



# CHAPTER 27

## Configuring DHCP Failover

---

DHCPv4 failover is a protocol designed to allow a backup DHCP server to take over for a main server if the main server is taken off the network for any reason. DHCPv4 failover applies to address leases, but is not applicable to on-demand address pools.

### See Also

[Failover Scenarios](#)  
[Failover Checklist, page 27-5](#)  
[Creating and Synchronizing Failover Server Pairs, page 27-6](#)  
[Confirming Failover, page 27-11](#)  
[State Transitions During Integration, page 27-11](#)  
[Setting Advanced Failover Attributes, page 27-15](#)  
[Changing Failover Server Roles, page 27-22](#)  
[Restoring a Standalone DHCP Failover Server to Backup State, page 27-25](#)  
[Repairing Partners to Their Original Roles, page 27-30](#)  
[Recovering in Failover Configuration, page 27-31](#)  
[Supporting BOOTP Clients in Failover, page 27-32](#)  
[DHCPLEASEQUERY and Failover, page 27-33](#)  
[Troubleshooting Failover, page 27-33](#)

## Failover Scenarios

There are three basic failover scenarios:

- **Simple failover (recommended)**—One server acting as main and its partner acting as backup (see the “[Simple Failover](#)” section).
- **Back office failover**—Two mains having the same backup server (see the “[Back Office Failover](#)” section on page 27-3).
- **Symmetrical failover**—Two servers acting as main and backup for each other (see the “[Symmetrical Failover](#)” section on page 27-4).



### Caution

---

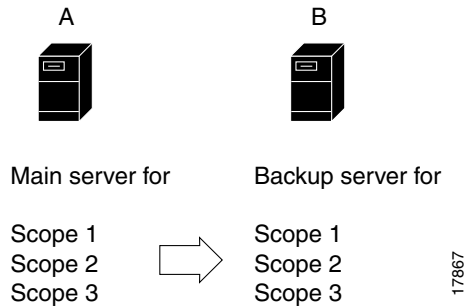
We recommend that you use the simple failover method. We do not recommend the use of back office failover or symmetrical failover.

---

## Simple Failover

Simple failover involves a main server and a single backup server pair (see [Figure 27-1 on page 27-2](#)). In the example, main server A has three scopes that must be configured identically on backup server B.

**Figure 27-1 Simple Failover Example**



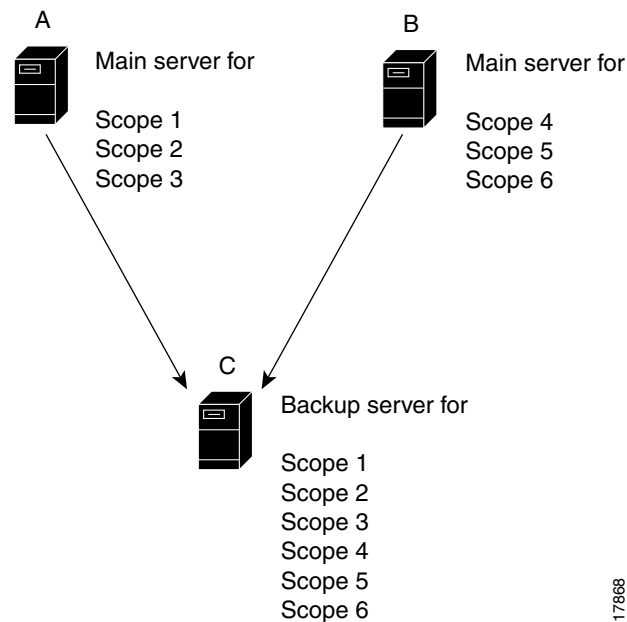
The advantages of simple failover over the other scenarios are:

- It is the easiest to manage as the network changes. It is fully supported by the web UI so that changes to the main server configuration can be duplicated on the backup server.
- Provides the greatest performance benefits.
- Having the additional load balancing feature eliminates the need for a back office or symmetrical scenario (see the [“Setting Load Balancing” section on page 27-21](#)).

## Back Office Failover

Back office failover involves two (or more) main servers that share the same backup server (see [Figure 27-2](#)). In the example, main servers A and B have different scopes, and backup server C must include all these scopes. This scenario is appropriate for scopes on the same LAN segment, which require the same main and backup servers, but with the sets of scopes on different LAN segments.

**Figure 27-2 Back Office Failover Example**



An advantage of back office failover over the other scenarios is that it reduces the number of servers managed. However, simple failover is still recommended, because in back office failover:

- The backup server must be sized to handle the sum of the configurations.
- Changes to any of the main servers must be duplicated on the backup server.
- The increased complexity can substantially reduce the actual availability.

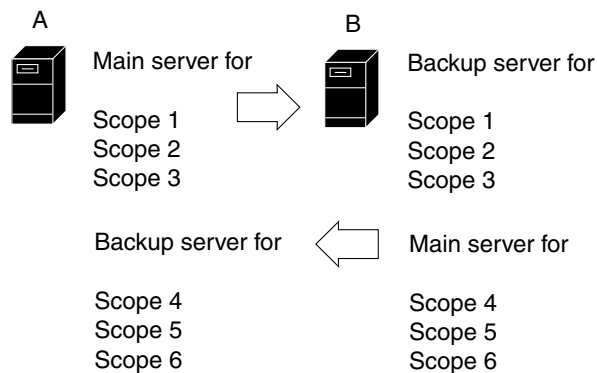
## Symmetrical Failover

Symmetrical failover involves servers that act as backups for each other (see [Figure 27-3](#)). This scenario is extremely tricky in that there can be no variance in scope attribute values between the servers for the relationship to work properly.

Symmetrical failover used to be a way of load balancing the servers, although this is now more effectively done in a simple failover scenario using the load balancing feature (see the [“Setting Load Balancing” section on page 27-21](#)).

Unfortunately, symmetrical failover provides little to no performance benefit over the simple or back office scenarios. A backup server operates at about 40% of the main server to keep its lease database synchronized. If the servers back each other up, a portion of their processing capacity goes to this task, with less capacity available to service clients. Moreover, because each scope must be configured individually, symmetrical failover is more prone to configuration errors.

**Figure 27-3 Symmetrical Failover Example**



17869

# Failover Checklist

Use this checklist to prepare for an effective failover configuration:

- Duplicate the scope, policy, DHCP option, and address configurations on the partner servers by configuring a failover server pair for a simple failover scenario. In the case of a back office or symmetrical failover scenario, you need to modify the Network Match List for the server pair configuration to include the network address or addresses that identify the correct scope or scopes.
- Ensure that both partners are configured with a wide enough range of addresses so that the backup server can provide leases for a reasonable amount of time while the main server is down.
- The following objects must have identical configurations on both servers:
  - Scopes and prefixes
  - Selection tags
  - Policies
  - IP addresses
  - Reservations
  - Clients
  - Client-classes
  - Dynamic DNS updates
  - Dynamic BOOTP
  - Virtual private networks (VPNs)
  - DHCP extensions
- If you use LDAP, direct the partner servers to the same LDAP server.
- If you use BOOTP relay (IP helpers), configure all BOOTP relay agents to point to both partners. Cisco Network Registrar does not automatically detect this.

You can detect BOOTP configuration errors only by performing live tests in which you periodically take the main server out of service to verify that the backup server is available to DHCP clients.

# Creating and Synchronizing Failover Server Pairs

You can create and synchronize failover pairs at the local and regional clusters.

A failover pair has two main elements, its configuration and the state information that the servers maintain. The key configuration attributes are the name of the failover pair, the role of the local server (main or backup), and the address of the partner. The failover state is defined when you reload the server and the server processes this state data at startup.

## See Also

[Adding Failover Pairs](#)  
[Synchronizing Failover Pairs, page 27-7](#)  
[Restarting the Failover Servers, page 27-10](#)

## Adding Failover Pairs

Create the DHCP failover pair based on the cluster of the main and backup servers. Then synchronize the failover pair so that the scopes, policies, and other DHCP properties match between the servers.

To add a failover pair:

### Local and Regional Web UI

- 
- Step 1** From the **DHCP** menu, choose **Failover** to open the List/Add DHCP Failover Pairs page.
- Step 2** Click **Add DHCP Failover Pair**.
- Step 3** On the Add DHCP Failover Pair page, add a failover pair name.  
 This is required, and can be any distinguishing name. (See also the [“Changing Failover Pair Server Addresses”](#) section on page 27-7.)
- Step 4** Proceed stepwise:
- Choose the cluster for the main DHCP server. This can be localhost or some other cluster you define. Whatever you select here becomes the IP address value for the *main-server* attribute once you add the failover pair.  
 You should not change the IP address values of the *main-server* and *backup-server* attributes unless you have multihomed hosts.  
 If you change the IP address of your local host, you must modify the localhost cluster (on the Edit Cluster page) to change the address in the IP Address field. Do not set the value to 127.0.0.1.
  - Choose the cluster for the backup DHCP server. This cannot be the same as the main server cluster, but it must be localhost if the main cluster is not localhost. Whatever you select here becomes the IP address value for the *backup-server* attribute once you add the failover pair.
  - For each address block to be included in the failover configuration, enter its IP address and mask, then click **Add Address Block**. (See the [“Adding Address Blocks”](#) section on page 9-5.)
  - If you have a more complex failover scenario, include the list of IP addresses in the Network Match List that identifies the subnets for the primary scopes (or secondary scopes with primary subnets) that you need for the configuration. Add the IP addresses, then click **Add Network Match**.

- e. You can set additional attributes, such as the maximum client lead time (*mclt*) or backup percentage (*backup-pct*). Most of the default values are optimized. Leave the *failover* attribute enabled by default unless you want to temporarily disable failover for the pair.

**Step 5** Click **Add Failover Pair**. You can edit the failover pair properties.

---

## See Also

[Failover Scenarios, page 27-1](#)  
[Failover Checklist, page 27-5](#)  
[Changing Failover Pair Server Addresses, page 27-7](#)  
[Synchronizing Failover Pairs, page 27-7](#)  
[Restarting the Failover Servers, page 27-10](#)

## Changing Failover Pair Server Addresses

If you need to change the name of a failover pair or an address associated with a cluster involved in a failover relationship, you can ensure that the failover state information is preserved. The failover partners should not enter recover state for the MCLT period. To change the name of a failover pair, you must remove the old object and add a new object.

If the cluster is configured for simple failover (only one failover pair exists), you can remove and add the renamed failover pair and change addresses, provided the failover role (main or backup) is not changed.

If there are multiple failover pairs, change either the name or the address, not both at the same time. Reload the DHCP server and allow the failover partners to return to normal state between name and address changes.



### Note

If a cluster role in a failover relationship is changed (main to backup or backup to main), any existing state information for that relationship is discarded.

---

## Synchronizing Failover Pairs

Once you create the failover pairs, you must synchronize the servers.




### Tip

In Expert mode, the List/Add DHCP Failover Pairs page provides a Resync CCM Failover Pairs button. For synchronization in the regional web UI, see the “[Managing DHCP Failover Pairs](#)” section on [page 6-20](#).

---

## Web UI

**Step 1** On the List/Add DHCP Failover Pairs page, click the Report icon () to open the Report Synchronize Failover Pair page.

For synchronization in the regional web UI, see the “[Managing DHCP Failover Pairs](#)” section on [page 6-20](#).

- Step 2** Choose the direction of synchronization. The direction of synchronization can be either from main to backup server or from backup to main server.
- Step 3** Choose the synchronization operation, depending on the degree to which you want the main server objects to replace those of the backup server. The following are the basic synchronization operations that can be performed on the servers:

- **Update operation**—This is the default and least radical operation. It is appropriate for update synchronizations in that it has the least effect on the unique properties of the backup server.
- **Complete operation**—This operation is appropriate for all initial synchronizations. It is more complete than an update operation, while still preserving many of the backup server unique properties. This operation is appropriate for back office failover configurations.
- **Exact operation**—This operation is appropriate for initial simple and symmetrical failover configurations, and is not appropriate for back office configurations.

It makes the two servers mirror images of each other, as much as possible, although this operation retains the unique DHCP server, and extension points on the backup server.



**Note** For initial failover configurations, use the Exact or Complete operation.

For a better understanding of the functions that are performed on the classes of the objects, consider the following example. Here, we have a main server and its backup server with the following objects:

On the main server	On the backup server
Name1=A	Name2=B
Name2=C	Name3=D



**Note** In this example, we consider failover synchronization from the main server to the backup server.

Each operation performs a different mix of functions on the classes of objects. The following are the four functions that are performed on the objects based on the operation selected.

- no change—Makes no change to the list of properties or their values on the backup server.  
For example, the result would be Name2=B, Name3=D.
- ensure—Ensures that a copy of the main server object exists on the backup. The target server objects with the same name as main server objects are left unchanged, the objects that are not on the target server are added to it, and the objects only on the target server are left unchanged.  
For example, the result would be Name1=A, Name2=B, Name3=D.
- replace—Ensures that the existing object in the target server is replaced by the main server object of the same name. Also the objects that are not on the target server are added to it and the objects only on the target server are left unchanged. The only exceptions to this are for policies and option definition sets, where the option lists are extracted to compare the list entries.  
For example, the result would be Name1=A, Name2=C, Name3=D.





**Note** After deleting the client on the main server and performing the failover synchronization Update or Complete operation to remove the entry on the backup, the client is not removed from the backup. The only failover synchronization operation that will delete the client entry on the backup, after it is removed from the main server, is the failover synchronization Exact operation.

- exact—Puts an exact copy of the main server object on the backup server and removes the unique ones. That is, the objects of target server are made identical to the objects of main server. For example, the result would be Name1=A, Name2=C.





For more information, see [Table 27-1 on page 27-9](#). This table provides the information on the functions (no change, ensure, replace, or exact) that are performed on the objects based on the operations (Update, Complete, Exact) you select.

**Table 27-1 Failover Pair Synchronization Functions**

Data Description	Update	Complete	Exact
DHCP Server (server level failover pair):	replace	replace	replace
Client-Class Properties			
Client Hostname Properties			
DNS Update Properties			
Failover Tuning Properties			
(See <a href="#">Table 27-2</a> for a list of the failover pair attributes affected by failover synchronization)			
All other properties	no change	replace	replace
DHCP Listener Configuration	ensure	replace	exact
LDAP Remote Server	ensure	replace	exact
Policy:			
Option List Properties	ensure	replace	exact
Packet Boot File Properties	ensure	replace	exact
All other properties	replace	replace	exact
Client	replace	replace	exact
Client-Class	replace	replace	exact
Scopes (related to failover pair)	exact	exact	exact
DNS Update Configuration	replace	replace	exact
Trap Configuration	ensure	replace	exact
VPN	replace	replace	exact
Key	replace	replace	exact
Extensions (You must copy extension files.)	ensure	replace	exact
Extension Point	replace	replace	replace
Option Information:	ensure	replace	exact
Custom options list			
Vendor options list			

**Table 27-2** Failover Pair Attributes Affected by Failover Synchronization**Affected Failover Pair Attributes**







*failover-bulking*  
*failover-poll-interval*  
*failover-poll-timeout*  
*recover*

- Step 4** Click **Run** on the Run Synchronize Failover page, or **Report** on the Report Synchronize Failover page:
- If you click **Run** and if the connection was accepted, the resulting View DHCP Failover Pair Sync Report page shows what change entries the synchronization added.
  - If you click **Report**, the resulting View DHCP Failover Pair Sync Report page shows what change entries the synchronization will apply if you run the synchronization. A **Run Update**, **Run Complete**, or **Run Exact** button indicates what kind of synchronization you want to perform. Click the applicable button to run the synchronization.
- Step 5** On the List/Add DHCP Failover Pairs page, click the View icon () in the Manage Servers column to open the Manage DHCP Failover Servers page.
- Step 6** Click the Reload icon () next to the backup server to reload the backup server.
- Step 7** Try to get a lease.
- Step 8** On the Manage DHCP Failover Servers page, look at the health of the servers (they should show as ). Also, click the Logs icon () to view the log entries on the Log for Server page, and ensure that the servers are in NORMAL failover mode. The log file should contain an item similar to the following:

```
06/19/2003 9:41:19 name/dhcp/1 Info Configuration 0 04092 Failover is enabled
server-wide. Main server name: '192.168.0.1', backup server name: '192.168.0.110', mclt =
3600, backup-pct = 10, dynamic-bootp-backup-pct = 0, use-safe-period: disabled,
safe-period = 0.
```

## Restarting the Failover Servers

For any failover synchronization to take effect, you must first connect to, and restart, both the main and backup failover servers.


- Step 1** On the List/Add DHCP Failover Pairs page, click the Go Local icon () in the Main DHCP Server column.
- Step 2** On the local Manage DHCP Server page for the main server, click the Reload icon () on the right-hand side of the page.
- Step 3** Click the Go Regional icon () at the top-right corner of the page.
- Step 4** On the regional List/Add DHCP Failover Pairs page, click the Go Local icon () in the Backup DHCP Server column.
- Step 5** On the local Manage DHCP Server page for the backup server, click the Reload icon () on the right-hand side of the page.
- Step 6** Click the Go Regional icon () at the top-right corner of the page.

**See Also**

[Confirming Failover, page 27-11](#)

## Confirming Failover

---

- Step 1** Ping from one server to the other to verify TCP/IP connectivity. Make sure that routers are configured to forward clients to both servers.
  - Step 2** Check that the server is in NORMAL mode by clicking the Related Servers icon () on the Manage DHCP Server or List/Add DHCP Failover Pairs page, or use **dhcp getRelatedServers** in the CLI.
  - Step 3** After startup, have a client attempt to get a lease.
  - Step 4** Set the log settings on the main server to include at least *failover-detail*.
  - Step 5** Confirm that the name\_dhcp\_1\_log log file (in *install-path/logs*) on the main server contains DHCPBNDACK or DHCPBNDUPD messages from each server.
  - Step 6** Confirm that the name\_dhcp\_1\_log log file on the backup server contains messages that the backup server is dropping requests because failover is in NORMAL state.
  - Step 7** Repeat [Step 2](#).
- 

**See Also**

[State Transitions During Integration](#)  
[Setting Advanced Failover Attributes, page 27-15](#)

## State Transitions During Integration

During normal operation, the failover partners transition between states. They stay in their current state until all the actions for the state transition are completed and, if communication fails, until the conditions for the next state are fulfilled. [Table 27-3](#) describes what happens when servers enter various states and how they initially integrate and later reintegrate with each other under certain conditions.

**Table 27-3** Failover State Transitions and Integration Processes

Integration	Results
Into NORMAL state, the first time the backup server contacts the main server	<ol style="list-style-type: none"> <li>1. The newly configured backup server contacts the main server, which starts in PARTNER-DOWN state.</li> <li>2. Because the backup server is a new partner, it goes into RECOVER state and sends a Binding Request message to the main server.</li> <li>3. The main server replies with Binding Update messages that include the leases in its lease state database.</li> <li>4. After the backup server acknowledges these messages, the main server responds with a Binding Complete message.</li> <li>5. The backup server goes into RECOVER-DONE state.</li> <li>6. Both servers go into NORMAL state.</li> <li>7. The backup server sends Pool Request messages.</li> <li>8. The main server responds with the leases to allocate to the backup server based on the <i>backup-pct</i> configured.</li> </ol>
After COMMUNICATIONS-INTERRUPTED state	<ol style="list-style-type: none"> <li>1. When a server comes back up and connects with a partner in this state, the returning server moves into the same state and then immediately into NORMAL state.</li> <li>2. The partner also moves into NORMAL state.</li> </ol>
After PARTNER-DOWN state	<p>When a server comes back up and connects with a partner in this state, the server compares the time it went down with the time the partner went into this state.</p> <ul style="list-style-type: none"> <li>• If the server finds that it went down and the partner subsequently went into this state: <ol style="list-style-type: none"> <li>a. The returning server moves into RECOVER state and sends an Update Request message to the partner.</li> <li>b. The partner returns all the binding data it was unable to send earlier and follows up with an Update Done message.</li> <li>c. The returning server moves into RECOVER-DONE state.</li> <li>d. Both servers move into NORMAL state.</li> </ol> </li> </ul>

**Table 27-3** *Failover State Transitions and Integration Processes (continued)*

Integration	Results
	<ul style="list-style-type: none"> <li>• If the returning server finds that it was still operating when the partner went into PARTNER-DOWN state:               <ol style="list-style-type: none"> <li>a. The server goes into POTENTIAL-CONFLICT state, which also causes the partner to go into this state.</li> <li>b. The main server sends an update request to the backup server.</li> <li>c. The backup server responds with all unacknowledged updates to the main server and finishes off with an Update Done message.</li> <li>d. The main server moves into NORMAL state.</li> <li>e. The backup server sends the main server an Update Request message requesting all unacknowledged updates.</li> <li>f. The main server sends these updates and finishes off with an Update Done message.</li> <li>g. The backup server goes into NORMAL state.</li> </ol> </li> </ul>
After the server loses its lease state database	<p>A returning server usually retains its lease state database. However, it can also lose it because of a catastrophic failure or intentional removal.</p> <ol style="list-style-type: none"> <li>1. When a server with a missing lease database returns with a partner that is in PARTNER-DOWN or COMMUNICATIONS-INTERRUPTED state, the server determines whether the partner ever communicated with it. If not, it assumes to have lost its database, moves into RECOVER state, and sends an Update Request All message to its partner.</li> <li>2. The partner responds with binding data about every lease in its database and follows up with an Update Done message.</li> <li>3. The returning server waits the maximum client lead time (MCLT) period, typically one hour, and moves into RECOVER-DONE state. For details on the MCLT, see the <a href="#">“Setting the Maximum Client Lead Time”</a> section on page 27-18.</li> <li>4. Both servers then move into NORMAL state.</li> </ol>

Table 27-3 Failover State Transitions and Integration Processes (continued)

Integration	Results
After a lease state database backup restoration	<p>When a returning server has its lease state database restored from backup, and if it reconnects with its partner without additional data, it only requests lease binding data that it has not yet seen. This data may be different from what it expects.</p> <ol style="list-style-type: none"> <li data-bbox="699 457 1482 514">1. In this case, you must configure the returning server with the <i>failover-recover</i> attribute set to the time the backup occurred.</li> <li data-bbox="699 533 1482 688">2. The server moves into RECOVER state and requests all its partner data. The server waits the MCLT period, typically one hour, from when the backup occurred and goes into RECOVER-DONE state. For details on the MCLT, see the <a href="#">“Setting the Maximum Client Lead Time”</a> section on page 27-18.</li> <li data-bbox="699 707 1482 764">3. Once the server returns to NORMAL state, you must unset its <i>failover-recover</i> attribute, or set it to zero.</li> </ol> <pre data-bbox="740 783 1174 804">nrcmd&gt; dhcp set failover-recover=0</pre>
After the operational server had failover disabled	<p>If the operating server had failover enabled, disabled, and subsequently reenabled, you must use special considerations when bringing a newly configured backup server into play. The backup server must have no lease state data and must have the <i>failover-recover</i> attribute set to the current time minus the MCLT interval, typically one hour. For details on the MCLT, see the <a href="#">“Setting the Maximum Client Lead Time”</a> section on page 27-18.</p> <ol style="list-style-type: none"> <li data-bbox="699 1087 1482 1243">1. The backup server then knows to request all the lease state data from the main server. Unlike what is described in “After the server loses its lease state database” section of this table, the backup server cannot request this data automatically because it has no record of having ever communicated with the main server.</li> <li data-bbox="699 1262 1482 1350">2. After reconnecting, the backup server goes into RECOVER state, requests all the main server lease data, and goes into RECOVER-DONE state.</li> <li data-bbox="699 1369 1482 1428">3. Both servers go into NORMAL state. At this point, you must unset the backup server <i>failover-recover</i> attribute, or set it to zero.</li> </ol> <pre data-bbox="740 1446 1174 1467">nrcmd&gt; dhcp set failover-recover=0</pre>

# Setting Advanced Failover Attributes

The advanced failover properties that are important to set are the following:

- Backup percentage (see the “[Setting Backup Percentages](#)” section on page 27-15)
- Backup allocation boundaries (see the “[Setting Backup Allocation Boundaries](#)” section on page 27-17)
- Maximum client lead time (MCLT) (see the “[Setting the Maximum Client Lead Time](#)” section on page 27-18)
- Safe period (see the “[Using the Failover Safe Period to Move Servers into PARTNER-DOWN State](#)” section on page 27-19)
- Request and response packet buffers (see the “[Setting DHCP Request and Response Packet Buffers](#)” section on page 27-20)
- Polling attributes (see the “[Changing Polling Attributes](#)” section on page 27-21)
- Network discovery (see the “[Setting the Network Discovery Attribute](#)” section on page 27-21)
- Load balancing (see the “[Setting Load Balancing](#)” section on page 27-21)

## Setting Backup Percentages

To keep failover partners operating despite a network partition (when both servers can communicate with clients, but not with each other), allocate more addresses than for a single server. Configure the main server to allocate a percentage of the currently available addresses in each scope to the backup server. This makes these addresses unavailable to the main server. The backup server uses these addresses when it cannot talk to the main server and cannot tell if it is down.



### Note

Cisco Network Registrar 6.3 and later server uses the *unavailable-timeout* value configured in the scope policy or **system\_default\_policy** policy as the timeout for the unavailable lease. When the lease times out, the policy causes the lease to transition to available in both failover partners.

You can set the percentage of currently available addresses by setting the *backup-pct* attribute on the failover pair or scope (**failover-pair name set backup-pct** or **scope name set backup-pct** in the CLI). Note that setting the backup percentage on the failover pair level sets the value for all scopes not set with that attribute. However, if set at the scope level, the backup percentage overrides the one at the failover pair level. If the *load-balancing* attribute is enabled for the failover pair (**failover-pair name enable load-balancing** in the CLI), the backup percentage is fixed at 50% and any of the backup percentage attributes (on a failover pair or scope) are ignored. (See the “[Load Balancing Compatibility with Earlier Cisco Network Registrar Versions](#)” section on page 27-22.)

The backup percentage should be set large enough to allow the backup server to continue serving new clients in the event that the main server fails. The backup percentage is calculated based on the number of available addresses. The default backup percentage is 10% (unless load balancing is in effect, then it is 50%). However, this number can safely be set to a larger value, if extended outages are expected, because the main server periodically reclaims addresses (once per hour) if, in the course of normal leasing activity, the main server's available address pool drops below its predefined percentage. For example, with the default 10% backup percentage, the main server will reclaim addresses if its address pool falls below 90%.

The percentage depends on the new client arrival rate and the network operator reaction time. The backup server needs enough addresses from each scope to satisfy all new clients requests arriving during the time it does not know if the main server is down. Even during PARTNER-DOWN state, the backup server waits for the maximum client lead time (MCLT) and lease time to expire before reallocating leases. See the “[Setting the Maximum Client Lead Time](#)” section on page 27-18. When these times expire, the backup server offers:

- Leases from its private pool.
- Leases from the main server pool.
- Expired leases to new clients.

During the day, an operator likely responds within two hours to COMMUNICATIONS-INTERRUPTED state to determine if the main server is working. The backup server then needs enough addresses to support a reasonable upper bound on the number of new clients that could arrive during those two hours.

During off-hours, the arrival rate of previously unknown clients is likely to be less. The operator can usually respond within 12 hours to the same situation. The backup server then needs enough addresses to support a reasonable upper bound on the number of clients that could arrive during those 12 hours.

The number of addresses over which the backup server requires sole control is the greater of the two numbers. You would express this number as a percentage of the currently available (unassigned) addresses in each scope. If you use client-classes, remember that some clients can only use some sets of scopes and not others.

**Note**


---

During failover, clients can sometimes obtain leases whose expiration times are shorter than the amount configured. This is a normal part of keeping the server partners synchronized. Typically this happens only for the first lease period, or during COMMUNICATIONS-INTERRUPTED state.

---

**See Also**

[Server and Scope Backup Percentages](#)  
[BOOTP Backup Percentage](#)

**Server and Scope Backup Percentages**

For all servers or scopes for which you enable failover, you must set the *backup-pct* attribute. This is the number of currently available (unreserved) leases that the backup server can use for allocations to new DHCP clients when the main server is down. You can use the preset value, which is 10 percent, or specify another value.

**Note**


---

When failover load balancing is in effect, the main and backup servers actively move available leases between them to maintain the backup percentage of available leases. See the “[Setting Load Balancing](#)” section on page 27-21.

---



## BOOTP Backup Percentage

For scopes for which you enable dynamic BOOTP, use the *dynamic-bootp-backup-pct* attribute rather than the *backup-pct* attribute for the failover pair. The *dynamic-bootp-backup-pct* is the percentage of available addresses that the main server should send to the backup server for use with BOOTP clients.

The *dynamic-bootp-backup-pct* is distinct from the *backup-pct* attribute, because if you enable BOOTP on a scope, a server, even in PARTNER-DOWN state, never grants leases on addresses that are available to the other server. Cisco Network Registrar does not grant leases because the partner might give them out using dynamic BOOTP, and you can never safely assume that they are available again.



### Note

You must define the dynamic BOOTP backup percentage on the main server. If you define it on the backup server, Cisco Network Registrar ignores it (to enable duplicating configuration through scripts). If you do not define it, Cisco Network Registrar uses the default *backup-pct* for the failover pair or scope.

To properly support dynamic BOOTP while using the failover protocol, do this on every LAN segment in which you want BOOTP support:

- Create one scope for dynamic BOOTP.
- Enable BOOTP and dynamic BOOTP.
- Disable DHCP for that scope.

## Setting Backup Allocation Boundaries

You can be more specific as to which addresses to allocate to the backup server by using the *failover-backup-allocation-boundary* attribute on the scope. The IP address set as this value is the upper boundary of addresses from which to allocate addresses to a backup server. Only addressees below this boundary are allocated to the backup. If there are none available below this boundary, then the addresses above it, if any, are allocated to the backup. The actual allocation works down from this address, while the normal allocation for DHCP clients works up from the lowest address in the scope.

If you set *failover-backup-allocation-boundary* for the scope, you must also enable the *allocate-first-available* attribute. If *failover-backup-allocation-boundary* is unset or set to zero, then the boundary used is halfway between the first and last addresses in the scope ranges. If there are no available addresses below this boundary, then the first available address is used.

## Setting the Maximum Client Lead Time

You can set a property for failover that controls an adjustment to the lease period, the maximum client lead time (MCLT). The MCLT adjusts for a potential period of uncertain connectivity between the servers. It is the maximum time one server can grant (or extend) a lease to a client without first negotiating a longer time with its partner. This time has the following implications:

- Clients may initially (or if the partners are not communicating) only receive leases of the MCLT length. This means that they need to renew leases sooner than they might otherwise without failover. At this renewal, the client should get a full lease time (unless the partners are not communicating).
- If a server enters PARTNER-DOWN state, it must wait until the MCLT after the later of the partner-down time or the latest lease expiration time communicated with the partner gets over. The latest lease expiration time communicated to the partner is typically 1.5 times the lease time from the last client lease request before communication was interrupted.
- If a failover recovery occurs where there is uncertainty about what one partner did (such as when it loses its lease database), the partners may have to restrict leasing activity for the MCLT period after they synchronize before they can resume normal failover operations.

The default MCLT is one hour, the optimum for most configurations. As defined by the failover protocol, the lease period given a client can never be more than the MCLT plus the most recently received potential expiration time from the failover partner, or the current time, whichever is later. That is why you sometimes see the initial lease period as only an hour, or an hour longer than expected for renewals. The actual lease time is recalculated when the main server comes back.

The MCLT is necessary because of failover use of lazy updates. Using lazy updates, the server can issue or renew leases to clients before updating its partner, which it can then do in batches of updates. If the server goes down and cannot communicate the lease information to its partner, the partner may try to reoffer the lease to another client based on what it last knew the expiration to be. The MCLT guarantees that there is an added window of opportunity for the client to renew. The way that a lease offer and renewal works with the MCLT is:

1. The client sends a DHCPDISCOVER to the server, requesting a desired lease period (say, three days). The server responds with a DHCPOFFER with an initial lease period of only the MCLT (one hour by default). The client then requests the MCLT lease period and the server acknowledges it.
2. The server sends its partner a bind update containing the lease expiration for the client as the current time plus the MCLT. The update also includes the potential expiration time as the current time plus the client desired period plus the MCLT (three days plus an hour). The partner acknowledges the potential expiration, thereby guaranteeing the transaction.
3. When the client sends a renewal request halfway through its lease (in one-half hour), the server acknowledges with the client desired lease period (three days). The server then updates its partner with the lease expiration as the current time plus the desired lease period (three days), and the potential expiration as the current time plus the desired period and another half of this period ( $3 + 1.5 = 4.5$  days). The partner acknowledges this potential expiration of 4.5 days. In this way, the main server tries to have its partner always lead the client in its understanding of the client lease period so that it can always offer it to the client.

There is no one correct value for the MCLT. There is an explicit trade-off between various factors in choosing one. Most people use the preset value of one hour effectively and it works well in almost all environments. Here are some of the trade-offs between a short and long MCLT:

- **Short MCLT**—A short MCLT value means that after entering PARTNER-DOWN state, a server only has to wait a short time before it can start allocating its partner IP addresses to DHCP clients. Furthermore, it only has to wait a short time after a lease expires before it can reallocate that address to another DHCP client. However, the down side is that the initial lease interval that is offered to

every new DHCP client will be short, which causes increased traffic, because those clients need to send their first renewal in a half of a short MCLT time. Also, the lease extensions that a server in COMMUNICATIONS-INTERRUPTED state can give is the MCLT only after the server has been in that state for around the desired client lease period. If a server stays in that state for that long, then the leases it hands out will be short, increasing the load on that server, possibly causing difficulty.

- **Long MCLT**—A long MCLT value means that the initial lease period will be longer and the time that a server in COMMUNICATIONS-INTERRUPTED state can extend leases (after it being in that state for around the desired client lease period) will be longer. However, a server entering PARTNER-DOWN state must wait the longer MCLT before being able to allocate its partner addresses to new DHCP clients. This may mean that additional addresses are required to cover this time period. Also, the server in PARTNER-DOWN state must wait the longer MCLT from every lease expiration before it can reallocate an address to a different DHCP client.

## Using the Failover Safe Period to Move Servers into PARTNER-DOWN State

One or both failover partners could potentially move into COMMUNICATIONS-INTERRUPTED state. Fortunately, they cannot issue duplicate addresses while in this state. However, having a server in this state over longer periods is not a good idea, because there are restrictions on what a server can do. The main server cannot reallocate expired leases and the backup server can run out of addresses from its pool. COMMUNICATIONS-INTERRUPTED state was designed for servers to easily survive transient communication failures of a few minutes to a few days. A server might function effectively in this state for only a short time, depending on the client arrival and departure rate. After that, it would be better to move a server into PARTNER-DOWN state so it can completely take over the lease functions until the servers resynchronize.

There are two ways a server can move into PARTNER-DOWN state:

- **User action**—An administrator sets a server into PARTNER-DOWN state based on an accurate assessment of reality. The failover protocol handles this correctly. Never set both partners to PARTNER-DOWN.
- **Failover safe period expires**—When the servers run unattended for longer periods, they need an automatic way to enter PARTNER-DOWN state.

Network operators might not sense in time that a server is down or uncommunicative. Hence, the failover safe period, which provides network operators some time to react to a server moving into COMMUNICATIONS-INTERRUPTED state. During the safe period, the only requirement is that the operators determine that both servers are still running and, if so, fix the network communications failure or take one of the servers down before the safe period expires.

During this safe period, either server allows renewals from any existing client, but there is a major risk of possibly issuing duplicate addresses. This is because one server can suddenly enter PARTNER-DOWN state while the other is still operating. Because of this risk, the failover safe period is disabled by default. That is why it is best to enable the safe period only if, during a server failure, it is more important to get an address than risk receiving a duplicate one.

The length of the safe period is installation-specific, and depends on the number of unallocated addresses in the pool and the expected arrival rate of previously unknown clients requiring addresses. The safe period is typically 24 hours, although many environments can support periods of several days.

The number of extra addresses required for the safe period should be the same as the expected total of new clients a server encounters. This depends on the arrival rate of new clients, not the total outstanding leases. Even if you can only afford a short safe period, because of a dearth of addresses or a high arrival

rate of new clients, you can benefit substantially by allowing DHCP to ride through minor problems that are fixable in an hour. There is minimum chance of duplicate address allocation, and reintegration after the solved failure is automatic and requires no operator intervention.

Here are some guidelines to follow, to help you decide whether to use manual intervention or the safe period for transitioning to PARTNER-DOWN state:

- If your corporate policy is to have minimal manual intervention, set the safe period. Enable the failover pair attribute *use-safe-period* to enable the safe period. Then, set the DHCP attribute *safe-period* to set the duration (86400 seconds, or 24 hours, by default). Set this duration long enough so that operations personnel can explore the cause of the communication failure and assure that the partner is truly down. At least 12 hours is recommended.
- If your corporate policy is to avoid conflict under any circumstances, then never let the backup server go into PARTNER-DOWN state unless by explicit command. Allocate sufficient addresses to the backup server so that it can handle new client arrivals during periods when there is no administrative coverage. You can set PARTNER-DOWN on the View Failover Related Server page of the regional cluster web UI, if the partner is in the Communications-interrupted failover state, you can click **Set Partner Down** in association with an input field for the PARTNER-DOWN date setting. This setting is initialized to the value of the *start-of-communications-interrupted* attribute. (In Normal web UI mode, you cannot set this date to be an earlier value than the initialized date. In Expert web UI mode, you can set this value to any date.) After clicking **Set Partner Down**, you return to the List Related Servers for DHCP Server page to view the result of the PARTNER-DOWN action. Never set both partners to PARTNER-DOWN.

Use **dhcp setPartnerDown** in the CLI, specifying the name of the partner server. This moves all the scopes running failover with the partner into PARTNER-DOWN state immediately, unless you specify a date and time with the command. This date and time should be when the partner was last known to be operational.

There are two conventions for specifying the date:

- *-num unit* (a time in the past), where *num* is a decimal number and *unit* is *s*, *m*, *h*, *d*, or *w* for seconds, minutes, hours, days or weeks respectively. For example, specify *-3d* for three days. Month (name or its first three letters), day, hour (24-hour convention), year (fully specified year or last two digits).



#### Note

Wherever you specify a date and time in the CLI, enter the time that is local to the **nrcmd** process. If the server is running in a different time zone than this process, disregard the time zone where the server is running and use local time instead.

## Setting DHCP Request and Response Packet Buffers

The number of request buffers (set through the *max-dhcp-requests* DHCP attribute) sets the maximum number of simultaneous requests that the server can accept. The default value is 500, which is suitable for most deployments, but can be tuned to the capacity of the server. This capacity relates to the leasing rate and average latency of the leasing transaction. For example, if clients receive new leases every 250 milliseconds, then a request buffer value of 500 is sufficient for the server to respond to 2000 clients per second. (This assumes sufficient processing capacity to service clients at that rate.) A lower value would throttle the server performance below this capacity. A higher value allows servicing a greater number of clients without retries during periods of burst load, but results in a higher average latency to each client. Average latency under two seconds is sufficient to properly service clients.

The number of response buffers (set through the *max-dhcp-responses* DHCP attribute) sets the maximum number of simultaneous requests that the server can complete by issuing a client response. When the network operates at a steady state, responses should track with the number of requests accepted. Because the same pool of response buffers serves for both lease and failover activity, when failover is enabled, the server adjusts the response buffer value to be at least four times the request buffers. This ensures that sufficient resources are available to process all pending client and failover activity simultaneously.

## Changing Polling Attributes

You can change some system defaults, such as the number of leases that the main server should send to the backup server, or the MCLT. See the “[Setting the Maximum Client Lead Time](#)” section on page 27-18. However, you need to change them on both servers.

On each server:

- **Change the poll interval (DHCP attribute *failover-poll-interval*)**—The interval that partners contact each other to confirm network connectivity. The preset value is 10 seconds.
- **Change the poll timeout (DHCP attribute *failover-poll-timeout*)**—Failover partners who cannot communicate during this timeout period will conclude that they lost network connectivity, and change their operational states appropriately. The preset value is 60 seconds.

Generally, you should not have to change the *failover-poll-timeout*. It is intimately linked to the *failover-poll-interval* and is based on real world experience. Note that for failover load balancing, until the backup detects that the main server is down (after the *failover-poll-timeout* period), the backup discards any requests it receives from clients targeting the main server.



### Note

To collect subnet utilization history for the failover pair, if you are configuring simple failover, disable individual polling of the main and backup DHCP servers, but enable failover pair polling by setting the failover pair attribute *poll-subnet-util-interval*, so as to collect one set of data from both servers.

## Setting the Network Discovery Attribute

If you enable failover on a UNIX system, you could set the *sms-network-discovery* attribute to enable the computing client os-type for leased addresses, which can help if you have a Windows partner server and want to use **dhcp updateSms** in the CLI on it.

## Setting Load Balancing

In normal failover mode, the main DHCP server bears most of the burden of servicing clients when the failover partners are in NORMAL communication mode. The main server not only services all new client requests, but has to handle renewal and rebinding requests and expired leases from the backup partner. To distribute the load more evenly between the two servers in a simple failover configuration scenario, Cisco Network Registrar introduced the load balancing feature (based on RFC 3074).

Failover load balancing allows both servers to actively service clients and determine which unique clients each will serve without running the risk of both servicing the same ones. Failover load balancing applies only while the servers are in NORMAL mode; in other states, both servers can respond to clients.

According to RFC 3074, the servers calculate a hash value for each request that the server receives, based on the client identifier option value or hardware address. The request is serviced if the hash value is assigned to that server.

With failover load balancing enabled, the servers split the client load evenly. The main partner processes 50% of the hash values and the backup partner the other 50%.

Each partner responds to all clients whenever a partner is not in NORMAL mode. Each partner responds only to the broadcast DHCPDISCOVER messages from clients that are in their assigned hash values.

For broadcast DHCPREQUESTs, the server responds only if it is the targeted one (based on the server identifier option); so, if the targeted server is the main server and it is down, the backup does not service the client (unless you release the lease). Broadcast BOOTP and DHCPINFORM requests are also load-balanced.

### See Also

[Load Balancing Compatibility with Earlier Cisco Network Registrar Versions](#)  
[Configuring Load Balancing](#)

## Load Balancing Compatibility with Earlier Cisco Network Registrar Versions

Failover load balancing is disabled by default to ensure backward compatibility with earlier Cisco Network Registrar releases, and is used only if both servers support load balancing. Hence, the failover pair load balancing is unset and the default value of disabled applies until you explicitly enable it. If you enable load balancing, each server services about 50% of the clients, and the free leases given to the backup will be 50%, regardless of the configured percentage (see the [“Setting Backup Percentages”](#) section on page 27-15).

## Configuring Load Balancing

In the web UI, when setting the failover properties for the pair (see the [“Creating and Synchronizing Failover Server Pairs”](#) section on page 27-6), enable or disable the *load-balancing* attribute in the Failover Settings attributes as desired to enable or disable failover load balancing. In the CLI, use **failover-pair *name* set load-balancing**.

# Changing Failover Server Roles



### Caution

Be careful when you change the role of a failover server. Remember that all address states in a scope are lost from a server if it is ever reloaded without that scope in its configuration.

### See Also

[Making Nonfailover Servers Failover Mains](#)  
[Replacing Servers Having Defective Storage, page 27-23](#)  
[Removing Backup Servers and Halting Failover Operation, page 27-24](#)  
[Adding Main Servers to Existing Backup Servers, page 27-24](#)  
[Configuring Failover on Multiple Interface Hosts, page 27-24](#)

## Making Nonfailover Servers Failover Mains

You can update an existing installation and increase the availability of the DHCP service it offers. You can use this procedure only if the original server never participated in failover.

- 
- Step 1** Install Cisco Network Registrar on the original server and ensure that it operates correctly after the installation.
  - Step 2** Install Cisco Network Registrar on the machine that is to be the backup server. Note the machine DNS name.
  - Step 3** Enable failover on the original server. Use the DNS name of the recently installed backup server. See the “Simple Failover” section on page 27-2.
  - Step 4** Reload the main server. It should go into PARTNER-DOWN state. It cannot locate the backup server, because it is not yet configured. There should be no change in main server operation at this point.
  - Step 5** Duplicate the main server configuration on the backup server, including scopes (including secondary), policies, and client-classes. If you use client-classes, make sure the clients are entered into each cluster or that each server can access an LDAP database with the client data.
  - Step 6** Enable failover on the backup server. Be sure to define the main server.
  - Step 7** Reconfigure all operational BOOTP relays to forward broadcast packets to the main and backup server.
  - Step 8** Reload the backup server.
- 

After you complete these steps:

1. The backup server detects the main server and moves into RECOVER state.
2. The backup server refreshes its stable storage with the main server lease data and, when complete, moves into RECOVER-DONE state.
3. The main server moves into NORMAL state.
4. The backup server moves into NORMAL state.
5. The backup server uses a pool request to ask the main server for addresses to allocate if communication is interrupted.
6. After allocating these addresses, the main server sends this data to the backup server.

## Replacing Servers Having Defective Storage

If a failover server loses its stable storage (hard disk), you can replace the server and have it recover its state information from its partner.

- 
- Step 1** Determine which server lost its stable storage.
  - Step 2** Use **dhcp setPartnerDown** in the CLI to tell the other server that its partner is down. If you do not specify a time, the current time is used.
  - Step 3** When the server is again operational, reinstall Cisco Network Registrar.
  - Step 4** Duplicate the server configuration from its partner. However, do not recover any lease databases from an earlier backup or the partner system.

**Step 5** Reload the replacement server.

---

After you complete these steps:

1. The recovered server moves into RECOVER state.
2. Its partner sends it all its data.
3. The server moves into RECOVER-DONE state when it reaches its maximum client lead time (and any time set for *failover-recover*).
4. Its partner moves into NORMAL state.
5. The recovered server moves into NORMAL state. It can request addresses, but can allocate few new ones, because its partner already sent it all its previously allocated addresses.

## Removing Backup Servers and Halting Failover Operation

Sometimes you might need to remove the backup server and halt all failover operations.

---

**Step 1** On the backup server, remove all the scopes that were designated as a backup to the main server.

**Step 2** On the main server, remove the failover capability from those scopes that were main for the backup server, or disable failover server-wide if that is how it was configured.

**Step 3** Reload both servers.

---

## Adding Main Servers to Existing Backup Servers

You can use an existing backup server for a main server.

---

**Step 1** Duplicate the main server scopes, policies, and other configurations on the backup server.

**Step 2** Configure the main server to enable failover and point to the backup server.

**Step 3** Configure the backup server to enable failover for the new scopes that point to the new main server.

**Step 4** Reload both servers. Cisco Network Registrar performs the same steps as those described in the [“Making Nonfailover Servers Failover Mains”](#) section on page 27-23.

---

## Configuring Failover on Multiple Interface Hosts

If you plan to use failover on a server host with multiple interfaces, you must explicitly configure the local server name or address. This requires an additional command. For example, if you have a host with two interfaces, serverA and serverB, and you want to make serverA the a main failover server, you must define serverA as the failover-main-server before you set the backup server name (external serverB). If you do not do this, failover might not initialize correctly and tries to use the wrong interface.

Set the DHCP server properties *failover-main-server* and *failover-backup-server*.



With multiple interfaces on one host, you must specify a hostname that points to only one address or a record. You cannot set up your servers for round-robin support.

## Restoring a Standalone DHCP Failover Server to Backup State

This section describes how to recreate a DHCP failover relationship between a main and backup server where a backup server was put in standalone mode. This situation does not come up very often.

An administrator may have to take the main failover server offline because of a hardware failure or manual shutdown of Cisco Network Registrar. The failover relationship with the main server will then be turned off, and the backup server will be pressed into service, temporarily, as a standalone DHCP server. Unfortunately, restoring the previous failover relationship from this condition can be hazardous to the lease state data.

According to the DHCP Failover protocol, if either of the partners maintained in a failover relationship fails, recovery is assured because the partners resynchronize. Even with a failed backup server, putting the main server in standalone mode would not be overly complicated to recover to failover mode.

However, restoring a standalone DHCP server to backup is not straightforward.

1. The standalone server assumes the role of the main server.
2. The original main server becomes the back up server.
3. The partners then synchronize.
4. Failover relationship to be intentionally broken to reverse the server roles.
5. Partners to resynchronize in their original failover roles.

### See Also

[Background](#)

[Repair Procedure](#)

[Restoring the Failover Pair with Reversed Roles, page 27-26](#)

[Starting with Server A Powered Off, page 27-27](#)

[Starting with Server A Powered On and Server Agent Disabled, page 27-28](#)

[Starting with Server A Replaced, page 27-29](#)

[Transferring Current Lease State to Server A, page 27-30](#)

## Background

For the remainder of this section, the main DHCP failover server is identified as Server A (with IP address 10.86.154.59), and the backup server as Server B (with IP address 10.86.154.60). Server A is administratively or otherwise shut down or its Cisco Network Registrar server agent gets stopped. At this point, Server B goes into the Communications-Interrupted mode.

The system administrator may then take one of the following approaches:

- **Continue running backup Server B in Communications-Interrupted mode**—The risk of running the backup server in this mode indefinitely is that it can exhaust the pool of typically 10% of the available addresses with which the backup server is allocated to service new clients.
- **Put Server B into Partner-Down mode without breaking the failover relationship**—One major caveat of giving the backup server full control of the address space, without suspending failover, is that the full transfer of the address space ownership does not occur until after the configured

Maximum Client Lead Time (MCLT). The MCLT is an additional time period set on the main server, which controls the duration for which the client lease expiration is ahead of what the backup server detects it to be. The MCLT is typically 60 minutes. Until the MCLT expires, the available address pool of the backup server is limited to its allocated reserve.

- **Put Server B into Partner-Down mode and break the failover relationship**—This approach puts the backup server in standalone mode, and is the approach that the administrator chose in this scenario. The deciding factors were that the main server was expected to be offline for an extended period, and the number of new devices coming online was higher than anticipated. Because the low percentage of available addresses that the backup server could service would soon cause an outage for new devices, the administrator put Server B in standalone mode. The disadvantage of this approach is the care and effort required to preserve the original state of the network when restoring the partners to their original relationship.

The first two approaches have distinct advantages over the third. In most cases, the backup server is expected to have enough addresses to cover newly arrived clients until the MCLT expires. Pursuing the third approach can incur unnecessary administrative burden and risk.

## Repair Procedure

The repair procedure is:

1. **Temporarily assign the backup Server B the role of the main failover server**—Reversing the failover partner roles effectively allows Server A to learn the current failover state from Server B.
2. **Migrate Server A and Server B back to their original failover roles**—The goal is for Server A to reacquire its original status as the main DHCP failover server.

The assumptions are:

- The Original main Server A is nonoperational and Cisco Network Registrar is stopped.
- The Original backup Server B is operational.
- Failover between the partners is administratively disabled.
- Decision was made not to permanently reverse the failover roles of the two partners.
- Domain Name Services (DNS) is not running on either of the failover partners.



### Note

The IP addresses used as examples are for demonstration purposes only.

## Restoring the Failover Pair with Reversed Roles

The following steps restore failover by temporarily moving Server B into the main server mode.

On **Server B** (10.86.154.60 or in cluster-B):

- Step 1** Ensure that failover is disabled. Modify the failover configuration, so that Server B becomes the main and Server A the backup:

- Cisco Network Registrar 6.3 and later:

```
nrcmd> failover-pair examplepair set failover=false
nrcmd> failover-pair examplepair set main=cluster-B
nrcmd> failover-pair examplepair set backup=cluster-A
```

**Step 2** Save the changes and reload the server:

```
nrcmd> save
nrcmd> dhcp reload
```

**Step 3** Reenable failover and reload the server again:

```
nrcmd> dhcp enable failover or failover-pair examplepair set failover=true
nrcmd> dhcp reload
```

Server B is now the main failover server, ready for its partner to become operational again. Any further action that you take to prevent Server A from beginning to give out addresses in the meantime depends on its current state.

If the Server A is:

- **Powered off**—See the “Starting with Server A Powered Off” section on page 27-27.
- **Powered on with the Cisco Network Registrar server agent disabled**—See the “Starting with Server A Powered On and Server Agent Disabled” section on page 27-28.
- **Replaced by another machine**—See the “Starting with Server A Replaced” section on page 27-29.

## Starting with Server A Powered Off

If Server A was powered off, you must power it on again to continue. The next steps ensure that Server A comes online while preventing IP address leakage.

On **Server A** (10.86.154.59 or in cluster-A):

**Step 1** Turn on the server, but prevent it from being active. How to do this is specific to your environment. Typically, if the machine is:

- Physically available, manually disconnect the network cable, then boot up the machine.
- Running SPARC Solaris and is managed remotely using reverse Telnet through a communication server, bring it online in single-user mode. This is enough to keep the DHCP server from starting:
  - Provided Cisco Network Registrar is not installed on one of the partitions that is mounted automatically in single-user mode.
  - After logging in as *root* in single-user mode, bring the partition on which Cisco Network Registrar is mounted online. This action makes the programs and data available without starting the server agent. (Normally, use the **mountall** command for this last step.)

**Step 2** Stop the Cisco Network Registrar server agent, if it is started:

- Solaris/Linux—`/etc/init.d/nwreglocal stop`
- Windows—`net stop nwreglocal`



**Note**

If it is not possible to bring the machine online without taking it off the network and starting the server agent, stop the server agent as quickly as possible. There might be a period during which the server can inadvertently give out addresses.

**Step 3** Go to the “Starting with Server A Powered On and Server Agent Disabled” section.

## Starting with Server A Powered On and Server Agent Disabled

Starting from a point where Server A is powered on, but the Cisco Network Registrar server agent is turned off, configure Cisco Network Registrar so that you can start the server agent without automatically enabling the DHCP server to give out addresses.

On **Server A** (10.86.154.59 or in cluster-A):

**Step 1** Prevent DHCP from starting on server agent startup:

a. Add the following content to a disableDHCP.txt file:

```
# version: 1.0
[config/cluster/1/trampolines/1/servers/2]
config_path = str:[0]servers/name/DHCP/1
enabled = int32:[0]0
```



**Note** The syntax, case, and spacing are critical for this text.

Setting enabled to 0 disables the DHCP service while allowing the DHCP server to start up.

**Step 2** Restart the server agent:

- Solaris/Linux—`/etc/init.d/nwreglocal start`
- Windows—`net start nwreglocal`

Cisco Network Registrar comes online, but the DHCP service will be inactive.

**Step 3** Examine the DHCP logs to confirm that the DHCP server is not running.

**Step 4** Bring Server A back on the network. If:

- The network cable is unplugged, restore the network connection.
- You are logged on in single-user mode, reboot the server.

**Step 5** Reverse the partner roles and remove the failover state data:

a. Modify the failover configuration so that Server A becomes the backup server and enable failover:

• Cisco Network Registrar 6.3 and later:

```
nrcmd> failover-pair examplepair set main=cluster-B
nrcmd> failover-pair examplepair set backup=cluster-A
nrcmd> failover-pair examplepair set failover=true
```

b. Set the DHCP service to be enabled on reboot and save the changes:

```
nrcmd> dhcp enable start-on-reboot
nrcmd> save
```



**Note** Do not reload the DHCP server at this point.

c. Remove all failover state data. To do this:

- Stop the server agent.
- Ensure that all Cisco Network Registrar processes are terminated.
- Kill any residual processes.

- Delete the event store and lease state databases.
- Delete the server level state
- Restart the server agent.

Solaris/Linux	<pre> /etc/init.d/nwreglocal stop ps -leaf   grep nwr kill -9 pid rm /var/nwreg2/local/data/dhcpeventstore/*.* rm -r /var/nwreg2/local/data/dhcp/ndb/*.* cd /opt/nwreg2/local/usrbin /etc/init.d/nwreglocal start </pre>
Windows	<pre> net stop nwreglocal cd install-path\local\data delete dhcpeventstore\*.* delete dhcp\ndb\*.* cd install-path\local\bin net start nwreglocal </pre>

**Step 6** Go to the “[Transferring Current Lease State to Server A](#)” section on page 27-30.

## Starting with Server A Replaced

If Server A was decommissioned and replaced, you must install Cisco Network Registrar and push the failover configuration from Server B to the new machine. Also, you must restore any customer configuration specific to Server A. After these steps, Cisco Network Registrar will start but not give out addresses:

- 
- Step 1** On **Server A** (10.86.154.59 or in cluster-A), install Cisco Network Registrar.
- Step 2** Reconstruct the Cisco Network Registrar operating environment by restoring the accompanying software, such as Cisco Broadband Access Center and its required DHCP extensions. Do not make any administrative changes to the configuration until after pushing the configuration to Server B.
- Step 3** On **Server B** (10.86.154.60 or in cluster-B), by using the Cisco Network Registrar web UI, push an exact failover configuration to Server A. This effectively makes Server A the backup partner.
- Step 4** On **Server A**:
- a. Save the new configuration, but do not reload the server:
 

```
nrcmd> save
```
  - b. If necessary, customize the Cisco Network Registrar configuration as required for the operating environment, which might include making administrative changes.
  - c. Ensure that the configuration is complete.
  - d. Reload the DHCP server:
 

```
nrcmd> dhcp reload
```
- Step 5** Go to the “[Transferring Current Lease State to Server A](#)” section.
-

## Transferring Current Lease State to Server A

- At this point, the failover partnership reestablishes itself, both servers will resynchronize their states.
- Server A becomes operational as the backup server.
- The operation will pause for the MCLT period (of one hour) and both partners resume their failover operations in normal communication mode.



### Note

Do not proceed to the “[Repairing Partners to Their Original Roles](#)” section until both partners synchronize and report normal communication.

## Repairing Partners to Their Original Roles

Assume that both partners are fully synchronized and report normal communication. To ensure that the failover partners can assume their original roles, you should:

**Step 1** On **Server A** (10.86.154.59 or in cluster-A), stop the DHCP server:

```
nrcmd> dhcp stop
```

**Step 2** On **Server B** (10.86.154.60 or in cluster-B), stop the DHCP server:

```
nrcmd> dhcp stop
```

**Step 3** On **Server A**:

a. Disable failover, then make Server A the main server and Server B the backup:

- Cisco Network Registrar 6.3 and later:

```
nrcmd> failover-pair examplepair set failover=false
nrcmd> failover-pair examplepair set main=cluster-A
nrcmd> failover-pair examplepair set backup=cluster-B
```

b. Save the changes and reload DHCP:

```
nrcmd> save
nrcmd> dhcp reload
```

c. Ensure that the configuration is in place and currently running. At this point, Server A is the sole operational DHCP server with 100% of the address pool.

d. Reenable failover:

```
nrcmd> dhcp enable failover or failover-pair examplepair set failover=true
```

e. Reload DHCP and double-check the configuration changes:

```
nrcmd> dhcp reload
```

Server A is now the failover main server awaiting Server B to become operational.

**Step 4** On **Server B**:

a. Make Server A the main server and Server B the backup, then enable failover:

- Cisco Network Registrar 6.3 and later:

```
nrcmd> failover-pair examplepair set main=cluster-A
```

```
nrcmd> failover-pair examplepair set backup=cluster-B
nrcmd> failover-pair examplepair set failover=true
```

- b. Save the new configuration, but do not reload the server:

```
nrcmd> save
```

- c. Remove all failover state data. To do this:

- Stop the server agent.
- Ensure that all Cisco Network Registrar processes are terminated.
- Kill any residual processes.
- Delete the event store and lease state databases.
- Delete the server level state.
- Restart the server agent.

Solaris/Linux	<pre>/etc/init.d/nwreglocal stop ps -leaf   grep nwr kill -9 pid rm /var/nwreg2/local/data/dhcpeventstore/*.* rm -r /var/nwreg2/local/data/dhcp/ndb/*.* cd /opt/nwreg2/local/usrbin /etc/init.d/nwreglocal start</pre>
Windows	<pre>net stop nwreglocal cd install-path\local\data delete dhcpeventstore\*.* delete dhcp\ndb\*.* cd install-path\local\bin net start nwreglocal</pre>

At this point, the failover partnership reestablishes itself in its original roles, both servers will resynchronize their states, and Server B becomes operational as the backup server. The operation will pause for the MCLT period (of one hour) and both partners resume their failover operations in normal communication mode.

#### Step 5 On Server A and Server B:

- a. Validate whether both partners are in normal failover state:

```
nrcmd> dhcp getRelatedservers
```

- b. Run a report and ensure that the results match on both partners, allowing a bit of skew for the difference in running times between the partners.

## Recovering in Failover Configuration

When you upgrade Cisco Network Registrar to the latest version, you can revert to the earlier version, in case the upgrade fails. You can upgrade one partner and when it has recovered to normal state and is working well, then upgrade the other partner.

You may be able to recover from the archive created during the upgrade, but if the upgrade is scheduled during a maintenance window then, you need to:

- Stop Cisco Network Registrar completely using `nwreglocal stop`.
- Tar up the Cisco Network Registrar DATADIR (`/var/nwreg2/local/data`) and save it in a safe location.

- Upgrade the server.

If it fails, then you need to:

- Stop Cisco Network Registrar completely using `nwreglocal stop`.
- Program `nwreg2` (Cisco Network Registrar).
- Delete the corrupt version of Cisco Network Registrar DATADIR (The location is: `/var/nwreg2/local/data`).
- Extract the saved Cisco Network Registrar DATADIR tarfile in the path the tarfile came from.
- Install original version of Cisco Network Registrar, which finds the existing DATADIR and use it.

## Supporting BOOTP Clients in Failover

You can configure scopes to support two types of BOOTP clients—static and dynamic.

### See Also

[Static BOOTP](#)

[Dynamic BOOTP](#)

[Configuring BOOTP Relays, page 27-33](#)

## Static BOOTP

You can support static BOOTP clients using DHCP reservations. When you enable failover, remember to configure both the main and the backup server with identical reservations.

## Dynamic BOOTP

You can enable dynamic BOOTP clients by enabling the *dynamic-bootp* attribute on a scope. When using failover, however, there are additional restrictions on address usage in such scopes, because BOOTP clients get permanent addresses and leases that never expire.

When a server whose scope does not have the *dynamic-bootp* option enabled goes to PARTNER-DOWN state, it can allocate any available (unassigned) address from that scope, whether or not it was initially available to any partner. However, when the *dynamic-bootp* option is set, each partner can only allocate its own addresses. Consequently, scopes that enable the *dynamic-bootp* option require more addresses to support failover.

When using dynamic BOOTP:

- Segregate dynamic BOOTP clients to a single scope. Disable DHCP clients from using that scope by disabling the *dhcp* attribute on the scope.
- Set the *dynamic-bootp-backup-pct* failover pair attribute to allocate a greater percentage of addresses to the backup server for this scope, as much as 50 percent higher than a regular backup percentage.



## Configuring BOOTP Relays

The Cisco Network Registrar failover protocol works with BOOTP relay (also called IP helper), a router capability that supports DHCP clients that are not locally connected to a server.

If you use BOOTP relay, ensure that the implementations point to both the main and backup servers. If they do not and the main fails, clients are not serviced, because the backup cannot see the required packets. If you cannot configure BOOTP relay to forward broadcast packets to two different servers, configure the router to forward the packets to a subnet-local broadcast address for a LAN segment, which could contain both the main and backup servers. Then, ensure that both the main and backup servers are on the same LAN segment.

## DHCPLEASEQUERY and Failover

To accommodate DHCPLEASEQUERY messages sent to a DHCP failover backup server when the master server is down, the master server must communicate the *relay-agent-info* (82) option values to its partner server. To accomplish this, the master server uses DHCP failover update messages.

## Troubleshooting Failover

This section describes how to avoid failover configuration mistakes, monitor failover operations, and detect and handle network problems.


### See Also

[Monitoring Failover Operations](#)  
[Detecting and Handling Network Failures, page 27-33](#)

## Monitoring Failover Operations

You can examine the DHCP server log files on both partner servers to verify your failover configuration.

You can make a few important log and debug settings to troubleshoot failover. Set the DHCP log settings to *failover-detail* to track the number and details of failover messages logged. To ensure that previous messages do not get overwritten, add the *failover-detail* attribute to the end of the list. Use the *no-failover-conflict* attribute to inhibit logging server failover conflicts, or the *no-failover-activity* attribute to inhibit logging normal server failover activity. Then, reload the server.

You can also isolate misconfigurations more easily by clicking the Related Servers icon () on the Manage DHCP Server or List/Add DHCP Failover Pairs page, or by using `dhcp getRelatedServers` in the CLI.

## Detecting and Handling Network Failures

[Table 27-4](#) describes some symptoms, causes, and solutions for failover problems.

**Table 27-4**     **Detecting and Handling Failures**

Symptom	Cause	Solution
New clients cannot get addresses	A backup server is in COMMUNICATIONS-INTERRUPTED state with too few addresses	Increase the backup percentage on the main server.
Error messages about mismatched scopes	There are mismatched scope configurations between partners	Reconfigure your servers.
Log messages about failure to communicate with partner	Server cannot communicate with its partner	Check the status of the server.
Main server fails. Some clients cannot renew or rebind leases. The leases expire even when the backup server is up and possibly processing some client requests.	Some BOOTP relay (ip-helper) was not configured to point at both servers; see the <a href="#">“Configuring BOOTP Relays” section on page 27-33</a> .	<ul style="list-style-type: none"> <li>• Reconfigure BOOTP relays to point at both main and backup server</li> <li>• Run a fire drill test—Take the main server down for a day or so and see if your user community can get and renew leases</li> </ul>
SNMP trap: other server not responding	Server cannot communicate with its partner	Check the status of the server.
SNMP trap: dhcp failover configuration mismatch	Mismatched scope configurations between partners	Reconfigure your servers.
Users complain that they cannot use services or system as expected	Mismatched policies and client-classes between partners	Reconfigure partners to have identical policies; possibly use LDAP for client registration if currently registering clients directly in partners.