Cisco Prime Network Registrar 11.2 Installation Guide

First Published: 2023-10-30

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED “AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

Cisco has more than 200 offices worldwide. Addresses and phone numbers are listed on the Cisco website at www.cisco.com/go/offices.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: https://www.cisco.com/c/en/us/about/legal/trademarks.html. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)

© 2023 Cisco Systems, Inc. All rights reserved.

CONTENTS

CHAPTER 1 **Installation Overview** 1

Overview 1

About Cisco Prime Network Registrar 1

Sensitive Data Exposure 2

CHAPTER 2 **Configuration Options** 3

Mixed DHCP and DNS Scenarios 3

One-Machine Mixed Configuration 3

Two-Machine Mixed Configuration 3

Three-Machine Mixed Configuration 4

Four-Machine Mixed Configuration 4

DHCP-Only Scenarios 4

One-Machine DHCP Configuration 5

Two-Machine DHCP Configuration 5

DNS-Only Scenarios 5

One-Machine DNS Configuration 5

Two-Machine DNS Configuration 5

Three-Machine DNS Configuration 5

CHAPTER 3 **Installation Requirements** 7

System Requirements 7

Recommendations 9

Installation Modes 10

License Files 10

CHAPTER 4 **Preparing for the Installation** 15

Installation Checklist 15
Before You Begin 16
Obtaining Cisco Prime Network Registrar License Files 16
Image Signing 17
Running Other Protocol Servers 18
Backup Software and Virus Scanning Guidelines 19

CHAPTER 5 **Installing and Upgrading Cisco Prime Network Registrar** 21
Installing Cisco Prime Network Registrar 21
Upgrade Considerations 24
Using Smart Licensing 24
Upgrading Cisco Prime Network Registrar 25
Reverting to an Earlier Product Version 27
Moving a Local Cluster to a New Machine 28
Moving a Regional Cluster to a New Machine 29
Installing Your Own Certificate for Web UI Access 30
Troubleshooting the Installation 31
Troubleshooting Local Cluster Licensing Issues 32

CHAPTER 6 **Next Steps** 35
Configuring Cisco Prime Network Registrar 35
Using Cisco Prime Network Registrar 35
Starting and Stopping Servers 37
Starting or Stopping Servers Using the Local Web UI 38
Starting and Stopping Servers Using the Regional Web UI 38
Server Event Logging 38
Disabling REST API 39
Local and Regional Advanced Web UI 39
CLI Commands 39

CHAPTER 7 **Uninstalling Cisco Prime Network Registrar** 41
Uninstalling Cisco Prime Network Registrar 41

CHAPTER 8 **Cisco Prime Network Registrar Virtual Appliance** 43

System Requirements 43
Installing and Upgrading Cisco Prime Network Registrar Virtual Appliance 44
Preparing to Deploy the Cisco Prime Network Registrar Virtual Appliance 44
Deploying the Regional Cluster OVA or Local Cluster OVA on VMware 45
Booting and Configuring Cisco Prime Network Registrar Virtual Appliance 47
Deploying the Regional Cluster or Local Cluster on OpenStack 47
Upgrading the Cisco Prime Network Registrar Virtual Appliance 50
Upgrading a Cisco Prime Network Registrar Installation to Run on a Cisco Prime Network Registrar Virtual Appliance 50
Upgrading to a new Version of the Virtual Appliance Operating System 51
Upgrading the Cisco Prime Network Registrar Application 52
Next Steps: Cisco Prime Network Registrar Virtual Appliance 52
Configuring Cisco Prime Network Registrar with the CLI on Virtual Appliance 52
Configuring the Virtual Appliance to Automatically Power Up 52
Managing the Cisco Prime Network Registrar Virtual Appliance 53
Post OVA Installation 54
Building Your Own Base Images for Deploying Virtual Machines 55
Disabling dnsmasq and libvirtd 55

CHAPTER 9 **Cisco Prime Network Registrar on Container** 57
Requirements on the Host Machine 57
Running Cisco Prime Network Registrar Docker Container 57
Moving an Existing Cisco Prime Network Registrar Cluster to Docker Container 60
Upgrading Cisco Prime Network Registrar Docker Container 61

CHAPTER 10 **Cisco Prime Network Registrar on Kubernetes** 63
Deploying Cisco Prime Network Registrar regional and local Instance on Kubernetes 63
Deploying Cisco Prime Network Registrar CDNS instance on Kubernetes 66
Container Logging 69
CDNS Metrics 69
CDNS Scaling 69
Upgrading Cisco Prime Network Registrar Kubernetes 70

APPENDIX A **Lab Evaluation Installations** 73

Lab Evaluation Installations 73
Installing Cisco Prime Network Registrar in a Lab 73
Testing the Lab Installation 74
Uninstalling in a Lab Environment 74

APPENDIX B **Installing the Cisco Prime Network Registrar SDK 75**
Installing Cisco Prime Network Registrar SDK 75
Testing Your Installation 76
Compatibility Considerations 76

APPENDIX C **Enhancing Security for Web UI 77**
Enhancing Security for Web UI 77

APPENDIX D **Hardening Guidelines 79**
Hardening Guidelines 79

APPENDIX E **Optimizing VM Performance 83**
Recommended UCS Settings 83
NUMA Optimization 83
Hyperthreading Considerations 83

APPENDIX F **Configuring Network Access on RHEL/AlmaLinux 8.x Using nmcli 85**
Configuring Network Access on RHEL/AlmaLinux 8.x Using nmcli 85

APPENDIX G **Authoritative DNS Capacity and Performance Guidelines 89**
DNS System Deployment Limits 89
DNS Database Architecture 90
DNS System Sizing 91

APPENDIX H **Caching DNS Capacity and Performance Guidelines 93**
DNS System Deployment Limits 93
Caching DNS System Sizing 94
Possible Impacts on Caching DNS Server Performance 95

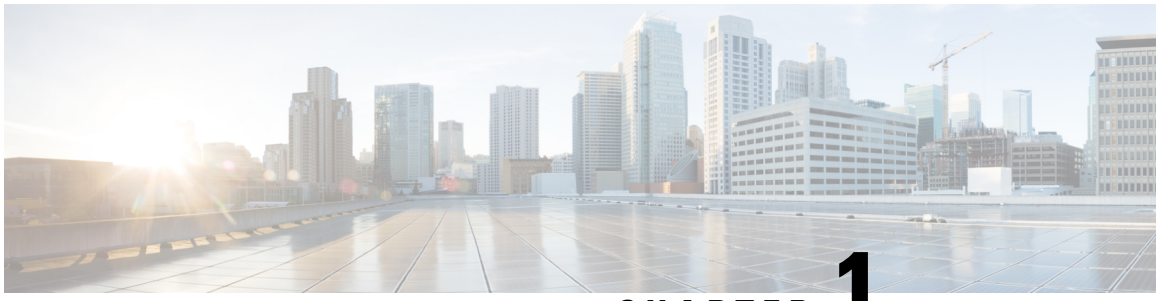
APPENDIX I **DHCP Capacity and Performance Guidelines** 97

Local Cluster DHCP Considerations 97

Number of Leases Allowed on a Single Server 97

Server Considerations 101

Regional Cluster DHCP Considerations 102



CHAPTER 1

Installation Overview

This chapter contains the following sections:

- Overview, on page 1
- About Cisco Prime Network Registrar, on page 1
- Sensitive Data Exposure, on page 2

Overview

This guide describes how to install Cisco Prime Network Registrar 11.2 on Linux operating system. You can also see the following documents for important information about configuring and managing Cisco Prime Network Registrar:

- For configuration and management procedures for Cisco Prime Network Registrar, see the *Cisco Prime Network Registrar 11.2 Administration Guide*.
- For details about commands available through the Command Line Interface (CLI), see the *Cisco Prime Network Registrar 11.2 CLI Reference Guide*.

About Cisco Prime Network Registrar

Cisco Prime Network Registrar is a network server suite that automates managing enterprise IP addresses. It provides a stable infrastructure that increases address assignment reliability and efficiency. It includes (refer the below figure).

- Dynamic Host Configuration Protocol (DHCP) server
- Domain Name System (DNS) server
- Caching Domain Name System (CDNS) server
- Simple Network Management Protocol (SNMP) server
- Trivial File Transfer Protocol (TFTP) server

You can control these servers by using the Cisco Prime Network Registrar web-based user interface (web UI) or the CLI. These user interfaces can also control server clusters that run on different platforms.

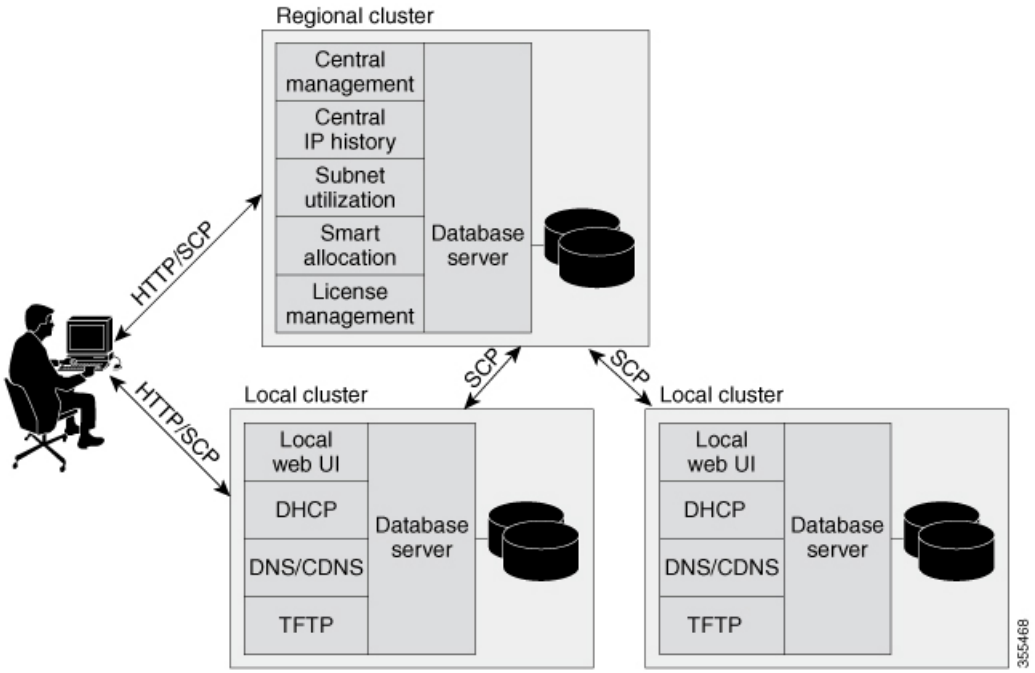
You can install Cisco Prime Network Registrar in either local or regional mode:

- Local mode is used for managing local cluster protocol servers.
- Regional mode is used for managing multiple local clusters through a central management model.

A regional cluster is required for licensing and can be used to centrally manage local cluster servers and their address spaces. The regional administrator can perform the following operations:

- Manage licenses for Cisco Prime Network Registrar. An installation must have at least one regional cluster for license management purposes.
- Push and pull configuration data to and from the local DNS and DHCP servers.
- Obtain DHCP utilization and IP lease history data from the local clusters.

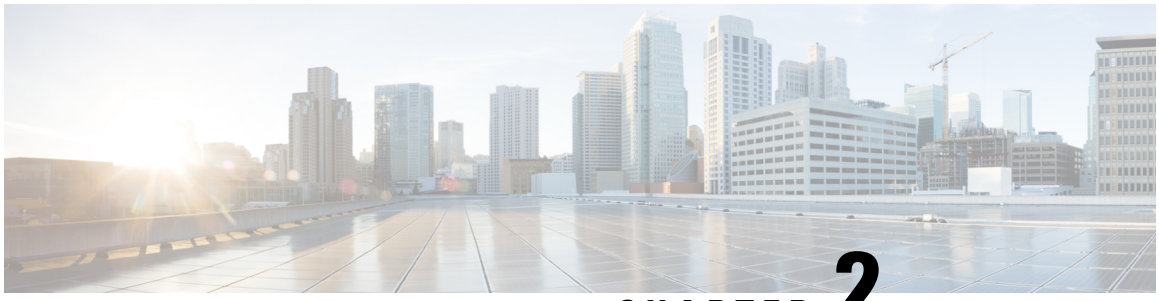
Figure 1: Cisco Prime Network Registrar User Interfaces and the Server Cluster



Sensitive Data Exposure

Most of the data that Cisco Prime Network Registrar deals with is sent over unencrypted networks (especially the last hop to client devices), and is designed by its nature to be shared and available to other devices on the network (either locally or across the Internet).

If you consider the data (or portions of it) that Cisco Prime Network Registrar has as sensitive, we highly recommend you to encrypt your disks using the Linux support for disk based encryption. This will help protect the data once the disks leave controlled space (that is, reach end of life, when it not possible to erase it properly, or is stolen). You also need to consider how to protect any backups, or other places you may move the data.



CHAPTER 2

Configuration Options

Cisco Prime Network Registrar DHCP, Authoritative DNS, and Caching DNS components are licensed and managed from the regional server. You need to have a regional server and all services in the local clusters are licensed through the regional cluster. Only a regional install asks for a license file and only the regional server accepts new license files. Then the regional server can authorize individual local clusters based on available licenses.

The sample configuration shown in this chapter is based on the typical use cases described in the following sections:

- Mixed DHCP and DNS Scenarios, on page 3
- DHCP-Only Scenarios, on page 4
- DNS-Only Scenarios, on page 5

Mixed DHCP and DNS Scenarios

You can set up Cisco Prime Network Registrar for a mixed DHCP and DNS configuration with different numbers of machines.

One-Machine Mixed Configuration

Configure both DHCP and Auth DNS servers on a single machine, initially enabling the servers as primaries, and enabling the TFTP server and SNMP traps. Then configure at least one forward zone and corresponding reverse zone, and at least one scope.

Configure both DHCP and Caching DNS servers on a single machine, initially enabling the servers as primaries, and enabling the TFTP server and SNMP traps. Then you can configure forwarders and exception lists.

Two-Machine Mixed Configuration

A mixed DHCP configuration on two machines offers a few alternatives:

- Configure one machine as primary DHCP and Auth DNS server, and the second machine as a secondary Auth DNS server. Then configure a zone distribution and DNS access controls on the first machine and optionally access controls on the second machine.
- Configure one machine as DHCP and Auth DNS main servers, and the second machine as DHCP and Auth DNS backup servers. Perform minimal configuration on the backup machine (changing the password,

enabling DHCP and Auth DNS, and selecting partner backup roles). On the main machine, build the configuration, creating server pairs and scheduling synchronization tasks with the backup machine.

- Configure one machine as a DHCP server and the second machine as a Auth DNS primary, and then configure either machine with DNS Update and push the configuration to the other machine.
- Configure one machine with both DHCP server and Auth DNS server, and the second machine as a Caching DNS server with the Auth DNS server as the Forwarder.

Three-Machine Mixed Configuration

A mixed configuration on three machines offers a few additional alternatives:

- Configure one machine as a DHCP server, the second machine as an Auth DNS primary, and the third machine as an Auth DNS secondary. Optionally revisit the machines to make the DHCP main the Auth DNS backup, and make the Auth DNS main the DHCP backup.
- Configure one machine as DHCP failover and Auth DNS High-Availability (HA) main servers, the second machine as DHCP failover and Auth DNS HA backup servers, and the third machine as an Auth DNS secondary server.
- Configure one machine as a DHCP server, the second machine as an Auth DNS server, and the third machine as a Caching DNS, with the Auth DNS as the Forwarder.
- Configure one machine as a DHCP primary server and Auth DNS primary, the second machine as a DHCP secondary and Auth DNS secondary server, and the third machine as a Caching DNS, with the primary Auth DNS of the first machine as the Forwarder.

Four-Machine Mixed Configuration

A mixed configuration on four machines could include:

- DHCP and Auth DNS main and backup pairs, with the first machine as a DHCP main, the second machine as a DHCP backup, the third machine as an Auth DNS main configured with DNS Update, and the fourth machine as an Auth DNS backup.
- An add-on to the three-machine scenario, with the first machine as a DHCP main, the second machine as an Auth DNS main, the third machine as DHCP and Auth DNS backups, and the fourth machine as an Auth DNS secondary.
- Configure the first machine as DHCP main, second machine as DHCP backup, third machine as Auth DNS, and Caching in fourth, with Auth DNS as Forwarder.

DHCP-Only Scenarios

A DHCP-only configuration could be on a single machine or two machines.

One-Machine DHCP Configuration

Initially configure only DHCP, skip the class-of-service and failover options, and revisit the setup to enable class-of-service and policy options.

Two-Machine DHCP Configuration

Configure the first machine as a DHCP main and the second machine as a backup, with minimal backup configuration (changing password, enabling DHCP, and selecting the backup role), and set up the first machine with failover load balancing, optionally scheduling failover synchronization tasks.

DNS-Only Scenarios

A DNS-only configuration could be on one, two, or three machines.

One-Machine DNS Configuration

Initially configure DNS as an Auth primary, Auth secondary, or caching server.

Two-Machine DNS Configuration

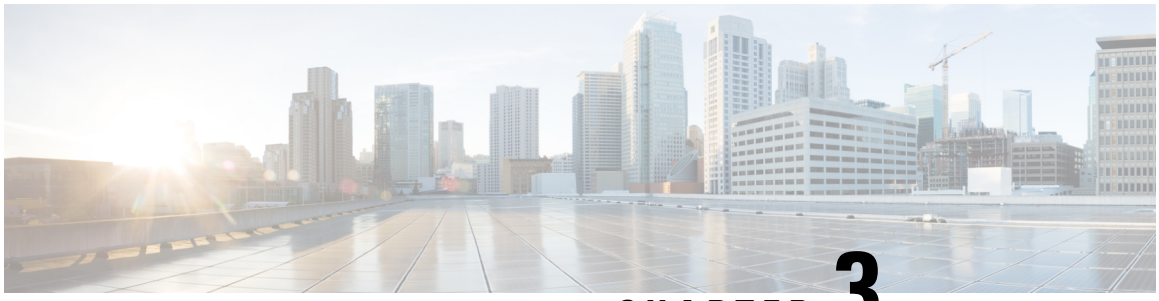
Configure the first machine as an Auth DNS primary and the second machine as a secondary, or the first machine as a main primary and the second machine as a backup primary.

Configure the first machine as an Auth DNS and the second machine as Caching DNS.

Three-Machine DNS Configuration

Configure the first machine as an Auth DNS main primary, the second machine as a backup primary, and the third machine as a secondary server.

Configure the first machine as Auth DNS primary, the second machine as secondary, and the third machine as Caching DNS.



CHAPTER 3

Installation Requirements

This chapter contains the following sections:

- System Requirements, on page 7
- Installation Modes, on page 10
- License Files, on page 10

System Requirements

Review the system requirements before installing the Cisco Prime Network Registrar 11.2 software:

- Java—Ensure that you have Java Development Kit (JDK) 11 installed on your system.

Note Any 64-bit Java 11 is supported for Cisco Prime Network Registrar 11.2 and it has been tested with the Java 11 OpenJDK package.

- Operating System—Ensure that your Cisco Prime Network Registrar machine runs on the Linux Operating Systems as described in the Server Requirements table below. Cisco Prime Network Registrar requires a 64-bit operating system.
- User Interface—Cisco Prime Network Registrar currently includes two user interfaces: a web UI and a CLI:
  - The web UI has been tested on Microsoft Edge 100, Mozilla Firefox 99, and Google Chrome 100. Internet Explorer is not supported.
  - The CLI runs in a Linux command window.

Tip Include a network time service in your configuration to avoid time differences between the local and regional clusters. This method ensures that the aggregated data at the regional server appears consistently. The maximum allowable time drift between the regional and local clusters is five minutes. If the time skew exceeds five minutes, then the installation process will not be able to correctly register the server with the regional. In this case, unset and set the password on the regional cluster, and sync again.

Table 1: Cisco Prime Network Registrar Server Requirements

Component	Requirement
OS version[1]	RHEL 8.x/AlmaLinux 8.x 64-bit **Note:** The newest level tested with this release are RHEL 8.8 and AlmaLinux 8.8.
Minimum disk space	200 GB For best performance, Cisco recommends use of SSD drives.
Minimum memory	16 GB
Minimum CPUs[2]	4 CPUs

[1] Cisco Prime Network Registrar 11.2 is only supported on 64-bit operating systems.
[2] Faster CPU and more memory typically result in higher peak performance.

Note The operating system version used for Cisco Prime Network Registrar 11.2 virtual appliance is AlmaLinux 8.8.

Note Cisco Prime Network Registrar 10.1 is the last release to support Windows.

Note Based on the type of clusters you are planning to deploy, see the Capacity and Performance Guidelines appendices for more details.

Important Treat these system requirements as minimal guidelines. We advise you to monitor your deployment and adjust based on the actual usage level you are seeing.

Cisco Network Registrar has been tested against Red Hat Enterprise Linux ES 8.8 and AlmaLinux 8.8. However, it is anticipated that the end users apply patches and maintenance releases to keep their OS upto date with OS-related bug fixes and security patches. Cisco does not anticipate that these patches/maintenance updates within the same OS major version will cause issues, but as always, it is highly recommended that any updates be lab tested before they are applied to production servers.

System Requirements for Linux OS

The following x86_64 (64-bit) packages (over and above the JDK) are required to install Cisco Prime Network Registrar on Red Hat Enterprise Linux or AlmaLinux. If you use the **yum** or **dnf** commands to install Cisco Prime Network Registrar, then these packages are installed as part of the installation process if required. If you use the **rpm** command to install Cisco Prime Network Registrar, then you must install these packages separately.

Table 2: Packages to Install

Package Name	Package Version
	For RHEL 8.x/AlmaLinux 8.x
glibc	2.28 or later
krb5-libs	1.17 or later
ldns	1.7.0 or later
libcurl (built with OpenSSL)	7.61.1 or later
libevent	2.1.8 or later
libgcc	8.3.1 or later
libicu	60.3 or later
libstdc++	8.3.1 or later
libxml2	2.9.7 or later
net-snmp-libs	5.8 or later
openldap	2.4.46 or later
openssl-libs	1.1.1c or later
tcl	8.6.8 or later
zlib	1.2.11 or later
libnghttp2	1.33.0 or later

To find out the packages required, you can also issue the following command on your Linux system if you have downloaded the RPM:

```
rpm -qpR rpm_package_file
```

The installer will report any packages that may be missing before beginning the installation process.

Note To know the kind of Linux system you are on, use the following command:

```
more /etc/redhat-release
```

Recommendations

When Cisco Prime Network Registrar is deployed on virtual machines, review the following recommendations:

- Do NOT deploy HA DNS or DHCP failover partners on the same physical server (in separate VMs). This will not provide high availability when the server goes down. Ideally, the high available/failover

partners should be sufficiently "separate" that when one fails (because of a hardware, power, or networking failure), the other does not.

- When deploying multiple Cisco Prime Network Registrar VMs on the same physical server (or servers served by a common set of disk resources), you should stagger the automatic nightly shadow backups (by default, they occur at 23:45 in the server's local time). To know how to alter this time, see the *"Setting Automatic Backup Time" section in the Cisco Prime Network Registrar 11.2 Administration Guide*.

Note It may be acceptable to not follow the above recommendations for lab environments; but they must be followed for production.

Installation Modes

The modes of installation that exist for the local and regional clusters are new installations and upgrades from a previous version. These installations or upgrades are performed by using the **yum install**, **rpm -i**, or **dnf install** command.

Note If you use the **rpm -i** command to install Cisco Prime Network Registrar, you may have to install the dependencies manually.

License Files

Cisco Prime Network Registrar 11.2 supports both Smart Licensing and traditional licensing.

Cisco Smart Licensing is a flexible licensing model that provides you with an easier, faster, and more consistent way to purchase and manage software across the Cisco portfolio and across your organization. And it’s secure – you control what users can access. With Smart Licensing you get:

- **Easy Activation**: Smart Licensing establishes a pool of software licenses that can be used across the entire organization—no more PAKs (Product Activation Keys).
- **Unified Management**: Cisco License Central provides a complete view into all of your Cisco products and services in an easy-to-use portal, so you always know what you have and what you are using.
- **License Flexibility**: Your software is not node-locked to your hardware, so you can easily use and transfer licenses as needed.

To use Smart Licensing, you must first set up a Smart Account on Cisco License Central. For a more detailed overview on Cisco Licensing, visit the Licensing Hub.

In case of traditional licenses (FLEXlm), you purchase a perpetual license for a version and use it until Cisco Prime Network Registrar servers are upgraded to a newer major version. In case of Smart Licensing, the licenses are not installed on the individual Cisco products, instead, they are kept in a centralized system called Cisco Smart Software Manager (CSSM) or CSSM On-Prem (Satellite), in customer specific Smart accounts.

For more details on the Licensing, see the *"Licensing" section in the Cisco Prime Network Registrar 11.2 Administration Guide*.

The The Cisco Prime Network Registrar 11.2 license file contains two sets of licenses that cover the permanent and subscription parts of the license. The permanent licenses are similar to the licenses issued for 8.x, 9.x, and 10.x versions. For Cisco Prime Network Registrar IP Express 11.2, the licensing is done according to the services that you require. The perpetual portion of the license will continue to use the mappings established for Cisco Prime Network Registrar 9.0 and later.

Following are the types of licenses available:

Smart Licensing:

- PNR-System—Licenses the CCM services. This license is mandatory if you want to run Cisco Prime Network Registrar.
- PNR-DHCP—Licenses DHCP/TFTP services and, optionally, an initial count of leases.
- PNR-DNS—Licenses the authoritative DNS services and, optionally, an initial count of RRs.
- PNR-Caching DNS—Licenses Caching DNS services and, optionally, an initial count of servers.
- PNR-DHCP Container—Licenses DHCP services on containers.
- PNR-DNS Container—Licenses authoritative DNS services on containers.
- PNR-Caching DNS Container—Licenses Caching DNS services on containers.

Traditional Licensing:

- base-system—Licenses the CCM services. This license is mandatory if you want to run Cisco Prime Network Registrar.
- base-dhcp—Licenses DHCP/TFTP services and, optionally, an initial count of leases.
- base-dns—Licenses the authoritative DNS services and, optionally, an initial count of RRs.
- base-cdns—Licenses Caching DNS services and, optionally, an initial count of servers.
- count-dhcp—Licenses an incremental number of active leases.
- count-dns—Licenses an incremental number of RRs.
- count-cdns—Licenses an incremental number of caching server instances.

A corresponding subscription license is issued for each permanent Cisco Prime Network Registrar 11.x license. The expiration date for each subscription license is set to the subscription period.

Following are the types of licenses available:

Smart Licensing:

- PNR-System SIA—Licenses the CCM services. This license is mandatory if you want to run Cisco Prime Network Registrar.
- PNR-DHCP SIA—Licenses DHCP/TFTP services and, optionally, an initial count of leases.
- PNR-DNS SIA—Licenses the authoritative DNS services and, optionally, an initial count of RRs.
- PNR-Caching DNS SIA—Licenses Caching DNS services and, optionally, an initial count of servers.

- PNR-DHCP Container SIA—Licenses DHCP services on containers.
- PNR-DNS Container SIA—Licenses authoritative DNS services on containers.
- PNR-Caching DNS Container SIA—Licenses Caching DNS services on containers.

Traditional Licensing:

- sub-system —Licenses the CCM services.
- sub-dhcp—Licenses the DHCP services.
- sub-count-dhcp—Licenses an incremental number of active leases.
- sub-dns—Licenses the authoritative DNS services.
- sub-count-dns—Licenses an incremental number of RRs.
- sub-cdns—Licenses Caching DNS services.

The different services provided by Cisco Prime Network Registrar are associated with the different license types as follows:

- CCM services—base-system and PNR-System
- DHCP services—base-dhcp, count-dhcp, and PNR-DHCP
- Authoritative DNS services—base-dns, count-dns, and PNR-DNS
- Caching DNS services—base-cdns, count-cdns, PNR-Caching DNS

Note Licenses for Cisco Prime Network Registrar 10.x or earlier are not valid for Cisco Prime Network Registrar 11.x. You should have a new license for Cisco Prime Network Registrar 11.x. For the 11.x regional, if one has 10.x CDNS clusters, the 10.x CDNS licenses must be added on the regional server (10.x CDNS clusters will use 10.x licenses, and 11.x CDNS clusters will use 11.x licenses).

Note You should not delete any of the individual licenses loaded from the file. If required, you may delete older versions of DNS and DHCP licenses after the upgrade. Older versions of CDNS licenses must be retained if the servers are not upgraded.

Note Subscription licenses, if provided, should be installed to assure upgrades to future releases.

Note You should have at least one base license for a server to enable that service.

License management is done from the regional cluster when Cisco Prime Network Registrar is installed. You must install the regional server first, and load all licenses in the regional server. When you install the local cluster, it registers with regional to obtain its license.

When you install the regional, you are prompted to provide the license file. You can store the license file in any location, provided the location and file are accessible during the installation.

The utilization of licenses are calculated by obtaining statistics from all the local clusters in the Cisco Prime Network Registrar system for all counted services (DHCP, DNS, and CDNS). The regional CCM server maintains the license utilization history for a predetermined time period.

Utilization is calculated for different services as:

- **DHCP services**—Total number of "active" DHCP leases (including v4 and v6)

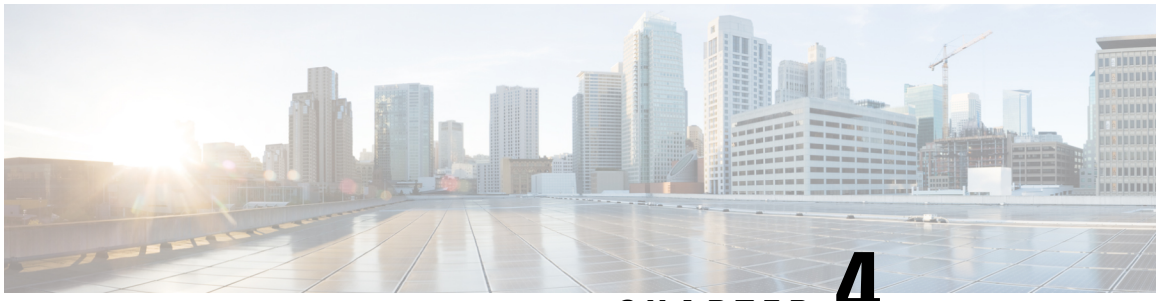
  Active leases include the number of leases in use by a client (and thus not available to another client) which also includes reservations and leases in transition.

- **Auth DNS services**—Total number of DNS resource records (all RR types)
- **Caching DNS services**—Total number of Caching DNS servers being run in the Cisco Prime Network Registrar system

The services on each local cluster will be restricted based on the services for which licenses are present.

When you configure DHCP failover, only simple failover is operational and supported (see the *"Failover Scenarios" section in the "Configuring DHCP Failover" chapter in the Cisco Prime Network Registrar 11.2 DHCP User Guide*).

To learn about obtaining the license files for Cisco Prime Network Registrar, see Obtaining Cisco Prime Network Registrar License Files, on page 16.



CHAPTER 4

Preparing for the Installation

This chapter covers any tasks that you have to perform before installing Cisco Prime Network Registrar.

- Installation Checklist, on page 15
- Before You Begin, on page 16
- Obtaining Cisco Prime Network Registrar License Files, on page 16
- Image Signing, on page 17
- Running Other Protocol Servers, on page 18
- Backup Software and Virus Scanning Guidelines, on page 19

Installation Checklist

This section explains the procedures you must follow to install Cisco Prime Network Registrar.

Before you perform the installation or upgrade, ensure that you are prepared by reviewing this checklist:

Table 3: Installation Checklist

Task	Checkoff
Does my operating system meet the minimum requirements to support Cisco Prime Network Registrar 11.2? (See System Requirements, on page 7)	☐
Does my hardware meet the minimum requirements? (See System Requirements, on page 7)	☐
If necessary, have I excluded Cisco Prime Network Registrar directories and subdirectories from virus scanning? (See Backup Software and Virus Scanning Guidelines, on page 19)	☐
Do I have the proper software license? (See License Files, on page 10)	☐
Am I authorized for the administrative privileges needed to install the software?	☐
Does the target installation server have enough disk space?	☐
Is this a new installation or an upgrade?	☐
Which type of installation is this - regional cluster, local cluster, or client-only?	☐
Is the 64-bit JDK installed on the system? If so, where?	☐

Task	Checkoff
Am I upgrading from an earlier version of Cisco Prime Network Registrar? If so:	☐
• Are there any active user interface sessions?	☐
• Is my database backed up?	☐
• Am I upgrading from a supported version (Cisco Prime Network Registrar 8.3 and later)?	☐
Are the required packages for Linux installed? (See System Requirements for Linux OS, on page 8)	☐
Is the signature for the Cisco Prime Network Registrar image verified? (see Image Signing, on page 17)	☐

Before You Begin

Verify that you are running a supported operating system and that your environment meets all other current system requirements (see System Requirements, on page 7).

To upgrade the operating system:

1. Use the currently installed Cisco Prime Network Registrar release to complete any configuration changes in progress, so that the existing database is consistent before you perform the upgrade.
2. Back up your database. The installation program tries to detect configuration data from an earlier installation and will upgrade the data.
3. Upgrade your operating system and install the prerequisite software.

Note In this document, when *install-path* is used, it refers to the path where Cisco Prime Network Registrar is installed (that is, /opt/nwreg2/{local | regional}.

Obtaining Cisco Prime Network Registrar License Files

Cisco Prime Network Registrar 11.2 supports both Smart Licensing and traditional licensing. However, it does not support the hybrid model, that is, you can use any one of the license types at a time. Smart Licensing is enabled by default in Cisco Prime Network Registrar. If you want to use traditional licenses, you must first disable Smart Licensing (see the *"Disabling Smart Licensing" section in the Cisco Prime Network Registrar 11.2 Administration Guide*.

Smart Licensing:

When you purchase Cisco Prime Network Registrar 11.2 with Smart License, the licenses get deposited to your Smart Account in the CSSM (or Satellite). You must register Cisco Prime Network Registrar with the CSSM (or Satellite) using web UI or CLI to use these licenses. See the *"Registering Cisco Prime Network Registrar with the CSSM" section in the Cisco Prime Network Registrar 11.2 Administration Guide*.

For a more detailed overview on Cisco Licensing, go to cisco.com/go/licensingguide.

Traditional Licensing:

When you purchase Cisco Prime Network Registrar 11.2, you receive a FLEXlm license file in an e-mail attachment from Cisco, after you register the software.

You must copy the license file to a location which will be accessible during the regional cluster installation before you attempt to install the software. The installation process will ask you for the location of the license file.

To obtain a license file:

1. Read the Software License Claim Certificate document packaged with the software.
2. Note the Product Authorization Key (PAK) number printed on the certificate.
3. Log in to one of the websites described on the certificate, and follow the registration instructions. The PAK number is required for the registration process.

   You should receive the license file through e-mail within one hour of registration.

A typical license file might look like:

```
INCREMENT base-system cisco 11.2 permanent uncounted \

 VENDOR_STRING=<Count>1</Count> HOSTID=ANY \

 NOTICE="<LicFileID>20110919130037832</LicFileID><LicLineID>4</LicLineID> \

 <PAK></PAK><CompanyName></CompanyName>" SIGN=521EA9F0925C
```

Image Signing

Starting from Cisco Prime Network Registrar 11.0, all the Cisco Prime Network Registrar images are signed. The RPM images have implicit signature, whereas the non-RPM images have a separate corresponding signature file. It is recommended that you verify the signature before installing Cisco Prime Network Registrar.

To verify the signature for the RPM images, do the following:

1. Import the GPG public key (**CPNR11-rel.gpg**) into the RPM using the following command. If you do not import the GPG public key into the RPM, then while installing, you will get a warning message.

   ```
   # rpm --import CPNR11-rel.gpg
   ```

2. Run the following command:

   ```
   # rpm -K file.rpm
   file.rpm: rsa sha1 (md5) pgp md5 OK
   ```

   Meaning: Package is signed, correct GPG key is imported

   Output from the above command shows that there are actually three distinct features of the package file that are checked by the -K option (use the -Kv option for verbose):

   - Size message indicates that the size of the packaged files has not changed.
   - PGP message indicates that the digital signature contained in the package file is a valid signature of the package file contents, and was produced by the organization that originally signed the package.

- MD5 message indicates that a checksum contained in the package file and calculated when the package was built, matches a checksum calculated by RPM during verification. Because the two checksums match, it is unlikely that the package has been modified.

OK means that each of these tests was successful.

Other possible outputs of **rpm –K** command are as follows:

- `# rpm –K file.rpm`
  `file.rpm: size md5 OK`

  Meaning: Package not signed.

- `# rpm –K file.rpm`
  `file.rpm: size (PGP) md5 OK (MISSING KEYS)`

  Meaning: Wrong public key.

- `# rpm –K file.rpm`
  `file.rpm:  size PGP MD5 NOT OK`

  Meaning: The RPM file has been changed or tampered with.

- `# rpm –K file.rpm`
  `file.rpm:  RSA sha1 ((MD5) PGP) md5 NOT OK (MISSING KEY)`

  Meaning: Package is signed, but GPG keys are not imported.

To run the signature verification program for the non-RPM images, do the following:

1. Download the verification file (**cpnr_image_verification.gtar.gz**) from the same location as the images. This file contains the public certificate, signature verification script, and the README file.
2. Run the signature verification script using the following command:

   ```
   ./cisco_x509_verify_release.py3 -e CNR_REL_KEY-CCO_RELEASE.pem -i image -s signature -v
    dgst -sha512
   ```

   For example:

   ```
   # ./cisco_x509_verify_release.py3 -e CNR_REL_KEY-CCO_RELEASE.pem -i
   cpnr-local-11.2-1.el8.x86_64_rhel_docker.tar.gz -s
   cpnr-local-11.2-1.el8.x86_64_rhel_docker.tar.gz.signature -v dgst -sha512
   ```

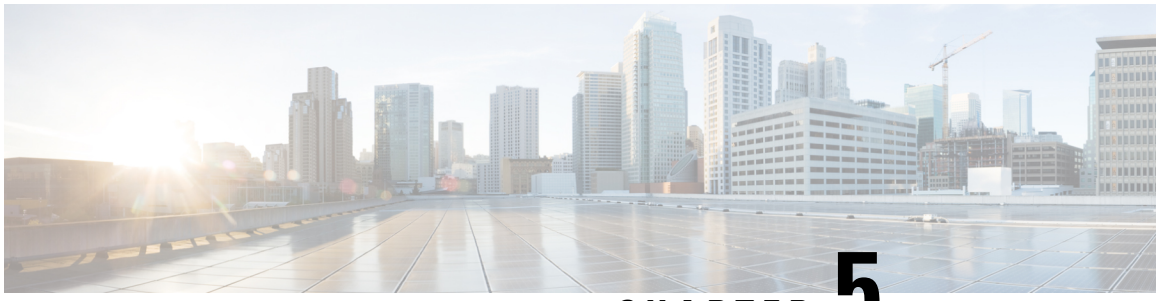
Running Other Protocol Servers

You cannot run the Cisco Prime Network Registrar DNS, CDNS, DHCP, or TFTP servers concurrently with any other DNS, DHCP, or TFTP servers. If there are port conflicts when a server starts, the server will log issues and not function correctly.

If you want to disable a protocol server and prevent the Cisco Prime Network Registrar server from starting automatically after a system reboot, use the **server** {**dns** | **cdns** | **dhcp** | **tftp**} **disable start-on-reboot** command in the CLI.

Backup Software and Virus Scanning Guidelines

If you have automatic backup or virus scanning software enabled on your system, exclude the Cisco Prime Network Registrar directories and their subdirectories from being scanned. If they are not excluded, file locking issues can corrupt the databases or make them unavailable to the Cisco Prime Network Registrar processes. If you are installing in the default locations, exclude the /var/nwreg2 directories and their subdirectories.



CHAPTER 5

Installing and Upgrading Cisco Prime Network Registrar

This chapter contains the following sections:

- Installing Cisco Prime Network Registrar, on page 21
- Upgrade Considerations, on page 24
- Upgrading Cisco Prime Network Registrar, on page 25
- Reverting to an Earlier Product Version, on page 27
- Moving a Local Cluster to a New Machine, on page 28
- Moving a Regional Cluster to a New Machine, on page 29
- Installing Your Own Certificate for Web UI Access, on page 30
- Troubleshooting the Installation, on page 31
- Troubleshooting Local Cluster Licensing Issues, on page 32

Installing Cisco Prime Network Registrar

Starting from Cisco Prime Network Registrar 11.0, there are no configuration questions asked during the installation. Also, it no longer asks for admin credentials and licensing details. You must provide these details when you connect to Cisco Prime Network Registrar for the first time (see Using Cisco Prime Network Registrar, on page 35).

Note that the following paths are used:

- Program files—/opt/nwreg2/{local | regional}

Warning You should NOT add or modify files in the /opt/nwreg2/* directories as these will be overwritten during upgrades or installations. You should add or modify the files only in the /var area. For example, you should add the extensions in the /var/nwreg2/local/extensions area and not in the /opt area.

- Data files—/var/nwreg2/{local | regional}/data
- Log files—/var/nwreg2/{local | regional}/logs
- cnr.conf file—/var/nwreg2/{local | regional}/conf

Also, Cisco Prime Network Registrar 11.2 installation configures the following by default:

- Type of web security—HTTPS only (default port, 8443 for local and 8453 for regional)
- Web services—REST API enabled (on HTTPS port, no separate port)
- Security mode—Required
- SCP Port number—CCM on default port (1234 for local and 1244 for regional)
- Run as root—Cisco Prime Network Registrar always runs as root, but with reduced permissions.
- Type of installation (local, regional, or client-only)—Depends on the RPM kit used. Following RPM kits are available for Cisco Prime Network Registrar 11.2:

Table 4: RPM Kits

	RHEL 8.x/AlmaLinux 8.x
Regional cluster	cpnr-regional-11.2.<minor version>.el8.x86_64.rpm
Local cluster	cpnr-local-11.2.<minor version>.el8.x86_64.rpm
Client only	cpnr-client-11.2.<minor version>.el8.x86_64.rpm

- Create a superuser administrator during first login (see Using Cisco Prime Network Registrar, on page 35).

The following steps are applicable for the new installations. To upgrade from earlier versions of Cisco Prime Network Registrar to 11.2, see Upgrading Cisco Prime Network Registrar, on page 25.

To install Cisco Prime Network Registrar, do the following:

Procedure

Step 1 Log in to the target machine.

Caution
Many distributions of Red Hat or AlmaLinux come with a firewall and connection tracking installed and enabled by default. Running a stateful firewall on the DNS server's operating system causes significant decrease in the server performance. Cisco strongly recommends **NOT** to use a firewall on the DNS server's operating system. If disabling the firewall is not possible, then connection tracking of DNS traffic MUST be disabled. For more information, see the *"DNS Performance and Firewall Connection Tracking" section in the Cisco Prime Network Registrar 11.2 Administration Guide*.

Step 2 Install the OpenJDK 11 if you have not already done so. Use the following command:

```
# yum install java-11-openjdk
```

Note that on some systems, you must use the **dnf install** command.

Step 3 Download the distribution file (RPM kit) from Cisco.com as per your requirement. For the list of RPM kits available for Cisco Prime Network Registrar 11.2, see RPM Kits above.

Cisco Prime Network Registrar 11.2 installs both client and server by default. For client only installation, use the appropriate kit as listed in RPM Kits above.

Note
Choose client only installation in a situation where you want the client software running on a different machine than the protocol servers. You must then set up a connection to the protocol servers from the client.

Step 4 Navigate to the directory in which you saved the downloaded distribution file.

Step 5 Install Cisco Prime Network Registrar using any of the following commands:

```
# yum install filename
```

OR

```
# rpm -i filename
```

OR

```
# dnf install filename
```

Where *filename* is the RPM kit name as listed in Table 4: RPM Kits, on page 22.

Note that RHEL 8.x kits have "el8" which indicates for which Linux version the package is intended.

For example, to install the regional cluster on AlmaLinux 8.8 and later, use any of the following commands:

```
# yum install cpnr-regional-11.2-1.el8.x86_64.rpm
```

OR

```
# rpm -i cpnr-regional-11.2-1.el8.x86_64.rpm
```

OR

```
# dnf install cpnr-regional-11.2-1.el8.x86_64.rpm
```

Note
Since a regional server is required for license management, install the regional server first so that you can register the local to the regional immediately after local installation.

Step 6 Start the Cisco Prime Network Registrar server agent using the following commands (or reboot the system as Cisco Prime Network Registrar is configured to start automatically):

For the local cluster:

```
# systemctl start nwreglocal
```

For the regional cluster:

```
# systemctl start nwregregional
```

During start-up, the /var/nwreg2/{local | regional} folder is created. The keystore file is created in the /var/nwreg2/{local | regional}/conf/priv folder and the keystore details are updated in the cnr-priv.conf file.

If you want to use your own certificate, see Installing Your Own Certificate for Web UI Access, on page 30.

Step 7 Verify the status of the Cisco Prime Network Registrar servers. Run either of the following commands:

`# ./cnr_status` (available in the *install-path*/usrbin directory)

OR

`# systemctl status nwreglocal` (for the local cluster)

`# systemctl status nwregregional` (for the regional cluster)

After the installation is complete, follow the steps in Using Cisco Prime Network Registrar, on page 35 to start using Cisco Prime Network Registrar. Make sure NOT to modify or add anything in the /opt folder as these files may be overwritten by a future upgrade. You can make changes in the /var folder.

Upgrade Considerations

Cisco Prime Network Registrar 11.2 supports direct upgrades from 9.0 and later.

Cisco Prime Network Registrar 11.2 can run on RHEL 8.x/AlmaLinux 8.x. If you are using an earlier version of the operating system, you will first need to upgrade your system to a supported version.

When you install the software, the installation program automatically detects an existing version and upgrades the software to the latest release. Archive the existing Cisco Prime Network Registrar data. If the upgrade fails and fails to start, then you should recover the backup that you made (and perhaps install the old Cisco Prime Network Registrar version). You can also find the backup of the data in the /var/nwreg2/{local | regional} directory, called **upgrade-backup-***date***.tar.gz**. If you did not create your own backup, you can use this backup to restore the databases.

The eventstore is no longer used to track the pending DNS updates. DHCPv4 lease objects are used for this purpose, similar to DHCPv6 DNS updates where leases are used. Therefore, when upgrading from Cisco Prime Network Registrar 10.x or earlier, it is best to upgrade when the DNS update backlog is low as any pending DHCPv4 DNS updates will be lost. The DHCP server logs the DNS update events that it drops using the log message 19669. This will report the lease, pending action, FQDNs, and DNS Update Configuration objects involved for each pending event. These are only logged once as the server removes these events from the eventstore.You can determine the DNS update backlog using the **dhcp getRelatedServers** command and by examining the "requests" count for the DNS servers.

Note If previous version is using smart license, make sure all the licenses are in-compliance before performing major or minor version upgrade. If any of the license is out-of-compliance, major or minor version upgrade fails.

Using Smart Licensing

Cisco Prime Network Registrar 11.x regional, working in Smart License mode, does not support pre-11.0 local clusters. Hence, you must perform the following steps to move to Smart Licensing:

Procedure

Step 1 Upgrade the Cisco Prime Network Registrar regional cluster to 11.x and disable Smart Licensing (post upgrade). To disable Smart Licensing, see the *"Disabling Smart Licensing" section in the Cisco Prime Network Registrar 11.2 Administration Guide*.

Step 2 Load the required traditional licenses on the Cisco Prime Network Registrar 11.x regional cluster.

Step 3 Re-register or resync all the local clusters with the upgraded regional cluster.

Warning
You must upgrade the Cisco Prime Network Registrar 10.x local clusters to 10.1.1 (or later version) before registering them with the 11.x regional cluster. The 10.x local clusters with lower than 10.1.1 version, have issues registering with the 11.x regional cluster.

Step 4 Upgrade all the local clusters to 11.x as per your schedule.

Step 5 After all the clusters are upgraded to 11.x, you can Enable Smart Licensing on regional if you want to move to Smart Licensing. You should perform this step only if you have required licenses in your Smart Account on the CSSM or Satellite. To enable Smart Licensing, see the *"Enabling Smart Licensing" section in the Cisco Prime Network Registrar 11.2 Administration Guide*.

Upgrading Cisco Prime Network Registrar

One major change introduced starting with Cisco Prime Network Registrar 11.0 is to better separate the distributed files (that is, those installed by the RPM) from those that are data and configuration files specific to your installation. Basically, the /opt/nwreg2 area should not include files that are not provided as part of the installation. Everything that is specific to your installation, should now be in the /var/nwreg2 area.

If you used the default paths when previously installing earlier versions of Cisco Prime Network Registrar, the following files will be automatically relocated the first time you start Cisco Prime Network Registrar after installing Cisco Prime Network Registrar 11.2:

- /opt/nwreg2/{local | regional}/conf/cnr.conf will be moved to /var/nwreg2/{local | regional}/conf
- /opt/nwreg2/{local | regional}/conf/priv (and its contents) will be moved to /var/nwreg2/{local | regional}/conf/priv
- /opt/nrweg2/{local | regional}/conf/cert (and its contents) will be moved to /var/nrweg2/{local | regional}/conf/cert
- Any paths in the cnr.conf and cnr-priv.conf will be updated to reflect this move

If the Cisco Prime Network Registrar data area is not in /var/nwreg2/{local | regional}/data, similar moves are done but the resulting paths will use a new conf directory in the parent directory of the data directory. Or, the files may be left where they are.

Note that after upgrading from earlier versions to Cisco Prime Network Registrar 11.2, the following changes will also occur in addition to the above mentioned changes:

- The nwreglocal.env (for local) or nwregregional.env (for regional) file in the /usr/lib/systemd/system directory is used instead of the /opt/nwreg2/{local | regional}/bin/cnr.env file. Therefore, after installing (before starting Cisco Prime Network Registrar), you may need to review if cnr.env changes (such as for LD_LIBRARY_PATH for extensions) need to be applied to the new .env file.
- The web UI keystore is used if one exists or a new one is generated. The existing priv/cnr-priv.conf is used and relocated to /var/nwreg2/{local | regional}.
- HTTPS is used for web UI and REST instead of HTTP. If no port was previously configured for HTTPS, the default (regional | local) ports are used.

- If REST was disabled in the previous install, it gets enabled after the upgrade. If you want to disable REST API, follow the steps in Disabling REST API, on page 39 after upgrading. If REST was previously using a different port than HTTPS, it is no longer supported and the same port is used for HTTPS (web UI) and REST.

Note From Cisco Prime Network Registrar 10.1 onwards, the keystore password is encrypted by default. Therefore, you do not have to encrypt the keystore password if you are upgrading from 10.1 to 11.0 and later. However, if you are upgrading from pre-10.1 version to 11.0 and later, then you must encrypt the keystore password manually.

To generate the encrypted password, use the encrypt script (**encrypt -s** <*plain-text password*>) present in the *install-path*/usrbin directory. You must update this encrypted password in server.xml and after making the change, you must restart Cisco Prime Network Registrar.

To upgrade to Cisco Prime Network Registrar 11.2:

Procedure

Step 1 Ensure that your environment meets the current system requirements (see System Requirements, on page 7).

Step 2 Remove the existing installation using the procedure described in Uninstalling Cisco Prime Network Registrar, on page 41. Ensure NOT to do the cleanup operations mentioned at the end (that is, retain the data, cnr.conf, and so on).

Step 3 If the old cnr.conf is in *install-path*/conf, you can upgrade without doing anything. If the old cnr.conf is elsewhere, then create a cnr.conf file in the *install-path*/conf directory that contains the following line:

cnr.confdir=*location of the old cnr.conf file*

Step 4 To avoid issues with Java upgrades, we recommend that you set JAVA_HOME variable correctly in your environment. Starting Cisco Prime Network Registrar 11.1.1, if JAVA_HOME is not set the rpm installer will try to search the best possible java by looking into packages in the environment and set that. To further reduce issues with Java upgrades, we highly recommend that you edit your cnr.conf file to replace the cnr.java-home entry path with /usr/bin/java (if this is the path that has the version of Java specified in cnr.conf). You can test this by doing:

```
/usr/bin/java -version
```

and

```
cnr.java-home-path/bin/java -version
```

If the two report the same result, you should change the cnr.java-home path to specify /usr/bin/java. If you do this, updates to Java will not require you to update the cnr.java-home path.

Step 5 Install Cisco Prime Network Registrar 11.2. For installation instructions, see Installing Cisco Prime Network Registrar, on page 21.

Step 6 Start the Cisco Prime Network Registrar server agent using the following commands:

- For the local cluster:

```
# systemctl start nwreglocal
```

- For the regional cluster:

```
# systemctl start nwregregional
```

The upgrade process may take some time depending on the size of the configuration and lease/resource record data, and the version from which you are upgrading. You can view the status using the **systemctl status nwreglocal** (for the local cluster) or **systemctl status nwregregional** (for the regional cluster) command. If this shows "trampoline startup, local mode" (for the local cluster) or "trampoline startup, regional mode" (for the regional cluster), it indicates that the services are up or have started. Follow the steps in Using Cisco Prime Network Registrar, on page 35 to start using Cisco Prime Network Registrar.

If the upgrade fails, you can revert to the earlier Cisco Prime Network Registrar version. For details about reverting to the earlier version, see the Reverting to an Earlier Product Version, on page 27.

Reverting to an Earlier Product Version

The Cisco Prime Network Registrar installation program archives the existing product configuration and data when you upgrade to a newer version. If the upgrade process fails, use the following procedure to revert to the earlier product version and configuration:

Caution To complete this process, you must have access to the product installer and license key or license file for the earlier Cisco Prime Network Registrar version. Any attempt to proceed otherwise may destabilize the product.

If the installer had successfully performed the upgrade but you want to roll back to the earlier version at some later point, this procedure can result in network destabilization and data loss; for example, you will lose updates made to the Cisco Prime Network Registrar database after the upgrade, including DHCP lease data and DNS dynamic updates.

Procedure

Step 1 Ensure that the archive file **upgrade-backup-***date***.tar.gz** is available in the /var/nwreg2/{local | regional} directory.

Step 2 Uninstall Cisco Prime Network Registrar using the procedure described in the Uninstalling Cisco Prime Network Registrar, on page 41.

Step 3 Other than the contents of the archive file, delete any remaining files and directories in the Cisco Prime Network Registrar installation paths.

Step 4 Restore the backup (either the one that you created or the archive file that is created in Step 7).

Step 5 Reinstall the original version of Cisco Prime Network Registrar. Ensure that you follow the reinstallation procedure described in *Cisco Prime Network Registrar Installation Guide* that is specific to the original product version.

Step 6 After the installation ends successfully, stop the Cisco Prime Network Registrar server agent:

- For the local cluster:

```
# systemctl stop nwreglocal
```

- For the regional cluster:

```
# systemctl stop nwregregional
```

Step 7 Extract the contents of the backup file to the reinstalled version of Cisco Prime Network Registrar.

a) Change to the root directory (/) of the filesystem.
b) Using the fully qualified path to the archive directory, extract the archive.

- Change to the root directory of the filesystem using **cd /**.
- Using the fully qualified path to the archive directory containing the **upgrade-backup-***date***.tar.gz** file, extract the archive.

```
tar xzf /var/nwreg2/{local | regional}/upgrade-backup-date.tar.gz
```

The above command creates the **opt** and **var** folders. The **opt** folder contains only the conf directory.

Step 8 Verify if the previous configuration, including scopes and zones, is intact.

Moving a Local Cluster to a New Machine

Before you begin, ensure that the new machine meets the current system requirements (see System Requirements, on page 7).

The following steps can be used to upgrade the cluster to a later Cisco Prime Network Registrar version (that is, it is not required that the same version of Cisco Prime Network Registrar be installed in Step 5; a later version that supports upgrades from the earlier version can be installed).

To move an existing Cisco Prime Network Registrar installation to a new machine on the same platform:

Procedure

Step 1 Stop the server agent on the old local server.

```
# systemctl stop nwreglocal
```

Step 2 Tar up /var/nwreg2/local, except /var/nwreg2/local/tomcat. You can also skip the /var/nwreg2/local/data.bak if you prefer not to copy over the latest backup.

Ensure that the cnr.conf file is added in tar file. Copy the cnr.conf along with /var/nwreg/local/data/, so that new cluster will use the same ‘cnr.localhost-uuid’ from the conf file.

Note
It becomes more important if DHCP server is using localhost-uuid as server-duid. Because if this duid changes, DHCP server will not renew existing DHCPv6 leases. This is not applicable if the DHCP servers are using previously generated (pre 9.2) DHCPv6 server-identifier (DUID) from database.

Step 3 Copy the tar file to the new server, and untar the files into the same location (/var/nwreg2/local). Ensure that there is no /var/nwreg2/local/tomcat directory (if so, remove it and anything in it).

Note
The Step 2 and Step 3 apply to Cisco Prime Network Registrar 11.0 and later. For earlier releases, refer to the documentation of that version.

Step 4 Move the /usr/lib/systemd/system/nwreglocal.env file to the new system.

Step 5 Install Cisco Prime Network Registrar (local cluster) on the new server. The installation will detect an upgrade and will do so based on the copied data.

This procedure preserves your original data on the old machine.

Re-apply any custom configuration changes (such as those outlined in Enhancing Security for Web UI, on page 77) after the installation.

Step 6 Log in to the web UI and navigate to the **Licenses** page under the **Administration** menu to open the List Licenses page.

Step 7 Edit the regional server information as necessary. Ensure that the regional server information provided is where you would like to register your new machine.

Step 8 Click the **Register** button to register with the regional server.

Step 9 If the IP address of the machine has changed, you may need to also update the failover/HA DNS partner to assure it also has the new address of the server. For DHCP, you may need to update the relay agent helper addresses and DNS server addresses.

Note
An address change can prevent DHCP clients from renewing promptly (they may not be able to renew until they reach the rebinding time) and can prevent DNS queries from being resolved until clients or other DNS servers receive the updated information.

Moving a Regional Cluster to a New Machine

License management is done from the regional cluster when Cisco Prime Network Registrar is installed. The regional server is installed first and all licenses are loaded in the regional server. When the local cluster is installed, it registers with the regional server to obtain its license.

When you want to move a regional cluster to a new machine, you need to back up the data on the old regional cluster and copy the data to the same location on the new machine.

Note When the regional server goes down or is taken out of service, the local cluster is not aware of this action. If the outage lasts for less than 24 hours, it results in no impact on the functioning of the local clusters. However, if the regional cluster is not restored for more than 24 hours, the local cluster may report warning messages that the local cluster is not properly licensed (in the web UI, CLI, or SDK). This does not impact the operation of the local clusters and the local clusters continue to work and service requests.

The following steps can be used to upgrade the cluster to a later Cisco Prime Network Registrar version (that is, it is not required that the same version of Cisco Prime Network Registrar be installed in Step 5; a later version that supports upgrades from the earlier version can be installed).

To move an existing Cisco Prime Network Registrar installation to a new machine:

Procedure

Step 1 Stop the server agent on the old regional server:

```
# systemctl stop nwregregional
```

Step 2 Tar up /var/nwreg2/regional, except /var/nwreg2/regional/tomcat. You can also skip the /var/nwreg2/regional/data.bak if you prefer not to copy over the latest backup.

Step 3 Copy the tar file to the new server, and untar the files into the same location (/var/nwreg2/regional). Ensure that there is no /var/nwreg2/regional/tomcat directory (if so, remove it and anything in it).

Note
The Step 2 and Step 3 apply to Cisco Prime Network Registrar 11.0 and later. For earlier releases, refer to the documentation of that version.

Step 4 Move the /usr/lib/systemd/system/nwregregional.env file to the new system.

Step 5 Install Cisco Prime Network Registrar (regional cluster) on the new server. For more information, see Installing Cisco Prime Network Registrar, on page 21.

The installation will detect an upgrade and will do so based on the copied data. This procedure preserves your original data from the old regional server.

Re-apply any custom configuration changes (such as those outlined in Enhancing Security for Web UI, on page 77) after the installation.

Note
When you install Cisco Prime Network Registrar on the new machine, you must choose the data directory on which you have copied the data from the old regional server.

Step 6 Start the Cisco Prime Network Registrar web UI or CLI. For more information, see Using Cisco Prime Network Registrar, on page 35.

Step 7 Log in as superuser to the CLI for the new regional cluster.

Step 8 To list the local clusters, use the following command:

```
nrcmd-R> cluster listnames
```

Step 9 To synchronize the data as well as the license information, use the following command:

```
nrcmd-R> cluster cluster-name sync
```

Installing Your Own Certificate for Web UI Access

If you want to use your own certificate for web UI access, do the following:

Procedure

Step 1 Create a keystore file with self-signed certificate by using **openssl** or **keytool**. Use the utility to define a self-signed certificate, or to request and later import a certificate from an external signing authority:

- To create a keystore file containing a self-signed certificate, run this command and respond to the prompts:

```
> keytool -genkey -alias tomcat -keyalg RSA -keystore k-file
```

```
Enter keystore password: password

What is your first and last name? [Unknown]: name

What is the name of your organizational unit? [Unknown]: org-unit

What is the name of your organization? [Unknown]: org-name

What is the name of your City or Locality? [Unknown]: local

What is the name of your State or Province? [Unknown]: state

What is the two-letter country code for this unit? [Unknown]: cc

Is CN=name, OU=org-unit, O=org-name, L=local, ST=state, C=cc correct? [no]: yes

Enter key password for <tomcat> (RETURN if same as keystore password):
```

Note
You must use 128-bit SSL to disable weak ciphers in the web UI. For more information, see Enhancing Security for Web UI, on page 77.

- To create a Certificate Signing Request (CSR) that you will submit to the Certificate Authority (CA) when you request a certificate, create the keystore file as in the previous step, then execute this command:

  ```
  > keytool -certreq -keyalg RSA -alias tomcat -file certreq.cer -keystore k-file
  ```

  Submit the resulting certreq.cer file to the CA. Once you receive the certificate from the CA, first download the Chain Certificate from the CA, then import the Chain Certificate and your new Certificate into the keystore file, as follows:

  ```
  > keytool -import -alias root -keystore k-file -trustcacerts -file chain-cert-file

  > keytool -import -alias tomcat -keystore k-file -trustcacerts -file new-cert-file
  ```

  For details on the **keytool** utility, see the documentation at the Java website of Oracle. For details on the **keystore** file and Tomcat, see the documentation at the website of the Apache Software Foundation.

- To create a self-signed certificate using openssl, use the following command:

  ```
  > openssl req -x509 -newkey rsa:4096 -keyout key.pem -out cert.pem -days 365
  ```

For more information on certificate management in Cisco Prime Network Registrar, see the *"Certificate Management" section in the Cisco Prime Network Registrar 11.2 Administration Guide*.

Step 2 Edit the cnr-priv.conf file (in /var/nwreg2/{local | regional}/conf/priv) as required to point to the new keystore and then specify the password (encrypted). To generate the encrypted password, use the encrypt script (**encrypt -s** <*plain-text password*>) present in the *install-path*/usrbin directory.

Step 3 Restart Cisco Prime Network Registrar.

Whenever Cisco Prime Network Registrar is restarted, the keystore details are applied to the Tomcat configuration.

Troubleshooting the Installation

The log directory is set to these locations by default:

- Local cluster: /var/nwreg2/local/logs
- Regional cluster: /var/nwreg2/regional/logs

If the installation or upgrade does not complete successfully, then:

- Check the contents of the above log files to help determine what might have failed. Some examples of possible causes of failure are:
  - An incorrect version of Java is installed.
  - Insufficient disk space is available.
  - Inconsistent data exists for an upgrade.
- Check the status of the service using the following commands:
  - For the local cluster:

    ```
    # systemctl status -l nwreglocal.service
    ```
  - For the regional cluster:

    ```
    # systemctl status -l nwregregional.service
    ```
- Check the systemd journal using the following commands:
  - For the local cluster:

    ```
    # journalctl -u nwreglocal --since=today
    ```
  - For the regional cluster:

    ```
    # journalctl -u nwregregional --since=today
    ```

  You can change the time interval used in since. For more details, use the **man journalctl** command.

Troubleshooting Local Cluster Licensing Issues

If your regional cluster and local cluster are located in isolated networks, are separated by a firewall, or the time skew between the regional and local clusters is more than five minutes, then the local cluster may be unable to register with the regional server. The firewall may block the return connection used to validate the local cluster admin credentials that are sent from the local cluster to the regional cluster.

To register a local cluster with the regional cluster:

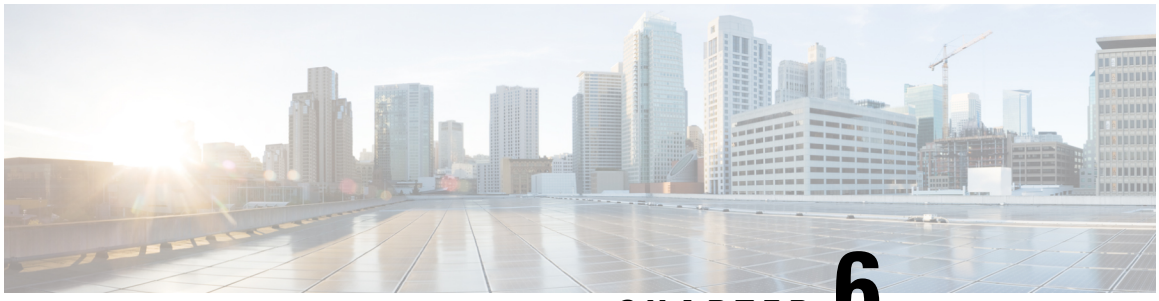
Procedure

Step 1 Install Cisco Prime Network Registrar (local cluster) on the server and create the admin user for the local cluster. For more information, see Installing and Upgrading Cisco Prime Network Registrar, on page 21.

When you try to log in for the first time (either in the web UI or CLI) after installing Cisco Prime Network Registrar on the local cluster, you will be prompted to create the superuser and to register with a regional cluster.

Step 2 Log in to the regional cluster and add the new local cluster to the regional cluster with the admin credentials. For more information, see the *"Adding Local Clusters" section in the Cisco Prime Network Registrar 11.2 Administration Guide*.

Step 3 To synchronize the data as well as the license information, click the **Resynchronize** icon.



CHAPTER 6

Next Steps

This chapter contains the following sections:

- Configuring Cisco Prime Network Registrar, on page 35
- Using Cisco Prime Network Registrar, on page 35
- Starting and Stopping Servers, on page 37
- Server Event Logging, on page 38
- Disabling REST API, on page 39

Configuring Cisco Prime Network Registrar

After installing Cisco Prime Network Registrar, you can perform the following tasks:

- Get started with Cisco Prime Network Registrar—See Cisco Prime Network Registrar 11.2 Quick Start Guide.
- Set up DHCP addresses, DHCP failover, and DNS update—See Cisco Prime Network Registrar 11.2 DHCP User Guide.
- Set up Authoritative and Caching DNS services—See Cisco Prime Network Registrar 11.2 Caching and Authoritative DNS User Guide.
- Perform administrative tasks, such as local and regional administration, set up Cisco Prime Network Registrar virtual appliance, and so on—See Cisco Prime Network Registrar 11.2 Administration Guide.
- Configure and manage Cisco Prime Network Registrar via CLI—See Cisco Prime Network Registrar 11.2 CLI Reference Guide.
- Configure and manage Cisco Prime Network Registrar via REST API—See Cisco Prime Network Registrar 11.2 REST APIs Reference Guide.

Using Cisco Prime Network Registrar

To administer the local and regional clusters that you have installed, you must create a superuser administrator and enter the appropriate license information. To do this, follow the below steps first time when you connect to Cisco Prime Network Registrar:

Procedure

Step 1 Start the Cisco Prime Network Registrar web UI or CLI:

- To access the web UI, open the web browser and use the HTTPS (secure login) website:

  `https://hostname:https-port`

  where:

  - *hostname* is the actual name of the target host.
  - *https-port* is the default HTTPS port (8443 for local and 8453 for regional).

- To start the CLI, launch nrcmd by entering:

  install-path`/usrbin/nrcmd -R -N` *username* `-P` *password*

  Specify the username and password of the administrator account that you want to create. You will be asked to confirm the password if the superuser administrator account needs to be created (during the first login).

  Note
  Specify -R only if you are connecting to a regional cluster.

Step 2 Create a superuser administrator by providing the username and password.

- Web UI—Enter the username and password in the **Admin** and **Password** fields respectively. Then, click the **Add** button.

Step 3 In Cisco Prime Network Registrar 11.2, Smart Licensing is enabled by default. Click the **Configure Smart Licensing** link in the alert window to open the Smart Software Licensing page and set up Smart Licensing. For details, see the *"Use Cisco Smart Licensing" section in the Cisco Prime Network Registrar 11.2 Administration Guide*.

If you want to use traditional licensing, then you must disable Smart Licensing first (see the *"Disabling Smart Licensing" section in the Cisco Prime Network Registrar 11.2 Administration Guide*). Then, click **Use Traditional Licensing** and enter the license information as follows:

- Web UI—Click **Browse** to navigate to the license file.
- CLI—Enter an absolute or relative path for the license filename, as follows:

  `nrcmd> license create` *filename*

Note
You must add the licenses in the regional cluster which means the regional should be installed first. The local cluster has to be registered with the regional cluster at the time of your first login. You can choose the services (dhcp, dns, and cdns) for the local based on the licenses added in the regional cluster.

Step 4 Enter the superuser username and password (created in Step 2) to log in to web UI and CLI.

You may create other administrator accounts to perform certain functions based on the assigned roles. For more information, see the *"Managing Administrators" chapter in the Cisco Prime Network Registrar 11.2 Administration Guide.*

Starting and Stopping Servers

If the installation completed successfully and you enabled the servers, the Cisco Prime Network Registrar DNS and DHCP servers start automatically each time you reboot the machine.

For the TFTP server, you must use the following Cisco Prime Network Registrar CLI command to enable it to restart on bootup:

```
nrcmd> tftp enable start-on-reboot
```

All servers in the cluster are controlled by the Cisco Prime Network Registrar regional or local server agent. You can stop or start the servers by stopping or starting the server agent.

For details on stopping and starting servers, see the *Cisco Prime Network Registrar 11.2 Administration Guide*.

The Cisco Prime Network Registrar servers automatically start up after a successful installation or upgrade. You do not need to reboot the system.

To start and stop servers:

Procedure

Step 1 Log in as superuser.

Step 2 Start the server agent by running the nwreglocal or nwregregional script with the *start* argument:

For the local cluster:

```
# systemctl start nwreglocal
```

For the regional cluster:

```
# systemctl start nwregregional
```

Step 3 Verify the status of the Cisco Prime Network Registrar servers. Run either of the following commands:

`# ./cnr_status` (available in the *install-path*/usrbin directory)

OR

`# systemctl status nwreglocal` (for the local cluster)

`# systemctl status nwregregional` (for the regional cluster)

Step 4 Stop the server agent by running the nwreglocal or nwregregional script with the stop argument:

For the local cluster:

```
# systemctl stop nwreglocal
```

For the regional cluster:

```
# systemctl stop nwregregional
```

Starting or Stopping Servers Using the Local Web UI

To start or stop servers in the local web UI:

Procedure

Step 1 From **Operate** menu, choose **Manage Servers** under the **Servers** submenu to open the Manage Servers page.

Step 2 To start or stop the DHCP, DNS, CDNS, TFTP, or SNMP servers, select the server in the Manage Servers pane and do any of the following:

- Click the **Start Server** button to start the server.
- Click the **Stop Server** button to stop the server.

Step 3 To reload the server, click the **Restart Server** button.

Starting and Stopping Servers Using the Regional Web UI

To start or stop servers in the regional web UI:

Procedure

Step 1 From **Operate** menu, choose **Manage Servers** under the **Servers** submenu to open the Manage Servers page.

Step 2 To start or stop the SNMP servers, select the server in the Manage Servers pane and do any of the following:

- Click the **Start Server** button to start the server.
- Click the **Stop Server** button to stop the server.

Step 3 To reload the server, click the **Restart Server** button.

Server Event Logging

System activity begins logging when you start Cisco Prime Network Registrar. The server maintains all the logs by default in the following directories:

- Local cluster: /var/nwreg2/local/logs
- Regional cluster: /var/nwreg2/regional/logs

To monitor the logs, use the **tail -f** command.

Disabling REST API

When you install Cisco Prime Network Registrar 11.2 or upgrade to 11.2 from previous versions, the REST API gets enabled by default. If you want to disable REST API, then do the following:

Local and Regional Advanced Web UI

Procedure

Step 1 From the **Operate** menu, choose **Manage Servers** under the **Servers** submenu to open the Manage Servers page.

Step 2 Click **CCM** in the Manage Servers pane on the left. The Edit Local CCM Server page appears. This page displays all the CCM server attributes.

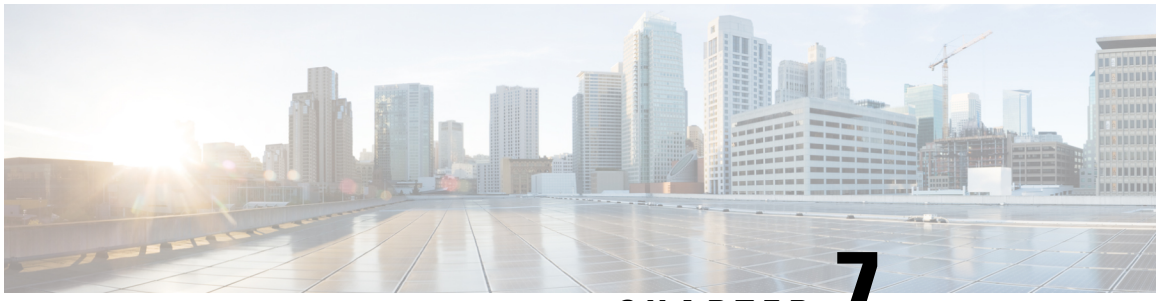
Step 3 Under the **Control Settings** section, set the *is-rest-enabled* attribute value to **false** to disable REST API.

Step 4 Click **Save** to save the changes.

CLI Commands

Use **ccm disable is-rest-enabled** to disable REST.

Use **ccm enable is-rest-enabled** to enable REST.



CHAPTER 7

Uninstalling Cisco Prime Network Registrar

You must have administrator or superuser privileges to uninstall Cisco Prime Network Registrar, just as you must to install it.

To back up your database before uninstalling Cisco Prime Network Registrar, see the *Cisco Prime Network Registrar 11.2 Administration Guide* for the procedure.

Note Uninstallation stops the Cisco Prime Network Registrar server agents first. If you find that the server processes are not shutting down, see Starting and Stopping Servers, on page 37.

• Uninstalling Cisco Prime Network Registrar, on page 41

Uninstalling Cisco Prime Network Registrar

To uninstall Cisco Prime Network Registrar, run any of the following commands:

```
# rpm -e kitname
```

OR

```
# yum remove kitname
```

OR

```
# dnf remove kitname
```

where *kitname* is either cpnr-local, cpnr-regional, or cpnr-client.

For example, to uninstall the regional cluster, use any of the following commands:

```
# rpm -e cpnr-regional
```

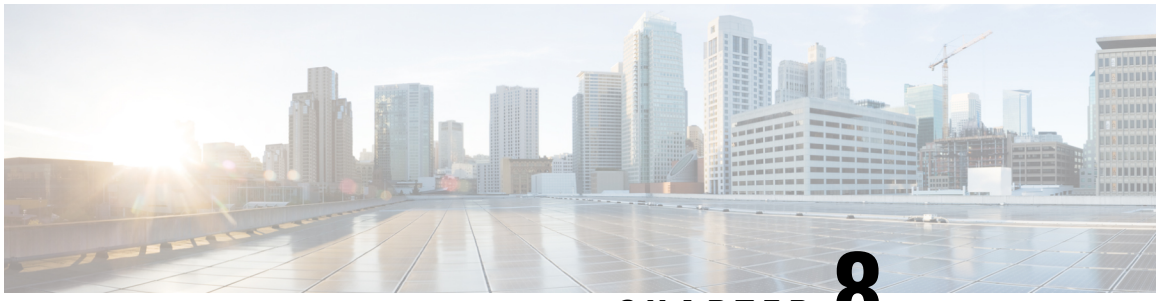
OR

```
# yum remove cpnr-regional
```

OR

```
# dnf remove cpnr-regional
```

Certain configuration and data files that are created during installation and operation remain deliberately after uninstallation. Optionally, delete the data files that are associated with Cisco Prime Network Registrar, as mentioned in the instructions at the end of the uninstallation message.



CHAPTER 8

Cisco Prime Network Registrar Virtual Appliance

The Cisco Prime Network Registrar virtual appliance includes all the functionality available in a version of Cisco Prime Network Registrar 11.2 installed on any Linux operating system.

This chapter describes how to install Cisco Prime Network Registrar virtual appliance and includes the following sections:

- System Requirements, on page 43
- Installing and Upgrading Cisco Prime Network Registrar Virtual Appliance, on page 44
- Upgrading the Cisco Prime Network Registrar Virtual Appliance, on page 50
- Next Steps: Cisco Prime Network Registrar Virtual Appliance, on page 52
- Building Your Own Base Images for Deploying Virtual Machines, on page 55
- Disabling dnsmasq and libvirtd, on page 55

System Requirements

There are two kits that can be used to install the virtual appliance:

- An OVA which runs on VMware ESXi 7.0
- A cloud image which can be deployed to OpenStack

These kits are effectively identical, and in this guide, when the OVA is discussed, the discussion applies to both the kits unless otherwise noted.

Each of these kits were created to require limited resources: 4 virtual CPU, 8 GB main memory, 6 GB swap partition, and a 42 GB system partition with 39 GB available (free). The total disk storage required is 48 GB. You will almost certainly want to increase the size of the system disk, and giving the virtual appliance additional virtual CPUs can increase the performance considerably. You should ensure that sufficient resources are available on the host that you are targeting for the deployment to meet these requirements.

Note that if you do not configure the virtual CPUs correctly, you may run into Non-Uniform Memory Access (NUMA) performance issues. To avoid this issue, assign less or equal amount of vCPUs to virtual machines than the total number of physical cores of a single CPU Socket (stay within 1 NUMA Node). Also, avoid single virtual machine consuming more vCPUs than the single NUMA node or it might be scheduled across multiple NUMA nodes causing memory access degradation.

You must increase the resources used by the virtual appliance or it will not function successfully. There are two different regimes: running a local cluster, or running a regional cluster and local cluster on the same

machine. The recommendations below are for running the virtual appliance(s) on a Jumpstart, but these are also useful starting points for any local or regional cluster deployment. For a local cluster:

- CPU: 1 socket, 8 CPUs
- Memory: 12 GB
- Disk: 100 GB or greater

For a regional cluster running on the same Jumpstart as a local cluster:

- CPU: 1 socket, 7 CPUs
- Memory: Minimum 8 GB
- Disk: 48 GB

You may need substantially more disk space than listed above based on the size of your deployment. You can increase the disk space by resizing the allocated disk and rebooting the appliance.

Installing and Upgrading Cisco Prime Network Registrar Virtual Appliance

You can deploy the virtual appliance in any of the three environments: VMware ESXi 7.0, or OpenStack. After discussing the information that you will need to determine for any deployment, the individual environments are discussed in detail.

Preparing to Deploy the Cisco Prime Network Registrar Virtual Appliance

In order to deploy the Cisco Prime Network Registrar virtual appliance and configure its network connection, you have to answer several questions. Some of these questions concern the networking environment in which the virtual appliance is being deployed, and some of them concern values which are unique to the particular virtual appliance being deployed.

The questions that are unique to the installation of this particular virtual appliance are listed below. You must decide on answers to these questions before you deploy the virtual appliance.

- A virtual machine name for the deployed virtual appliance.
- A root password for the underlying AlmaLinux operating system.
- An IPv4 address for the virtual appliance.
- A DNS name associated with the IPv4 address of the virtual appliance.
- A username and password for the initial administrator account for the Cisco Prime Network Registrar application.

Note From Cisco Prime Network Registrar 9.1 and later, you can copy an existing VM to create a new local cluster (snapshot), you must generate a new UUID and re-register it with the regional cluster to avoid duplication of UUID, see the *"Generating new UUID" section in the Cisco Prime Network Registrar 11.2 Administration Guide*.

The questions concerning the networking environment are as follows. The answers to these questions are not unique to the virtual appliance, but are instead values that are determined by the environment in which you will deploy the virtual appliance:

- The network mask associated with the IP address of the virtual appliance itself
- The default gateway address for the virtual appliance
- The IP address of at least one DNS server that can be accessed by the virtual appliance, although it is best if you have the IP addresses of two DNS servers to provide additional availability.
- Any proxy values necessary for the virtual appliance to access the Internet (if you want the virtual appliance to have access to the Internet).
- If this is a local cluster installation, you will need to determine the IP address of the Cisco Prime Network Registrar regional cluster to which this local cluster will connect in order to receive its license information. If this is a regional cluster installation, you can ignore this requirement.

Deploying the Regional Cluster OVA or Local Cluster OVA on VMware

The Cisco Prime Network Registrar virtual appliance is supported for production use on VMware ESXi 7.0 and can be accessed or managed using the VMware vSphere client. The Cisco Prime Network Registrar virtual appliance is made available in an Open Virtual Appliance (OVA) package.

The VMware vSphere client can be connected directly to your ESXi installation, or it can be connected to a vCenter server which in turn is connected to your vSphere installation. Connecting through vCenter provides a number of capabilities that connecting directly to ESXi does not. If a vCenter server is available and associated with the ESXi installation, it should be used.

To install the Cisco Prime Network Registrar virtual appliance, you must first download the correct installation file. There are two files available: a regional virtual appliance and a local cluster virtual appliance. Each of these virtual appliances is provided as a .ova file.

The names are:

- **cpnr_version_local.ova** for the local virtual appliance
- **cpnr_version_regional.ova** for the regional virtual appliance

Download the virtual appliance of your choice. Every Cisco Prime Network Registrar local cluster installation must connect to a Cisco Prime Network Registrar regional cluster in order to receive the necessary license information required to operate. Thus, before you install a Cisco Prime Network Registrar local virtual appliance, you must identify the IP address of the regional cluster to which it will connect to receive the license information.

Using vSphere, connect directly to the ESXi installation or the vCenter server, and select the ESXi installation where the OVA is to be deployed.

If you have a vCenter server available, you can connect the ESXi hypervisor to your existing vCenter server and manage it through that vCenter server. Managing all your VMware hypervisors through a common vCenter server provides many benefits.

The screens that you see while managing the ESXi hypervisor with a vSphere client through a vCenter server are different from the screens that you see while connecting the vSphere client directly to the ESXi hypervisor. You can see additional screens if connected through vCenter server. These screens do not actually provide any benefit for the operations in which you will engage to deploy the Cisco Prime Network Registrar virtual appliance. The benefits to using the vCenter server approach come after the initial deployment of the virtual appliance.

To deploy a regional cluster OVA or local cluster OVA:

Procedure

Step 1 From vSphere menu, choose **File > Deploy OVF Template**.

The Deploy OVF Template Source window appears.

Step 2 To deploy the OVA file, click **Browse** and navigate to select the OVA file (.ova) available on the local machine where vSphere is running.

Note
You cannot browse for URLs and you must enter the full path to the file.

Step 3 Click **Next**.

The OVF Template Details window appears. It displays the product name, the size of the OVA file, and the amount of disk space that needs to be available for the virtual appliance.

Step 4 Verify the OVA template details and click **Next**.

Step 5 Provide a name to the new virtual appliance and click **Next**.

Note
You must enter the same name while configuring the virtual appliance, so make sure you remember this name.

The Deployment Options window appears.

The **Thin** provisioned format is selected by default. You should select the **Thick** regardless of the default value.

Step 6 Click **Next** to continue.

Note
The virtual appliance is only supported when deployed with thick provisioning.

Step 7 To map the networks used in this OVA template to the networks in your inventory, select the current destination network, and choose the destination network from the **Destination Networks** drop-down list. Click **Next**.

The Ready to Complete window appears.

Step 8 Click **Finish** to begin deployment of the OVF Template.

Booting and Configuring Cisco Prime Network Registrar Virtual Appliance

To boot and then configure the Cisco Prime Network Registrar virtual appliance:

Note You must set the memory and CPUs based on the requirements before clicking the **Power on** button ( ). Once you start the VM, you cannot change the memory or CPU settings until you shut down.

Procedure

Step 1 After deploying the Virtual Appliance OVA, select the virtual machine name in vSphere, right-click on it, and select **Open Console**.

Step 2 Click the **Power on** button ( ) on the console and click in the window after clicking the Power on button.

During the initial boot of the newly deployed machine, you will be prompted to enter a root (system) password, which is not the Cisco Prime Network Registrar application password.

Note
This is the root password for the underlying Linux operating system on which the Cisco Prime Network Registrar 11.2 application is installed. You will be asked to enter this password twice. You will need root access to the underlying Linux operating system at various times in the future, so make sure you remember this password.

The boot process can take a while, both before you are asked for a root password, as well as after you enter the root password.

The End User License Agreement window appears on the first boot. Read the license agreement completely, and only if you understand and accept the license terms, enter **y** (Yes).

Step 3 Log into the server as the root user.

Step 4 To configure the network for the Virtual Appliance, see Configuring Network Access on RHEL/AlmaLinux 8.x Using nmcli, on page 85.

Deploying the Regional Cluster or Local Cluster on OpenStack

To install the Cisco Prime Network Registrar virtual appliance, you must first download the correct installation file. There are two files available: a regional virtual appliance and a local cluster virtual appliance. Each of these virtual appliances is provided as a .qcow2 file.

The names are as follows:

- **cpnr_version_local.qcow2** for the local virtual appliance
- **cpnr_version_regional.qcow2** for the regional virtual appliance

Download the virtual appliance of your choice. Every Cisco Prime Network Registrar local cluster installation must connect to a Cisco Prime Network Registrar regional cluster in order to receive the necessary license information required to operate. Thus, before you install a Cisco Prime Network Registrar local virtual

appliance, you must identify the IP address of the regional cluster to which it will connect to receive the license information.

To run the local cluster or regional cluster on OpenStack, you must first create a local or regional image out using the **.qcow2** distribution kit.

After this image exists, you may launch an instance of the local or regional cluster. The Flavor you associate with the instance needs at least 4 VCPU, 8 GB of RAM, and at least 48 GB of root disk storage. In order to have an operational instance of Cisco Prime Network Registrar, you must allocate more than the absolute minimum of 48 GB of root disk storage. See System Requirements, on page 43 for the amount of disk space needed for a local or regional cluster.

An instance of Cisco Prime Network Registrar will be created with a fixed IP address. Cisco Prime Network Registrar will automatically use any IP addresses associated with interfaces that it can detect when it is started. If the interface available to Cisco Prime Network Registrar has an IP address allocated to it from a provider network (that is, it is accessible to the clients that need the DHCP or DNS capabilities provided by Cisco Prime Network Registrar), then you can configure Cisco Prime Network Registrar normally.

Note When you install Cisco Prime Network Registrar qcow2 image on the recent versions of OpenStack (such as Victoria), the Cisco Prime Network Registrar processes (DNS, DHCP, and CCM) run fine, but the Java process and web UI may take hours or days to run. This issue is specific to Java and the QEMU hypervisor. Therefore, we recommend to use the KVM virtualization to have the proper CPU setting.

You can verify the support for KVM virtualization using the following commands:

```
# lscpu | grep -i "Model Name"
Model name: Intel(R) Xeon(R) CPU E5-2640 v3 @ 2.60GHz

# grep -i -o vmx /proc/cpuinfo | uniq
vmx
```

The presence of "vmx" in the above output shows that necessary CPU extensions are in place on the Intel processor.

When you install a Cisco Prime Network Registrar virtual appliance on VMware, you configure the root password for the underlying Linux system on the system console when the virtual machine is first booted. However, usually OpenStack instances are created and deployed in such a way as to only allow logins with SSH using an SSH key-pair that is configured as part of the OpenStack instance. Many OpenStack instances do not allow root password login at all, and only allow login using SSH with an SSH key-pair.

A Cisco Prime Network Registrar OpenStack instance can be configured to operate in either of these two regimes:

Option 1: Require root password configuration and allow root login using a password.

Option 2: Disable the root password configuration and login. An SSH key-pair is required to login.

Option 1:

This is the default approach for all Cisco Prime Network Registrar virtual appliance kits, and requires no additional actions. You will launch an instance from the Cisco Prime Network Registrar virtual appliance image, and on first boot you will have to bring up a console window for the Cisco Prime Network Registrar instance, enter a root password for the Linux system, and accept the End User License Agreement. After the first boot, you will not need to access the console. You can also access this instance with an SSH key-pair.

Option 2:

If you wish to deploy the Cisco Prime Network Registrar virtual appliance instance in a way that is more in accordance with the usual practice for OpenStack instance deployment, you can configure the Cisco Prime Network Registrar OpenStack instance to not allow root logins with a password, and require an SSH key-pair to login. If you also wish to allow a password based login for a user other than root with root permissions, instructions on how to configure are listed below.

When you launch an OpenStack instance from web UI, to prevent root password login, you will have to perform specific configuration in the **Configuration** section of the Launch instance dialog. You will need to provide a customization script - which is analogous to User Data in other systems. You will need to configure a script (provided below) which will make the OpenStack instance disable the root password based login. After you deploy an instance configured with this customization script, the only way to gain access to the Linux operating system on the instance is to login via SSH using the **ssh key pair** associated with the instance at the time of launch.

For example, you might login with **ssh -i keypairname.pem root@a.b.c.d**. If you did not associate a key pair with the instance, or have lost access to the key pair, you will not be able to login to the instance. There is no default root password when the instance is created in this way, and the root password login is disabled.

To configure option 2, enter the following in the **Customization Script** text box:

```
# cloud-boothook
 # !/bin/bash
 if [ ! -f /etc/cloud/cloud.cfg.orig ]; then
 cp /etc/cloud/cloud.cfg /etc/cloud/cloud.cfg.orig
 cp /etc/cloud/cloud.cfg.norootpasswd /etc/cloud/cloud.cfg
 fi
```

Note If you choose option 2 and once you gained access to the instance using the **ssh key pair**, if you would like to login with a password as well, you can create a new Linux user using the **useradd** command and make that user a member of the group wheel. You must also give that user a secure password using the **passwd** command. Then you can always login with **ssh** or to the console as that user, and have root privileges.

To create a user to allow password login, use the following command:

```
useradd safeuser -g wheel
passwd safeuser
```

Then, if you need root access, login as **safeuser** and use the following command:

```
sudo su
```

Enter the password for **safeuser**, and you will become a root user.

If the IP addresses that are associated with the available interfaces are fixed addresses (that is, they are only accessible to other instances in OpenStack), then you need to associate a floating address with Cisco Prime Network Registrar instance. This floating address must then be accessible to the clients of the DHCP or DNS service to be provided by the Cisco Prime Network Registrar instance. You will have to configure the DHCP server provided by Cisco Prime Network Registrar to return the IP address of the floating address as its server-id, instead of the fixed IP address that Cisco Prime Network Registrar can detect that is associated with the interface built into the instance. To configure DHCP for this situation, you need to be in Expert mode, and configure the DHCP Policy attribute *dhcp-server-identifier-address* with the floating address allocated to this instance. Then the DHCP server will return the configured IP address (which will be the externally visible IP address of this instance) instead of the IP address that the DHCP server can detect from examining the interface that it is using for communications with clients (which is the fixed IP address).

A local cluster needs to be registered with a regional cluster. After this registration, the regional cluster needs to be able to connect to the local cluster. When the local cluster initially registers with the regional cluster, it sends its IP address to the regional cluster. If the regional cluster can contact the local cluster by using the IP address that the local cluster sees is configured to its network interface, then no action is required. This would be the case if the local cluster has a fixed IP address that is only visible within the OpenStack cloud, but the regional cluster was also in the same cloud. If the regional cluster can ping the IP address that the local cluster sees as the IP address on its network interface, then no additional steps are required. However, in the event that the regional cluster is not local to the OpenStack cloud on which the local cluster is running, and the local cluster has a floating address in addition to a fixed address, then the regional cluster's configuration for the local cluster needs to have its IP address updated to be that of the floating address (and not the fixed address, which is what it will have from the initial registration).

When allocating a local cluster, you should consider allocating 4 or even 8 VCPUs and at least 12 GB of RAM, with more for large systems. Local clusters will absolutely need more than the 7+ GB free space available in the minimal installation. Regional clusters will probably need an additional disk space, but 2 to 4 VCPUs and 8 to 12 GB of RAM will suffice for many installations.

Upgrading the Cisco Prime Network Registrar Virtual Appliance

This section describes the procedure for upgrading Cisco Prime Network Registrar to Cisco Prime Network Registrar virtual appliance and upgrading the operating system to AlmaLinux 8.6 using the data from an existing virtual appliance.

Note The newest operating system version used for Cisco Prime Network Registrar 11.2 virtual appliance is AlmaLinux 8.8.

Upgrading a Cisco Prime Network Registrar Installation to Run on a Cisco Prime Network Registrar Virtual Appliance

This section describes how to upgrade an existing installation of Cisco Prime Network Registrar to become a Cisco Prime Network Registrar virtual appliance.

Note This procedure upgrades a current version of Cisco Prime Network Registrar running on a Linux operating system to a current version of the Cisco Prime Network Registrar virtual appliance. If you need to move from a different platform, you have to first convert to the Linux platform prior to upgrading to a virtual appliance. If you need to move from a different version of Cisco Prime Network Registrar to the current version of the virtual appliance, you have to first upgrade to the current version of Cisco Prime Network Registrar on an external Linux system before upgrading to the virtual appliance. See Installing and Upgrading Cisco Prime Network Registrar, on page 21.

Procedure

Step 1 Install the Cisco Prime Network Registrar virtual appliance.

Step 2 Shut down the Cisco Prime Network Registrar application being upgraded using the following command: **systemctl stop nwreglocal**

Step 3 Tar the existing /var/nwreg2/local/data directory using the following command:

```
tar cvf tarfile.tar data
```

Step 4 Copy the tar file created to the new virtual appliance.

Step 5 Shut down Cisco Prime Network Registrar on the new virtual appliance using the following command:

```
systemctl stop nwreglocal
```

Step 6 Rename the existing database to **.orig** using the following command:

```
mv /var/nwreg2/local/data /var/nwreg2/local/data.orig
```

Step 7 Untar the latest database, transferred in Step 3, using **tar xvf tarfile.tar**.

Step 8 Copy any existing extensions from the system being upgraded to the correct directories on the new virtual appliance.

Step 9 Start the Cisco Prime Network Registrar on the new virtual appliance using the following command:

```
systemctl start nwreglocal
```

Upgrading to a new Version of the Virtual Appliance Operating System

To upgrade and to use a new version of the Cisco Prime Network Registrar virtual appliance, install a new virtual appliance which has the new operating system version on it, and then move the data and configuration from the existing virtual appliance to the new virtual appliance.

To do this, follow the steps in Upgrading a Cisco Prime Network Registrar Installation to Run on a Cisco Prime Network Registrar Virtual Appliance, on page 50.

You can now start the new virtual machine. It will have the entire data directory of the existing virtual machine.

Note The new virtual machine with the upgraded operating system will pause during the boot process and instruct you to upgrade the Cisco Prime Network Registrar database to match the database version of the Cisco Prime Network Registrar application that resides on the new virtual machine. Whenever this pause during the boot process and message appears, Cisco Prime Network Registrar will not be able to start until after the script /opt/nwreg2/local/usrbin/upgrade_cnr (or /opt/nwreg2/regional/usrbin/upgrade_cnr for a regional cluster) has been run. Cisco Prime Network Registrar has been masked using systemctl, and the **upgrade_cnr** script will unmask it before performing the upgrade.

Procedure

Step 1 Press return on the console to complete the boot process.

Step 2 Log in as root and run the displayed command.
After boot completion, you should see your existing configuration running with the new version of Cisco Prime Network Registrar on the new virtual machine.

Upgrading the Cisco Prime Network Registrar Application

If you want to upgrade the installation of Cisco Prime Network Registrar that currently exists on the virtual appliance to a new version of Cisco Prime Network Registrar, follow the procedure in this document to perform a straightforward software product upgrade. The installation of Cisco Prime Network Registrar delivered on the virtual appliance is a regular installation of the Cisco Prime Network Registrar software product.

Next Steps: Cisco Prime Network Registrar Virtual Appliance

Configuring Cisco Prime Network Registrar with the CLI on Virtual Appliance

The Cisco Prime Network Registrar CLI can be used to configure the virtual appliance in two ways:

- You can use the nrcmd CLI on the virtual appliance directly by first using SSH to connect into the underlying Linux operating system on the virtual appliance. You can use any username and password which you have created on the virtual appliance for the SSH login. You must use an administrator username and password for the Cisco Prime Network Registrar to use the nrcmd CLI to configure Cisco Prime Network Registrar.

Note As distributed, there is only one valid user for the Linux operating system—root. While you can login as root to use the Cisco Prime Network Registrar CLI, you might want to add additional users to the system. Use the **useradd** program to add additional users. You can type **man useradd** for more information on how to add additional users.

- Alternatively, you can use the nrcmd CLI on some other system in the network to configure and manage Cisco Prime Network Registrar on the virtual appliance the same way that you would use it to manage any remote installation of Cisco Prime Network Registrar. This requires installing Cisco Prime Network Registrar (typically only the client-only installation) on the other system.

Configuring the Virtual Appliance to Automatically Power Up

You can configure the ESXi hypervisor to automatically power up the Cisco Prime Network Registrar virtual appliance when power is restored to the ESXi hypervisor layer.

To configure automatic power up:

Procedure

Step 1 In the vSphere client, select the ESXi machine to which you are connected. It is not a specific virtual machine that you have to select but the ESXi hypervisor on which they reside.

Step 2 Select the **Configuration** tab.

Step 3 Click the **Virtual Machine Startup/Shutdown** link under the **Software** area. You should see the virtual machine in the list shown in window.

Step 4 Click the **Properties...** link present at the top right corner of the page. If you do not see that, resize the window until you do.

The Virtual Machine Startup and Shutdown page is displayed.

Step 5 Check the **Allow virtual machines to start and stop automatically with the system** check box.

Step 6 Select the virtual machine running the Cisco Prime Network Registrar virtual appliance and use the **Move Up** button on the right to move it up into the group labelled **Automatic Startup**.

Step 7 Click **OK**.

This ensures that whenever power is restored to the ESXi hypervisor. The Cisco Prime Network Registrar appliance powers up automatically.

Managing the Cisco Prime Network Registrar Virtual Appliance

You can manage the underlying Linux operating system, which is based on AlmaLinux 8.8, by logging in as the root user. You may use SSH to log into the virtual appliance with the username root and the root password you specified when you first booted the virtual appliance. On Openstack, you may use the key pair created when you launched the instance.

You will probably want to create additional users on the Linux system so that people can access the Linux system with a username other than root.

The Linux system which is included on the virtual appliance is stripped down to a considerable degree and thus does not include things that are not required to run or manage the Cisco Prime Network Registrar application, such as a Windows system manager and its associated GUI user interface. However, all the tools necessary to support and manage the Cisco Prime Network Registrar application are included on the Linux operating system used inside of the virtual appliance.

You may also want to take additional steps to secure the SSH connection. For instance, configuring it to prevent logging on as root, and requiring a user to **su** to gain root privileges after logging on as another user.

You may wish to perform other configuration changes on the underlying Linux operating system in order to lock it down in ways appropriate to your environment.

Note The newest operating system version used for Cisco Prime Network Registrar 11.2 virtual appliance is AlmaLinux 8.8.

Note Cisco Prime Network Registrar customers are solely responsible for keeping their OS up to date regarding patches that they desire to apply and Cisco is not responsible for the same.

Note The Linux distribution packaged with the optional pre-built VM downloads for Cisco Prime Network Registrar is Open Source software and not owned or supported by Cisco. Customers seeking support for Linux should reach out to third party software providers.

Post OVA Installation

Follow the below steps to get the latest AlmaLinux updates, latest version of installed packages and security updates before configuring the Cisco Prime Network Registrar.

Note The command **yum update or dnf update** will update the running system with new and changed software that in most cases did not exist when the Cisco Prime Network Registrar application was tested with the operating system shipped on the virtual appliance. The updates that are installed as part of the **yum update or dnf update** command do not cause any problems for the Cisco Prime Network Registrar application. However, Cisco cannot guarantee that the Cisco Prime Network Registrar application will perform without any problems when interfacing with software that was not available when our testing was performed. You should perform your own testing to ensure that everything is operating correctly in your environment after a **yum update or dnf update** command has been executed before placing the updated virtual appliance into production.

Procedure

Step 1 Login as root.

Step 2 Configure networking.

Step 3 Go to the root prompt and enter the following command:

```
# yum update
```

Step 4 Reboot the system and then configure Cisco Prime Network Registrar.

Building Your Own Base Images for Deploying Virtual Machines

Disabling dnsmasq and libvirtd

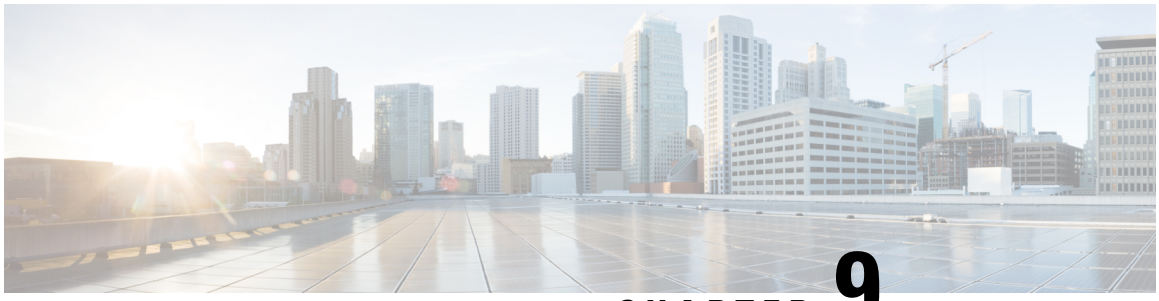
The dnsmasq and libvirtd services sometimes come preinstalled with Linux 7 and 8 distributions. These use ports 53 and 67, and may prevent Cisco Prime Network Registrar from operating properly. You must disable dnsmasq and libvirtd for DHCP and DNS servers to start up and run properly.

To disable the dnsmasq and libvirtd services, execute the following commands and then reboot the virtual machine:

```
# systemctl disable dnsmasq.service

# systemctl disable libvirtd.service

# reboot
```



CHAPTER 9

Cisco Prime Network Registrar on Container

Cisco Prime Network Registrar 11.2 can be run as a Docker container that you can install in your own infrastructure.

The following Docker images are provided for Cisco Prime Network Registrar 11.2:

- Regional container: **cpnr-regional-11.2-1.el8.x86_64_rhel_docker.tar.gz**
- DHCP and DNS Local container: **cpnr-local-11.2-1.el8.x86_64_rhel_docker.tar.gz**
- CDNS local container: **cpnr-cdns-11.2-1.el8.x86_64_rhel_docker.tar.gz**

This chapter contains the following sections:

- Requirements on the Host Machine, on page 57
- Running Cisco Prime Network Registrar Docker Container, on page 57
- Moving an Existing Cisco Prime Network Registrar Cluster to Docker Container, on page 60
- Upgrading Cisco Prime Network Registrar Docker Container, on page 61

Requirements on the Host Machine

- Identify the ports on the host machine to be exposed to ports required by Cisco Prime Network Registrar container. For a complete list of ports used by Cisco Prime Network Registrar services, see the *"Default Ports for Cisco Prime Network Registrar Services" section in the Cisco Prime Network Registrar 11.2 Administration Guide*.
- Select an option to persist data of Cisco Prime Network Registrar container on the host machines: Bind mount (where a directory on the host machine is used) or Volume (which is managed by Docker).
- For IPv4, you can use either bridged or macvlan network; Cisco recommends macvlan for better performance.
- For IPv6, you will need to configure the container to have an IPv6 address.

Running Cisco Prime Network Registrar Docker Container

To run Cisco Prime Network Registrar as Docker container, you must first download the Docker image of your choice. Then, do the following:

Procedure

Step 1 Load the Docker image using the following command:

- For the regional container:

```
# docker load -i cpnr-regional-11.2-1.el8.x86_64_rhel_docker.tar.gz
```

- For the local container:

```
# docker load -i cpnr-local-11.2-1.el8.x86_64_rhel_docker.tar.gz
```

Step 2 Verify that the image is successfully loaded using the following command:

```
# docker image ls
```

Step 3 Run the Docker container using the following command:

- For the regional container:

```
# docker run -d --name cpnr_regional_container --privileged=true -p 8453:8453 -p 1244:1244 --mount
 type=bind,source=/data/cpnr_regional_data,target=/var/nwreg2/regional cpnr-regional:11.2
/usr/sbin/init
```

In the above command:

- Default bridge networking driver of Docker is used. Ports needed by container are exposed–8453 is for regional web UI, 1244 for regional configuration management.
- Data directory of Cisco Prime Network Registrar is /var/nwreg2/regional and mountpoint on the host is /data/cpnr_regional_data
- The command to be run is /usr/sbin/init

If it is required to synchronize the timezone of the host and the Docker container, then add the **-v /etc/localtime:/etc/localtime** option to the above Docker run command.

By default, the core files are available in the /var/lib/systemd/coredump directory of the Docker host machine. To collect the core files via the **cnr_tactool** utility, run the following commands on the Docker host machine:

```
# echo '/data/cpnr_regional_data/core.%p' > /proc/sys/kernel/core_pattern'
# ulimit -c unlimited
```

After running the above commands, the core files will be available in the /data/cpnr_regional_data directory and you can use **cnr_tactool** to collect them.

- For the local container:

```
# docker run -d --name cpnr_local_container --privileged=true -p 8443:8443 -p 1234:1234 -p
67:67/udp -p 53:53/udp --mount type=bind,source=/data/cpnr_local1_data,target=/var/nwreg2/local
 cpnr-local:11.2 /usr/sbin/init
```

In the above command:

- Default bridge networking driver of Docker is used. Ports needed by container are exposed–8443 is for the web UI, 1234 for local configuration management, 67 for DHCP, and 53 for DNS. For other services like SNMP, TFTP see the *"Default Ports for Cisco Prime Network Registrar Services" section in the Cisco Prime Network Registrar 11.2 Administration Guide*.

- Data directory of Cisco Prime Network Registrar is /var/nwreg2/local and mountpoint on the host is /data/cpnr_local1_data
- The command to be run is /usr/sbin/init

If it is required to synchronize the timezone of the host and the Docker container, then add the **-v /etc/localtime:/etc/localtime** option to the above Docker run command.

By default, the core files are available in the /var/lib/systemd/coredump directory of the Docker host machine. To collect the core files via the **cnr_tactool** utility, run the following commands on the Docker host machine:

```
# echo '/data/cpnr_local1_data/core.%p' > /proc/sys/kernel/core_pattern'
# ulimit -c unlimited
```

After running the above commands, the core files will be available in the /data/cpnr_local1_data directory and you can use **cnr_tactool** to collect them.

Step 4 Start configuring Cisco Prime Network Registrar.

- For the regional container:
  - To connect using web UI, use https://*hostip*:8453
  - To connect using CLI, use the following command:

    *install-path***/usrbin/nrcmd -R -C** *hostip***:1244 -N** *username* **-P** *password*

- For the local container:
  - To connect using web UI, use https://*hostip*:8443
  - To connect using CLI, use the following command:

    *install-path***/usrbin/nrcmd -C** *hostip***:1234 -N** *username* **-P** *password*

For running DHCP failover and HA DNS, we recommend running two Cisco Prime Network Registrar containers (Main and Backup) in separate hosts, as this avoids single point of failure. Given that bridge network is limited to a single host, using macvlan as networking driver is the best choice. With macvlan, container appears to be directly connected to physical network.

If the Docker daemon allows IPv6, you can use dual-stack macvlan networks, that is, both IPv4 and IPv6:

```
# docker network create --driver=macvlan --ipv6 --subnet=2001:db8:1:1::/64
--gateway=2001:db8:1:1::1 --subnet=10.0.0.0/24 --gateway=10.0.0.1 -o macvlan_mode=bridge
-o parent=eth0 cpnr_macvlan
```

Run Cisco Prime Network Registrar container and attach it to the macvlan network created above:

```
# docker run -d --name cpnr_dhcp_main --network=cpnr_macvlan --ip 10.0.0.20 --ip6
2001:db8:1:1::20 --privileged=true --mount type=bind,source=/data/cpnr_dhcp_main_data,
target=/var/nwreg2/local cpnr-local:11.2 /usr/sbin/init
```

This Cisco Prime Network Registrar container (local) can now be reached at 10.0.0.20 and 2001:db8:1:1::20.

- To connect using web UI over IPv4, use https://10.0.0.20:8443
- To connect using CLI over IPv6, use the following command:

  *install-path***/usrbin/nrcmd -C [2001:db8:1:1::20]:1234 -N** *username* **-P** *password*

Moving an Existing Cisco Prime Network Registrar Cluster to Docker Container

To move to Cisco Prime Network Registrar 11.2 Docker container from an existing Cisco Prime Network Registrar 9.0 or later cluster, do the following:

Procedure

Step 1 Remove the existing installation using the procedure described in Uninstalling Cisco Prime Network Registrar, on page 41.

Step 2 Delete the /opt/nwreg2 folder. Ensure NOT to delete the /var/nwreg2 folder after uninstallation.

If you are upgrading to Cisco Prime Network Registrar 11.2 Docker Container on the same machine, then skip this Step 3 and proceed to Step 4.

Step 3 If you are upgrading to Cisco Prime Network Registrar 11.2 Docker Container on a different machine, then create the source directory tree (for example, /data/cpnr_local1_data for local cluster and/data/cpnr_regional_data for regional cluster) in the machine where you want to create the Docker instance (target machine). Then, transfer the original cluster's /var/nwreg2/{local | regional} directory to this directory. Use the following commands:

- For regional cluster:

```
# mkdir  -p  /data/cpnr_regional_data
# mv  /var/nwreg2/regional  /data/cpnr_regional_data
```

- For local cluster:

```
# mkdir  -p  /data/cpnr_local1_data
# mv  /var/nwreg2/local  /data/cpnr_local1_data
```

Note
In case of pre-11.0 releases, copy the cnr.conf file from the /opt/nwreg2/{local | regional}/conf directory to the conf folder in the source directory of the target machine. Use the following commands:

- For regional cluster:

```
# mv /opt/nwreg2/regional/conf /data/cpnr_regional_data/conf
```

- For local cluster:

```
# mv /opt/nwreg2/local/conf /data/cpnr_local1_data/conf
```

Step 4 Create the Docker instance using the following command:

- For regional container:

```
$ docker run -d --name cpnr_container -v /etc/localtime:/etc/localtime --network=mymacvlan
--ip hostip --ip6 ipv6address --privileged=true --hostname=hostip --mount type=bind,
source=/data/cpnr_regional_data,target=/var/nwreg2/regional cpnr_regional:11.2 /usr/sbin/init
```

- For local container:

```
$ docker run -d --name cpnr_container -v /etc/localtime:/etc/localtime --network=mymacvlan
--ip hostip --ip6 ipv6address --privileged=true --hostname=hostip --mount type=bind,
source=/data/cpnr_local1_data,target=/var/nwreg2/local cpnr_local:11.2 /usr/sbin/init
```

Step 5 Verify if the previous configuration, including scopes and zones, is intact in Cisco Prime Network Registrar 11.2 server. Also, verify that the data.bak folder is created which contains the pre-upgrade version's database version as backup.

Note After performing the above steps, all settings will take their defaults and you may have to take additional steps to reinstall the certificates or change ports. For information, see Installing Your Own Certificate for Web UI Access, on page 30.

Upgrading Cisco Prime Network Registrar Docker Container

To upgrade the existing Cisco Prime Network Registrar Docker Container image to the latest version of Cisco Prime Network Registrar Docker Container image, do the following:

Procedure

Step 1 Check the running Docker container using the following command:

```
# docker ps -a
```

Step 2 Stop the running Docker container using the following command:

- For Cisco Prime Network Registrar local instance:

  ```
  # docker stop local_1
  ```

- For Cisco Prime Network Registrar regional instance:

  ```
  # docker stop regional_1
  ```

Step 3 Remove the existing Docker container using the following command:

- For Cisco Prime Network Registrar local instance:

  ```
  # docker rm local_1
  ```

- For Cisco Prime Network Registrar regional instance:

  ```
  # docker rm regional_1
  ```

Step 4 List the images using the following command:

```
# docker images
```

Step 5 Delete the images using the following command:

```
# docker rmi f6c7bafa04f3
```

Note

Do not remove the data folder of Cisco Prime Network Registrar.

Step 6 Load the new Docker image using the following command:

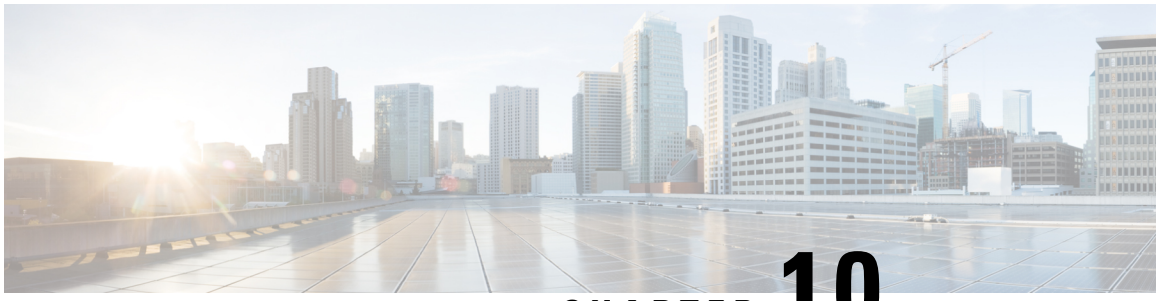
- For Cisco Prime Network Registrar local instance:

```
# docker load -i cpnr-local-11.2.el8.x86_64_rhel_docker.tar.gz
```

- For Cisco Prime Network Registrar regional instance:

```
# docker load -i cpnr-regional-11.2.el8.x86_64_rhel_docker.tar.gz
```

Step 7 Run the Cisco Prime Network Registrar docker container using the procedure described in Running Cisco Prime Network Registrar Docker Container, on page 57.



CHAPTER 10

Cisco Prime Network Registrar on Kubernetes

Kubernetes is an open source container orchestration system for automating software deployment, scaling, and management. Starting from Cisco Prime Network Registrar 11.2, you can deploy the Cisco Prime Network Registrar instances on Kubernetes using the following Docker images:

- For regional instance deployment: **cpnr-regional-11.2-1.el8.x86_64_rhel_docker.tar.gz**
- For DHCP and DNS local instance deployment: **cpnr-local-11.2-1.el8.x86_64_rhel_docker.tar.gz**
- For CDNS local instance deployment: **cpnr-cdns-11.2-1.el8.x86_64_rhel_docker.tar.gz**

Note The names of the images will change with releases in future.

You need to have a private docker registry in Kubernetes to load the container image.

This chapter contains the following section:

- Deploying Cisco Prime Network Registrar regional and local Instance on Kubernetes, on page 63
- Deploying Cisco Prime Network Registrar CDNS instance on Kubernetes, on page 66
- Upgrading Cisco Prime Network Registrar Kubernetes, on page 70

Deploying Cisco Prime Network Registrar regional and local Instance on Kubernetes

You can deploy the Cisco Prime Network Registrar instances on Kubernetes using the YAML files. YAML is the standard format used in a Kubernetes configuration file. Cisco Prime Network Registrar kit **cpnr-11.2-1.el8.x86_64_kubernetes.tar.gz** contains examples of YAML files (cpnr-local-statefulset.yaml and cpnr-regional-statefulset.yaml), which depict one of the ways of deploying Cisco Prime Network Registrar on Kubernetes.

Note No special license is required to deploy Cisco Prime Network Registrar in the Kubernetes environment. It uses the existing container licenses.

For example, you can use the cpnr-local-statefulset.yaml in the Cisco Prime Network Registrar kit to create the Cisco Prime Network Registrar local instance on Kubernetes. This configuration uses StatefulSet and hostNetwork deployment. Instance created using this YAML will bind the Cisco Prime Network Registrar local instance to the configured worker node. This instance will run only on the configured worker node and it will not run on the other node.

When a pod is configured with **hostNetwork: true**, the applications running in the pod can directly see the network interfaces of the host machine where the pod was started. An application that is configured to listen on all network interfaces will in turn be accessible on all the network interfaces of the host machine.

YAML file consists of the following Kubernetes resources:

- Service

  In cpnr-local-statefulset.yaml, cpnr-local is the service name and since Headless service is used, the clusterIp is set to None.

- StatefulSet

  In cpnr-local-statefulset.yaml, cpnr-local is the StatefulSet name and is used to run the Cisco Prime Network Registrar 11.2 docker image with replica of one.

  StatefulSet is used in Cisco Prime Network Registrar for two reasons:

  - Constant pod names
  - When pods created using deployment are deleted, it creates a new pod before the old pod is completely terminated. Since hostNetwork is used before the old Cisco Prime Network Registrar process on the host node is termianted completely, the new Cisco Prime Network Registrar pod is created and the Cisco Prime Network Registrar services on the new pod is down since the Cisco Prime Network Registrar process on the old pod is not termianted completely.

    This is overcome by StatefulSet where the old pod is terminated completely and only then the new pod is created.

Note Cisco Prime Network Registrar has been tested using the hostNetwork, as other networking modes had issues with HA and failover pair. In the hostNetwork mode, when a pod starts, it uses the host network namespace and host IP address. This basically means that the pod can see all the network interfaces of the host. With hostNetwork mode, only one Cisco Prime Network Registrar instance can be deployed in a node and hence the replica in the YAML is set to 1. If multiple Cisco Prime Network Registrar pods are to be deployed in the same node using hostNetwork mode, then for each pod, you need to change all Cisco Prime Network Registrar related ports and adjust the replicas accordingly. However, this may not be useful.

Procedure

Step 1 Configure the following parameters in the YAML file:

- NODE_NAME: Worker node name, where the Cisco Prime Network Registrar instance will run. For example, “cnr-k8s-worker2.*server*.com”. Use the **kubectl get nodes** command to get the node name.

- IMAGE: The Docker image location. For example “cnr-k8s-worker1.server.com/cpnr-local:11.2”, where cnr-k8s-worker1.*server*.com is the private registry and the image name is cpnr-local with tag 11.2.
- HOST_MOUNT_PATH: Directory path on the host machine. This directory will be used to store configuration files and data on the Cisco Prime Network Registrar instance. /var/nwreg2 of the pod is mapped to HOST_MOUNT_PATH in the host machine. This is required to persist the data ofCisco Prime Network Registrar instance on the host machines.

Step 2 Create the Cisco Prime Network Registrar instance on Kubernetes using the following command:

- For Cisco Prime Network Registrar local instance deployment:

```
# kubectl create -f cpnr-local-statefulset.yaml
```

- For Cisco Prime Network Registrar regional instance deployment:

```
# kubectl create -f cpnr-regional-statefulset.yaml
```

Step 3 Check the Cisco Prime Network Registrar instance details on Kubernetes using the following command:

```
# kubectl get all
```

Step 4 Log into the Cisco Prime Network Registrar instance pod using the following command:

```
# kubectl exec -it <pod name> -- bash
```

For example:

```
# kubectl exec -it cpnr-dhcp-dns-0 -- bash
# /opt/nwreg2/local/usrbin/nrcmd -s
100 Ok
```

Step 5 Set the user name and password, and register the local pods to regional pods.

If you want to delete the Cisco Prime Network Registrar instance on Kubernetes, use the following command:

- For Cisco Prime Network Registrar local instance:

```
# kubectl delete -f cpnr-local-statefulset.yaml
```

- For Cisco Prime Network Registrar regional instance:

```
# kubectl delete -f cpnr-regional-statefulset.yaml
```

To debug any pod failures, use the **kubectl logs** *podname* or **kubectl describe pod** *podname* command.

Note If you want to deploy Cisco Prime Network Registrar pod in another worker node, you need to make changes in the YAML file (for example, you need to change service.metadata.name and statefulset.metadata.name.).

Note Cisco Prime Network Registrar 11.2 Docker images have been tested with Kubernetes version 1.27. CNI used is Calico 3.22.0. The entire testing is carried out using the YAML files provided as examples. If you want to change the YAML file, it is your responsibility to test it before moving it to production.

Deploying Cisco Prime Network Registrar CDNS instance on Kubernetes

You can deploy the Cisco Prime Network Registrar Caching DNS Server instances on Kubernetes using the YAML files. **cpnr-11.2-1.el8.x86_64_kubernetes.tar.gz** contains examples of YAML files (cpnr-cdns-deployment.yaml).

Starting Cisco Prime Network Registrar 11.2, a separate CDNS container is deployed to make CDNS service more cloud native (smaller, stateless, and, horizontally scalable). CDNS on Kubernetes now supports auto scaling, CDNS service will start only if corresponding license is available on Regional Cluster, so it is highly recommended to use Smart License for CDNS deployment on Kubernetes to make scaling seamless. Before deploying the CDNS pod, make sure to which regional server this CDNS pod will be registered.

Note No special license is required to deploy Cisco Prime Network Registrar in the Kubernetes environment. It uses the existing container licenses.

In case if you prefer to use Flexlm license, there must be sufficient buffer of licenses on Regional Cluster to avoid CDNS registration/deregistration issues as license update is not instant. In this scenario if services exceed the licenses and the pods are stuck, you might need to manually clean-up the CDNS clusters which are not in use but have not been automatically deregistered due to license mismatch.

The CDNS container does not have any of the database such as CCM and no client interfaces like Web UI, or CLI, or REST API. All the configuration and management for CDNS happens from the regional and whatever configuration is required for CDNS, it has to be done from regional.

Note Hybrid mode and its configuration is not applicable to a CDNS service pod.

SNMP Server is not applicable for CDNS service pod, it is recommended to use the Open Source or Kubernetes native technologies like Prometheus, Grafana etc for monitoring, stats collection, and alerting.

The utilities like cnr_exim and cnr_tactool are available and they connects to regional for configuration related data. A new utility cnr_cmd used is to reload the server without restarting the CDNS pod. The examples of the utilities used and debug settings is also shown below:

```
./cnr_tactool
user: admin
password:
Collecting CPNR Information:
Finding CNR Version...
Getting CNR Configuration File...
Searching for core files in '/var/nwreg2/cores/'...
Searching for Java HotSpot Error Logs...
Copying CNR Logs...
Exporting CNR Configuration...
Collecting System Information:
Gathering System Information...
Copying System Logs...
Getting System Processor Information...
Getting System Memory...
Getting Resource Limitation Information...
```

```
Getting IP Configuration Information...
Getting Cron Information...
Getting Process List...
Getting Services List...
Getting Socket Stats...
Getting Disk Space Utilization Information...
Getting Network Statistics...
Getting Kernel Routing Tables...
Getting Network Statistics...
Getting Kernel Parameters...
Getting IP Tables Information...
Getting nwreglocal.service journal entries...
Getting Curl Version Information...
Generating HTML File...
Successfully processed and archived data in
'/opt/nwreg2/local/temp/cnr_tac-9-19-2023-01.tar.gz'
```

```
./cnr_exim -e -x export-data
user: admin
password:
-----
```

```
./cnr_cmd -c cdns reload
```

```
./cnr_cmd -c "cdns setDebug AQW=1"
cdns debug set to AQW=1.
```

```
./cnr_cmd -c "cdns unsetDebug"
cdns debug cleared.
```

For CDNS container, the YAML file consists of the Kubernetes resource Service where there are multiple CDNS services available and that is called the node port type wherein it just runs as a normal Kubernetes cloud native service and can be changed it to a load balancer.

Refer to the sample code for the Kubernetes resource Service:

```
apiVersion: v1
kind: Service
metadata:
  name: cpnr-cdns-svc
  labels:
    app: cpnr-cdns-svc
spec:
  type: NodePort
  clusterIP: 10.8.7.6
  ports:
    - protocol: UDP
      name: cdns-tcp
      port: 53
      targetPort: 53
      nodePort: 53
    - protocol: TCP
      name: cdns-udp
      port: 53
      targetPort: 53
      nodePort: 53
    - protocol: TCP
      name: cdns-metrics
      port: 8000
      targetPort: 8000
      nodePort: 8000
  selector:
    app: cpnr-cdns
```

Procedure

Step 1 Configure the following parameters in the YAML file:

- Regional IP: The regional IP where the CDNS server will be registered.
- Image: The Docker image location. For example “cnr-k8s-worker1.server.com/cpnr-local:11.2”, where cnr-k8s-worker1.server.com is the private registry and the image name is cpnr-local with tag 11.2.
- Host mount path: Directory path on the host machine. This directory will be used to store configuration files and data on the Cisco Prime Network Registrar instance. /var/nwreg2 of the pod is mapped to HOST_MOUNT_PATH in the host machine. This is required to persist the data of Cisco Prime Network Registrar instance on the host machines.

Step 2 Create the Cisco Prime Network Registrar CDNS local instance deployment on Kubernetes using the following command:

- **# kubectl create -f cpnr-cdns-deployment.yaml**

Step 3 Check the Cisco Prime Network Registrar CDNS local instance deployment on Kubernetes using the following command:

- **# kubectl get all**

Step 4 Log into the Cisco Prime Network Registrar instance pod using the following command:

```
# kubectl exec -it <pod name> -- bash
```

For example:

```
# kubectl exec -it cpnr-cdns-5795ff8d47-9zzzq -- bash
# /opt/nwreg2/local/usrbin/nrcmd -s
100 Ok
```

Step 5 Set the user name and password, and register the local pods to regional pods.

If you want to delete the Cisco Prime Network Registrar instance on Kubernetes, use the following command:

- For Cisco Prime Network Registrar CDNS instance:

  ```
  # kubectl delete -f cpnr-cdns-deployment.yaml
  ```

To debug any pod failures, use the **kubectl logs** *podname* or **kubectl describe pod** *podname* command.

Note Force deletion of CDNS service pod using the following command does not remove cluster or service-pod entry on regional cluster.

```
#kubectl delete pod <name> --force
```

Cisco Prime Network Registrar 11.2 Docker images have been tested with Kubernetes version 1.28.2. CNI used is Calico 3.26.1. The entire testing is carried out using the YAML files provided as examples. If you want to change the YAML file, it is your responsibility to test it before moving it to production.

Container Logging

CDNS supports Kubernetes native logging and the logs like kubectl logs, and CDNS pods can be accessed. The CDNS logs can be configured with external logging system for proper log management. The CDNS logs are deployed and tested with Elastic search, Fluentd and Kibana (EFK) stack.

The command that can be used to retrieve the logs for a specific CDNS pod running in a Kubernetes cluster is:

```
kubectl logs <cdns-pod>
```

The CDNS logs example in Kubernetes cluster:

```
kubectl logs cpnr-cdns-7dfbc4bb79-68r6t
```

CDNS Metrics

The CDNS metrics (stats) are exposed in the cloud native way to make it accessible to a monitoring system such as Prometheus.

Note CDNS writes its stats periodically to a file named cdns_metrics for every 10 seconds.

All CDNS stats will be exported. Upstream services such as Prometheus and Grafana can do further filtering to fit the customer's monitoring requirements. This allows the customer to have the most flexibility over what metrics are monitored. Additionally, since the metrics are stored long term, it also allows the customer to make changes on the fly without needing to change anything about their deployment. The CDNS metrics has been tested using Prometheus.

CDNS Scaling

CDNS Service can be scaled up/down horizontally using Kubernetes HPA (Horizontal Pod Autoscaling). HPA can be configured for the CDNS deployment. Customers can create their own HPA based on the metric(s) they want to monitor for scaling the CDNS service.

Below example depicts HPA based on ‘avg_queries_per_second’ custom metric. This metric has been generated based on ‘queries_total’ CPNR metric by configuring appropriate ConfigMap in the Prometheus Adapter.

```
apiVersion: autoscaling/v2beta2
kind: HorizontalPodAutoscaler
metadata:
  name: cdns-hpa
spec:
  scaleTargetRef:
    apiVersion: apps/v1
    kind: Deployment
    name: cpnr-cdns
  minReplicas: 2
  maxReplicas: 6
  metrics:
  - type: Pods
    pods:
      metric:
        name: avg_queries_per_second
      target:
        type: AverageValue
```

```
        averageValue: 75000

apiVersion: v1
kind: ConfigMap
metadata:
  name: adapter-config
  namespace: monitoring
data:
  config.yaml: |
    rules:
    - seriesQuery: 'queries_total'
      resources:
        overrides:
          kubernetes_namespace:
            resource: namespace
          kubernetes_pod_name:
            resource: pod
      name:
        matches: "queries_total"
        as: "queries_per_second_avg"
      metricsQuery: sum(rate(<<.Series>>{<<.LabelMatchers>>}[1m])) by (<<.GroupBy>>)
```

Upgrading Cisco Prime Network Registrar Kubernetes

To upgrade the existing Cisco Prime Network Registrar Kubernetes to the latest version of Cisco Prime Network Registrar Kubernetes, do the following:

Procedure

Step 1 Check the Cisco Prime Network Registrar instance details on Kubernetes using the following command:

```
#kubectl get all
```

Step 2 Remove the existing Cisco Prime Network Registrar instance on Kubernetes using the following steps:

- For Cisco Prime Network Registrar local instance:

  ```
  #kubectl delete -f cpnr-local-statefulset.yaml
  ```

- For Cisco Prime Network Registrar regional instance:

  ```
  #kubectl delete -f cpnr-regional-statefulset.yaml
  ```

Note
Make sure not to do the cleanup operations (that is, retain the data, cnr.conf, and so on). Directory mentioned as HOST_MOUNT_PATH in the yaml file must not be deleted.

Step 3 To avoid issues with Java upgrades, it is highly recommended to edit the cnr.conf file (under HOST_MOUNT_PATH) to replace the cnr.java-home entry path with /usr/bin/java.

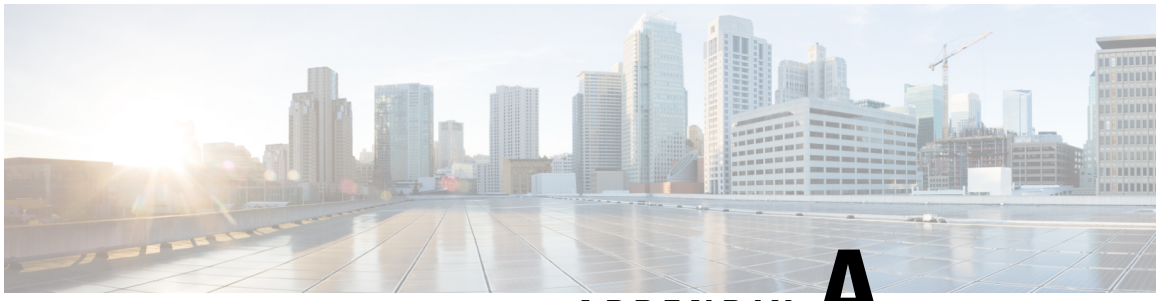
Step 4 Deploy Cisco Prime Network Registrar 11.2 instance in Kubernetes. For installation instructions, see Deploying Cisco Prime Network Registrar CDNS instance on Kubernetes, on page 66.

Step 5 To debug any pod failures, use the following commands and take corrective actions based on the failure:

```
#kubectl logs podname
```

or

```
#kubectl describe pod podname
```



APPENDIX A

Lab Evaluation Installations

This appendix contains the following sections:

- Lab Evaluation Installations, on page 73
- Installing Cisco Prime Network Registrar in a Lab, on page 73
- Testing the Lab Installation, on page 74
- Uninstalling in a Lab Environment, on page 74

Lab Evaluation Installations

This appendix describes how to install, upgrade, and uninstall Cisco Prime Network Registrar regional and local clusters on a single machine to support smaller test configurations for evaluation purposes.

Caution Installing the regional and local cluster on a single machine is intended only for lab evaluations, and should not be chosen for production environments. The aggregated regional cluster databases are expected to be too large to be reasonably located with a local server that is also running DNS or DHCP services. Running out of free disk space causes these servers to fail.

Installing Cisco Prime Network Registrar in a Lab

To install Cisco Prime Network Registrar on a single machine for evaluation purposes:

Procedure

Step 1 Check whether the machine has enough disk space to accommodate two separate installations of Cisco Prime Network Registrar.

Step 2 Install or upgrade the local cluster, according to the procedure in Installing Cisco Prime Network Registrar, on page 21. Use the cpnr-local kit.

Step 3 Install or upgrade the regional cluster on the same machine, according to the same procedure. Use the cpnr-regional kit.

Testing the Lab Installation

To test the installation:

Procedure

Step 1 Start and log in to the web UI for the local cluster. By default, the local port number is **8443** for HTTPS (secure) connections.

Step 2 Add DNS zones and DHCP scopes, templates, client-classes, or virtual private networks (VPNs) as a test to pull data to the regional cluster.

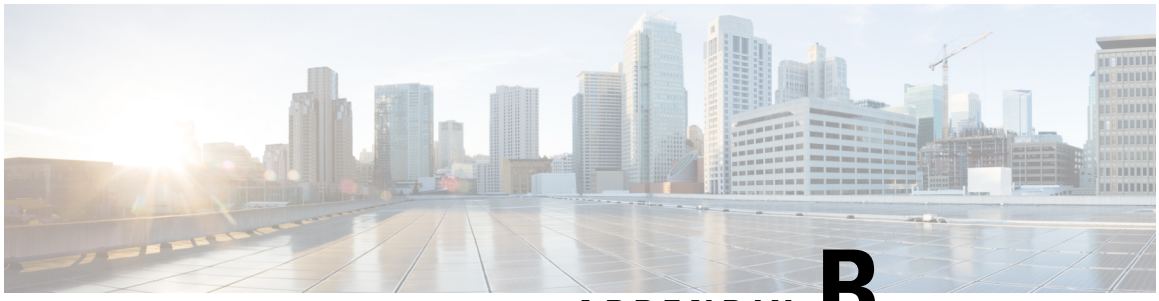
Step 3 Start and log in to the web UI for the regional cluster. By default, the regional port number is **8453** for HTTPS (secure) connections.

Step 4 Test the regional cluster for single sign-on connectivity to the local cluster. Try to pull DNS zone distributions, DHCP scopes, templates, client-classes, or VPNs from the local cluster to the regional replica database.

Uninstalling in a Lab Environment

To remove the local cluster, follow the steps in Uninstalling Cisco Prime Network Registrar, on page 41 and specify cpnr-local for the kit.

To remove the regional cluster, follow the steps in Uninstalling Cisco Prime Network Registrar, on page 41 and specify cpnr-regional for the kit.



APPENDIX B

Installing the Cisco Prime Network Registrar SDK

This section documents how to install the Cisco Prime Network Registrar SDK. Before installing the SDK, ensure that you have JDK 11 installed on your system. The Cisco Prime Network Registrar SDK is a separate product and is sold separately.

This appendix contains the following sections:

- Installing Cisco Prime Network Registrar SDK, on page 75
- Testing Your Installation, on page 76
- Compatibility Considerations, on page 76

Installing Cisco Prime Network Registrar SDK

To install the Cisco Prime Network Registrar SDK:

Procedure

Step 1 Extract the contents of the distribution .tar file.

a) Create the SDK directory:

```
% mkdir /cnr-sdk
```

b) Change to the directory that you just created and extract the .tar file contents:

```
% cd /cnr-sdk
% tar xvf sdk_tar_file_location/cnrsdk.tar
```

Step 2 Export your LD_LIBRARY_PATH and CLASSPATH environment variable:

```
% export LD_LIBRARY_PATH=/cnr-sdk/lib
% export CLASSPATH=/cnr-sdk/classes/cnrsdk.jar:.
```

Note
If you have Cisco Prime Network Registrar installed on the system, then use the /opt/nwreg2/{local | regional}/lib path for LD_LIBRARY_PATH. If you do not have Cisco Prime Network Registrar installed, then you must point to the lib directory into which you untarred the files. If the system is not running as a local or regional cluster, you may want to

consider installing the cpnr-client kit (to get access to other command line utilities). You can then specify LD_LIBRARY_PATH as /opt/nwreg2/client/lib.

Testing Your Installation

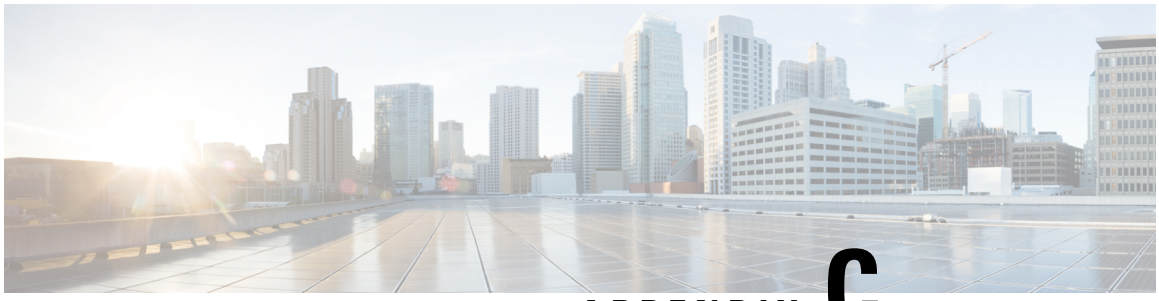
The following test program verifies that you have set your PATH or LD_LIBRARY_PATH correctly:

```
% java -jar /cnr-sdk/classes/cnrsdk.jar
```

Compatibility Considerations

For Java SDK client code developed with an earlier version of the SDK, you can simply recompile most code with the latest JAR file to connect to an upgraded server.

Review the *"SDK Compatibility Considerations" sections of the Cisco Prime Network Registrar 11.2 Release Notes* for the intervening Cisco Prime Network Registrar versions, as these highlight any significant SDK compatibility considerations.



APPENDIX C

Enhancing Security for Web UI

This appendix contains the following section:

- Enhancing Security for Web UI, on page 77

Enhancing Security for Web UI

When connected through the Secured Socket Layer (SSL) protocol using HTTPS, the web UI uses the default ciphers for the Java Virtual Machine (JVM). These ciphers usually include weak cipher session keys and can affect system security. In case you want to harden the system, adjust the ciphers as below:

Note The default installation of Cisco Prime Network Registrar 11.2 works with Transport Layer Security (TLS) 1.2. You can change the configuration to make it work with the older TLS versions, if needed.

Procedure

Step 1 Open the **server.xml** file in the /var/nwreg2/{local | regional}/tomcat/conf folder.

Step 2 Use the below recommended sslEnabledProtocols and ciphers, or configure it as per your security requirement. For more details, refer the tomcat SSL/TLS Configuration document available online.

```
<Connector port="${cnrui.https.port}" protocol="com.cisco.cnr.webui.tomcat.SecureHTTP"

relaxedQueryChars='[]'

maxConnections="1024" maxThreads="150" SSLEnabled="true" scheme="https" secure="true"

clientAuth="false"

keystoreFile="..."

keystorePass="..."

ciphers="TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256, TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256,
TLS_RSA_WITH_AES_128_GCM_SHA256, TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256,
TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256, TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384,
TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384, TLS_RSA_WITH_AES_256_GCM_SHA384,
TLS_DHE_RSA_WITH_AES_128_GCM_SHA256, TLS_DHE_RSA_WITH_AES_128_CBC_SHA,
```

```
TLS_DHE_RSA_WITH_AES_128_CBC_SHA256, TLS_DHE_RSA_WITH_AES_256_GCM_SHA384,
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA, TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256,
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA, TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256,
TLS_RSA_WITH_AES_128_CBC_SHA256, TLS_RSA_WITH_AES_128_CBC_SHA,
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384, TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384,
TLS_RSA_WITH_AES_256_CBC_SHA256, TLS_RSA_WITH_AES_256_CBC_SHA"

compression="on"

compressionMinSize="2048"

noCompressionUserAgents="gozilla, traviata"

URIEncoding="UTF-8"

compressableMimeType="text/html,text/xml,text/plain, text/css,text/javascript,
application/x-javascript,application/javascript"

sslEnabledProtocols="TLSv1.2"/>
```

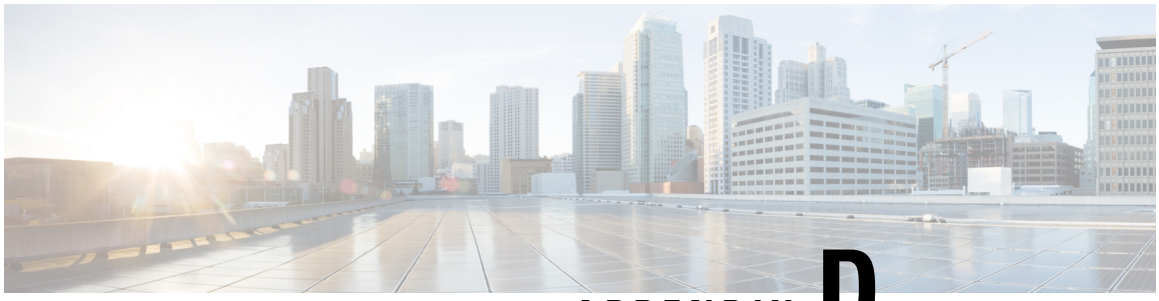
Note
The **keystoreFile** and **keystorePass** values are specific to your installation. You should not change these values as it will be overwritten each time Cisco Prime Network Registrar is started.

Step 3 Restart Cisco Prime Network Registrar for the changes to take effect.

Note Cisco Prime Network Registrar 11.2 supports TLS 1.3 with below ciphers:

- TLS_AES_256_GCM_SHA384
- TLS_CHACHA20_POLY1305_SHA256
- TLS_AES_128_GCM_SHA256
- TLS_AES_128_CCM_8_SHA256
- TLS_AES_128_CCM_SHA256

If you want to leverage TLS 1.3, you need to update the server.xml file appropriately and restart Cisco Prime Network Registrar.



APPENDIX D

Hardening Guidelines

This appendix contains the following section:

- Hardening Guidelines, on page 79

Hardening Guidelines

If you consider hardening the system, you should consider the following hardening guidelines:

- Refer to the host platform's hardening guides. For example:
  - RHEL 8.x:

    https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/pdf/security_hardening/Red_Hat_Enterprise_Linux-8-Security_hardening-en-US.pdf

    https://www.cisecurity.org/benchmark/red_hat_linux/
  - NSA hardening guide collection:

    https://www.nsa.gov/ia/mitigation_guidance/security_configuration_guides/operating_systems.shtml

Note The above links reference external websites and Cisco is not responsible for keeping them up-to-date. They are provided for reference only. If you find that the content is outdated or if you cannot access the links, please contact the website owner for updated information.

- Disable or block the ports that are not used by Cisco Prime Network Registrar. The Cisco Prime Network Registrar documentation outlines the port usage and also the issues with using firewall items, such as connection tracking.
  - For a list of ports used by Cisco Prime Network Registrar, see the *"Default Ports for Cisco Prime Network Registrar Services" section in the Cisco Prime Network Registrar 11.2 Administration Guide*. Note that some are defaults and may have been changed during install or configuration.
  - For connection tracking related issues, see the *"DNS Performance and Firewall Connection Tracking" section in the Cisco Prime Network Registrar 11.2 Administration Guide*.

- Confirm that any product directories (primarily, /opt/nwreg2/* and /var/nwreg2/*) are locked down as appropriate. Note that you may need to adjust the protection based on your needs (such as for performing offline backups and viewing logs).
- DNS specific considerations include:
  - Use DNS Security Extensions (DNSSEC):

    DNSSEC provides origin authority, data integrity, and authenticated denial of existence. With DNSSEC, the DNS protocol is much less susceptible to certain types of attacks, particularly DNS spoofing attacks. DNSSEC provides protection against malicious or forged answers by adding digital signatures into DNS data, so each DNS response can be verified for integrity and authenticity.

    Cisco Prime Network Registrar 9.0 and earlier Authoritative DNS Server do not support signing of zones. Starting from Cisco Prime Network Registrar 10.0, Authoritative DNSSEC support adds authentication and integrity to DNS zones. With this support, Cisco Prime Network Registrar DNS server is able to support both secure and unsecure zones. For more information, see the *"Managing Authoritative DNSSEC" section in the Cisco Prime Network Registrar 11.2 Authoritative and Caching DNS User Guide*.
  - Secure DNS server activity with ACLs:
    - Restricting Zone Queries—The *restrict-query-acl* attribute on the DNS server serves as a default value for zones that do not have *restrict-query-acl* explicitly set.
    - Restricting Zone Transfer Requests—Use the *restrict-xfer-acl* attribute to filter the zone transfer request to the known secondary servers.
    - Restricting DDNS Updates—Use the *update-acl* attribute to filter DDNS packet from the known DHCP servers.
  - Secure zone transfers and DNS updates using TSIG or GSS-TSIG:

    Zone transfer in secure mode supports both HMAC-MD5 based TSIG and GSS-TSIG. You can add an optional TSIG key or GSS-TSIG keys (see the *"Transaction Security" or "GSS-TSIG " sections in the Cisco Prime Network Registrar 11.2 DHCP User Guide*) to the primary server address by hyphenating the entry in the format *address–key*. For each entry, click **Add IP Key**.

    For more information, see the *"Creating a Zone Distribution" section in the Cisco Prime Network Registrar 11.2 Authoritative and Caching DNS User Guide*
  - Randomize Query IDs and Source Ports.
  - DNS Rate Limiting—See the *"Managing Caching Rate Limiting" section in the Cisco Prime Network Registrar 11.2 Authoritative and Caching DNS User Guide*.
  - Separate Recursive Server and Authoritative Server roles.
- DHCP specific considerations include:
  - Assure DHCPv4 and DHCPv6 traffic from the "external" sources is blocked on routers and that only valid relay agents are enabled to forward packets to the DHCP servers.
  - Use DHCP Guard and similar services on switches:

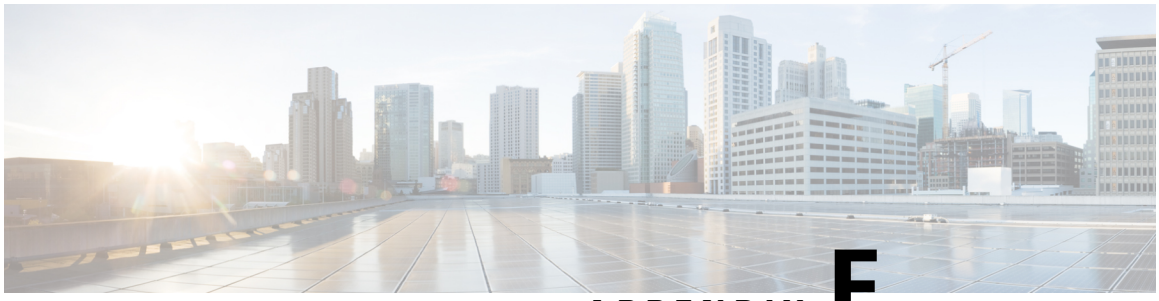
    See https://www.cisco.com/c/en/us/td/docs/switches/datacenter/sw/4_1/nx-os/security/configuration/guide/sec_nx-os-cfg/sec_dhcpsnoop.html

See https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/ipaddr_dhcp/configuration/15-sy/dhcp-15-sy-book/ip6-dhcpv6-guard.pdf

- Use the Chatty Client Filter—See the *"Preventing Chatty Clients by Using an Extension" section in the Cisco Prime Network Registrar 11.2 DHCP User Guide*.

- Consider using external user authentication as password rules (that is, change frequency, length, and difficulty checks) can typically be implemented for Active Directory (LDAP) and RADIUS users. See the *"External Authentication Servers" section in the Cisco Prime Network Registrar 11.2 Administration Guide*.

Hardening Guidelines



APPENDIX E

Optimizing VM Performance

See the following sections for optimizing VM performance:

- Recommended UCS Settings, on page 83
- NUMA Optimization, on page 83
- Hyperthreading Considerations, on page 83

Recommended UCS Settings

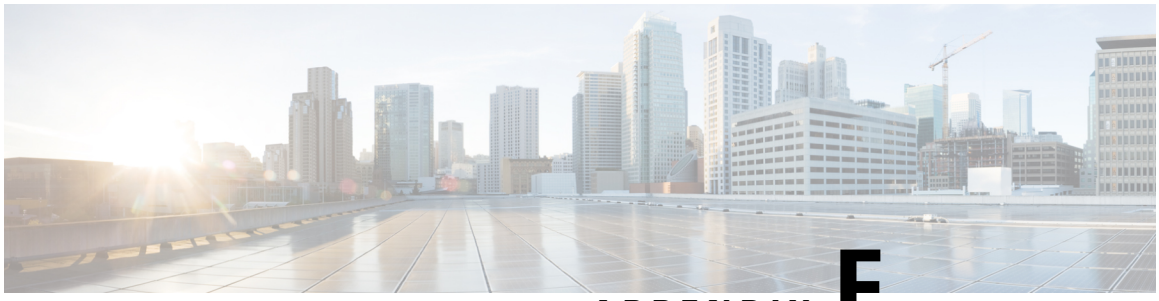
On UCS servers with RAID configured, for improved performance, it is recommended to set the Requested Write Cache Policy on the RAID controller as **Write Back** instead of **Write Through** (the default setting). The downside of using the Write Back option is that you may lose some data if a system failure occurs before the data in the cache is written to disk. Therefore, we recommend to set the Requested Write Cache Policy on the RAID controller to **Write Back Good BBU**. In this mode, the controller enables Write Back caching when the Battery Backup Unit (BBU) is installed and charged. It provides a good balance between data protection and performance.

NUMA Optimization

If you do not configure the virtual CPUs correctly, you may run into Non-Uniform Memory Access (NUMA) performance issues. To avoid this issue, do not configure a virtual machine from using more virtual CPUs than a single NUMA node. Otherwise, it will be scheduled across multiple NUMA nodes causing memory access degradation. Generally, this means to assign no more virtual CPUs to a virtual machine than the total number of physical cores of a single CPU socket.

Hyperthreading Considerations

When using hyperthreading virtual CPUs, note that the general CPUs utilization is 30% not 100% as threading allows for other work to be done when the main thread is stalled waiting for something. The exact numbers may be different as it depends on the workloads.



APPENDIX F

Configuring Network Access on RHEL/AlmaLinux 8.x Using nmcli

This appendix contains the following section:

- Configuring Network Access on RHEL/AlmaLinux 8.x Using nmcli, on page 85

Configuring Network Access on RHEL/AlmaLinux 8.x Using nmcli

The **NetworkManager** command-line tool (**nmcli**) provides a command line way to configure networking by controlling NetworkManager. This section provides only an overview with some examples to help you learn how to use nmcli to configure network access on the virtual appliance.

In a departure from previous approaches to network interface configuration, NetworkManager deals with both connections and interfaces (also known as devices). Connections are configured with IP addresses, gateways, DNS servers, and then applied to interfaces (devices).

There are two nmcli commands that are of general usefulness:

- The **nmcli d** command lists all available network interfaces (devices).
- The **nmcli c** command lists all available configurations.

Use the above two commands frequently as you learn to use nmcli.

Follow the steps below to configure an IP address for an interface on your virtual appliance. Typically, these commands are typed directly into the console of the virtual appliance. If you are already connected through the network (for example, by **ssh**), then making changes to the network interface configuration can be problematic, as you may also lose network connectivity (and thereby your ability to issue nmcli commands) at any point in the process.

Procedure

Step 1 Make sure that the interface does not block nmcli. The **nmcli d** command lists the existing interfaces. If the interface you want to configure is listed as **unmanaged**, then NetworkManager has been explicitly blocked from configuring this interface. Until you remove this blockage, no nmcli command will have any effect on this interface. Note that you may

not need to perform this procedure unless the interface is listed as **unmanaged**. Follow the steps below to allow it to be managed by NetworkManager:

a) Remove the line NM_CONTROLLED-no from the file **/etc/sysconfig/network-scripts/ifcfg-***interface*, where *interface* is the interface name listed in the **nmcli d** command. If there is no file with this name, then you do not need to perform this procedure.
b) Tell the NetworkManager to read the configuration files again by using the following command:

```
nmcli connection reload
```

Note
Manual changes to any **ifcfg** file will not be noticed by NetworkManager until the **nmcli connection reload** command is issued.

Step 2 Make sure that there is no current configuration for the interface that you want to configure. If you want the configuration that you create to be the default for the interface and there are multiple configurations associated with an interface, it may lead to confusion when the system reboots. The **nmcli c** command lists the existing configurations. If you see any existing configurations, examine them to see if they apply to the interface you want to configure. An easy way to do this is to use the following command:

```
nmcli con show config | grep interface
```

If you see any output, you should remove the configuration *config* using the following command:

```
nmcli con delete config
```

Note
There is often a configuration called "Wired connection 1" which needs to be deleted.

Step 3 Create the configuration and associate it with the interface (device) in one command. This command only creates the configuration and associates it with the interface, it does not apply it to the interface.

```
nmcli con add type ethernet con-name config ifname interface ip4 ip/netmaskwidth gw4 gateway
```

where *config* is the name of the configuration, which can be anything (including the name of the interface), *interface* is the name of the interface (device), *ip* is the IPv4 address, *netmaskwidth* is the network mask width, and *gateway* is the IPv4 gateway address.

For example (type all in one line):

```
nmcli con add type ethernet con-name my-office ifname ens160 ip4 10.10.24.25/24 gw4 10.10.20.174
```

Step 4 Add the DNS server to the configuration for the interface (device):

```
nmcli con mod config ipv4.dns dnsip
```

where *dnsip* is the IPv4 address of the DNS server and *config* is the name of the configuration.

For example:

```
nmcli con mod my-office ipv4.dns 72.63.128.140
```

You can add two DNS addresses as given below:

```
nmcli con mod my-office ipv4.dns "72.63.128.140 72.63.111.120"
```

Note
This will replace any previously set DNS servers. To add to an previously set DNS entry, use the + before ipv4.dns as shown below:

```
nmcli con mod test-lab +ipv4.dns "72.63.128.140 72.63.111.120"
```

Step 5 Apply the configuration to the interface, which will bring up the interface if it was not already running:

```
nmcli con up config
```

where *config* is the name of the configuration.

Step 6 Use the following command to examine information about a connection:

```
nmcli -p con show config
```

This will typically scroll off of the console screen, leaving the beginning unreadable. To allow you to move back and forth and examine the output easily, use the following command:

```
nmcli -p con show config | less
```

From this, you can see the entire configuration. You can modify things in the configuration using the following command:

```
nmcli con mod config something.other new-value
```

For example:

```
nmcli con mod my-office wifi-min.key-cntl wpa-psk
```

Step 7 Use the **set-hostname** command to set the hostname for the system:

```
hostnamectl set-hostname hostname.domain
```

Note
This must be done before registering the local to the regional. Otherwise, an error will result about "localhost" already existing.

where *hostname* is the hostname you want to use and *domain* is the domain name, ending with .com, .org, and so on. It is important to include the domain name (along with the .com, .org, or whatever ending is appropriate), since this is used as the default for DNS lookups.

For example:

```
hostnamectl set-hostname my-server.gooddomain.com
```

Step 8 After you configure the networking, you **must** restart Cisco Prime Network Registrar for the interfaces to be properly discovered by Cisco Prime Network Registrar. Use the following commands to restart:

- For local cluster:

  ```
  # systemctl restart nwreglocal
  ```

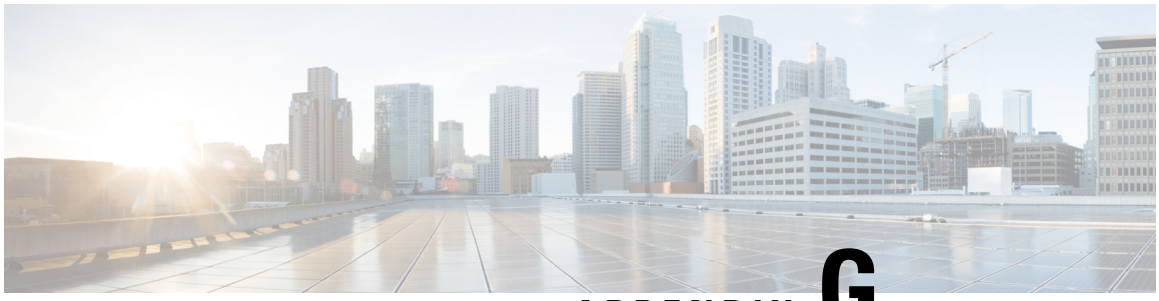
- For regional cluster:

  ```
  # systemctl restart nwregregional
  ```

If you fail to restart, it will result in a misconfigured registration at the regional.

To develop a complete understanding of the usage of nmcli, search the Internet for online resources on nmcli and AlmaLinux 8.x.

Configuring Network Access on RHEL/AlmaLinux 8.x Using nmcli



APPENDIX G

Authoritative DNS Capacity and Performance Guidelines

This chapter provides information on Authoritative DNS capacity and performance guidelines for Cisco Prime Network Registrar.

- DNS System Deployment Limits, on page 89
- DNS Database Architecture, on page 90
- DNS System Sizing, on page 91

DNS System Deployment Limits

Cisco Prime Network Registrar makes the following recommendations on maximum Authoritative DNS System configuration sizes. The following recommendations are as per Cisco Prime Network Registrar Authoritative DNS server which can be a primary, primary HA, or secondary server. A redundant DNS architecture will contain multiple of these types of servers all servicing the same data. Therefore, the capacity can be expanded horizontally by introducing a new set of servers. These recommendations are guidelines to ensure a properly functioning DNS deployment.

Note DNSSEC enabled zones (Cisco Prime Network Registrar 9.1 and later versions) will include auto-generated RRs that significantly increase the number of RRs in the zone.

- Maximum of 25 million RRs per Authoritative DNS server (primary, HA pair, or secondary server), ideally not to exceed 2 million RRs per zone. Multiple DNS primary servers can be used for deployments requiring more RRs.
- Maximum of 10000 zones per Authoritative DNS server (primary, HA pair, or secondary server). Multiple DNS primary servers can be used for deployments requiring more zones.
- Maximum of 4 secondary servers per primary or HA pair.
- Maximum of 2 tiers of secondary servers (first tier secondaries and second tier secondaries).
- Maximum of 2 second tier secondary servers per first tier secondary server.

DNS Database Architecture

The Authoritative DNS servers utilize a combination of in-memory cache and on-disk databases to store and maintain authoritative RR data. For sizing purpose, assume an each RR requires 300 bytes of memory for the RR cache and 300 bytes of disk space for the RR DB. The CSET DB has a higher disk space requirement for each RR since it records changes to the RR set, but those changes are capped to the number of history changes kept per zone.

RR DB

- Database that stores all RRs (protected and unprotected) for the zones configured on a DNS server.
- On primary DNS servers, RR data edits are written to the RR DB either through administrative actions (that is, RR adds), or DNS updates and zone scavenging. On secondaries, the RR DB is written through zone transfers.
- The RR DB is required for all ADNS servers (primary/secondary).

RR Cache

- Increases query performance by storing a subset of the RR DB data (stores entire name sets).
- Most active RR data is stored to RR cache dynamically as part of RR DB lookups generated by DNS query processing.
- The memory foot print of the RR Cache is capped by a configurable DNS server attribute (*mem-cache-size*). When the maximum cache size has been reached, the DNS server will remove older entries from the cache to make room for newer entries. Each RR requires approximately 300 bytes of memory.
- DNS server reload/restart causes the RR cache to be deleted. When the server starts up again, it is rebuilt based on query traffic.
- The RR cache is required for all ADNS servers (primary/secondary).

CSET DB

- Database that stores RR changes (adds, deletes, protection changes, and refreshes) needed to respond to the incremental zone transfer requests (IXFRs).
- RR changes are first stored in the RR DB and then persisted to the CSET DB.
- For DNS servers that do not need to service incremental zone transfers (that is, secondaries that do not send outbound IXFRs), server performance can be increased by disabling persisted change sets (*csetdb-persist-csets*). By default, changes are automatically persisted to the CSET DB.
- DNS maintains only a limited configurable number of changes (*csetdb-htrim-max-cset-kept*) and automatically trims entries when the maximum has been reached. Trimming helps limit the database size. For deployments with DNS updates, it is recommended that the number of changes kept is increased to avoid full zone transfers.
- If the CSET DB is deleted, the DNS server will create an empty database and respond with full zone transfers (AXFRs) until new zone history data is populated into the database.

HA DB

- Database that stores state information about the DNS HA pair as well as data about RR changes during a communications interrupted or partner down event.
- Only applicable on primary HA DNS servers (main and backup).
- If the HA DB is deleted, HA synchronization causes all zone data to be pushed from the HA main to the HA backup.

DNS System Sizing

A Cisco Prime Network Registrar DNS deployment can be categorized as small, medium, or large depending on the number of RRs/zones, DNS update activity, and recovery time during an outage or update. The number of zones can have an impact on the size of the deployment, primarily the number of RRs is the deciding factor. Also, if the DNS deployment requires a large number of RRs/zones, it is recommend that multiple DNS deployments be used - ideally segregating the data appropriately so that related zones/RRs are configured together.

Note To ensure a properly functioning Authoritative DNS system, it is important to monitor system disk space and memory. If the Authoritative DNS server runs out of memory, it will crash. If it runs out of disk space, it will no longer be able to service requests and the databases may become corrupt and unusable.

Regional Management of DNS Deployments

The regional server provides license management of all Cisco Prime Network Registrar local clusters, and allows for central management and replication of Cisco Prime Network Registrar DNS deployments. Follow the below recommendations for system sizing and configuration adjustments to be made when using regional DNS cluster management:

- A minimum of 4 CPUs
- A minimum of 8 GB of RAM
- Disk space should be at minimum an aggregate of the disk size of all the managed DNS (main) primary clusters.
- On large DNS deployments, replication of unprotected RRs should be disabled (*poll-replica-rrs*).

Small Deployment

- 1-1000 RRs and 1-100 zones
- Mainly static data; zone edits are primarily done by administrators.
- Typically consists of one primary and a secondary server.
- DNS Caching server is not required or can be handled by hybrid mode.
- DNS can be recovered from a shadow backup within a matter of minutes with little to no impact on production.

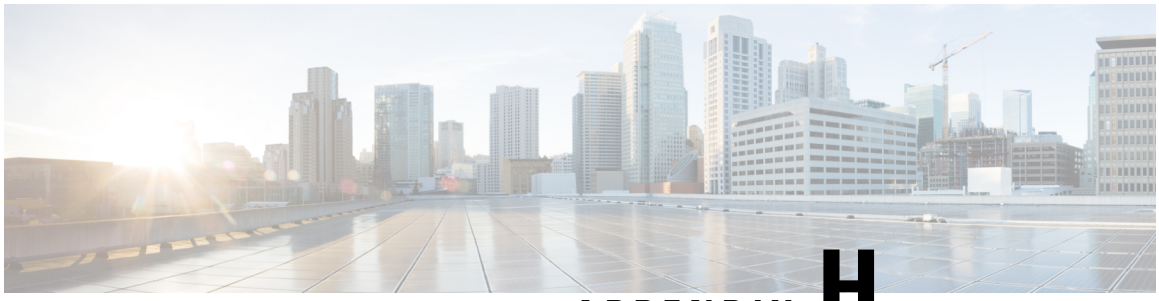
- A minimum of 2 CPUs
- A minimum of 4 GB of RAM
- A minimum of 10 GB of disk space

Medium Deployment

- 1000-100,000 RRs and 100-1000 zones
- A pretty even mix of static and dynamic data; 100 updates per second or less.
- Typically consists of one primary and two to four secondaries.
- Typically consists of two to four DNS Caching Servers. DNS Caching Servers must be deployed on separate machines or VMs.
- DNS can be recovered from a shadow backup within an hour with minimal impact to production.
- A minimum of 4 CPUs
- A minimum of 8 GB of RAM
- A minimum of 25 GB of disk space. On the primaries, the number of change sets kept (*csetdb-htrim-max-cset-kept*) should be increased. The value will depend on how many DNS updates are handled by the system, but should be between 1000 and 5000.

Large Deployment

- 100,000-25,000,000 RRs and 1000-10,000 zones
- Dynamic data makes up a larger percentage of the data; thousands of updates per second.
- Typically consists of two primaries (DNS HA pair) and four secondaries.
- Typically consists of four or more DNS Caching servers.
- DNS recovery is complex and must be done during a maintenance window; DNS servers can take an hour or more to recover from a shadow backup.
- A minimum of 8 CPUs
- A minimum of 16 GB of RAM. The DNS RR cache memory size (*mem-cache-size*) should be increased (approximately 300 bytes per RR, but not to exceed 2,000,000 KB).
- A minimum of 100 GB of disk space. On the primaries, the number of change sets kept (*csetdb-htrim-max-cset-kept*) should be increased. The value will depend on how many DNS updates are handled by the system, but should be between 5000 and 10,000.



APPENDIX H

Caching DNS Capacity and Performance Guidelines

This chapter provides information on Caching DNS capacity and performance guidelines for Cisco Prime Network Registrar.

- DNS System Deployment Limits, on page 93
- Caching DNS System Sizing, on page 94
- Possible Impacts on Caching DNS Server Performance, on page 95

DNS System Deployment Limits

Cisco Prime Network Registrar makes the following recommendations on maximum Caching DNS System configuration sizes. A redundant DNS architecture will contain multiple servers, therefore the capacity can be expanded horizontally by adding on new servers. Although Cisco Prime Network Registrar does not put hard limits on many of its configuration objects, these recommended maximums are to ensure a properly functioning DNS deployment.

- Maximum of 100 DNS Views
- Maximum of 500 Exceptions and Forwarders
- Maximum of 3 DNS RPZ Firewall Objects. Note that the RPZ zones can have many thousands of entries.
- Maximum of 12 DNS Firewall Objects (non-RPZ) with no more than 200 domains each
- Maximum of 30 DNS64 Objects

Note To account for situations where one or more servers are unavailable due to maintenance or outage, it is recommended to include excess capacity in the deployment architecture to accommodate the additional load that must be borne by the remaining live systems. The excess capacity to be deployed or the number of backup systems, will depend on the level of redundancy that you want to achieve. A minimum of n+1 redundancy is recommended.

Caching DNS System Sizing

A Cisco Prime Network Registrar Caching DNS deployment can be categorized as small, medium, or large depending on the number of servers and query load. The following sections are an indication of how to provision the Caching DNS server based on the deployment size.

Note To ensure a properly functioning DNS system, it is important to monitor system disk space and memory.

Small Deployment

- Typically consists of 2-4 DNS Caching servers. DNS Caching server maybe co-located with the DNS Authoritative server using hybrid mode.
- Typically less than 1,000 Queries per second
- A minimum of 2 CPUs
- A minimum of 4 GB of RAM
- A minimum of 10 GB of disk space

Medium Deployment

- Typically consists of 2-4 DNS Caching servers. DNS Caching servers must be deployed on separate machines or VMs.
- Typically between 1,000 and 50,000 queries per second
- A minimum of 4 CPUs
- A minimum of 8 GB of RAM
- A minimum of 25 GB of disk space

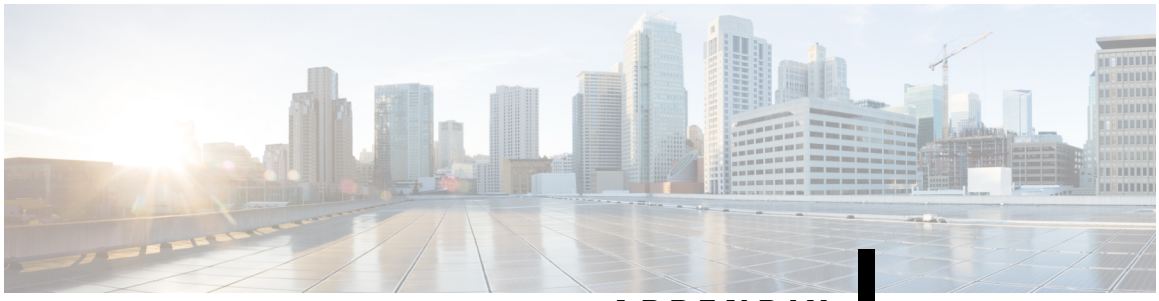
Large Deployment

- Typically consists of 4 or more DNS Caching servers.
- Typically more than 50,000 queries per second
- A minimum of 8 CPUs
- A minimum of 32 GB of RAM. The Caching DNS RR cache settings are *msg-cache-size* and *rrset-cache-size*, and they may both be increased to 4,294,967,295 bytes.
- A minimum of 100 GB of disk space

Possible Impacts on Caching DNS Server Performance

The following is a list of common system components and Cisco Prime Network Registrar configurations that may have an impact on performance:

- Firewalls and Connection Tracking may have a negative impact on performance especially in medium to large deployments where the firewall may drop a significant amount of DNS traffic.
- Excessive logging—Either enabling too many log settings, packet logging, or debug logging can decrease server performance.
- IPv6 only networks configured to also use IPv4. IPv6 networks should be configured in IPv6 only mode in order to prevent the server wasting cycles on failed IPv4 communication.
- IPv4 only networks configured to also use IPv6. IPv4 networks should be configured in IPv4 only mode in order to prevent the server wasting cycles on failed IPv6 communication.



APPENDIX

DHCP Capacity and Performance Guidelines

This section provides information on DHCP capacity and performance guidelines for Cisco Prime Network Registrar.

The goal of this section is to provide an understanding of what influences the capacity and performance of the servers to help in planning how to deploy the product and what to consider when purchasing hardware for these systems.

When multiple clusters are running on virtual machines, the underlying physical hardware needs to be at least the sum of the individual virtual machine requirements. Also, it should be noted that high availability solutions (that is, HA-DNS or DHCP failover) should not have both partners located on the same physical machine in virtual environments, as that makes the hardware a single point of failure.

Note These are just guidelines, as actual performance may vary based on variances in the live deployment.

- Local Cluster DHCP Considerations, on page 97
- Regional Cluster DHCP Considerations, on page 102

Local Cluster DHCP Considerations

There are two common questions concerning DHCP capacity:

1. How many leases can I put on a single server?
2. If I want to put *n* leases on a server, what sort of server should I purchase or virtual machine should I configure?

Number of Leases Allowed on a Single Server

When discussing about the capacity of a server, the number of DHCP operations per second that the server can support is the most important issue. There are two regimes that affect the operations per second that the server will be required to support:

- **Steady state**: Made up of existing DHCP clients renewing their leases and the arrival of DHCP clients not previously seen by the server.

- **Avalanche**: Made up of a large (possibly vast) quantity of existing DHCP clients, all contending at the DHCP server to get an address. This situation can occur with restoration of power after a failure or perhaps a blanket reset of many customer devices. This can often consist of tens of thousands of DHCP clients all trying to get an IP address from the DHCP server at the same time. It can even be hundreds of thousands of DHCP clients trying to get an IP address.

For the steady state situation, the number of DHCP clients and the lease times of the leases they are granted will dominate the load.

The operations per second required by a DHCP client population is largely driven by the size of that client population coupled with the lease times (both expiration and renewal times) that are granted to that population. These values are all configurable, and so the actual requirements can vary dramatically.

Following table presents a range of these data points showing the operations per second required for various client populations and differing lease times:

Table 5: Client lease Times

Operations per Second						
	Client Lease Times					
Active Leases	30 min	1 hr	1 day	1 week	2 weeks	30 days
1,000	1	1	-	-	-	-
10,000	11	6	-	-	-	-
100,000	111	56	2	-	-	-
500, 000	556	278	12	2	1	-
1,000,000	1,111	556	23	4	2	1
1,500,000	1,667	833	35	5	2	1
2,000,000	2,222	1,111	46	7	3	2
4,000,000	4,444	2,222	93	13	7	3
6,000,000	6,667	3,333	139	20	10	5

The lease times granted to the clients has an overwhelming influence on the steady state operations per second required on the DHCP server. A server's operations likely include a mix of lease times, as lease times for clients without an existing lease are limited by the failover Maximum Client Lead Time (MCLT), and there may be other operations (such as from "bad" clients or lease query requests).

The DHCP server will not collapse under any client load, but it can take seconds to minutes to work through tens or hundreds of thousands of clients. It is for this reason that our recommendations for the operations per second that the server is required to support in steady state tends to be on the lower side; so that the server has plenty of headroom to process the eventual avalanche.

DHCP operations per second

It is difficult to give concrete recommendations regarding the operations per second that the DHCP server can deliver to DHCP clients, since there are many factors that are involved in this aspect of DHCP server performance.

Cisco has measured DHCP server performance in the lab well above 20,000 operations per second. However, that was a DHCP server which was configured specifically for maximal performance (no failover, no logging, no lease history, no extensions, and no LDAP). Almost every feature that you configure in the DHCP server costs some amount of performance; frequently trimming 10 percent or so off of the previous performance. Some features, for instance LDAP lookup or running with the Prime Cable Provisioning (PCP) product, can have a much bigger effect on performance; since the LDAP lookup or PCP interaction with the DPE requires interlocking with a separate server and the round-trip delays that entails, prior to even processing the incoming DHCP request. Failover costs at least 10 percent, basic logging can also cost 10 percent of performance or more. Extensions will cost an unpredictable amount on top of a constant overhead to just call the extension. The time spent in the extension is also synchronous and additive to the time it takes to process every DHCP request.

The upshot of all of this is that there is no way to reasonably predict the operations per second that the DHCP server will be able to supply given a particular load when running on a particular hardware configuration with a particular software configuration.

Also, the operations per second load placed on the DHCP server by the constant requirement to process DHCP RENEW requests from DHCP clients ("steady state") is frequently overshadowed by the requirements to process large "avalanche" loads, where many thousands to tens of thousands of DHCP clients attempt to get service from the DHCP sever in a very short time. These events can be generated by a power outage among the DHCP clients or network element resets that will provoke many thousands of DHCP clients to re-DISCOVER / re-SOLICIT for IP addresses. The DHCP server needs to be able to process these loads, which typically dwarf the loads generated by the steady state RENEWAL traffic.

Cisco recommends that the steady state load on the DHCP server be limited to a few hundred operations per second, in part to ensure that headroom exists to process the avalanche loads presented to the DHCP server in unusual circumstances. We have customers which have high performance hardware and excellent monitoring regimes that run with several hundred operations per second and sometimes more with constant load. They are running successfully, in part because they are careful to ensure that they do not let the avalanche load size get too large; by limiting the number of active leases on each server.

The DHCP server has several features to reduce the load on the server and help it service requests as quickly as possible, especially under avalanche conditions:

- **Defer-lease-extension**

  By default, the server will defer extending a lease to a client if the client "renews" before its expected renewal time. This usually helps out with avalanches if the outage that triggered it was short (less than 1/2 the lease time) as a large number of clients will avoid the need for a disk write (and failover update).

- **Reduced logging when overloaded**

  By default, the server will reduce the logging when the request buffers in use exceeds 67 percent of the configured buffers. As logging can be costly, this allows the server to handle additional capacity when very busy. This feature can be disabled. Note that the server dropping requests under avalanche conditions should be expected, as that is the only way that the server can shed load, and the client will re-transmit the request. Under steady state conditions, if a server is frequently dropping requests, that is probably an indication that it is unable to handle the load.

- **Chatty Client Filter**

Use of this provided extension is highly recommended in all service provider networks. This extension monitors client activity and blocks those clients that are considered to be "chatty". Once a client is blocked, it is unblocked if it quiets down. In many service provider networks, the Chatty Client Filter can reduce the requests to the server by about 50 percent. However, the Chatty Client Filter requires careful tuning and requires reviewing that tuning periodically to assure traffic patterns have not changed. For more details, see the *"Preventing Chatty Clients by Using an Extension" section in the Cisco Prime Network Registrar 11.2 DHCP User Guide*.

- **Discriminating Rate-Limiter**

  The Discriminating Rate-Limiter reduces downtime after an outage in service networks by restricting the rate of DISCOVER and SOLICIT requests while still honoring all RENEW requests. The basic concept is to assure a client that was offered a lease is able to complete getting that lease. For more details, see the *"Setting Advanced DHCP Server Attributes" section in the Cisco Prime Network Registrar 11.2 DHCP User Guide*.

Number of leases you want on a server

If the only thing that mattered was the steady state operations per second load, then looking at the table above and with a one week lease time, you could imagine 12 million or even 24 million leases would pose no problem. However, there are other factors as follows:

- **Avalanche load**: Which may or may not scale with the total leases on a server.
- **Reload time**: The server needs to refresh its in-memory cache whenever it is reloaded, and the reload time scales linearly with the number of active leases in the server.
- **Service interruption impact**: If you have millions of leases to start with, then there is probably a relationship between DHCP clients and customers of some sort. You probably want to avoid having a DHCP server have so many leases that having an entire DHCP failover pair out of service for a few hours would cause an unacceptable risk to your business. While DHCP failover will prevent almost all service interruptions and you probably have no single points of failure, sometimes two things do fail at the same time. It is possible that both servers in a DHCP failover pair will fail for a while, and in the unlikely event that this should happen, the difference between having 2 million DHCP clients on a server and 10 million DHCP clients on a server could be very important. With the reasonable DHCP lease times, only some small percentage of DHCP clients will have their leases expire every hour that a failover pair is out of service.

Recommendations

Cisco strongly recommends that you limit the total active leases on a single DHCP server (or server failover pair) to 6 million leases. In addition, Cisco strongly recommends that you limit the steady-state operations per second requirement to 500 operations per second, in order to have sufficient bandwidth to handle avalanche and other exceptional conditions.

Scale out, not up, beyond some point!

Instead of loading vast quantities of leases into a single DHCP server or failover pair, consider keeping the number of leases to a more modest number, say 3 to 5 million leases. Cisco resource limits set the warning level to be 6 million leases, and it is wise to configure more like 4 million leases per server to allow for growth in the future. While managing multiple failover pairs is more work than just managing one failover pair, the ease of management of a server that is more modestly loaded with 3 to 4 million leases will pay long term

dividends, to say nothing of the impact on your business in the unlikely event that an entire server pair should fail for a couple of hours.

Request Latency

It should be noted that the DHCP server's design is optimized to respond to large numbers of requests quickly – it is not optimized to have the lowest latency for each request. This often complicates testing for scale as the server's performance with a few simultaneous requests may not show its true processing power.

Server Considerations

If you do not need a lot of operations per second and do not have a lot of leases on the server, pretty much any server will do. For the purpose of this discussion, we will assume that you want to get the maximum performance possible.

For DHCP, the general recommendations in terms of physical or virtual server considerations are as follows:

1. Disk write performance is the primary consideration. SAN storage or SSD disks are recommended. The DHCP server is disk write performance limited, because it must commit to disk any changes to leases (primarily assigning a lease to a new client and extending the lease times on a lease) before responding to a client. Configuration options, such as failover, lease history, and DNS updates also increase the disk write load on the server, as each of these require additional write operations. There are up to 4 writes for a lease on the server that grants, extends (renew/rebind), releases, or expires a lease plus 1 more write on the failover partner as follows:
   - The lease itself (before responding to the client). Generally, this also results in a failover binding update if failover is used.
   - A history record (this only occurs if lease history is enabled and the lease was leased but is no longer).
   - The partner writes the lease when it receives a failover binding update (if failover used).
   - The lease after the receipt of the failover binding update acknowledgement (if failover used).
   - The lease after the DNS update completes (if configured and initiated for the lease).

   A server may also initiate writes at other times for a lease, such as for failover state transitions for the lease, when balancing failover pools, and because of user action (such as to force a lease available). The DHCP server lease state database disk space requirements are generally as follows:
   - 1 KB for each configured or active lease, and
   - If lease history is enabled, 1 KB for each historical record.

   These numbers can be reduced about 30 percent if the lease record compression is enabled (see the DHCP server's *server-flags* attribute).

Note These numbers need to be multiplied by 3 to accommodate the shadow backups. These numbers just reflect the lease state database and no other system requirements.

2. Memory (RAM) is secondary, with 64-bit support, memory limits are not generally a concern provided the system has sufficient memory. It is important to have sufficient "free" memory for the file system to

be able to keep the entire DHCP lease state database in memory to avoid the need for disk reads. A rough rule of thumb is to assume:

- 1 KB for each configured or active lease for the DHCP server's memory usage. Configuration options, such as DNS update and the length of host and domain names and the amount of option-82 (DHCPv4) or Relay-forward message (DHCPv6) data can influence this rule of thumb.
- 1 KB of "free" memory for the file system cache for each lease (configured or active) and,
- If lease history is enabled, 1 KB of "free" memory for the file system cache for each history record (this will be more difficult to judge as it depends on how frequently leases expire or are released).

3. CPU performance is the least significant as the processing required to service requests is generally low. On the other hand, avalanche processing is largely handled with just CPU cycles and minimal disk writes. So, if you have a large avalanche possibility, invest in a system with good CPU capability and fast network interfaces. Most modern multi-processor systems should be sufficient for modest avalanche loads. For higher capacity/performance applications, both the CPU speed and number of effective processors should be higher. The DHCP server is highly multi-threaded, so that, additional CPU cores will help DHCP server performance up to a point. Due to the requirements for some minimal amount of locking inside the DHCP server, performance will improve when adding up to 12 CPU cores. Beyond 12 CPU cores, there is not much of any performance improvement due to the requirements for synchronization.

Regional Cluster DHCP Considerations

The regional cluster disk space requirements are dictated by several factors for DHCP:

1. **Lease history**—When lease history is enabled at the local clusters, by default, the regional cluster collects this history from the local clusters for longer term storage (the default is to retain these records for 24 weeks, see the CCM server's *trim-lease-hist-age* attribute). As mentioned above for the DHCP server, each lease record (active and historic) should be assumed to require about 1KB, but this should be multiplied by 3 to accommodate backup requirements – thus, 3 KB/lease record. The regional cluster disk space needed will depend on the total number of lease history records, which depends on the number of servers, their lease counts and client activity levels, and the period of time over which the history is to be retained. In very large service provider networks, this can easily be 100 GB or more.

Note These disk space requirements can be reduced to 30 percent for the lease history data by enabling lease record compression in Cisco Prime Network Registrar 9.0 and later (see the CCM server's *lease-hist-compression* attribute).

2. **Network utilization**—The regional cluster also collects subnet and prefix utilization data from the local clusters (by default, every hour and retained for 24 weeks; see the CCM server's *addrutil-poll-interval* and *addrutil-trim-age* attributes). While each record is about 1/2 KB (the scope/prefix names, owner, region, selection tags, and other data cause the size to vary), this can add up if there are many subnets and prefixes, a 10,000 scope/prefix deployment can use 10 GB over a 24 week period (not considering the backup requirements, which make this 30 GB).

INDEX

A

Add License page 35

C

certificate file 30–31
  importing 31
  keytool 30
  openssl 31
checking status 23
ciphers 77
  adjusting 77
CLI 1, 7, 35
  license 35
  requirements 7
  starting 35
cnr_status 37
cnr_status utility 37
command line interface 1
container 57

D

DHCP servers 2
disk space requirements 8
DNS servers 2
Docker container 57

E

error logging 38
excluding directories for virus scanning 19

I

image signing 17
installation 1, 10, 15, 21–23, 30–31, 73
  checklist 15
  directory 21
  Java 22
  lab evaluation 73
  local directory 21
  logs 31
installation *(continued)*
  modes 10
    new 10
    upgrade with data migration 10
    upgrade without data migration 10
  overview 1
  regional directory 21
  rpm command 23
  secure login 30
  troubleshooting 31
  upgrade 15
    license keys 15
  yum command 23
installation procedure 21

J

Java 7
  requirements 7

K

keystore 30
keystore file 30
keytool 30
keytool utility 30–31
Kubernetes 63
  deploying CPNR 63

L

lab evaluation installations 73
license command (CLI) 35
license keys 10, 35
Linux 8, 37
  cnr_status 37
  requirement 8
logging 38
  server events 38
  startups 38

N

network distribution 22

Network Registrar 1
about 1
nwreglocal and nwregregional 37
nwreglocal utility 37
nwregregional utility 37

O

OpenJDK 22
openssl 30
operating system 7–8
requirements 7
versions 8
OVA 43
overview 1

R

RAM requirements 8
RPM kit 22

S

sdk 75–76
compatibility considerations 76
installing 75
secure login 30
self-signed certificate 30
self-signed certificates 30
servers 2, 18, 37–38
DHCP 2
DNS 2
logging events 38
running with other 18
starting 37
starting/stopping 37
stopping 37
starting 35
CLI 35
Web UI 35

T

tail command 38

U

uninstallation 41, 74
lab evaluation 74
upgrade 1, 21–22, 30, 73
lab evaluation 73
network distribution 22
overview 1
secure login 30

V

viewing server logs 38
virtual appliance 44–45, 47, 50
booting and configuring 47
deploying 45, 47
installing and configuring 44
managing 50
upgrading 50
virus scanning 19
excluding directories 19
VMWare vCenter 46
VMWare vSphere 45

W

Web UI 1, 7, 35, 77
Add License page 35
ciphers 77
requirements 7
starting 35
web-based user interface 1

Y

yum install 23