

Stateful Network Address Translation 64 Interchassis Redundancy

The Stateful Network Address Translation 64 Interchassis Redundancy feature adds interchassis redundancy support to stateful Network Address Translation 64 (NAT64). The stateful interchassis redundancy enables you to configure pairs of devices to act as backups for each other.

This module describes how to configure stateful NAT64 interchassis redundancy.

- Restrictions for Stateful Network Address Translation 64 Interchassis Redundancy, on page 1
- Information About Stateful Network Address Translation 64 Interchassis Redundancy, on page 1
- How to Configure Stateful Network Translation 64 Interchassis Redundancy, on page 6
- Configuration Examples for Stateful Network Address Translation 64 Interchassis Redundancy, on page 15
- Additional References, on page 17

Restrictions for Stateful Network Address Translation 64 Interchassis Redundancy

- Asymmetric routing is not supported.
- Box-to-box (B2B) redundancy along with intrachassis redundancy is not supported.
- NAT interface overload configuration is not supported.

Information About Stateful Network Address Translation 64 Interchassis Redundancy

Stateful Interchassis Redundancy Operation

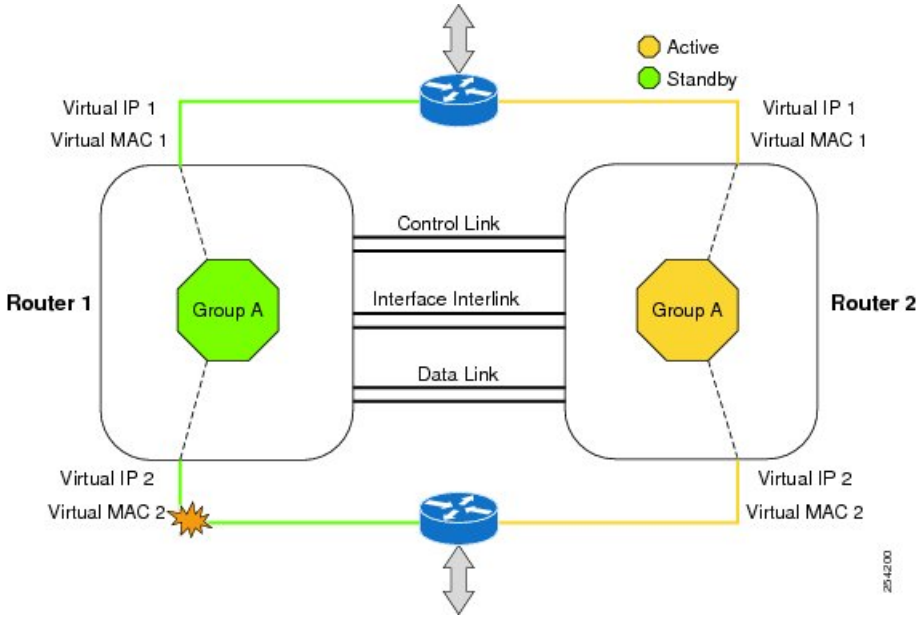
You can configure pairs of devices to act as hot standbys for each other. Redundancy is configured on an interface basis. Pairs of redundant interfaces are known as redundancy groups (RGs). Redundancy occurs at an application level and does not require a complete physical failure of the interface or device for a switchover

of the application to occur. When a switchover occurs, the application activity continues to run seamlessly on the redundant interface.

The figure below depicts an active/standby load-sharing scenario. The figure shows how an RG is configured for a pair of devices that has one outgoing interface. Group A on Router 1 is the active RG and Group A on Router 2 is the standby RG.

Redundant devices are joined by a configurable control link and a data synchronization link. The control link is used to communicate the status of devices. The data synchronization link is used to transfer stateful information from Network Address Translation (NAT) and the firewall and synchronize the stateful database. The pairs of redundant interfaces are configured with the same unique ID number known as the redundant interface identifier (RII).

Figure 1: Redundancy Group Configuration—One Outgoing Interface



The status of redundancy group members is determined through the use of hello messages sent over the control link. The software considers either device not responding to a hello message within a configurable amount of time to be a failure and initiates a switchover. For the software to detect a failure in milliseconds, control links run the failover protocol that is integrated with the Bidirectional Forwarding Detection (BFD) protocol. You can configure the following parameters for hello messages:

- Hello time—Interval at which hello messages are sent.
- Hold time—Amount of time before which the active or standby device is declared to be down.

The hello time defaults to 3 seconds to align with the Hot Standby Router Protocol (HSRP), and the hold time defaults to 10 seconds. You can also configure these timers in milliseconds by using the **timers hellotime msec** command.

To determine the pairs of interfaces that are affected by the switchover, you must configure a unique ID for each pair of redundant interfaces. This ID is known as the RII that is associated with the interface.

A switchover to the standby device can occur when the priority setting that is configured on each device changes. The device with the highest priority value acts as the active device. If a fault occurs on either the active or standby device, the priority of the device is decremented by a configurable amount known as the weight. If the priority of the active device falls below the priority of the standby device, a switchover occurs

and the standby device becomes the active device. This default behavior can be overridden by disabling the preemption attribute for the RG. You can also configure each interface to decrease the priority when the Layer 1 state of the interface goes down. The priority that is configured overrides the default priority of an RG.

Each failure event that causes a modification of an RG priority generates a syslog entry that contains a time stamp, the RG that was affected, the previous priority, the new priority, and a description of the failure event cause.

A switchover also can occur when the priority of a device or interface falls below a configurable threshold level.

A switchover to the standby device occurs under the following circumstances:

- Power loss or a reload occurs on the active device (including reloads).
- The run-time priority of the active device goes below that of the standby device (with preempt configured).
- The run-time priority of the active device goes below that of the configured threshold.
- The redundancy group on the active device is reloaded manually. Use the **redundancy application reload group** *rg-number* command for a manual reload.

Active/Active Failover

In an active/active failover configuration, both devices can process network traffic. Active/active failover generates virtual MAC (VMAC) addresses for interfaces in each redundancy group (RG).

One device in an active/active failover pair is designated as the primary (active) device, and the other is designated as the secondary (standby) device. Unlike with active/standby failover, this designation does not indicate which device becomes active when both devices start simultaneously. Instead, the primary/secondary designation determines the following:

- The device that provides the running configuration to the failover pair when they start simultaneously.
- The device on which the failover RG appears in the active state when devices start simultaneously. Each failover RG in the configuration is configured with a primary or secondary device preference. You can configure both failover RGs to be in the active state on a single device and the standby failover RGs to be on the other device. You can also configure one failover RG to be in the active state and the other RG to be in the standby state on a single device.

Active/Standby Failover

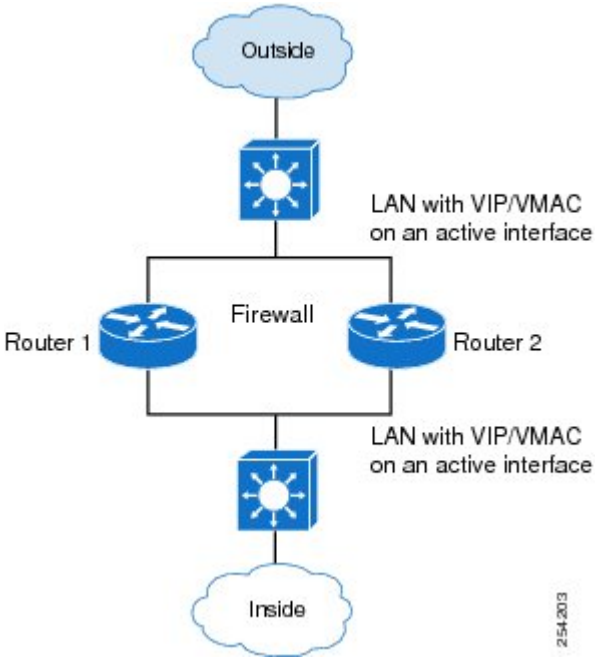
Active/standby failover enables you to use a standby device to take over the functionality of a failed device. A failed active device changes to the standby state, and the standby device changes to the active state. The device that is now in the active state takes over IP addresses and MAC addresses of the failed device and starts processing traffic. The device that is now in the standby state takes over standby IP addresses and MAC addresses. Because network devices do not see any change in the MAC-to-IP address pairing, Address Resolution Protocol (ARP) entries do not change or time out anywhere on the network.

In an active/standby scenario, the main difference between two devices in a failover pair depends on which device is active and which device is a standby, namely which IP addresses to use and which device actively passes the traffic. The active device always becomes the active device if both devices start up at the same time (and are of equal operational health). MAC addresses of the active device are always paired with active IP addresses.

LAN-LAN Topology

In a LAN-LAN topology, all participating devices are connected to each other through LAN interfaces on both the inside and the outside. In this scenario, the traffic is often directed to the correct firewall if static routing is configured on the upstream or downstream devices to an appropriate virtual IP address. The dynamic routing configuration supported on LAN-facing interfaces must not introduce a dependency on routing protocol convergence; otherwise, fast failover requirements will not be met. The figure below shows a LAN-LAN topology.

Figure 2: LAN-LAN Scenario



Redundancy Groups for Stateful NAT64

To support stateful Network Address Translation 64 (NAT64) box-to-box (B2B) redundancy, all stateful NAT64 mappings must be associated with a redundancy group (RG). You can associate multiple stateful NAT64 mappings with one RG. Any session or bind that is created from a stateful NAT64 mapping is associated with the RG to which the stateful NAT64 is mapped. In B2B redundancy, stateful NAT64 checks the state of the created, changed, or destroyed session or bind in the RG to determine whether the stateful NAT64 high availability (HA) message should be sent to the standby device.

NAT binding is a one-to-one association between a local IP address and a global IP address. Sessions are identified by the 5-tuple (the source IP address, the destination IP address, the protocol, the source port, and the destination port) information. Sessions are normally created and destroyed at a much faster rate than bindings.

Translation Filtering

RFC 4787 provides translation filtering behaviors for Network Address Translation (NAT). The following options are used by NAT to filter packets that originate from specific external endpoints:

- Endpoint-independent filtering—Filters out packets that are not destined to an internal IP address and port regardless of the external IP address and port source.
- Address-dependent filtering—Filters out packets that are not destined to an internal IP address. NAT also filters out packets that are destined for an internal endpoint.
- Address- and port-dependent filtering—Filters out packets that are not destined to an internal IP address. NAT also filters out packets that are destined for an internal endpoint if packets were not sent to the endpoint previously.

FTP64 Application-Level Gateway Support

The FTP64 (or service FTP) application-level gateway (ALG) helps stateful Network Address Translation 64 (NAT64) to operate on Layer 7 data. FTP64 ALG translates IP addresses and the TCP port information embedded in the payload of an FTP control session.

NAT translates any TCP/UDP traffic that does not carry source and destination IP addresses in the application data stream. Protocols that embed the IP address information within the payload (or in the application data stream) require the support of an ALG. ALGs handle application data stream (Layer 7) protocol-specific services, such as translating embedded IP addresses and port numbers in the packet payload and extracting new connection or session information from control channels.

FTP64 is automatically enabled when Stateful NAT64 is enabled. Use the **no nat64 service ftp** command to disable the NAT64 FTP service.



Note The FTP64 ALG is not supported in Stateless NAT64 translation.



Note The FTP64 ALG does not support IPv4-compatible IPv6 addresses.

Based on *IPv6-to-IPv4 translation FTP considerations draft-ietf-behave-ftp64-02* and RFC 2228, the FTP64 ALG must switch to transparent mode (a device in a transparent mode is invisible in the network; however, this device can act as a bridge and inspect or filter packets), when commands and responses flow between the FTP client and the FTP server. When a client issues the FTP AUTH command, the FTP64 ALG transparently forwards all data on the control channel in both (ingress and egress) directions, until the end of the control channel session. Similarly, during an AUTH negotiation, the ALG must be in transparent mode, whether the negotiation is successful or not.

Based on RFC 6384, the behavior of the FTP64 ALG during a client-server communication is different. During an IPv6-to-IPv4 translation, the FTP64 ALG must transparently copy data transmitted over the control channel so that the transport layer security (TLS) session works correctly. However, the client commands and server responses are hidden from the FTP64 ALG. To ensure a consistent behavior, as soon as the initial FTP AUTH command is issued by a client, the FTP64 ALG must stop translating commands and responses and start transparently copying TCP data that is sent by the server to the client and vice versa. The FTP64 ALG must ignore the AUTH command and not go into transparent mode if the server response is in the 4*xx* or 5*xx* ranges, which comprise FTP error/warning messages.

Prior to CSCtu37975, when an IPv6 FTP client issues an FTP AUTH command, irrespective of whether the IPv4 FTP server accepts or rejects that authorization negotiation, the FTP64 ALG moves the AUTH session

to transparent mode (or bypass mode). When a session is in transparent mode, NAT cannot perform translation on the packets within the session. With CSCtu37975, during a client-server communication, the FTP64 ALG's behavior is compliant with RFC 6384.

How to Configure Stateful Network Translation 64 Interchassis Redundancy

Configuring Redundancy Group Protocols

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **redundancy**
4. **application redundancy**
5. **protocol** *id*
6. **name** *group-name*
7. Repeat Steps 3 to 6 to configure a redundancy group protocol on another device.
8. **timers hellotime** *seconds* **holdtime** *seconds*
9. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**redundancy** **Example:** `Device(config)# redundancy`	Enters redundancy configuration mode.
Step 4	**application redundancy** **Example:** `Device(red)# application redundancy`	Configures application redundancy and enters redundancy application configuration mode.

	Command or Action	Purpose
Step 5	**protocol** *id* **Example:** `Device(config-red-app)# protocol 1`	Defines a protocol instance for a redundancy group and enters redundancy application protocol configuration mode.
Step 6	**name** *group-name* **Example:** `Device(config-red-app-prtcl)# name RG1`	Configures a name for the redundancy group.
Step 7	Repeat Steps 3 to 6 to configure a redundancy group protocol on another device.	—
Step 8	**timers hellotime** *seconds* **holdtime** *seconds* **Example:** `Device(config-red-app-prtcl)# timers hellotime 1 holdtime 3`	Configures timers for hellotime and holdtime messages for a redundancy group.
Step 9	**end** **Example:** `Device(config-red-app-prtcl)# end`	Exits redundancy application protocol configuration mode and enters privileged EXEC mode.

Configuring Redundancy Groups for Active/Standby Load Sharing

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **redundancy**
4. **application redundancy**
5. **group** *id*
6. **name** *group-name*
7. **control** *interface-type interface-number* **protocol** *id*
8. **data** *interface-type interface-number*
9. Repeat Steps 3 to 8 to configure another redundancy group.
10. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**redundancy** **Example:** `Device(config)# redundancy`	Enters redundancy configuration mode.
Step 4	**application redundancy** **Example:** `Device(config-red)# application redundancy`	Configures application redundancy and enters redundancy application configuration mode.
Step 5	**group** *id* **Example:** `Device(config-red-app)# group 1`	Configures a redundancy application group and enters redundancy application group configuration mode.
Step 6	**name** *group-name* **Example:** `Device(config-red-app-grp)# name RG1`	Configures a name for the redundancy application group.
Step 7	**control** *interface-type interface-number* **protocol** *id* **Example:** `Device(config-red-app-grp)# control gigabitethernet 0/0/1 protocol 1`	Configures a control interface type and number for the redundancy application group.
Step 8	**data** *interface-type interface-number* **Example:** `Device(config-red-app-grp)# data gigabitethernet 0/2/2`	Configures a data interface type and number for the redundancy application group.
Step 9	Repeat Steps 3 to 8 to configure another redundancy group.	—
Step 10	**end** **Example:** `Device(config-red-app-grp)# end`	Exits redundancy application group configuration mode and enters privileged EXEC mode.

Configuring Redundancy Groups for Active/Active Load Sharing

Perform this task to configure two redundancy groups (RGs) on the same device for active/active load sharing.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **redundancy**

4. **application redundancy**
5. **group** *id*
6. **name** *group-name*
7. **priority** *value* [**failover-threshold** *value*]
8. **control** *interface-type interface-number* **protocol** *id*
9. **data** *interface-type interface-number*
10. **end**
11. **configure terminal**
12. **redundancy**
13. **application redundancy**
14. **group** *id*
15. **name** *group-name*
16. **priority** *value* [**failover-threshold** *value*]
17. **control** *interface-type interface-number* **protocol** *id*
18. **data** *interface-type interface-number*
19. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**redundancy** **Example:** `Device(config)# redundancy`	Enters redundancy configuration mode.
Step 4	**application redundancy** **Example:** `Device(config-red)# application redundancy`	Configures application redundancy and enters redundancy application configuration mode.
Step 5	**group** *id* **Example:** `Device(config-red-app)# group 1`	Configures a redundancy application group and enters redundancy application group configuration mode.
Step 6	**name** *group-name* **Example:** `Device(config-red-app-grp)# name RG1`	Configures a name for the redundancy application group.

	Command or Action	Purpose
Step 7	**priority** *value* [**failover-threshold** *value*] **Example:** `Device(config-red-app-grp)# priority 195 failover-threshold 190`	Specifies a group priority and failover threshold value for the redundancy group.
Step 8	**control** *interface-type interface-number* **protocol** *id* **Example:** `Device(config-red-app-grp)# control gigabitethernet 0/0/1 protocol 1`	Configures a control interface type and number for the redundancy application group.
Step 9	**data** *interface-type interface-number* **Example:** `Device(config-red-app-grp)# data gigabitethernet 0/2/2`	Configures a data interface type and number for the redundancy application group.
Step 10	**end** **Example:** `Device(config-red-app-grp)# end`	Exits redundancy application group configuration mode and enters privileged EXEC mode.
Step 11	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 12	**redundancy** **Example:** `Device(config)# redundancy`	Enters redundancy configuration mode.
Step 13	**application redundancy** **Example:** `Device(config-red)# application redundancy`	Configures application redundancy and enters redundancy application configuration mode.
Step 14	**group** *id* **Example:** `Device(config-red-app)# group 2`	Configures a redundancy application group and enters redundancy application group configuration mode.
Step 15	**name** *group-name* **Example:** `Device(config-red-app-grp)# name RG2`	Configures a name for the redundancy application group.
Step 16	**priority** *value* [**failover-threshold** *value*] **Example:** `Device(config-red-app-grp)# priority 205 failover-threshold 200`	Specifies a group priority and failover threshold value for the redundancy group.

	Command or Action	Purpose
Step 17	**control** *interface-type interface-number* **protocol** *id* **Example:** `Device(config-red-app-grp)# control gigabitethernet 0/0/1 protocol 2`	Configures a control interface type and number for the redundancy application group.
Step 18	**data** *interface-type interface-number* **Example:** `Device(config-red-app-grp)# data gigabitethernet 0/2/2`	Configures a data interface type and number for the redundancy application group.
Step 19	**end** **Example:** `Device(config-red-app-grp)# end`	Exits redundancy application group configuration mode and enters privileged EXEC mode.

Configuring a Traffic Interface for Stateful NAT64 Interchassis Redundancy

This task applies to a LAN-LAN scenario.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **redundancy rii** *id*
5. **redundancy group** *group-id* **ipv6** *ipv6-prefix/prefix-length* **exclusive decrement** *value*
6. **exit**
7. **interface** *type number*
8. **redundancy rii** *id*
9. **redundancy group** *group-id* **ipv6** *ipv6-prefix/prefix-length* **exclusive decrement** *value*
10. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**interface** *type number* **Example:** `Device(config)# interface gigabitethernet 0/0/0`	Configures an interface and enters interface configuration mode.
Step 4	**redundancy rii** *id* **Example:** `Device(config-if)# redundancy rii 100`	Configures a redundancy interface identifier (RII) for a redundancy group-protected traffic interfaces.
Step 5	**redundancy group** *group-id* **ipv6** *ipv6-prefix*/*prefix-length* **exclusive decrement** *value* **Example:** `Device(config-if)# redundancy group 1 ipv6 2001:DB8:1::1:100/64 exclusive decrement 50`	Enables IPv6 redundancy.
Step 6	**exit** **Example:** `Device(config-if)# exit`	Exits interface configuration mode and enters global configuration mode.
Step 7	**interface** *type number* **Example:** `Device(config)# interface gigabitethernet 0/1/1`	Configures an interface and enters interface configuration mode.
Step 8	**redundancy rii** *id* **Example:** `Device(config-if)# redundancy rii 120`	Configures an RII for a redundancy group-protected traffic interfaces.
Step 9	**redundancy group** *group-id* **ipv6** *ipv6-prefix*/*prefix-length* **exclusive decrement** *value* **Example:** `Device(config-if)# redundancy group 1 ipv6 2001:DB8:2::1:100/64 exclusive decrement 50`	Enables IPv6 redundancy.
Step 10	**end** **Example:** `Device(config-if)# end`	Exits interface configuration mode and enters privileged EXEC mode.

Configuring Static Stateful NAT64 for Interchassis Redundancy

Perform this task to configure a static stateful NAT64 with interchassis redundancy. You can configure interchassis redundancy with the following types of NAT configurations: dynamic, static, and Port Address Translation (PAT) translations.

SUMMARY STEPS

1. **enable**

2. **configure terminal**
3. **ipv6 unicast-routing**
4. **interface** *type number*
5. **ipv6 enable**
6. **ipv6 address** *ipv6-address*/*prefix-length*
7. **nat64 enable**
8. **exit**
9. Repeat Steps 3 to 8 to configure NAT64 on another interface.
10. **nat64 prefix stateful** *ipv6-prefix*/*length*
11. **nat64 v6v4 static** *ipv6-address ipv6-address* [**redundancy** *group-id* **mapping-id** *id*]
12. **nat64 v6v4 tcp** *ipv6-address ipv6-port ipv4-address ipv4-port* [**redundancy** *group-id* **mapping-id** *id*]
13. **end**
14. **show nat64 translations protocol tcp**
15. **show nat64 translations redundancy** *group-id*

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**ipv6 unicast-routing** **Example:** `Device(config)# ipv6 unicast-routing`	Enables the forwarding of IPv6 unicast datagrams.
Step 4	**interface** *type number* **Example:** `Device(config)# interface gigabitethernet 0/0/1`	Configures an interface and enters interface configuration mode.
Step 5	**ipv6 enable** **Example:** `Device(config-if)# ipv6 enable`	Enables IPv6 processing on an interface.
Step 6	**ipv6 address** *ipv6-address*/*prefix-length* **Example:** `Device(config-if)# ipv6 address 2001:DB8:1::1/96`	Configures an IPv6 address based on an IPv6 general prefix and enables IPv6 processing on an interface.

	Command or Action	Purpose
Step 7	**nat64 enable** **Example:** `Device(config-if)# nat64 enable`	Enables NAT64 translation on an IPv6 interface.
Step 8	**exit** **Example:** `Device(config-if)# exit`	Exits interface configuration mode and enters global configuration mode.
Step 9	Repeat Steps 3 to 8 to configure NAT64 on another interface.	—
Step 10	**nat64 prefix stateful** *ipv6-prefix/length* **Example:** `Device(config)# nat64 prefix stateful 2001:DB8:1::1/96`	Defines the stateful NAT64 prefix that is to be added to IPv4 hosts to translate the IPv4 address into an IPv6 address. • The stateful NAT64 prefix can be configured at the global configuration level or at the interface configuration level.
Step 11	**nat64 v6v4 static** *ipv6-address ipv6-address* [**redundancy** *group-id* **mapping-id** *id*] **Example:** `Device(config)# nat64 v6v4 static 2001:DB8:1::FFFE 209.165.201.1 redundancy 1 mapping-id 30`	Enables NAT64 IPv6-to-IPv4 static address mapping and interchassis redundancy.
Step 12	**nat64 v6v4 tcp** *ipv6-address ipv6-port ipv4-address ipv4-port* [**redundancy** *group-id* **mapping-id** *id*] **Example:** `Device(config)# nat64 v6v4 tcp 2001:DB8:1::1 redundancy 1 mapping-id 1`	Applies static mapping to TCP protocol packets and enables interchassis redundancy.
Step 13	**end** **Example:** `Device(config)# end`	Exits global configuration mode and enters privileged EXEC mode.
Step 14	**show nat64 translations protocol tcp** **Example:** `Device# show nat64 translations protocol tcp`	Displays information about NAT 64 protocol translations.
Step 15	**show nat64 translations redundancy** *group-id* **Example:** `Device# show nat64 translations redundancy 1`	Displays information about NAT64 redundancy translations.

Example:

The following is sample output from the **show nat64 translations protocol tcp** command:

```
Device# show nat64 translations protocol tcp

 Proto  Original IPv4         Translated IPv4
        Translated IPv6       Original IPv6
 ------------------------------------------------------------------------------

 tcp    209.165.201.2:21      [2001:DB8:1::103]:32847
        10.2.1.1:80           [2001::11]:80
 tcp    209.165.201.2:21      [2001:DB8:1::104]:32848
        10.2.1.1:80           [2001::11]:80

 Total number of translations: 2
```

The following is sample output from the **show nat64 translations redundancy** command:

```
Device# show nat64 translations redundancy 1

 Proto  Original IPv4         Translated IPv4
        Translated IPv6       Original IPv6
 ------------------------------------------------------------------------------

        209.165.201.2:21      [2001:DB8:1::103]:32847

 tcp    10.2.1.11:32863       [2001::3201:10b]:32863
        10.1.1.1:80           [2001::11]:80
 tcp    209.165.201.2:21      [2001:DB8:1::104]:32848
        10.1.1.1:80           [2001::11]:80

 Total number of translations: 3
```

Configuration Examples for Stateful Network Address Translation 64 Interchassis Redundancy

Example: Configuring Redundancy Group Protocols

```
Device# configure terminal
Device(config)# redundancy
Device(red)# application redundancy
Device(config-red-app)# protocol 1
Device(config-red-app-prtcl)# name RG1
Device(config-red-app-prtcl)# timers hellotime 1 holdtime 3
Device(config-red-app-prtcl)# end
Device# configure terminal
Device(config)# redundancy
Device(red)# application redundancy
Device(config-red-app)# protocol 2
Device(config-red-app-prtcl)# name RG1
Device(config-red-app-prtcl)# end
```

Example: Configuring Redundancy Groups for Active/Standby Load Sharing

The following example shows how to configure redundancy groups (RGs) on two devices for active/standby load sharing:

```
Device1# configure terminal
Device1(config)# redundancy
Device1(config-red)# application redundancy
Device1(config-red-app)# group 1
Device1(config-red-app-grp)# name RG1
Device1(config-red-app-grp)# control gigabitethernet 0/0/1 protocol 1
Device1(config-red-app-grp)# data gigabitethernet 0/2/2
Device1(config-red-app-grp)# end

Device2# configure terminal
Device2(config)# redundancy
Device2(config-red)# application redundancy
Device2(config-red-app)# group 1
Device2(config-red-app-grp)# name RG1
Device2(config-red-app-grp)# control gigabitethernet 0/0/1 protocol 1
Device2(config-red-app-grp)# data gigabitethernet 0/2/2
Device2(config-red-app-grp)# end
```

Example: Configuring Redundancy Groups for Active/Active Load Sharing

The following example shows how to configure two redundancy groups (RGs) on the same device for active/active load sharing:

```
Device1# configure terminal
Device1(config)# redundancy
Device1(config-red)# application redundancy
Device1(config-red-app)# group 1
Device1(config-red-app-grp)# name RG1
Device1(config-red-app-grp)# priority 195 failover-threshold 190
Device1(config-red-app-grp)# control gigabitethernet 0/0/1 protocol 1
Device1(config-red-app-grp)# data gigabitethernet 0/2/2
Device1(config-red-app-grp)# end
Device1# configure terminal
Device1(config)# redundancy
Device1(config-red)# application redundancy
Device1(config-red-app)# group 2
Device1(config-red-app-grp)# name RG2
Device1(config-red-app-grp)# priority 205 failover-threshold 200
Device1(config-red-app-grp)# control gigabitethernet 0/0/1 protocol 2
Device1(config-red-app-grp)# data gigabitethernet 0/2/2
Device1(config-red-app-grp)# end

Device2# configure terminal
Device2(config)# redundancy
Device2(config-red)# application redundancy
Device2(config-red-app)# group 1
Device2(config-red-app-grp)# name RG1
Device2(config-red-app-grp)# priority 195 failover-threshold 190
Device2(config-red-app-grp)# control gigabitethernet 0/0/1 protocol 1
Device2(config-red-app-grp)# data gigabitethernet 0/2/2
Device2(config-red-app-grp)# end
Device2# configure terminal
Device2(config)# redundancy
Device2(config-red)# application redundancy
Device2(config-red-app)# group 2
Device2(config-red-app-grp)# name RG2
Device2(config-red-app-grp)# priority 205 failover-threshold 200
Device2(config-red-app-grp)# control gigabitethernet 0/0/1 protocol 2
```

```
Device2(config-red-app-grp)# data gigabitethernet 0/2/2
Device2(config-red-app-grp)# end
```

Example: Configuring a Traffic Interface for Stateful NAT64 Interchassis Redundancy

```
Device# configure terminal
Device(config)# interface gigabitethernet 0/0/0
Device(config-if)# redundancy rii 100
Device(config-if)# redundancy group 1 ipv6 2001:DB8:1::1:100/64 exclusive decrement 50
Device(config-if)# exit
Device(config)# interface gigabitethernet 0/1/1
Device(config-if)# redundancy rii 100
Device(config-if)# redundancy group 1 ipv6 2001:DB8::2:1:100/64 exclusive decrement 50
Device(config-if)# end
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Master Command List, All Releases
NAT commands	IP Addressing Services Command Reference

Standards/RFCs

Standard/RFC	Title
RFC 4787	*Network Address Translation (NAT) Behavioral Requirements for Unicast UDP*

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html