



## About the nrcmd Program

You can use the Web-based user interface (Web UI) or the **nrcmd** command line interface (CLI) to configure and manage your DNS, DHCP, and TFTP servers. This chapter describes how to use the **nrcmd** command line interface. It specifically describes:

- Invoking the command in batch and interactive modes
- Command organization and syntax
- Special keyboard navigation characters

## Invoking the nrcmd Command

You can use the **nrcmd** command in batch mode by executing scripts that use the commands or by using the interactive mode in which you enter commands at the **nrcmd** command prompt. By default, the **nrcmd** command is located in C:\Program Files\Network Registrar\Local\bin on Windows and in /opt/nwreg2/local/usrbin on Solaris and Linux.



### Note

In Windows, if you want to run the **nrcmd** program from outside the installed path, you must set the CNR\_HOME environment variable.

On Windows, you can invoke the **nrcmd** command window from the Start menu:

**Start > Programs > Network Registrar 6.1 > Network Registrar 6.1 CLI**

This method prompts for your user name and password. On Solaris and Linux (as well as Windows alternatively), invoke the command from the command prompt using this syntax:

```
nrcmd [general-options] [command] [options]
```

[Table 1-1](#) describes the general options when invoking from the command prompt. [Chapter 2, “Using the nrcmd Commands,”](#) describes the commands and their specific options.

**Table 1-1 General Options to nrcmd Command**

Option	Description
-C <i>cluster</i>	<i>cluster</i> is the name of the machine on which the Network Registrar servers are running. If not specified, the cluster name defaults to localhost.
-N <i>user</i>	<i>user</i> is the Network Registrar user name.
-P <i>password</i>	<i>password</i> is the password of the Network Registrar user.

**Table 1-1** General Options to nrcmd Command (continued)

Option	Description
<b>-h</b>	prints help text.
<b>-r</b>	logs in as a read-only user.
<b>-b file.txt</b>	file.txt is the name of a file of <b>nrcmd</b> commands that run in batch mode; reading a line at a time and printing a new line after the prompt.

## Batch Mode

The program goes into batch mode if you include a functional command or the **-b file.txt** option on the line. The text file can include any number of **nrcmd** commands, and you can include comment lines preceded by the pound sign (#). In batch mode, you return to the normal system prompt. Note that display in batch mode is intended for parsing by an external program and, therefore, includes only command attributes that have values.



### Note

The last line of code in the input file must end with an end-of-line character. It is also a good practice to make the last line of code an explicit **exit** command.

## Interactive Mode

The program goes into interactive mode if you enter just the **nrcmd** command, or include the *cluster*, *user*, or *password* options. To execute the CLI in interactive mode, enter:

```
nrcmd [-C cluster] [-N user] [-P password]
```

This displays the interactive `nrcmd>` prompt, at which you enter a functional command and any optional parameters:

```
nrcmd> command [parameter,parameter,...]
system-response
```

To enter a series of attribute values, insert commas between them. Do not add a space after the comma. If the value is a string containing one or more space characters, enclose the value in quotes:

```
nrcmd> zone example.com. set auth-servers=192.168.50.1,10.0.0.1
100 Ok
auth-servers=192.168.50.1,10.0.0.1
```

To terminate an interactive session, enter the **exit** command. To view the online help, enter the **help** command.

## Registry and Environment Variables

If you omit the general options, Network Registrar gets them from the Registry or environment variables. If Network Registrar cannot find values for these parameters, it prompts you for them. If you omit the cluster name on a system where Network Registrar servers are installed, the **nrcmd** program assumes access to **localhost** and does not prompt you.

The environment variables that you can set that are recognized by the **nrcmd** program are **CNR\_NAME** for the name, **CNR\_PASSWORD** for the password, and **CNR\_CLUSTER** for the cluster name.

# Command Organization

The **nrcmd** commands specify a class of object, which you can create, delete, or list. Each of these objects in turn has attributes, which you can enable, disable, set, get, and unset, depending on data type. These objects may also have common methods, which are specific to the type of object, and that let you perform operations on groups of attributes.

When you use the **nrcmd** commands to configure Network Registrar, you manipulate:

- Classes—Things that you can create, delete, show, or list, such as scopes, policies, or zones.
  - **create**—Creates an entry. If the entry already exists, this command returns an error.
  - **delete**—Removes an entry.
  - **list**—Displays all the objects of a given type, including all attributes.
  - **listnames**—Displays only the names of all objects of a given type.
  - **show**—Displays the values of all the attributes.
- Attributes—Things that you can enable or disable, or whose value you can set or display using these common methods:
  - **enable**—enables a Boolean type of attribute.
  - **disable**—disables a Boolean type of attribute.
  - **set**—sets the value of an attribute.
  - **get**—displays the value of an attribute.
  - **unset**—makes an attribute have no value. You cannot unset required attributes.

As of Network Registrar 6.0, attributes that have default values are displayed in interactive mode in the form *attribute* = [default=*value*]. In interactive mode, all the attributes are displayed. In batch mode, only those attributes having values are displayed, and no default values are displayed. The display in batch mode is less user-friendly, but is more easily parsable by a program. These examples show how output compares in interactive and batch modes, respectively:

```
nrcmd> zone example.com show
100 Ok
example.com. (primary):
  checkpoint-interval =
  checkpoint-min-interval =
  defttl = 12h
  dynamic = [default=true]
  dynupdate-set =
  expire = 7d
  ...

$ nrcmd -N admin -P changeme zone example.com show
100 Ok
example.com.: defttl=12h; expire=7d; minttl=10m; nameservers={{0 rr2.example.com.}};
ns=rr2.; origin=example.com.; person=rr1.; refresh=3h; retry=60m; serial=1;
update-acl="key myKey";
```

- Other custom methods—these are specific operations that you can perform on an object, beyond editing its attributes. Examples are adding a range of IP addresses to a scope, or removing hosts from a zone.

## Command Usage

How you specify a series of arguments depends on the type of command you are using. The following subsections describe the differences between using the **create**, **set**, and **enable** commands.

### Create Keyword

When you use the **create** keyword and there are required arguments, you must supply them. You can also supply additional arguments. You must supply the required arguments in the specified order; however, you can specify the optional arguments in any order with the syntax *attribute=value*.

For example, the syntax for creating a scope is:

```
scope name create ipaddress mask [attribute=value]
```

This means that you must supply an IP address and mask when you create a scope, and you can optionally specify other attributes of the scope.

This example creates the scope testScope with an IP address of 192.168.50.0 and a mask of 255.255.255.0:

```
nrcmd> scope testScope create 192.168.50.0 255.255.255.0
100 Ok
testScope:
  addr = 192.168.50.0
  bootp = disabled
  deactivated =
  ...
```

You can also include attribute definitions on the same line. This example creates the same scope, but also specifies the name of the DNS zone to which a DHCP client's host name should be added:

```
nrcmd> scope testScope create 192.168.50.0 255.255.255.0 dns-zone-name=example.com.
100 Ok
dns-zone-name=example.com.
testScope:
  addr = 192.168.50.0
  bootp = disabled
  deactivated =
  ...
```

After the **create** keyword creates and assigns all specified parameters to the object, it checks that all the required attributes have values (either default or user-specified). If you omit the required attributes, Network Registrar returns an error.

### Set Keyword

You use the **set** keyword to set the value of an attribute that is already created. If you want to set a list of values, such as DNS servers or IP addresses, you can separate them with commas. You can also use the **set** keyword to set several attributes on a single line—just specify the attribute and its value followed by a space and the next attribute and value pair.

This example specifies the name of the DNS zone to which a DHCP client's host name should be added:

```
nrcmd> scope testScope set dns-zone-name=example.com.
100 Ok
dns-zone-name=example.com.
```

This example specifies the list of IP addresses for zone transfers for a zone:

```
nrcmd> zone example.com. set auth-servers=192.168.50.1,10.0.0.1
100 Ok
auth-servers=192.168.50.1,10.0.0.1
```

This example sets a client's client-class and domain name:

```
nrcmd> client 00:d0:ba:d3:bd:3b set client-class-name=internal
domain-name=example.com.
100 Ok
client-class-name=internal
domain-name=example.com.
```

The **unset** keyword places an attribute in the undefined state. The **get** keyword displays the value for an attribute.

## Enable Keyword

You use the **enable** keyword to enable a boolean attribute. After you enable one boolean attribute, you may need to set its associated attributes. Use the **disable** keyword to disable a boolean attribute. You can use the **unset** keyword to remove the enabled or disabled state of the boolean attribute.

This example enables incremental transfer processing for the DNS server:

```
nrcmd> dns enable ixfr-enable
100 Ok
ixfr-enable=enabled
```

Once incremental transfer is enabled, this example changes its expiration interval:

```
nrcmd> dns set ixfr-expire-interval=10d
100 Ok
ixfr-expire-interval=1w3d
```



### Note

You cannot add **set** keywords to an **enable** command line. You need to first enable the boolean attribute, and then, on the next command line, set the associated attributes.

## Attribute Flags

Command attributes are described as:

- **Required**—The attribute is required for the object. You must set the attribute or accept its default, and you can modify the value. You cannot use the **unset** keyword to set a required attribute to undefined. Trying to do so returns the error message “386 - Required attribute cannot be deleted.”
- **Optional**—The attribute is optional and does not require a value. You can set and reset the attribute, and you can use the **unset** keyword to make it undefined.
- **Read-only**—The attribute is immutable and read-only. You can use the **get** keyword with the attribute, but you cannot set or unset it. Trying to set or unset a read-only attribute returns the error message “385 - Read-only attribute cannot be modified.”

## Saving Your Changes

The CLI waits for one of these events to occur before it saves your changes to the database:

- Invoking the **save** command
- Exiting from **nrcmd**
- Reloading a server
- Adding a resource record or host to a zone

## Refreshing and Clearing the CLI Cache

The CLI caches many configuration objects that it reads. If multiple users are making changes simultaneously, one CLI instance might have cached an out of date version of an object. The **session cache refresh** command causes the CLI to clear its local cache of all unmodified objects, forcing it to reread objects from the configuration database. The **session cache clear** command forces the CLI to clear all cached data, whether or not unsaved changes were made.

## Navigation Keys

Table 1-2 lists keyboard navigation key combinations that are useful when entering **nrcmd** commands.

**Table 1-2** *nrcmd Navigation Key Combinations*

Key Combination	Action
Control-a	Go to the beginning of the line
Control-b	Back one character
Control-d	Delete one character
Control-e	Go to the end of the line
Control-f	Forward one character
Control-k	Kill to the end of the line
Control-l	Redraw the line
Control-n	Next line in the history
Control-p	Previous line in the history
Control-t	Shift an individual character left
Control-u	Delete the line and move the cursor to the beginning of the line
Control-w	Delete one word backwards
Esc-b	Back one word
Esc-f	Forward one word

# Command List

Table 1-3 lists the **nrcmd** commands, alphabetically. You can use these commands on the command line or insert them into scripts.

**Table 1-3 nrcmd Commands**

Command	Description
<b>acl</b>	Creates access control lists (ACLs)
<b>address-block</b>	Creates and sets properties for address blocks
<b>address-block-policy</b>	Configures DHCP embedded policies for address blocks
<b>admin</b>	Creates administrators and assigns them passwords
<b>client</b>	Creates clients and assigns them to client-classes
<b>client-class</b>	Creates client-classes
<b>client-class policy</b>	Sets embedded client-class policies
<b>client-policy</b>	Sets embedded client policies
<b>custom-option</b>	Creates a custom DHCP option
<b>dhcp</b>	Specifies the DHCP server attributes
<b>dhcp-interface</b>	Specifies the IP address of the DHCP server's hardware card
<b>dns</b>	Specifies the DNS server attributes
<b>exit</b> or <b>quit</b>	Exits the <b>nrcmd</b> command and saves the current configuration changes
<b>export</b>	Writes the state of the lease or a zone to a file
<b>extension</b>	Integrates DHCP extensions into the Network Registrar DHCP server
<b>force-lock</b>	Obtains an exclusive lock for the <b>nrcmd</b> command session
<b>help</b>	Provides online help
<b>import</b>	Loads configuration information from a file
<b>key</b>	Creates transaction signature (TSIG) keys for dynamic DNS updates
<b>ldap</b>	Specifies the LDAP remote server attributes
<b>lease</b>	Retrieves information about DHCP leases
<b>lease-notification</b>	Notifies you when you run out of available leases in a scope
<b>license</b>	Views and updates license information
<b>namespace</b>	Creates and sets properties for namespaces
<b>option-datatype</b>	Defines data types for use in defining vendor-supplied DHCP options
<b>policy</b>	Specifies the policy information
<b>remote-dns</b>	Specifies information about remote DNS servers
<b>report</b>	Creates a summary of the IP address utilization for one or more clusters
<b>save</b>	Saves the current configuration changes
<b>scope</b>	Specifies scope attributes
<b>scope-policy</b>	Sets embedded scope attributes
<b>scope-selection-tag</b>	Creates scope selection tags

**Table 1-3** *nrcmd* Commands (continued)

<b>Command</b>	<b>Description</b>
<b>server</b>	Affects server behavior
<b>session</b>	Configures session parameters
<b>subnet</b>	Retrieves information about subnets
<b>tftp</b>	Specifies the Trivial File Transport Protocol (TFTP) server attributes
<b>trap</b>	Activates Simple Network Management Protocol (SNMP) traps
<b>vendor-option</b>	Defines vendor-supplied DHCP options
<b>zone</b>	Specifies DNS zone attributes