Cluster Manager* task in iManager. This opens up the Cluster Manager window.
- 2 Click the *Search* icon to open the eDirectory Object Selector. Browse or search the list to locate and select the cluster object. The Cluster state is displayed.
- 3 Select the check box next to the DNS Cluster and select *Online*. Select the cluster node where you want the resource to load, then click *OK* to load the resource on that node.

After a few seconds you see the server status as *Online*, indicating that the server has been loaded.

After the Virtual NCP server is successfully loaded on a node, use the `rcnovell-named status` command to check the status of the server.

14.3.2 Unloading the DNS Server

- 1 Click the *Cluster > Cluster Manager* task in iManager. This opens up the Cluster Manager window.
- 2 Click the *Search* icon to open the eDirectory Object Selector. Browse or search the list to locate the cluster object. The Cluster state is displayed.
- 3 Select the check box next to the DNS Cluster and select *Offline*. After a few seconds you see the server status as *Offline*, indicating that the server has been unloaded.

14.4 What's Next

The next section discusses the security issues and recommendations for DNS on a Novell Open Enterprise Server 2 Linux server.

- ♦ [Chapter 15, “Security Considerations for DNS,” on page 175](#)

This section describes security issues and recommendations for DNS on a Novell® Open Enterprise Server 2 Linux server. It is intended for security administrators or anyone who is using DNS for Linux and is responsible for the security of the system. It requires a basic understanding of the DNS protocol. It also requires the organizational authorization and the administrative rights to effect the configuration recommendations.

- ♦ [Section 15.1, “Logging,” on page 175](#)
- ♦ [Section 15.2, “Commands,” on page 175](#)
- ♦ [Section 15.3, “Cryptographic Algorithms,” on page 175](#)
- ♦ [Section 15.4, “Best Practices,” on page 175](#)
- ♦ [Section 15.5, “What’s Next,” on page 176](#)

15.1 Logging

`novell-named` supports different log levels. For more information on the debug levels, see [Section 13.4, “novell-named Command Line Options,” on page 165](#).

For more detailed logging information, see *BIND 9 Administrator Reference Manual* (<http://www.isc.org/index.pl?sw/bind/arm93/>).

15.2 Commands

The DNS server can be administered by using command line options. For details, see [Chapter 13, “Administering and Managing a DNS Server,” on page 133](#).

15.3 Cryptographic Algorithms

AES from Open SSL implementation is used for encryption and decryption of proxy user credentials.

15.4 Best Practices

- ♦ It is preferable to run the DNS server installation process with the YaST tool. For details on the DNS Server installation process, see [Chapter 11, “Installing and Configuring DNS,” on page 125](#)
- ♦ During the DNS server installation ensure that the *Use secure channel for configuration* option is selected. This ensures that the authentication mechanism is secured.
- ♦ To load `novell-named` with the `-t` (chroot) option, make sure that the following directories are created under the chroot directory with permissions to the user specified with the `-u` option:
 - ♦ The configuration file directory – `/etc/opt/novell/named`
 - ♦ The log file directory – `/var/opt/novell/log/named`
 - ♦ The pid directory – `/var/opt/novell/run/named`

- ♦ By default `novell-named` is loaded by using the existing non-root user, which is `named`. You should load `novell-named` with the `-u <non-root user>` option.
- ♦ It is recommended to load `named` with a log level specific for your needs. For more details, see [Section 13.4, “novell-named Command Line Options,” on page 165](#)
- ♦ You should configure Apparmor* profiles for `novell-named` according to your needs. The default profile is stored at `/etc/apparmor.d/opt.novell.named.bin.novell-named` and includes only minimal configuration.

After making changes to the profile, reload Apparmor with the `rcapparmor` command.

- ♦ Zone security factors: To secure DNS, BIND provides different options. This includes IP-based access control and secure queries using Keys (recommended). The `allow-query` option is used to restrict queries to a particular set of hosts or keys.
 - ♦ For non-authoritative zones (zones not served by the server, so the responses are cached), restrict the query access at the server level (using `allow-query`) to your own network.
 - ♦ For authoritative zones (zones served by the server), access can be restricted either to your local network or to any other network.

NOTE: Restrict DNS zone transfers to only the servers that absolutely need it.

15.5 What's Next

The next section discusses potential issues and workarounds for Novell DNS and DHCP services on OES 2 Linux.

- ♦ [Chapter 16, “DNS/DHCP Advanced Features,” on page 177](#)

This section describes how to configure the Import-Convert-Export (ICE) zone handler.

- ♦ [Section 16.1, “Configuring the ICE Zone Handler,” on page 177](#)
- ♦ [Section 16.2, “What’s Next,” on page 181](#)

16.1 Configuring the ICE Zone Handler

- ♦ [Section 16.1.1, “Modifying the ice.conf File,” on page 177](#)
- ♦ [Section 16.1.2, “Enabling Clear-Text Passwords,” on page 177](#)
- ♦ [Section 16.1.3, “Importing Configuration and Script Files,” on page 178](#)
- ♦ [Section 16.1.4, “Exporting Configuration and Script Information,” on page 179](#)

16.1.1 Modifying the ice.conf File

The source and destination handlers available to the application, with other information such as the version of the handlers and the modes in which they operate, must be provided in the `ice.conf` file in the `/etc/opt/novell/eDirectory/conf/` directory. You modify the `ice.conf` file by appending the zone handler information.

```
[Zone]
Version: 1.0
Mode: FromFile, FromServer, ToFile
Module name: zone
Flags: 1
```

The mode is used to convey the information about the functionality supported by the handler. In the example above, the mode is `FromFile, FromServer, ToFile` because the zone handler can read from the file, read from the server, and write to the file.

The LDAP handler is used to write to the directory. Ensure that `ice.conf` also contains the following:

```
[LDAP]
Version: 1.0
Mode: FromServer, ToServer
Module Name: ldaphdlr
Flags: 1
```

The module name specifies the handler name. Flags specifies the flags that should be sent to the destination handler. Currently, the only flag available is for `LBURP`.

16.1.2 Enabling Clear-Text Passwords

Clear-text passwords should be enabled in the LDAP group object to avoid LDAP bind operation failure. You can do this by using `iManager`.

16.1.3 Importing Configuration and Script Files

You can use the ICE zone handler, (`named.conf` file) along with the corresponding zone master files can be migrated to Novell® eDirectory™, or a script file can be formed in a particular format. This script file is used to migrate the zone master files of the desired zones, without changing the server and zone configuration information.

The import operation generates an output script file that indicates the status of the zone import with a “done:” token at the beginning of zones imported successfully. If an import fails for a particular zone, the corresponding output script file generated not have a “done:” token for that particular zone and the script file can be reused to import the failed zone later.

Command Line Parameters for ICE Zone Import

```
ice -S ZONE -f <input file> [-t scr | conf] -x < zone context> -b  
<DNS server DN> [-l <log file name>] [-r] [-s <LDAP server name>]  
[-p <port no>] [-d <bind dn>] [-w <password>] -D {Destination  
Handler with options }
```

| Options | Descriptions |
|-----------------------|--|
| -f <input file> | The absolute name of the input file. The input file can be either a configuration file (typically <code>named.conf</code>) or a script file. The type of the file passed is specified with the <code>-t</code> option. |
| -t {scr conf} | The type of the file passed with the <code>-f</code> option. <code>scr</code> is used to indicate that a script file is being passed and <code>conf</code> is used to indicate that a configuration file is being passed. <code>scr</code> is the default option used when <code>-t</code> is not specified. |
| -l <log file name> | The name of the log file where the messages are logged. By default, the <code>/etc/opt/novell/named/zoneimp.log</code> file is created. If any error is encountered, the important messages are printed on the ICE screen. |
| -x <zone context> | The context under which the zone objects are created. |
| -b <DNS Server DN> | The distinguished name of the DNS server in Novell eDirectory. The imported zones are associated with this DNS server. This is required to link the imported zone objects to the DNS server and vice versa. |
| -r | The zone object, if already present, should be replaced. If this option is not specified, the existing zone objects are not disturbed. |
| -s <LDAP server name> | The LDAP server name or IP address to which the zone and configuration information are imported. The default is the local machine (127.0.0.1/"local host"). NOTE: The server name specified here should be the same as the name specified in the destination LDAP handler options (<code>-s</code> option). |
| -p <port no> | The port number where the LDAP server is listening. The default value is 389. NOTE: The port number specified here should be the same as the port specified in the destination LDAP handler options (<code>-p</code> option). |

| Options | Descriptions |
|---------------|--|
| -d <bind dn> | The distinguished name with which you want to bind to the LDAP server. NOTE: The fully distinguished name specified here should be the same as the name specified in the destination LDAP handler options (-d option). |
| -w <password> | The password for the Bind DN. NOTE: The password specified here should be the same as the password specified in the destination LDAP handler options (-w option). If you do not specify the password for bind DN, only those LDAP operations that do not need authentication will pass and the rest will fail. |

Example for Command Line Options: `ice -S ZONE -f /home/user/db/named.conf -t conf -s 164.10.1.1 -x o=novell -b cn=DNS_MYSERVER,o=novell -d cn=admin,o=novell -w mypassword -D LDAP -s 164.10.1.1 -d cn=admin,o=novell -w mypassword`

Script File Format: A typical line from a script file contains the following fields.

```
<type of zone> <zone name> [master server IP] <master file name> [zone context]
[comments] /* end of line */
```

Type of Zone: Primary or Secondary.

Zone Name: The domain name for which the resource records are to be imported.

Master Server IP: The IP address of the master server, if the zone is a secondary zone.

Master File Name: The file that contains the resource records.

Zone Context: The context where the zone object should be created.

Comments: Any ASCII pattern, the first character being a semicolon (;)

```
For example, primary novell.com /home/user/db/novell.com.db; primary
zone secondary novell.com 164.1.1.1 /home/user/db/novell.com.db;
```

Named.conf File Format: The handler supports BIND 9.2 `named.conf` format only. It interoperates with Novell extended attributes in the `named.conf` file. That is, it ignores those attributes during import. The existing BIND4 and BIND8 `conf` files must be converted to BIND9 format before passing them to this utility.

16.1.4 Exporting Configuration and Script Information

You can use the ICE zone handler, to export the DNS server, zone configuration information, and data from eDirectory and write it to the files.

- ♦ [“Command Line Parameters for ICE Zone Export” on page 180](#)

Command Line Parameters for ICE Zone Export

- ♦ “Source Handler Options” on page 180
- ♦ “Destination Handler Options:” on page 181

Source Handler Options

```
ice -S ZONE -s<source server> [-p<source LDAP port>] [-d<user name  
in source server>] [-w<password for source server>] <[-b <DNS  
Server DN>] [-x <Zone context>]> [-F <LDAP filter>] -D {Destination  
Handler with options}
```

| Options | Descriptions |
|--------------------|--|
| -s <server name> | <p>Specify the LDAP server name or IP address to which the zone and configuration information. The default is the local machine (127.0.0.1/"local host")</p> <p>The server name specified here should be the same as specified in the destination LDAP handler options (-s option).</p> |
| -p <port no> | <p>Specify the port number where the server is listening. The default value is 389.</p> <p>The port number specified here should be the same as specified in the destination LDAP handler options (-p option).</p> |
| -d <bind dn> | <p>Specify the distinguished name with which you want to bind to the LDAP server.</p> <p>The fully distinguished name specified here should be the same as specified in the destination LDAP handler options (-d option).</p> |
| -w <password> | <p>Specify the password for the Bind DN.</p> <p>The password specified here should be the same as specified in the destination LDAP handler options (-w option). If you do not specify the password for bind DN, only those LDAP operations that do not need authentication will pass and the rest will fail.</p> |
| -b <DNS Server DN> | <p>Specify the FDN of the DNS server object.</p> <p>The handler uses this information to read the configuration information and also to detect zone objects that fall under the administrative domain of this server.</p> <p>If -b option is not specified, the configuration information is not exported and only the zone master files will be formed.</p> |
| -x <Zone Context> | <p>Specify the context, from which the zone objects will be exported.</p> <p>x or b option must be specified. If b option is specified without the x option, all zones belonging to that DNS server will be exported. If both these options are specified, the configuration information is exported from the specified DNS server and the zone data with configuration from the specified zone objects.</p> |

| Options | Descriptions |
|------------------|---|
| -F <LDAP filter> | Specify the LDAP-compliant filter. This acts in conjunction with the <code>-x</code> option described above to specify the zone objects to export. The default value is <code>objectClass=*</code> The <code>-F</code> options works only with the <code>-x</code> option, to export all zones under the given context that match the given filter, and not when both <code>-b</code> and <code>-x</code> are specified. |

Destination Handler Options:

`D ZONE -p <path>`

`<path>` is the path where the output files are created. The files that are created are `named.conf` and the zone master files, with the corresponding names of the zone objects as they are in the eDirectory.

By default, all zone information is created in the current directory if the `-p` option is not specified.

For example, `ice -S ZONE -b cn=DNS_MYSERVER,o=novell -s 164.99.1.1 -p 389 -d cn=admin,o=novell -w mypassword -D ZONE -p /home/user/db/`

16.2 What's Next

The next section provides information on installing DNS with the Domain Services for Windows pattern.

- ♦ [Chapter 17, “DNS-DSfW Integration,” on page 183](#)

Novell® Domain Services for Windows (DSfW), a component of Open Enterprise Server (OES) 2 SP1, creates seamless cross-authentication capabilities between Windows/Active Directory* and Novell OES 2 Linux/eDirectory™ servers. This suite of technologies allows Novell customers with Windows networking environments to set up one or more “virtual” Active Directory domains in an eDirectory tree. Users can then log in and authenticate to both eDirectory and Active Directory from a Windows workstation without requiring multiple logins or having the Novell Client™ for Windows installed.

These technologies also enable the user to access Novell File and Print services without a Novell Client on their windows workstations. Administrators can use Novell iManager or Microsoft* management consoles to manage users and groups in the directory.

Active Directory implementation is based on domain naming standards. Service registration and queries are processed through a DNS server. Microsoft integrates its own directory-based DNS, and DHCP services Active Directory deployments.

In DSfW, bind was used as the DNS server. With DNS-DSfW integration, bind is replaced by novell-bind to be used as the DNS server in OES 2 SP1.

As a part of this integration, the following changes are observed in the installation scenarios:

IMPORTANT: In a Domain Services for Windows (DSfW) environment, if the services persist difficulties, then the novell-named, ntpd, and nscd AppArmor® profiles should be loaded in complain mode.

- ♦ [Section 17.1, “Normal eDirectory with DNS,” on page 183](#)
- ♦ [Section 17.2, “DSfW with DNS,” on page 183](#)
- ♦ [Section 17.3, “DSfW with Remote DNS \(Child Domains\),” on page 185](#)
- ♦ [Section 17.4, “Scenarios,” on page 185](#)
- ♦ [Section 17.5, “FAQs,” on page 186](#)
- ♦ [Section 17.6, “What’s Next,” on page 186](#)

17.1 Normal eDirectory with DNS

For more information on eDirectory with DNS installation, refer [Section 11.2, “Installing the DNS Server,” on page 127](#)

17.2 DSfW with DNS

- ♦ [Section 17.2.1, “Changes for DNS,” on page 184](#)
- ♦ [Section 17.2.2, “Local DNS Server Installation,” on page 184](#)

17.2.1 Changes for DNS

Domain Services for Windows no longer uses bind from the SUSE® Linux Enterprise Server. Novell-bind is installed by default when the DSfW pattern is selected. DSfW configures novell-bind on the forest root domain (FRD) by default. The first DSfW server in the eDirectory tree (forest root domain) has novell-bind configured.

Additional domain controllers have novell-bind configured; However, it is not required for DSfW to function. Existing novell-bind servers in the tree can be configured as a forward lookup server with novell-bind on the DSfW server.

DSfW configuration will configure novell-bind on other Domain Controllers if the *Configure this server to be a primary DNS server* option is selected. DSfW automatically populates the zone information whenever a domain controller or domain is configured. Administrators do not require a different utility.

On Novell Cluster Services™, Novell DNS is bound to a virtual NCP™ server and not to physical nodes. Because of this dependency, DSfW provisioning is not automated for Novell Cluster Services.

For more information on DSfW, see the *OES 2 SP1: Domain Services for Windows Administration Guide*.

17.2.2 Local DNS Server Installation

For all the supported DSfW scenarios with local DNS, an installation screen for DNS is not used. All the inputs are gathered by eDirectory installation page and is passed to DNS.

Local DNS server installation can be determined by the status of the check box in the DSfW installation page. If the box is selected, the installation for the local DNS server proceeds. YaST collates the input and then calls the dns-inst utility to install the DNS server after configuring eDirectory (ndsconfig call) but before calling ndsdcinit to install DsfW. The order of the call is:

- ♦ ndsconfig (to configure eDirectory).
- ♦ extend the DNS schema
- ♦ dns-inst (to configure the DNS server)
- ♦ ndsdcinit (to configure DSfW)

YaST displays the DNS server input page with the following changes:

- ♦ The eDirectory server IP address is dimmed and the field is populated with the local server IP address.
- ♦ The contexts are dimmed for the Locator, Group object, rootserverinfo, and NCP server object. This is similar to the Server context; that is, ou=novell.
For example, In *Domain name in DC format*, if the domain being installed is widget.com, then the server context is ou=novell,dc=widget,dc=com.
- ♦ The DNS Host Name and Domain Name for DNS Server can be edited. However, populate the host name from `/etc/hosts` as the default value and the domain name from `/etc/resolv.conf` or from `/etc/hosts`.
- ♦ **DNS Proxy User:** The DNS screen displays a proxy user or DNS admin.

17.3 DSfW with Remote DNS (Child Domains)

In this scenario, DNS is not installed on the local system and remote DNS is used to serve the child domain.

- ♦ [Section 17.3.1, “DSfW on a Remote DNS Server,” on page 185](#)

17.3.1 DSfW on a Remote DNS Server

YaST determines the remote DNS server configuration only when the *Configure this server as DNS server* option is not selected by the user. In this case, YaST does not call `dns-inst` utility to configure the DNS server on the local machine, but it gets the details of the remote DNS server by displaying a new screen to the user. This new screen prompts for the following objects:

- ♦ Locator object (DNS-DHCP): Fully qualified DN
- ♦ DNS group object (DNSDHCP-GROUP): Fully qualified DN

While calling the utility, YaST needs to pass the following objects to `ndsdcinit`:

- ♦ Locator object FQDN
- ♦ DNS group FQDN
- ♦ Local or remote server configuration (LOCAL_DNS)

17.4 Scenarios

Table 17-1 Local and Remote DNS scenarios

| Case | Local DNS | Remote DNS |
|------------------------------|-----------|------------|
| Non-named mapped FRD | Yes | No |
| Non-named mapped child | Yes | Yes |
| Name-mapped FRD | Yes | Yes |
| Name-mapped child domain | Yes | Yes |
| Additional Domain Controller | Yes | Yes |

The following items provide more information on DSfW configuration with DNS:

- ♦ FRD servers are always configured with DNS.
- ♦ For non-name mapped FRD installation, the locator, group, and rootserverinfo objects are created within the novell container under the domain context. These contexts cannot be edited.
- ♦ For named mapped FRD, the contexts for locator, group, and rootserverinfo are not populated.
 - ♦ If it is the first DNS server in the tree, then by default these objects are created under the novell container within the domain context.
 - ♦ If it is not the first DNS server in the tree, then the context pointing to the existing locator, group, and rootserver must be entered.

Entering different context results in multiple instances of these objects, which leads to management difficulties.

- ♦ For child domains, the DNS server can be either local or one of the parent or grandparent DNS servers. For remote DNS servers, the locator and group object contexts are required as inputs.
- ♦ DNS is not installed for additional domain controllers.
- ♦ For workstations to join the domain, an Update Policy must be enabled on the zone. Allow-update and update policies are mutually exclusive, so allow-update cannot be used on these zones.

Table 17-2 *DNS Deployment Scenarios*

| Deployment Scenarios | Description |
|------------------------------|--|
| Non-name Mapped FRD | The DSfW service is configured for the new tree. The DNS server is configured on the local server. Because this is a new eDirectory tree being configured for DSfW-DNS, it does not require remote DNS server configuration. |
| Non-name Mapped Child | The DNS server is configured locally or configured to point to the remote DNS server (parent DNS server) in the tree. |
| Name Mapped FRD | DSfW is configured for an existing eDirectory tree. The DNS server is either configured locally or is configured to point to the remote DNS server in the tree. |
| Name Mapped Child Domain | The DNS server is configured for the child domain on the local server or it is configured to point to the remote DNS server (parent DNS server) in the tree. |
| Additional Domain Controller | The DNS server is configured locally or is configured to point to the remote DNS server. |

17.5 FAQs

How can I configure DSfW DNS as primary DNS server for any other non-Novell DNS Server?

Novell DNS can work and co-exist with any DNS server such as Windows, Linux, and NetWare®.

Does Novell DNS handle interoperability?

Novell DNS works with all DHCP servers, such as Linux, NetWare, and Windows. With the support for GSS-TSIG-based updates added for OES2 SP1, SLES BIND and Novell BIND can be only implementations supporting secure updates from Microsoft DHCP servers.

17.6 What's Next

The next section provides methods to troubleshoot various DNS and DHCP services.

- ♦ [Chapter 18, “Troubleshooting DNS and DHCP Services,” on page 187](#)

Troubleshooting DNS and DHCP Services

18

This section discusses potential issues and workarounds for Novell® DNS and DHCP services on OES 2 Linux.

- ♦ [Section 18.1, “DHCP,” on page 187](#)
- ♦ [Section 18.2, “DNS,” on page 190](#)
- ♦ [Section 18.3, “What’s Next,” on page 192](#)

18.1 DHCP

- ♦ [Section 18.1.1, “DHCP Server fails to load and records a “Cannot find host LDAP entry DHCP” error in the log file,” on page 187](#)
- ♦ [Section 18.1.2, “Installing an OES Server inside a container with a separate partition on an existing tree that already has DHCP Server installed on it results in a constraint violation error.” on page 188](#)
- ♦ [Section 18.1.3, “The dhcpd.log file is empty,” on page 188](#)
- ♦ [Section 18.1.4, “The DHCP server failed to Start,” on page 188](#)
- ♦ [Section 18.1.5, “The DHCP Server displays “Unknown Error” on the console,” on page 189](#)
- ♦ [Section 18.1.6, “Permission denied to DHCP server,” on page 189](#)
- ♦ [Section 18.1.7, “segfault dhcpd - You get an error “dhcpd: Can't create new lease file: Permission denied” and “dhcpd\[8249\]: segfault at 0000000000000000 rip 00002abbf999db7f rsp 00007fffb18ea5e0 error 4”,” on page 190](#)

18.1.1 DHCP Server fails to load and records a “Cannot find host LDAP entry DHCP” error in the log file

Cause: The DHCP Server and DHCP Service objects do not exist.

Action: Use iManager to create the DHCP Server and DHCP Service object. See [Section 7.1, “Using iManager to Manage DHCP,” on page 81](#)

Cause: The DHCP server and DHCP Service have been created but the association between both of these objects is not set.

Action: Set the DHCP server and DHCP Service association by using the [Viewing or Modifying a Service](#) task.

18.1.2 Installing an OES Server inside a container with a separate partition on an existing tree that already has DHCP Server installed on it results in a constraint violation error.

Cause: The Locator object contains a reference to the DHCP servers installed on the eDirectory tree. Information for the newly installed DHCP Server is added to the Locator object. Because of the delay in synchronizing the replica, the Locator object does not find the newly installed DHCP server object inside its own replica.

Action: To resolve this problem, perform one of the following actions:

- ♦ Delete the replica and reinstall the DHCP server.
- ♦ While configuring the newly installed DHCP server, provide the IP address of the OES machine where eDirectory is installed.

18.1.3 The dhcpd.log file is empty

Cause: The `dhcpd.log` file was probably deleted and the file you see now is the new file.

Action: Execute the `rcsyslog restart` command

Cause: The `syslog-ng.conf` file has been modified

Action: Check the `/etc/syslog-ng/syslog-ng.conf` file for the following lines:

```
filter f_dhcpd { facility(daemon) and match('^dhcpd:'); };
destination dhcpmessages { file(var/log/dhcpd.log); };
log { source(src); filter(f_dhcpd); destination(dhcpmessages);};
```

Action: If you want to log only to the `/var/log/dhcpd.log` file, then edit `syslog-ng.conf` files

Restart the syslog daemon by using the `rcsyslog restart` command.

18.1.4 The DHCP server failed to Start

Cause: The DHCP server might have been blocked by the AppArmor process.

Action: Run AppArmor in complain mode to see if DHCP server has been blocked. For details on running AppArmor in complain mode, see the *Novell AppArmor Administration Guide* (http://www.novell.com/documentation/apparmor/apparmor201_sp10_admin/index.html?page=/documentation/apparmor/apparmor201_sp10_admin/data/bx5bml8.html#bx5bmld)

Cause: No credentials are set in CASA

Action: Verify the credentials set in CASA by using the `CASACli -l` command in the console. If no information is displayed, then set the CASA credentials by using the following command:

```
KEYVALUE=<DN of runtime user> CASACli -s -n dhcp-ldap -k CN
```

```
KEYVALUE=<password of runtime user> CASACli -s -n dhcp-ldap -k
Password
```

Action: Verify the credentials set in CASA by using the `CASACli -l` command. If `dhcp-ldap` information is displayed, stop the CASA daemon by using the following command:

```
rcmicasad stop
```

Store the username and password details in the `/etc/dhcpd.conf` file.

For example,

```
ldap-username "cn=runtimeuser,o=novell";  
ldap-password "novell";
```

Cause: Wrong credentials are set in CASA

Action: Reset the CASA credentials by using the CASACli tool.

Cause: DHCP fails to start and throws an error “No subnet declaration for ethx”. Subnet declaration for the interface to which DHCP listens is missing in the configuration.

Action: Declare the same subnet as the DHCP listening interface.

Example:

If DHCP is listening on the eth0 interface 192.168.1.1, then declare a subnet for 192.168.1.0.

NOTE: If you decide not to manage this subnet by this DHCP server, leave the subnet declaration empty i.e, do not create any pools within this subnet.

18.1.5 The DHCP Server displays “Unknown Error” on the console

Cause: The details for user (`- user`) and group (`-group`) passed as command line arguments do not exist on the local machine.

Action: In the `/etc/sysconfig/dhcpd` file, set the value of `DHCPD_RUN_AS` as the local system user.

For example, `DHCPD_RUN_AS=<user>`.

The `<user>` parameter corresponds to the local system user. You can create a local system user with the *Security and Users > User Management* option in YaST.

18.1.6 Permission denied to DHCP server

Cause: The DHCP server might have been blocked by the AppArmor process.

Action: Run AppArmor in complain mode to see if DHCP server has been blocked. For details on running AppArmor in complain mode, see the *Novell AppArmor Administration Guide* (http://www.novell.com/documentation/apparmor/apparmor201_sp10_admin/index.html?page=/documentation/apparmor/apparmor201_sp10_admin/data/bx5bml8.html#bx5bmld)

18.1.7 segfault dhcpd - You get an error “dhcpd: Can't create new lease file: Permission denied” and “dhcpd[8249]: segfault at 0000000000000000 rip 00002abbf999db7f rsp 00007fffb18ea5e0 error 4”

Cause: The db directory might not be owned by the user that is used in `/etc/sysconfig/dhcpd` in the parameter `DHCPD_RUN_AS="dhcpd"`.

Action: Verify that there is a dhcpd user. For details on resolution see [TID 7001158](http://www.novell.com/support/search.do?cmd=displayKC&docType=kc&externalId=7001158&sliceId=1&docTypeID=DT_TID_1_1&dialogID=14520091&stateId=0%200%203951888) (http://www.novell.com/support/search.do?cmd=displayKC&docType=kc&externalId=7001158&sliceId=1&docTypeID=DT_TID_1_1&dialogID=14520091&stateId=0%200%203951888)

18.2 DNS

- ♦ Section 18.2.1, “Failed to Configure DNS Server,” on page 190
- ♦ Section 18.2.2, “Insufficient Permissions for LDAP Admin User,” on page 190
- ♦ Section 18.2.3, “Failed to create the DNS Server object for the Virtual NCP Server,” on page 191
- ♦ Section 18.2.4, “novell-named Failed to Start,” on page 191
- ♦ Section 18.2.5, “rcnovell-named and rnamed interfere in their individual status/stop query functionality,” on page 191
- ♦ Section 18.2.6, “The DNS Server failed to load and provides critical error messages for NWCallsInit/NWCLXInit/NWNetInit,” on page 192
- ♦ Section 18.2.7, “Error message when you add RootServInfo that gives an undefined attribute,” on page 192
- ♦ Section 18.2.8, “Removal of DNS schema post usage of Remove Schema option of dns-maint,” on page 192

18.2.1 Failed to Configure DNS Server

Cause: The DNS schema is not synchronized in the eDirectory tree.

Action: If you are attaching the Linux server to an existing NetWare[®] tree where DNS is not installed, make sure you extend the DNS schema before installing DNS on the Linux tree.

If you are attaching the Linux server to an existing NetWare tree where DNS is installed, make sure the RootServerInfo object on NetWare does not have redundant create and delete permissions for the DNSDHCP-Group object at the entry level permissions.

18.2.2 Insufficient Permissions for LDAP Admin User

Cause: The LDAP user has insufficient permissions for eDirectory objects.

Action: Ensure that adequate permissions are assigned to the user as per [Section 6.1.2, “eDirectory Permissions,” on page 73](#)

18.2.3 Failed to create the DNS Server object for the Virtual NCP Server

Cause: Preferred nodes are not set to the Virtual NCP server

Action: Set the association between the preferred node and the Virtual NCP server.

18.2.4 novell-named Failed to Start

Cause: The daemons required for `novell-named` to start have not been loaded.

Action: Make sure you have loaded all the dependent daemons. For a list of dependent daemons, see [Section 13.5, “Starting the DNS Server,” on page 167](#)

Cause: No credentials are set in CASA.

Action: Verify the credentials set in CASA by using the `CASACli -l` command in the console. If no information is displayed, set the CASA credentials by using the following command:

```
KEYVALUE=<DN of runtime user> CASACli -s -n dns-ldap -k CN
```

```
KEYVALUE=<password of runtime user> CASACli -s -n dns-ldap -k  
Password
```

To reset the CASA credential for DNS, use `dns-maint`, `dns-inst`, or `YaST2` to reset the existing user password. However, for an existing DNS runtime user in eDirectory, you must have the correct password before trying to reset. For a new DNS runtime user and update to the CASA store, use `dns-maint`, `dns-inst`, or `YaST2`.

Cause: On the Cluster setup, the `/etc/rndc.key` file is not same on all the cluster nodes.

Action: Ensure that the `/etc/rndc.key` file is same by copying it across all the nodes on the cluster setup.

Cause: You are loading `novell-named` with the `chroot (-t)` option.

Action: Include the `sys_chroot` capability in the DNS AppArmor profile.

Restart AppArmor using `rcapparmor reload` command.

Cause: `novell-named` might have been blocked by the AppArmor process.

Action: Run AppArmor in complain mode to see if `novell-named` has been blocked. For details on running AppArmor in complain mode, see the [Novell AppArmor Administration Guide \(http://www.novell.com/documentation/apparmor/apparmor201_sp10_admin/index.html?page=/documentation/apparmor/apparmor201_sp10_admin/data/bx5bml8.html#bx5bml8\)](http://www.novell.com/documentation/apparmor/apparmor201_sp10_admin/index.html?page=/documentation/apparmor/apparmor201_sp10_admin/data/bx5bml8.html#bx5bml8)

18.2.5 rcnovell-named and rcnamed interfere in their individual status/stop query functionality

Cause: `novell-named` and BIND DNS (`named`) both leverage `rndcbin` to perform the start, stop, and status queries. `rndc`, a popular Linux command line tool, remotely manages DNS. `novell-bind` continued to support the same feature. For the query operations (start/stop/status), `rndcbin` sends a command to standard DNS control port and waits for the response. However, it does not check which DNS server is listening on this port. If both the servers are loaded in your system (which is

highly unlikely), the server that starts first gets the control port access and listens to it. The status is shown for the server that is loaded and that acquired control over the port. This is because bind-based rndc does not expect two DNS servers running in the system.

This behavior is an rndc restriction and is the way DNS servers dictate the control port.

Action: Running two DNS servers on same system is highly unlikely. Leveraging rndcbin for query functionality either for named or novell-named doesn't cause an issue, provided you explicitly ensure that only one DNS server is running on the system. This ensures that the rndcbin queries correctly identify the DNS server on the system.

18.2.6 The DNS Server failed to load and provides critical error messages for NWCallsInit/NWCLXInit/NWNNetInit

Cause: novell-xregd is not running.

Action: Load novell-xregd if it is not running. Then load novell-named again.

18.2.7 Error message when you add RootServInfo that gives an undefined attribute

Error: [bash] ShellCommand.cc(shellcommand):78. Adding DNS RootServerInfo object failed error_code:17:error_message: Undefined attribute type

Cause: This is a schema synchronization issue.

Action: Ensure that the schema is extended properly, wait for the schema synchronization to occur, then try it again.

18.2.8 Removal of DNS schema post usage of Remove Schema option of dns-maint

Action: See “[Troubleshooting Schema](#)” in the *Novell eDirectory 8.8 Troubleshooting Guide*.

18.3 What's Next

The next section provides information on the various issues and limitations that are specific to the DNS service on OES 2 Linux.

- ♦ [Chapter 19, “Linux Notes,” on page 193](#)

This section describes the various issues and limitations that are specific to the DNS service on OES 2 Linux.

- ♦ [Section 19.1, “DNS Notes,” on page 193](#)
- ♦ [Section 19.2, “What’s Next,” on page 193](#)

19.1 DNS Notes

- ♦ The `-n number of cpus` option in `novell-named` is not supported in the current release.
- ♦ If you are attaching the target server to an existing tree where a DNS server is not installed, you need to extend the DNS schema on the eDirectory™ tree before installing DNS on Linux.

19.2 What’s Next

The next sections provide details on the following:

- ♦ [Section A.1, “Supported RFCs,” on page 195](#)
- ♦ [Section A.2, “Types of Resource Records,” on page 196](#)
- ♦ [Section A.3, “DHCP Option Descriptions,” on page 198](#)
- ♦ [Section A.4, “DNS Root Servers,” on page 205](#)
- ♦ [Section A.5, “DNS Server Configuration Utility,” on page 206](#)
- ♦ [Section A.6, “DNS Server Maintenance Utility,” on page 208](#)
- ♦ [Section A.7, “DHCP Server Maintenance Utility,” on page 212](#)

Appendix

A

This section provides the following information:

- ♦ [Section A.1, “Supported RFCs,” on page 195](#)
- ♦ [Section A.2, “Types of Resource Records,” on page 196](#)
- ♦ [Section A.3, “DHCP Option Descriptions,” on page 198](#)
- ♦ [Section A.4, “DNS Root Servers,” on page 205](#)
- ♦ [Section A.5, “DNS Server Configuration Utility,” on page 206](#)
- ♦ [Section A.6, “DNS Server Maintenance Utility,” on page 208](#)
- ♦ [Section A.7, “DHCP Server Maintenance Utility,” on page 212](#)
- ♦ [Section A.8, “Post-Install Maintenance Tools,” on page 216](#)

A.1 Supported RFCs

Novell® DNS/DHCP services supports the following RFCs:

- ♦ RFC 819—Domain Naming Convention for Internet User Applications
- ♦ RFC 920—Domain Requirements
- ♦ RFC 974—Mail Routing and Domain System
- ♦ RFC 1032—Domain Administrator’s Guide
- ♦ RFC 1033—Domain Administrator’s Operations Guide
- ♦ RFC 1034—Domain Names - Concepts and Facilities
- ♦ RFC 1035—Domain Names - Implementation and Specification
- ♦ RFC 1036—Standard Interchange of USENET Messages
- ♦ RFC 1101—DNS Encoding of Network Names and other Types
- ♦ RFC 1122—Requirements for Internet Hosts - Communications Layers
- ♦ RFC 1123—Requirements for Internet Hosts - Application and Support
- ♦ RFC 1183—New DNS RR Definitions
- ♦ RFC 1535—A Security Problem and Proposed Correction with Widely Deployed DNS Software
- ♦ RFC 1536—Common DNS Implementation Errors and Suggested Fixes
- ♦ RFC 1537—Common DNS Data File Configuration Errors
- ♦ RFC 1591—Domain Name System Structure and Delegation
- ♦ RFC 1597—Address Allocation for Private Internets
- ♦ RFC 1627—Network 10 Considered Harmful (Some Practices Shouldn’t Be Codified)
- ♦ RFC 1884—IP Version 6 Addressing Architecture
- ♦ RFC 1876—Location Information in the DNS
- ♦ RFC 1886—DNS Extensions to Support IP Version 6

- ♦ RFC 1912—Common DNS Operations and Configurations Errors
- ♦ RFC 1995—Incremental Zone Transfer in DNS
- ♦ RFC 1996—A mechanism for Prompt Notification of Zone Changes (DNS NOTIFY)
- ♦ RFC 2010—Operations Criteria for Root Name servers
- ♦ RFC 2136—Dynamic Updates in the DNS
- ♦ RFC 2137—Secure Domain Name System (DNS) Dynamic Update
- ♦ RFC 2163—Using the Internet DNS to distribute MIXER Conformant Global Address Mapping (MCGAM)
- ♦ RFC 2308—Negative Caching of DNS Queries (DNS NCACHE)
- ♦ RFC 2317—Classless IN-ADDR.ARPA delegation
- ♦ RFC 2672—Non-Terminal DNS Name Redirection
- ♦ RFC 2782—A DNS RR for specifying the location of services (DNS SRV)
- ♦ RFC 2874—DNS Extensions to Support IPv6 Address Aggregation and Renumbering

A.2 Types of Resource Records

A resource record (RR) contains data associated with domain names. This data is represented by and is subordinate to the resource record set (RRset) of a zone container.

The following types of resource records can be created:

- ♦ **A6:** Maps a domain name to an IPv6 address.

Specify the following for this option:

- ♦ **Prefix Length:** This can be any value between 0 and 128.
- ♦ **Address Suffix:** The address suffix need not be specified if the prefix length is 128.
- ♦ **Prefix Name:** The domain name of the server. This need not be specified if the prefix length is 0.
- ♦ **A:** Maps a domain name to an IP address.
You must specify the 32-bit IPv4 address that maps to the associated domain for this option.
- ♦ **AAAA:** A 128-bit IPv6 address that is encoded in the data portion of the resource record in the network byte order.
- ♦ **AFSDB:** The AFS system uses DNS to map from a domain name to the name of an AFS cell database server. It contains the following data:
 - ♦ **Subtype:** A 16-bit integer.
 - ♦ **Hostname:** The domain name of the host server.
- ♦ **CNAME:** Specifies the canonical or primary name for the owner. Because the owner name is an alias, you must specify the domain name of the aliased host if you select this option.
- ♦ **DNAME:** Specifies an alternate name to map the entire subtree of the domain namespace.
You must specify the target domain name for this option.
- ♦ **HINFO:** Specifies host information in the form of the following parameters:
 - ♦ **CPU:** A character string that specifies the CPU type.
 - ♦ **OS:** A character string that specifies the operating system type.

- ♦ **ISDN:** An ISDN (Integrated Service Digital Network) number is a telephone number integrating the telephone and data network service to a common service. It contains the following parameters:
 - ♦ ISDN Address: A character string specifying the ISDN address.
 - ♦ Sub Address: A character string specifying the subaddress. This parameter is optional.
- ♦ **LOC:** Specifies the location information in the globe. If you select this option, you must specify the following:
 - ♦ Latitude: Specify the degree of the latitude, minutes, and seconds of the globe enclosing the specified host. Select the direction.
 - ♦ Longitude: Specify the degree of the latitude, minutes, and seconds of the globe enclosing the specified host. Select the direction.
 - ♦ Altitude: Specify the altitude of the globe enclosing the specified host in meters.
 - ♦ Size: Specify the diameter of the globe enclosing the specified host in meters.
 - ♦ Horizontal precision: Specify the horizontal precision of the globe enclosing the specified host in meters.
 - ♦ Vertical precision: Specify the vertical precision of the globe enclosing the specified host in meters.
- ♦ **MB:** Specifies the domain name of the mailbox address.
- ♦ **MG:** A domain name that specifies a mailbox that is a member of the mail group specified by the domain name.
- ♦ **MINFO:** Specifies mailbox or mail list information in the form of the following parameters:
 - ♦ Responsible MailBox: A domain name that specifies the mailbox that is responsible for the mailing list or mailbox.
 - ♦ Error Message MailBox: A domain name that specifies the mailbox that is to receive error messages related to the mailing list or mailbox.
- ♦ **MR:** A domain name that specifies the mailbox that is the proper rename of the specified mailbox.
- ♦ **MX:** Specifies a mail exchange resource record in the form of the following parameters:
 - ♦ Preference: A 16-bit integer that specifies the preference given to this resource record among others at the same owner.
 - ♦ Exchange: A domain name that specifies a host willing to act as a mail exchange for the owner name.
- ♦ **NS:** Specifies a domain name for an authoritative name server for the specified class and domain.
- ♦ **PTR:** A domain name that points to some location in the domain namespace.
- ♦ **PX:** Contains X.400 mail mapping information with the following parameters:
 - ♦ Preference: A 16-bit integer that specifies the preference given to this resource record among others at the same owner.
 - ♦ MAP822: A domain name element containing the RFC822 part of the MCGAM.
 - ♦ MAPX400: A domain name element containing the value derived from the X.400 part of the MCGAM.

- ♦ **RP:** Provides a standard method of associating responsible person identification with any name in the DNS with the following parameters:
 - ♦ Responsible Person's Mailbox: A domain name that specifies the mailbox for the responsible person.
 - ♦ TXT-RR Domain Name: A domain name for the TXT RRs that exist.
- ♦ **RT:** Specifies the routing information with the following parameters:
 - ♦ Preference: A 16-bit integer representing the preference of the route.
 - ♦ Intermediate: The domain name of a host that serves as an intermediate in reaching the host specified by the owner.
- ♦ **SRV:** Specifies the location of services with the following parameters:
 - ♦ Service: The symbolic name of the desired service.
 - ♦ Proto: The TCP and UDP are the most useful values for this field.
 - ♦ Priority: The priority of the target host.
 - ♦ Weight: A load balancing mechanism when selecting a target host among those that have the same priority.
 - ♦ Port: The port on the target host running the service.
 - ♦ Target: The domain name of the target host.
- ♦ **TXT:** Specifies text data in the form of a character string.
- ♦ **WKS:** Describes the well-known services supported by a protocol on a particular host with the following parameters:
 - ♦ Address: A 32-bit Internet address.
 - ♦ Protocol: An 8-bit IP protocol number.
 - ♦ Available Services: A variable-length bitmap.
- ♦ **X25:** Specifies a PSDN (Public Switched Data Network) address.

A.3 DHCP Option Descriptions

The following table describes the DHCP option codes and names:

Table A-1 *DHCP Option Codes and Names*

| Code | Option Name | Description |
|------|-------------|---|
| 1 | Subnet Mask | The subnet mask option specifies the client's subnet mask as per RFC 950. If no subnet mask option is provided, the DHCP server uses the subnet mask from the subnet declaration for the network on which an address is being assigned. |
| 2 | Time Offset | Specifies the offset time of client's subnet in seconds. This is expressed as a two's complement 32-bit integer preference value. Two types of offset can be set: positive and negative. A positive offset indicates a location to the east of the zero meridian and a negative offset indicates a location to the west of the zero meridian. |

| Code | Option Name | Description |
|------|------------------------------|---|
| 3 | Router | Specifies the routers on the client's subnet as a list of IP addresses. The routers should be listed in the order of preference. |
| 4 | Time Server | Specifies a list of RFC time servers available to the client. The servers should be listed in the order of preference. |
| 5 | Name Server | Specifies a list of IEN 116 [7] name servers available to the client. The name servers should be listed in the order of preference. |
| 6 | Domain Name Server | Specifies a list of DNS name servers available to the client. The DNS servers should be listed in the order of preference. |
| 7 | Log Server | Specifies a list of MIT-LCS UDP log servers available to the client. The log servers should be listed in the order of preference. |
| 8 | Cookie Server | Specifies a list of RFC 865 cookie servers available to the client. The cookie servers should be listed in the order of preference. |
| 9 | LPR Server | Specifies a list of RFC 1179 line printer servers available to the client. The LPR servers should be listed in the order of preference. |
| 10 | Impress Server | Specifies a list of Imagen Impress servers available to the client. The impress servers should be listed in the order of preference. |
| 11 | Resource Location Server | Specifies a list of RFC 887 resource location servers available to the client. The resource location servers should be listed in the order of preference. |
| 12 | Host Name | The Host object represents a client in the network with a statically assigned IP address and is identified by a host name. |
| 13 | Boot File Size | Specifies the length of the boot image for the client, in 512-octet blocks. The length is specified as an unsigned 16-bit integer. |
| 14 | Merit Dump File | Specifies the location of a file where the core image of the client should be dumped if the client crashes. |
| 16 | Swap Server | Specifies the IP address of the swap server for the client. |
| 17 | Root Path | Specifies the path name for the client's root disk. |
| 18 | Extension Paths | |
| 19 | IP Forwarding Enable/Disable | Specifies whether the client should forward an IP address. Values can be either True or False. True indicates that IP forwarding should be enabled and False indicates that IP forwarding should be disabled. |
| 20 | Non-Local Source Routing | Specifies whether the client should forward datagrams with non-local source routing. Values can be either True or False. True indicates to enable datagram forwarding and False indicates to disable datagram forwarding. |

| Code | Option Name | Description |
|------|-----------------------------------|--|
| 21 | Policy Filter | Specifies the policy filters for non-local source routing. The policy filters consist of a list of IP addresses and masks that filter the incoming source routes. |
| 22 | Maximum Datagram Re-assembly size | Specifies the maximum size of the datagram that the client should reassemble. |
| 23 | Default IP Time-to-live | Specifies the time-to-live used by the client on outgoing datagrams. |
| 24 | Path MTU Aging Time-out | Specifies the time-out (in seconds) used when the aging path MTU values are discovered by the mechanism defined in RFC 1191. |
| 25 | Path MTU Plateau Table | Specifies a table of MTU sizes used when performing path MTU discovery as defined in RFC 1191. |
| 26 | Interface MTU | Specifies the MTU used on this interface. The minimum value for the MTU is 68. |
| 27 | All subnets are local | Specifies whether the client can assume that all subnets of IP network connected to the client use the same MTU as the subnet of the network to which the client is directly connected. Values can be either True or False. True indicates that all subnets share the same MTU. False indicates that some subnets of the network that is directly connected have smaller MTU values. |
| 28 | Broadcast Address | Specifies the broadcast address being used on the client's subnet. |
| 29 | Perform Mask Discovery | Specifies whether the client should perform subnet mask discovery by using ICMP. Values can be either True or False. True indicates that the client should perform subnet mask discovery. False indicates that the client should not perform subnet mask discovery. |
| 30 | Mask Supplier | Specifies whether the client should respond to subnet mask requests by using ICMP. Values can be either True or False. True indicates that the client should respond and False indicates that the client should not respond. |
| 31 | Perform Router Discovery | Specifies whether the client should solicit routers by using the Router Discovery mechanism as defined in RFC 1256. Values can be either True or False. True indicates that the client should perform router discovery and False indicates that the client should not perform router discovery. |
| 32 | Router Solicitation Address | Specifies the IP address to which the client can send router solicitation requests. |
| 33 | Static Route | Specifies a list of static routes that the client can install in its routing cache. Multiple routes to the same destination are listed in descending order. Static routes consists of a list of IP address in pairs. The first address in the pair is the destination address and the second is the router for the destination. |

| Code | Option Name | Description |
|------|--|---|
| 34 | Trailer Encapsulation | Specifies whether the client can negotiate encapsulating trailers when using the ARP protocol. Values can be either True or False. True indicates that the client should use trailers and False indicates that the client should not use trailers. |
| 35 | ARP Cache Time-out | Specifies the time-out (in seconds) for ARP cache entries. |
| 36 | Ethernet Encapsulation | Specifies whether the client can use Ethernet version 2.0 (RFC 894) or IEEE 802.3 (RFC 1042) encapsulation if it is an Ethernet interface. Values can be either True or False. True indicates that the client should use RFC 1042 encapsulation and False indicates that the client should use RFC 894 encapsulation. |
| 37 | TCP Default TTL | Specifies the default TTL that the client should use when sending TCP segments. |
| 38 | TCP Keep-alive interval | Specifies the interval (in seconds) that the TCP client should wait before sending a keep-alive message on a TCP connection. |
| 39 | TCP Keep-alive garbage | Specifies whether the client should send TCP keep-alive messages with a garbage octet for compatibility with older implementations. Values can be either True or False. True indicates that a garbage octet should be sent and False indicates that a garbage octet should not be sent. |
| 40 | NIS Domain | Specifies the NIS domain name of the client. |
| 41 | NIS Servers | Specifies a list of the NIS server's IP addresses available to the client. The NIS servers should be listed in the order of preference. |
| 42 | Network Time Protocol Servers | Specifies a list of IP addresses indicating Network Time Protocol Servers (NTP servers) available to the client. The NTP servers should be listed in the order of preference. |
| 43 | Vendor Specific Option | Specifies the vendor-specific information that can be used by the clients and servers. |
| 44 | NetBIOS over TCP/IP options-name server | Specifies a list of RFC 1001 and RFC 1002 NetBIOS over TCP/IP name servers listed in order of preference. |
| 45 | NetBIOS over TCP/IP options-datagram distribution server | Specifies a list of RFC 1001 and RFC 1002 NetBIOS over Datagram Distribution servers listed in order of preference. |
| 46 | NetBIOS over TCP/IP options-node type | Allows NetBIOS over TCP/IP clients that can be configured as described in RFC 1001 and RFC 1002. Node types include B-node, P-node, M-node, and H-node. |
| 47 | NetBIOS over TCP/IP options-Scope | Specifies the NetBIOS over TCP/IP scope parameter for the client as specified in RFC 1001 and RFC 1002. |
| 48 | X Window System Font Server | Specifies a list of X Window System Font servers available to the client. These servers should be listed in order of preference. |

| Code | Option Name | Description |
|-------|---------------------------------|--|
| 49 | X Window System Display Manager | Specifies a list of IP addresses of systems that use the X Window System Display Manager and are available to the client. The IP addresses should be listed in order of preference. |
| 58 | Renewal (T1) Time | Specifies the time interval from the address assignment until the client reaches the renewing state. |
| 59 | Renewal (T2) Time | Specifies the time interval from the address assignment until the client reaches the rebinding state. |
| 60 | Vendor Class Identifier | Specifies the vendor type and configuration of a DHCP client. |
| 62 | NWIP Domain Name | Enables the server to convey the NetWare/IP domain name used by the NetWare/IP product. |
| 63-05 | Perform NSQ Broadcast | Specifies whether the client should perform a NetWare Nearest Server Query (NSQ) to find out its nearest NetWare/IP server. Values can be either True or False. True indicates that the client should perform a NetWare NSQ and False indicates that the client should not perform a NetWare NSQ. |
| 63-06 | Preferred DSS | Specifies a list of addresses for a NetWare Domain SAP/RIP Server (DSS). |
| 63-07 | Nearest NWIP Server(s) | Specifies a list of addresses for the Nearest NetWare/IP (NWIP) servers. |
| 63-08 | Number of Auto Retries | Specifies the number of times a NetWare/IP client can attempt to communicate with a DSS server at startup. |
| 63-09 | Auto Retry Interval | Specifies the delay interval (in seconds) that each NetWare/IP client uses when attempting to communicate with a DSS server at startup. |
| 63-10 | Support NWIP 1.1 | Specifies whether the NetWare/IP client should support NetWare/IP version 1.1. This is required only if it contacts a NetWare/IP version 1.1 server. Values can either be True or False. True indicates that the client should support NetWare/IP version 1.1 and False indicates that the client should not support NetWare/IP version 1.1. |
| 63-11 | Primary DDS | Specifies the Primary Domain SAP/RIP Service server (DSS) for the NetWare/IP domain. The NetWare/IP administration utility uses this as the Primary DSS server when configuring a secondary DSS server. |
| 64 | NIS+ Domain | Specifies the name of the client's NIS+ domain. |
| 65 | NIS Servers | Specifies a list of IP addresses indicating Network Information Service (NIS)+ servers available to the client. The NIS+ servers should be listed in order of preference. |
| 66 | TFTP Server Name | Specifies the name of the TFTP server for the client. |
| 67 | Boot File Name | Specifies the name of the boot file for the client. |
| 68 | Mobile IP Home Agent | Specifies a list of IP addresses that indicates the mobile IP home agents available to the client. These agents should be listed in order of preference. |

| Code | Option Name | Description |
|-------|---|---|
| 69 | SMTP Server | Specifies a list of Simple Mail Transport Protocol (SMTP) servers available to the client. The SMTP servers should be listed in order of preference. |
| 70 | POP3 Server | Specifies a list of POP3 servers available to the client. The POP3 servers should be listed in order of preference. |
| 71 | NNTP Server | Specifies a list of Network News Transport Protocol (NNTP) servers available to the client. The NNTP servers should be listed in order of preference. |
| 72 | WWW Server | Specifies a list of World Wide Web (WWW) servers available to the client. The WWW servers should be listed in order of preference. |
| 73 | Default Finger Server | Specifies a list of Finger servers available to the client. The Finger servers should be listed in order of preference. |
| 74 | Default IRC Server | Specifies a list of Internet Relay Chat (IRC) servers available to the client. The IRC servers should be listed in order of preference. |
| 75 | Street Talk Server | Specifies a list of StreetTalk* servers available to the client. The StreetTalk servers should be listed in order of preference. |
| 76 | Street Talk Directory Assistance server | Specifies a list of StreetTalk Directory Assistance (STDA) servers available to the client. The STDA servers should be listed in order of preference. |
| 78 | SLP Directory Agent | Specifies a list of IP addresses for Directory Agents. The Directory Agents should be listed in order of preference. |
| 79 | SLP Service Scope | Specifies the scope that an agent is configured to use. |
| 82-01 | Agent Circuit ID | This sub-option can be added by DHCP relay agents that terminate switched or permanent circuits. It encodes an agent-local identifier of the circuit from which a DHCP client-to-server packet was received. It is intended for use by agents in relaying DHCP responses to the proper circuit. |
| 82-02 | Agent Remote ID | This sub-option can be added by DHCP relay agents that terminate switched or permanent circuits and have mechanisms to identify the remote host end of the circuit. |
| 82-04 | Agent DOCSIS Device Class | Specifies the Device Class encoding within the payload of the Device Class Identification Request (DCI-REQ) message. The relay agent must pass the Device Class value unchanged to the DHCP server. |
| 85 | NDS Servers | Specifies one or more NDS® servers for the client to contact to access the NDS database. The NDS servers should be listed in order of preference. |
| 86 | NDS Tree Name | Specifies the name of the NDS tree that the client contacts. |
| 87 | NDS Context | Specifies the initial NDS context that the client should use. |

A.3.1 Assigning Options

DHCP and BOOTP options can be assigned at these levels:

- ♦ DHCP Service
- ♦ Shared Network
- ♦ Subnet
- ♦ Pool
- ♦ Host
- ♦ Class

The DHCP server's options inheritance rules specify that options assigned at the lowest level override options set at a higher level. For example, options have been assigned at these levels for the client on the subnet, as shown in the following table.

Table A-2 *Options at Different Levels for the Client on the Subnet*

| Level | Option | Value |
|---------|--------------------|--------------|
| Service | 1, Subnet Mask | 255.255.0.0 |
| | 3, Router | 132.57.3.8 |
| | 4, Time Server | 129.23.120.5 |
| Subnet | 1, Subnet Mask | 255.254.0.0 |
| | 5, Name Server | 10.73.57.251 |
| | 7, Log Server | 10.73.58.2 |
| | 13, Boot File Size | 1024 |
| Host | 7, Log Server | Null |
| | 13, Boot File Size | 256 |

The following table lists the effective options for the client with the IP address referred to in the preceding table.

Table A-3 *Options for the Client with IP Address*

| Option | Value |
|--------------------|--------------|
| 1, Subnet Mask | 255.254.0.0 |
| 3, Router | 132.57.3.8 |
| 4, Time Server | 129.23.120.5 |
| 5, Name Server | 10.73.57.251 |
| 7, Log Server | Null |
| 13, Boot File Size | 256 |

A.4 DNS Root Servers

The `/etc/opt/novell/named/root.hint` file contains information about DNS root servers. When the DNS services are installed, the RootSrvrInfo Zone object is created by reading the contents from this file. The RootSrvrInfo zone holds information on the root name servers needed to initialize the cache of Internet domain name servers. The *rootsrvr.dat* file is the Novell version of *named.root*, which is made available by InterNIC registration services.

```
under anonymous FTP as
file                /domain/named.root
on server           FTP.RS.INTERNIC.NET
-OR-
under Gopher at     RS.INTERNIC.NET
under menu          InterNIC Registration Services (NSI)
submenu            InterNIC Registration Archives
file               named.root
```

The related version number of Root Zone is 2004012900 and was last updated on Jan 29, 2004.

The data for the root servers is as follows:

SOA and NS RRs

```
$ORIGIN RootServerInfo.
@      SOA server. root. (
  2004012900  3600 3600 604800 86400 )
.                3600000    NS      A.ROOT-SERVERS.NET.
.                3600000    NS      B.ROOT-SERVERS.NET.
.                3600000    NS      C.ROOT-SERVERS.NET.
.                3600000    NS      D.ROOT-SERVERS.NET.
.                3600000    NS      E.ROOT-SERVERS.NET.
.                3600000    NS      F.ROOT-SERVERS.NET.
.                3600000    NS      G.ROOT-SERVERS.NET.
.                3600000    NS      H.ROOT-SERVERS.NET.
.                3600000    NS      I.ROOT-SERVERS.NET.
.                3600000    NS      J.ROOT-SERVERS.NET.
.                3600000    NS      K.ROOT-SERVERS.NET.
.                3600000    NS      L.ROOT-SERVERS.NET.
.                3600000    NS      M.ROOT-SERVERS.NET.
```

Glue RRs

| | | | |
|---------------------|---------|---|----------------|
| A.ROOT-SERVERS.NET. | 3600000 | A | 198.41.0.4 |
| B.ROOT-SERVERS.NET. | 3600000 | A | 192.228.79.201 |
| C.ROOT-SERVERS.NET. | 3600000 | A | 192.33.4.12 |
| D.ROOT-SERVERS.NET. | 3600000 | A | 128.8.10.90 |
| E.ROOT-SERVERS.NET. | 3600000 | A | 192.203.230.10 |
| F.ROOT-SERVERS.NET. | 3600000 | A | 192.5.5.241 |
| G.ROOT-SERVERS.NET. | 3600000 | A | 192.112.36.4 |
| H.ROOT-SERVERS.NET. | 3600000 | A | 128.63.2.53 |
| I.ROOT-SERVERS.NET. | 3600000 | A | 192.36.148.17 |
| J.ROOT-SERVERS.NET. | 3600000 | A | 192.58.128.30 |
| K.ROOT-SERVERS.NET. | 3600000 | A | 193.0.14.129 |
| L.ROOT-SERVERS.NET. | 3600000 | A | 199.7.83.42 |
| M.ROOT-SERVERS.NET. | 3600000 | A | 202.12.27.33 |

NOTE: Please check the Root Server IP addresses to see that they include the latest IP addresses for all the root zones; otherwise, update the `root.hint` file with the latest update from the [InterNIC site \(http://www.internic.net/zones/named.root\)](http://www.internic.net/zones/named.root) and run the `/opt/novell/named/bin/dns-inst` tool to update the root server addresses after deleting the existing root zones from eDirectory

A.5 DNS Server Configuration Utility

- ♦ “dns-inst” on page 207

dns-inst

The DNS server configuration utility.

Description

dns-inst is the utility to create the DNS server object in the eDirectory tree associated with the NCP server.

Currently, the Novell DNS server configuration utility uses `/opt/novell/named/schema/DNIP.SCH` to extend the DNS schema in the eDirectory tree.

NOTE: In a cluster setup, you cannot create the DNS server object on all the nodes because it needs to refer to the virtual NCP server. The Create Server option cannot be used in cluster setup, it can be used in normal scenarios. Cluster setup requires manual DNS Server object creation.

Syntax

Configuration Option

```
dns-inst <LDAP host name or IP> <LDAP port number> <Admin DN> <password>  
<eDirectory user DN for DNS> <password> <DNS Locator object container name> <DNS  
group object container name> <RootServerInfo container name> <Local NCP server  
context> <Create DNS server object (1->create)*> <Host Name*> <Domain Name for DNS  
Server*> <Secure LDAP or Not (0 -> non-SSL, 1->SSL)>
```

* parameters are optional.

Example

```
dns-inst Acme.com 389 cn=admin,o=Acme secret cn=dns-admin,o=dns-domain secret  
ou=Sales,o=Acme ou=Finance,o=Acme o=acme o=acme 1 acme-host acme.americas.com 0
```

Other Parameters for Server Object Creation

The following parameters are optional. If you do not want to create DNS server object, then do not consider these parameters.

- ♦ **Create DNS server object:** The value is 1 for object creation.
- ♦ **Host Name:** Specify a unique hostname for the DNS Server object, such as acme-host.
- ♦ **Domain Name for DNS Server:** Specify a domain name for the Server object, such as acme.americas.com.

Definitions

- ♦ **LDAP host name:** The IP address of the default LDAP server for the service.
- ♦ **LDAP port number:** The secure or non-secure LDAP port to connect to the LDAP server.
- ♦ **Admin DN:** The LDAP administrator distinguished name to authenticate against the LDAP host.

- ♦ **password:** The password for the LDAP Admin.
- ♦ **eDirectory user DN for DNS:** The user authenticates to eDirectory to access information for DNS during runtime. The user must have eDirectory read, write, and browse rights under the specified context.
- ♦ **DNS Locator object container name:** The context/container for the DNS Locator object. For example: o=novell. The DNS Locator object contains global defaults, DHCP options, and a list of all DNS and DHCP servers, subnets, and zones in the tree.
- ♦ **DNS group object container name:** The context for the DNS Group object. For example: o=novell. This object is used to grant DNS servers the necessary rights to other data within the eDirectory tree.
- ♦ **RootServerInfo container name:** The context for the DNS Services RootServerInfo object. For example: o=novell. The RootServerInfo Zone is an eDirectory container object that contains resource records for the DNS root servers.
- ♦ **Local NCP server context:** Specify a context for the local NCP Server object. The DNS Server reference is stored in this object. For example: o=novell.
- ♦ **Secure LDAP or Not (0 -> non-SSL, 1 -> SSL):** Set this option to 1 to ensure that the data transferred by this service is secure and private. Set this option to 0 to transfer the data in clear text format.

Authors

Copyright 2008, Novell, Inc. All rights reserved. <http://www.novell.com>

A.6 DNS Server Maintenance Utility

- ♦ “dns-maint(8)” on page 209

dns-maint(8)

Name

`dns-maint` - The DNS server maintenance utility.

Description

`dns-maint` is the utility to create and maintain the DNS server object in the eDirectory tree associated with the NCP server.

NOTE: In a cluster setup, you cannot create the DNS server object on all the nodes because it needs to refer to virtual NCP server. The Create Server option cannot be used in cluster setup; it can be used only in normal scenarios. Cluster setup requires manual DNS Server object creation.

Syntax

DNS Configuration Options

Default Option

```
dns-maint <LDAP host name or IP> <LDAP port number> <Admin DN> <Admin password>  
<DNS Proxy user DN> <DNS Proxy user password> <DNS Locator object container name>  
<DNS group object container name> <RootServerInfo container name> <Local NCP server  
context> <Create DNS server object (1->create)*> <Host Name*> <Domain Name for DNS  
Server*> <Secure LDAP or Not (0 -> non-SSL, 1->SSL)>
```

* parameters are optional.

Advanced option

Configure All

```
dns-maint -ca <Tree-name> <LDAP host name> <LDAP port number> <Admin DN> <Admin  
password> <DNS Proxy user DN> <DNS Proxy user password> <DNS Locator object  
container name> <DNS group object container name> <RootServerInfo container name>  
<Local NCP server context> <Create DNS server object (1->create)*> <Host Name*>  
<Domain Name for DNS Server*> <Secure LDAP or Not (0 -> non-SSL, 1->SSL)>
```

* parameters are optional.

Currently, `dns-maint` uses `/opt/novell/named/schema/DNIP.SCH` to extend the DNS schema in eDirectory tree.

IMPORTANT: In the default and advanced options for `dns-maint`, if you choose to configure an existing user as a proxy user, ensure that you provide the correct password for this user. A wrong password creates casa store corruption for this user and causes DNS server load failure.

Other Parameters for Server Object Creation

The following parameters are optional. If you do not want to create a DNS server object, do not consider these parameters.

- ♦ **Create DNS server object:** The value is 1 for object creation.
- ♦ **Host Name:** Specify a unique hostname for the DNS Server object. For example, acme-host.
- ♦ **Domain Name for DNS Server:** Specify a domain name for the Server object. For example, acme.americas.com.

Object and Schema Removal Options

Remove All

```
dns-maint -ra <LDAP host name> <LDAP port number> <Admin DN> <DNS Locator object  
container name> <Delete Common Objects=1 Do not delete Common Objects=0> <Secure  
LDAP or Not (0 -> non-SSL, 1->SSL)>
```

Remove Schema

```
dns-maint -rs <LDAP host name> <LDAP port number> <Admin DN> <Secure LDAP or Not (0  
-> non-SSL, 1->SSL)>
```

WARNING: During installation, removing DNS in Domain Services for Windows setup means that DSfW Services fails or demands reconfiguration to some other DNS server. Reconfiguring DNS on the DSfW setup is not possible because of its integration with DSfW. To reconfigure, you need to completely re-run the DSfW-DNS install script to re-populate the DNS zones and resource records in the tree. dns-maint should not be used to remove all (-ra), reconfig (-ca), or remove schema (-rs) for a DSfW deployment.

Options

Usage Options:

-ca, Configure All

```
dns-maint -ca <Tree-name> <LDAP host name> <LDAP port number> <Admin DN> <Admin  
password> <DNS Proxy user DN> <DNS Proxy user password> <DNS Locator object  
container name> <DNS group object container name> <RootServerInfo container  
name> <Local NCP server context> <Create DNS server object (1->create)*> <Host  
Name*> <Domain Name for DNS Server*> <Secure LDAP or Not (0 -> non-SSL, 1-  
>SSL)>
```

* parameters are optional. If the DNS runtime admin already exists in the tree, then the Configure All option resets the existing user password to the specified password in the command. The password is reset only in the CASA store but not in eDirectory. Because of this inconsistency, the DNS service might not work properly.

The Configure All option extends the DNS schema in the tree, refreshes the tree, and creates DNS objects such as DNSDHCP-Group, DNS-DHCP (Locator), and RootServerInfo in the specified input context in the tree.

It also creates the Runtime Admin (Proxy User) if it does not exist in the tree, and adds it to the CASA store.

For secure updates, specify the SSL port number for the LDAP and SSL option as 1.

For non-secure updates, specify the SSL option as 0.

For example:

```
dns-maint -ca <Acme-tree> Acme.com 636 cn=admin,o=Acme secret cn=dns-  
admin,o=dns-domain secret ou=Sales,o=Acme ou=Finance,o=Acme o=acme o=acme 1  
acme-host acme.americas.com 0
```

-ra, Remove All

```
dns-maint -ra <LDAP host name> <LDAP port number> <Admin DN> <DNS Locator  
object container name> <Delete Common Objects=1 Do not delete Common  
Objects=0> <Secure LDAP or Not (0 -> non-SSL, 1->SSL)>
```

The Remove All option removes the Rootserverinfo, Zone objects, resource record details of the zones, and the DNS-Server objects from the tree for the specified DNS Locator object.

1. To delete the Group Object and the Locator Object, specify the option as 1
2. To retain the Group Object and the Locator Object, specify the option as 0

It removes the DNS credentials from CASA. It removes the .conf, .db, .jnl, .pid files from the system from their respective directories (/etc/opt/novell/named, /var/opt/novell/log/named, /var/opt/novell/run/named) under the current user root.

For secure updates, specify the SSL port number for the LDAP and SSL option as 1.

For non secure updates, specify the SSL option as 0.

For example:

```
dns-maint -ra Acme.com 636 cn=admin,o=Acme ou=Sales,o=Acme 1
```

-rs, Remove Schema

```
dns-maint -rs <LDAP host name> <LDAP port number> <Admin DN> <Secure LDAP or  
Not (0 -> non-SSL, 1->SSL)>
```

The Remove Schema option uses the /opt/novell/named/schema/DNIP.SCH schema file to remove schema from eDirectory tree by using the /opt/novell/named/bin/removeschema.sh script.

If the DNS schema is not in use, the Remove Schema option removes the DNS schema from the tree.

For secure updates, specify the SSL port number for the LDAP and SSL option as 1.

For non-secure updates, specify the SSL option as 0.

For example:

```
dns-maint -rs Acme.com 636 cn=admin,o=Acme
```

Definitions

1. **LDAP host name:** The IP address of the default LDAP server for the service.
2. **LDAP port number:** The secure or non-secure LDAP port to connect to the LDAP server.

3. **Admin DN:** LDAP administrator distinguished name to authenticate against the LDAP host.
4. **password:** Password for the LDAP Admin.
5. **eDirectory user DN for DNS:** The user authenticates to eDirectory to access information for DNS during runtime. The user must have eDirectory read, write, and browse rights under the specified context.
6. **DNS Locator object container name:** The context/container for the DNS Locator object. For example: o=novell. The DNS Locator object contains global defaults, DNS options, and a list of all DNS and DHCP servers, subnets, and zones in the tree.
7. **DNS Group object container name:** The context for the DNS Group object. For example: o=novell. This object is used to grant DNS servers the necessary rights to other data within the eDirectory tree.
8. **RootServerInfo container name:** The context for the DNS Services RootServerInfo object. For example: o=novell. The RootServerInfo Zone is an eDirectory container object that contains resource records for the DNS root servers.
9. **Local NCP server context:** Specify a context for the local NCP Server object. The DNS Server reference is stored in this object. For example: o=novell.
10. **Secure LDAP or Not (0 -> non-SSL, 1 -> SSL):** Set this option to 1 to ensure that the data transferred by this service is secure and private. Set this option to 0, to transfer the data in clear text format.
11. **Tree name:** eDirectory tree name of your NCP server.
12. **Delete Common Objects=1 Do not delete Common Objects=0:** This option specifies the DNS objects to be deleted from the eDirectory Tree.
 - ♦ To delete the Group Object and the Locator Object, specify the option as 1.
 - ♦ To retain the Group Object and the Locator Object, specify the option as 0.

The Group Objects and the Locator Objects are common for DNS and DHCP services on NetWare. Cleaning up these objects from dns-maint is not recommended if you want to clean up only the DNS objects and then retain the DHCP objects in the tree. The Delete Common Objects=1 Do not delete Common Objects=0 option gives you the choice to delete or retain the common objects for DNS-DHCP in the tree.

Authors

Copyright 2008, Novell, Inc. All rights reserved. <http://www.novell.com>

A.7 DHCP Server Maintenance Utility

- ♦ “dhcp-maint(8)” on page 213

dhcp-maint(8)

Name

dhcp-maint - The DHCP server maintenance utility.

Description

dhcp-maint is the utility to create and maintain the DHCP configuration.

Syntax

DHCP Configuration Options

DHCP Default Configuration Option

```
dhcp-maint <LDAP host name or IP> <LDAP port number> <Admin DN> <Admin password>  
<server object container name> <Locator object container name> <group object  
container name> <server object name> <Secure LDAP or Not (0 -> non-SSL, 1->SSL)>  
<DHCP Proxy user DN> <DHCP Proxy user password>
```

DHCP Advanced Configuration Option

Configure All

```
dhcp-maint -ca <LDAP host name or IP> <LDAP port number> <Admin DN> <Admin  
password> <server object container name> <locator object container name> <group  
object container name> <server object name> <Secure LDAP or Not (0 -> non-SSL, 1-  
>SSL)> <DHCP Proxy user DN> <DHCP Proxy user password>
```

If the DHCP runtime admin already exists in the tree, then the Advanced option resets the existing user password to the specified password in the command. The password is reset only in the CASA store but not in eDirectory. Because of this inconsistency, the DHCP service might not work properly.

IMPORTANT: In the default and advanced options for dhcp-maint, if you choose to configure an existing user by using dhcp-maint, ensure that you provide the correct password for this user. A wrong/new password creates CASA store corruption for this user and causes DHCP server load failure.

Object and Schema Removal Options

Remove All

```
dhcp-maint -ra <LDAP host name or IP> <LDAP port number> <Admin DN> <Locator object  
container name> <Delete Group=1 Locator=2 Both=3 None=0> <Secure LDAP or Not (0 ->  
non-SSL, 1->SSL)>
```

Remove Schema

```
dhcp-maint -rs <LDAP host name or IP> <LDAP port number> <Admin DN> <Secure LDAP or Not (0 -> non-SSL, 1->SSL)>
```

Options

Usage Options

-ca, Configure All

```
dhcp-maint -ca <LDAP host name or IP> <LDAP port number> <Admin DN> <Admin password> <server object container name> <locator object container name> <group object container name> <server object name> <Secure LDAP or Not (0 -> non-SSL, 1->SSL)> <DHCP Proxy user DN> <DHCP Proxy user password>
```

The Configure All option extends the DHCP schema in the tree, refreshes the tree, and creates DHCP objects such as DHCPGroup, dhcpLocator, and DHCP Server in the specified input context in the tree.

It also creates the Runtime Admin (Proxy User) if it does not exist in the tree and adds it to the CASA store.

For a secure connection, specify the SSL port number for LDAP and SSL option as 1.

For a non-secure connection, specify the SSL option as 0.

If the DHCP runtime admin already exists in the tree, the Advanced option resets the existing user password to the specified password in the command. The password is reset only in the CASA store but not in eDirectory. Because of this inconsistency, the DHCP service might not work properly.

For example:

```
dhcp-maint -ca Acme.com 636 cn=admin,o=Acme secret ou=Sales,o=Acme ou=Finance,o=Acme o=Acme dhcp_ncpsrvr 1 cn=dhcp-admin,o=Acme secretou=Sales,o=Acme ou=Finance,o=Acme o=Acme o=Acme 0
```

-ra, Remove All

```
dhcp-maint -ra <LDAP host name or IP> <LDAP port number> <Admin DN> <locator object container name> <Delete Group=1 Locator=2 Both=3 None=0> <Secure LDAP or Not (0 -> non-SSL, 1->SSL)>
```

The Remove All option removes all the DHCP configuration objects like service, subnet, host, pool, failover peer, DHCP Server etc from the tree for the specified Locator object.

1. To delete the Group Object, specify the option as 1.
2. To delete the Locator Object, specify the option as 2.
3. To delete both the Locator and the Group Objects, specify the option as 3.
4. Specify 0 for no deletions.

The Remove All option performs a complete cleanup of eDirectory for DHCP and no check is done for local or remote server. After the cleanup it removes the DHCP files (dhcpd.log, dhcpd.conf, dhcpd.pid) from the local machine from their respective directories (/var/log, /etc, /var/run) under the current user root. The DHCP CASA credentials are also removed from the local CASA store only.

NOTE: The server connected here is for LDAP bind and should not be considered as cleanup for the remote server.

To delete files and CASA credentials on a remote system, dhcp-maint should be run on that system locally. However, the CASA credentials can also be removed by using the CASAcli tool.

For a secure connection, specify the SSL port number for the LDAP and SSL options as 1.

For a non-secure connection, specify the SSL option as 0.

For example:

```
dhcp-maint -ra Acme.com 636 cn=admin,o=Acme o=Acme 3 1
```

-rs, Remove Schema

```
dhcp-maint -rs <LDAP host name> <LDAP port number> <Admin DN> <Secure LDAP or Not (0 -> non-SSL, 1->SSL)>
```

If the DHCP schema is not in use, then the Remove Schema option removes the DHCP schema from the tree.

For a secure connection, specify the SSL port number for the LDAP and SSL options as 1.

For a non-secure connection, specify the SSL option as 0.

For example:

```
dhcp-maint -rs Acme.com 636 cn=admin,o=Acme
```

Definitions

1. **LDAP host name or IP:** The IP address of the default LDAP server for the service.
2. **LDAP port number:** The secure or non-secure LDAP port to connect to the LDAP server.
3. **Admin DN:** The LDAP administrator distinguished name to authenticate against the LDAP host.
4. **password:** The password for the LDAP Admin.
5. **eDirectory user DN for DHCP:** The user authenticates to eDirectory to access information for DHCP during runtime. The user must have eDirectory read, write, and browse rights under the specified context.
6. **locator object container name:** The context/container for the DHCP Locator object. For example: o=novell. The DHCP Locator object contains global defaults, DHCP options, and a list of all DHCP servers, subnets, and zones in the tree.
7. **group object container name:** The context for the DHCP Group object. For example: o=novell. This object is used to grant DHCP servers the necessary rights to other data within the eDirectory tree.

- 8. Secure LDAP or Not (0 -> non-SSL, 1 -> SSL):** Set this option to 1 to ensure that the data transferred by this service is secure and private. Set this option to 0, to transfer the data in clear text format.
- 9. Delete Group=1 Locator=2 Both=3 None=0:** This option specifies the DHCP objects to be deleted from the eDirectory tree.
- a. To delete the DHCPGroup Object, specify the option as 1.
 - b. To delete the dhcpLocator Object, specify the option as 2.
 - c. To delete both the DHCPGroup and dhcpLocator Objects, specify the option as 3.
- The Delete Group=1 Locator=2 Both=3 None=0 option gives you the choice to retain or delete the common objects for DNS-DHCP in the tree.

Authors

Copyright 2008, Novell, Inc. All rights reserved. <http://www.novell.com>

A.8 Post-Install Maintenance Tools

For any post-install operations using dns-maint and dhcp-maint, ports 1389/1636 should be used to avoid any issues because of changed behavior of the DSFW server. Many access/search errors happen if you run dns-maint and dhcp-maint to use 389/636.

Documentation Updates

B

This section contains information on documentation content changes that were made in this DNS/DHCP Administration Guide for Linux after the initial release. The changes are listed according to the date they were published.

The documentation for this product is provided on the Web in two formats: HTML and PDF. The HTML and PDF documentation are both kept up-to-date with the changes listed in this section.

If you need to know whether a copy of the PDF documentation that you are using is the most recent, the PDF document includes a publication date on the title page.

The documentation was updated on the following dates:

- ♦ [Section B.1, “November, 2008,” on page 217](#)
- ♦ [Section B.2, “December, 2008,” on page 218](#)
- ♦ [Section B.3, “October, 2008,” on page 218](#)
- ♦ [Section B.4, “September, 2008,” on page 218](#)
- ♦ [Section B.5, “August, 2008,” on page 219](#)
- ♦ [Section B.6, “July, 2008,” on page 219](#)
- ♦ [Section B.7, “June, 2008,” on page 219](#)
- ♦ [Section B.8, “April, 2008,” on page 219](#)
- ♦ [Section B.9, “March, 2008,” on page 219](#)

B.1 November, 2008

- ♦ In [Table 1-3 on page 38](#), the following content was deleted as it was not applicable for DNS:

| | |
|----------------------|---|
| Our Network | DHCP Server, Subnet, Shared Network |
| DHCP Server | DHCP Server, Subnet, Shared Network |
| Subnet | Subnet Address Range, DHCP Server, IP Address, Subnet, Shared Network |
| Subnet Address Range | DHCP Server, Subnet, Shared Network |
| IP Address | DHCP Server, Subnet, Shared Network |
| Shared Network | DHCP Server, Subnet, Shared Network |

- ♦ In [“Managing Roles and Tasks” on page 36](#), TSIG Key management was included with the following description:
The TSIG Key Management allows you to create, modify, and delete TSIG Key objects. TSIG Key provides a means of authentication for dynamic DNS updates and for queries to a secured DNS Server.
- ♦ The iManager [Figure 1-12 on page 36](#) and Java Console [Figure 1-13 on page 37](#) were changed to DNS management interface.

- ♦ The note in the section “DNS Zone Object” on page 29, was revised to:
Novell DNS support underscore character in domain names and Resource records using Novell DNS server "check-names" statement for co-existence with Windows DNS servers.
- ♦ Section 11.4, “Verifying Installation,” on page 129 was included with the following information:
Verify that CASA is updated with the new run time admin credentials using CASACli option.
- ♦ Editorial corrections were made throughout the book, and some sections were reorganized.
- ♦ YaST screen for DNS installation was updated with the latest screen in the Section 11.2, “Installing the DNS Server,” on page 127.

B.2 December, 2008

- ♦ The following note is included in the Section 11.3, “Setting Runtime Credentials,” on page 129:

NOTE: In cluster setup, Runtime Credentials must be set on all the nodes in the DNS cluster.

- ♦ Step 3 is included in the Section 13.7, “Running DNS Server as a Non-Root User,” on page 168.

B.3 October, 2008

- ♦ The section “Removal of DNS schema post usage of Remove Schema option of dns-maint” on page 192 was included to provide information for troubleshooting the eDirectory™ schema.
- ♦ Important note included in the Section , “Loading or Unloading a Server,” on page 84 as follows:

IMPORTANT: To load/unload a server, the user must be LUM-enabled and should have supervisor rights at the entry level to the UNIX workstation object.

B.4 September, 2008

- ♦ Section 6.1.2, “eDirectory Permissions,” on page 73 was are re-written.
- ♦ A new section Chapter 12, “Migrating DNS/DHCP from OES 1 NetWare to OES 2 SP1 Linux,” on page 131 was included to refer to Migration chapters.
- ♦ Added an update in the Section 18.2.4, “novell-named Failed to Start,” on page 191 with the following content:
 - ♦ KEYVALUE=<DN of runtime user> CASACli -s -n dns-ldap -k CN
KEYVALUE=<password of runtime user> CASACli -s -n dns-ldap -k Password
To reset the CASA credential for DNS, use dns-maint, dns-inst, or YaST2 to reset the existing user password. However, for an existing DNS runtime user in eDirectory, you must have the correct password before trying to reset. For a new DNS runtime user and update to the CASA store, use dns-maint, dns-inst, or Yast2.
- ♦ A new section “Associating a Zone to Specific DNS Servers” on page 145 was included to associate a zone serving a specific DNS Server.

B.5 August, 2008

- ♦ TSIG Key (DHCP) description changed.
- ♦ Note included for max-journal-size for Linux iManager.
- ♦ Chapter 14, “Configuring DNS with Novell Cluster Services,” on page 169 reviewed for user comments and changed the cluster setup and configuring sections.
- ♦ Chapter 8, “Configuring DHCP with Novell Cluster Services for the NSS File System,” on page 109 and Chapter 9, “Configuring DHCP with Novell Cluster Services for the Linux File System,” on page 115 reviewed for user comments and sections to load and unload scripts refined.
- ♦ `dns-maint(8)` and `dhcp-maint(8)` NOTE included for default and `-ca` options. IMPORTANT also included for default and `-ca` options.

B.6 July, 2008

- ♦ Updated the document with edits and common structure. Book updated to the July 11, 2008 template.
- ♦ Added Chapter 17, “DNS-DSfW Integration,” on page 183.
- ♦ Changes made in Chapter 14, “Configuring DNS with Novell Cluster Services,” on page 169 and Chapter 9, “Configuring DHCP with Novell Cluster Services for the Linux File System,” on page 115.
- ♦ Migration sections for DNS and DHCP were moved to the Migration Framework guide for OES2 SP1.

B.7 June, 2008

- ♦ Added Section 7.2, “Using the Java Management Console for DHCP (OES Linux),” on page 99.

B.8 April, 2008

- ♦ Updated the book to April 28, 2008 template.

B.9 March, 2008

- ♦ Updated the template to March 11, 2008.

Glossary

C

This section describes the most commonly used terms in DNS/DHCP Services.

Additional Options: An attribute of the DNS server and the zone, which allows fine-tuning of options for server performance. The values specified at the zone level override the values specified in the DNS server.

Additional from Auth: Controls the behavior of an authoritative server when answering queries that have additional data, or when following CNAME and DNAME chains. When this option is set to yes, and when a query is being answered from authoritative data, the additional data section of the reply is filled in with data from other authoritative zones.

Additional from Cache: Controls the behavior of an authoritative server when answering queries that have additional data, or when following CNAME and DNAME chains. When this option is set to yes, and when a query is being answered from authoritative data, the additional data section of the reply is filled in using data from the cache.

Allow Notify: Specifies the hosts that are allowed to notify slaves of a zone change in addition to the zone masters. This can be configured only for a secondary zone.

Allow Recursion: A list of IP addresses or networks from which the DNS server accepts queries recursively. If a value is not specified, the default is to allow recursive queries from all hosts.

Also Notify: A list of IP addresses of name servers that are also sent notify messages when a new copy of the zone is loaded, in addition to the servers listed in the zone's NS records. This is primarily meant to converge stealth servers. The default is an empty list (no additional notification list). The value specified in the zone overrides the value specified in the server.

Authoritative: DNS data that is served by the resident DNS server. The server can be either primary or secondary. This is the DNS data that belongs to a resident domain and is managed by the administrator of that domain, or it is the DNS data that is imported through a zone transfer.

Blacklist servers: These are fake servers; the DNS server does not answer queries from or forward queries to these servers. This list is maintained in the `dnipBlacklistServers` attribute of the DNS server object in Novell® eDirectory™.

Bulk Zone Export: An action that transfers one or more zones' configuration and data from eDirectory to files. This can be done by using the Import-Convert-Export (ICE) utility, which uses ICE zone handlers.

Bulk Zone Import: An action that transfers one or more zones' configuration and data from files to eDirectory. This can be done using the Import-Convert-Export (ICE) utility, which uses ICE zone handlers.

Cleaning Interval: With this option set, the server removes expired resource records from the cache after every cleaning interval. If it is set to 0, no periodic cleaning occurs.

Cluster: Novell DNS cluster services is a server clustering solution that provides high availability and manageability of critical network resources including data, applications, and services. It is enabled for eDirectory and supports failover, fallback, and migration (load balancing) of individual managed services.

Co-existence: Both the old and new servers interoperate on the same eDirectory configuration. The underlying platform / configuration is the same and allows different versions of the server to work on it. Co-existence of a new DNS server with a current DNS server allows customers to do a phased upgrade and migration from old DNS servers to new DNS servers.

Designated Primary Server: The master primary server, which serves a primary zone and honors zone-out transfer requests. It is the only primary server in the zone that accepts dynamic updates. There is only one designated primary server per zone.

Designated Secondary Server: The master secondary server, which serves a secondary zone and honors zone-out transfer requests. It is the only secondary server in the zone that performs in-bound zone transfer requests to the primary server in the zone. There is only one designated secondary server per zone.

Dynamic Reconfiguration: Detects the changes in the DNS server and DNS Zone configuration data and applies it from eDirectory to the DNS server in-memory while the DNS server is in running mode (without shutting down the server).

Dynamic Update - Novell proprietary: The Novell DHCP server sends the updates to DNS server by using a Novell proprietary update format. The proprietary dynamic update message has a different format from the standard RFC 2136 and also has security signature associated with the message. There is a logic for establishing credentials for each connection from DHCP to DNS. After the credentials are established, the DHCP server sends the actual packet of DNS data for update to the DNS server.

Dynamic Update - RFC 2136: The new DNS servers accept dynamic update requests in standard RFC 2136 format. For more information, refer to [RFC 2136 \(http://www.ietf.org/rfc/rfc2136.txt?number=2136\)](http://www.ietf.org/rfc/rfc2136.txt?number=2136).

Event: The occurrence of an action on an object of interest.

Fault tolerance: Handles temporary disruptions of eDirectory unavailability, with graceful degradation in functionality. This can be categorized as:

- ♦ **Full Fault Tolerance:** The state when eDirectory is down and access to the DNS server object is broken. The server does not accept the following:
 - ♦ Dynamic Update
 - ♦ Zone-in
 - ♦ Notify

No write operation can be performed until eDirectory is up.

The server resolves only normal queries and zone-out transfers. Full Fault-Tolerance mode is applicable to all zones that are being serviced by the DNS server.

- ♦ **Partial Fault Tolerance:** The state when eDirectory is up but access to some zones is broken (because some eDirectory partitions are down or are not accessible). The server does not accept:
 - ♦ Dynamic Update
 - ♦ Zone-in
 - ♦ Notify

No write operation to these zones is performed until the partition is down

The server only resolves normal queries for this zone. The queries are resolved only until the expiration for a secondary zone and forever for a primary zone.

Forward: This option can be configured only if the forwarding list is not empty. A value of First, which is the default, causes the server to query the forwarders first. If that does not answer the query, the server then looks for the answer. If Only is specified, the server queries only the forwarders.

Empty Forwarder: This option is used for domain delegation (child zones). With an Empty Forward list, global forwarders are ignored and NS records are used for domain delegation.

Forwarder: A DNS server that forwards queries to other DNS servers, if the requested information is not found on the local server.

FQDN: Fully qualified distinguished name

Group: The DNS/DHCP Group object is a standard eDirectory group object. The DNS and DHCP servers gain the rights to DNS and DHCP data within the tree through the Group object.

ICE: The Import-Convert-Export utility to import or export the DNS server, zone configuration information, and data to or from the eDirectory database.

Journal Log: All changes made to a zone through dynamic update are stored in the zone's journal log. The server automatically creates this log when the first dynamic update takes place. The extension .jnl is appended to the name of the corresponding zone to form the journal log file. The journal log is in a binary format and should not be edited manually.

Lame TTL: Sets the number of seconds to cache a lame server indication (these are misconfigurations in the remote servers, discovered by the DNS service when trying to query those servers during resolution). 0 disables caching (not recommended). The maximum value is 1800 (30 minutes).

Listen On: Specifies the interfaces and ports that the server answers queries from. It takes an optional port and an address match list. If a port is not specified, port 53 is used.

Locator: The DNS/DHCP Locator object contains a reference to global defaults and DHCP options, and list of all DNS and DHCP servers, subnets, and zones in the tree.

Maximum Cache Size: The maximum amount of memory (in bytes) used for the server's cache. When the amount of data in the cache reaches this limit, the server causes records to expire prematurely so that the limit is not exceeded. The default is 0 (unlimited cache).

Maximum Cache TTL: Sets the maximum time for which the server caches ordinary (positive) answers.

Maximum NCache TTL: Sets a maximum retention time for negative answers in the server. The server stores negative answers to reduce network traffic and increase performance. The maximum value is 7 days.

Maximum Recursion Lookups: The maximum number of simultaneous recursive lookups that the server performs on behalf of the clients. This allows you to set limits on the servers' resource consumption. The default value is 1000.

NOTE: Each recursive client uses about 20 KB of memory

Minimal Responses: Allows the server to add records to the authority sections, and optionally to the additional section depending on the value set for this option. If this is set to No, the server adds records to both the authority and additional sections when generating responses. If this is set to Yes, the server adds records only to the authority section when generating responses. The performance of the server increases if this option is set to No.

Non-Authoritative: DNS data that is not served by the resident DNS server. This is the DNS data that belongs to a foreign domain and is not managed by the resident DNS administrator. This data is cached through responses to forwarded queries.

Notify: When this option is set to yes, DNS notify messages are sent when the contents of a zone for which the server is authoritative changes. The messages are sent to the servers listed in the zone's NS records (except the master server identified in the SOA MNAME field), and to any servers listed in the also-notify option.

Notify Source: Determines the local source address, and optionally the UDP port that is used to send notify messages. The slave should also be configured to receive notify messages from this address.

Novell Dynamic Reconfigure: Specifies the time interval at which dynamic reconfiguration takes place. The minimum value is 10 minutes and the maximum is 24 hours.

Out-of-band update: Any update to DNS Zone data in eDirectory that by-passes the DNS server (that is, all updates except dynamic updates).

Passive Primary Server: A DNS server that serves a primary zone and honors zone-out transfer requests. This server is passive because it cannot update the zone data. There can be multiple passive primary servers serving the same primary zone.

Passive Secondary Server: A DNS server that serves a secondary zone and does not issue in-bound zone transfer requests to the primary server of the zone. It answers queries to the zone and honors zone-out transfers requests to the zone. There can be multiple passive secondary servers serving the secondary zone.

Performance: This parameter measures the throughput of the server in handling requests and is indicated as the response time for processing concurrent requests (queries, updates, zone transfers, etc.).

Primary Zone: A zone that is authoritative and is serviced by a designated primary DNS server and one or more passive primary DNS servers.

Provide IXFR: Determines whether the local server, acting as the master, responds with an incremental zone transfer when the given remote server, a slave, requests it. If it is set to Yes, incremental transfer is provided whenever possible. If it is set to No, all transfers to the remote server are non-incremental (AXFR). The default is Yes.

Query Filter: List of IP addresses or networks from which the DNS server accepts queries. If this option is not specified, the default is to allow queries from all hosts. The value specified for this option in the zone overrides the value specified in the server.

Query Source: Specifies the address and port used for querying other name servers, if the server does not know the answer to a query.

Recursion: If this option is set to Yes, and a DNS query requests recursion, then the server attempts to do everything required to answer the query. If this option is set to No and the server does not already know the answer, it returns a referral response.

Role: An object in the iManager framework that is associated with user objects in eDirectory.

Roll Back: To revert to the previous state if a transaction fails.

RootSrvrInfo: The RootSrvrInfo Zone is a Zone object, which is an eDirectory container object that contains RRsets for the DNS Root servers. The RootSrvrInfo Zone object is the equivalent of the BIND `db.root` file.

Request IXFR: Determines whether the local server, acting as a slave, requests incremental zone transfers from the given remote server, a master. The default is True.

RR Set Order: Permits ordering of the records in a multiple record response in an RRset. Currently, Novell DNS server supports two orders: random-cyclic and fixed. The default is random-cyclic.

Scalability: This parameter measures how the server scales with load in terms of the number of zones, number of RRs per zone, number of DNS queries, and zone transfers or dynamic updates handled by the server in a typical deployment scenario. It also identifies the limits of the parameters to which the server offers consistent performance without degradation.

Scope settings: Setting the scope and context of Locator object in the eDirectory tree enables better search responses for DNS-DHCP objects. This avoids searching the entire tree by limiting the search within the current scope set.

Secondary Zone: A zone that is serviced by a designated secondary DNS server and one or more passive secondary DNS servers.

Serial Query Rate: Through this option, the slave servers periodically query master servers to find out if the zone serial numbers have changed. Each such query uses a small amount of the slave server's network bandwidth. In order to limit the amount of bandwidth used, limit the rate at which queries are sent. The value of the serial-query-rate option is an integer, which is the maximum number of queries sent per second.

Slave Server: A DNS server that answers queries from its authoritative data and cached data, but relies completely on the forwarders for external information. It does not contact other servers if the forwarders do not give it an answer. A slave server can be a primary or secondary for its authoritative data.

Task: A task is an object in the iManager framework that is associated with a role object in eDirectory. Each task describes some action that a role can play to create, modify, or delete objects in eDirectory.

TCP Clients: Specifies the maximum number of simultaneous client TCP connections that the server accepts.

Transaction support: The DNS server supports transaction for a dynamic update request. This means committing the update to eDirectory, in-memory rbt (red-black tree) database, and in the journal log. If the transaction to any of these fails, the update is rolled back and a negative response is sent to the dynamic update request.

Transfer Format: Through this option, zone transfers can be done using two different formats, one-answer and many-answers. This option is used on the master server to determine which format it sends. One-answer uses one DNS message per resource record transferred; many-answers places as many resource records as possible into a message. Many-answers is more efficient.

Transfers In: Specifies the maximum number of inbound zone transfers that can run concurrently. Increasing the transfers-in might speed up the convergence of slave zones, but it might also increase the load on the local system.

Transfers Out: Specifies the maximum number of outbound zone transfers that can run concurrently. The zone transfer requests that are in excess of the limit are refused.

Transfers per NS: Specifies the maximum number of inbound zone transfers that can be transferred concurrently from a given remote name server. Increasing the value of this option might speed up the convergence of slave zones, but it might also increase the load on the remote name server.

Transfer Source: Determines the local address that is bound to the IPv4 TCP connections used to fetch the zones transferred inbound by the server. It also determines the source IPv4 address, and optionally the UDP port, used for the refresh queries and forwarded dynamic updates.

Update Filter: List of IP addresses or networks from which the DNS server accepts dynamic DNS updates for primary zones. The default is to deny updates from all hosts. This attribute is effective only on a primary designated server.

Write-through: Writing the dynamic update data immediately to eDirectory (primary data), server in-memory, and the journal log at the time of request (that is, before replying to the dynamic update request).

Zone Export: Transfers a single zone configuration/data from eDirectory into a file. This can be done with the DNS/DHCP Management utilities.

Zone Import: Transfers a single zone configuration and data from a file into eDirectory. This can be done with the DNS/DHCP Management utilities.

Zone-in: Zone data received by a secondary server from a primary server.

Zone-out: Transfer of data from a primary server to a secondary server.

Zone Statistics: If this option is set to ON, the DNS server collects statistical data on all zones in the server.