High-Availability Cluster Communications

The Tail-f HCC package implements High-Availability Framework (HAFW) and extends the built-in HA functionality by providing virtual IP addresses (VIPs) that can be used to connect to the NSO HA group primary node. HCC ensures that the VIP addresses are always bound by the HA group primary and never bound by a secondary. Each time a node transitions between primary and secondary states HCC reacts by binding (primary) or unbinding (secondary) the VIP addresses.

HCC manages IP addresses at the link-layer (OSI layer 2) for Ethernet interfaces, and, optionally, also at network-layer (OSI layer 3) using BGP router advertisements. The layer 2 and layer 3 functions are mostly independent, and this document describes the details of each one separately. However, the layer-3 function builds on top of the layer-2 function. The layer-2 function is always necessary. Otherwise, the Linux kernel on the primary node would not recognize the VIP address or accept traffic directed to it.

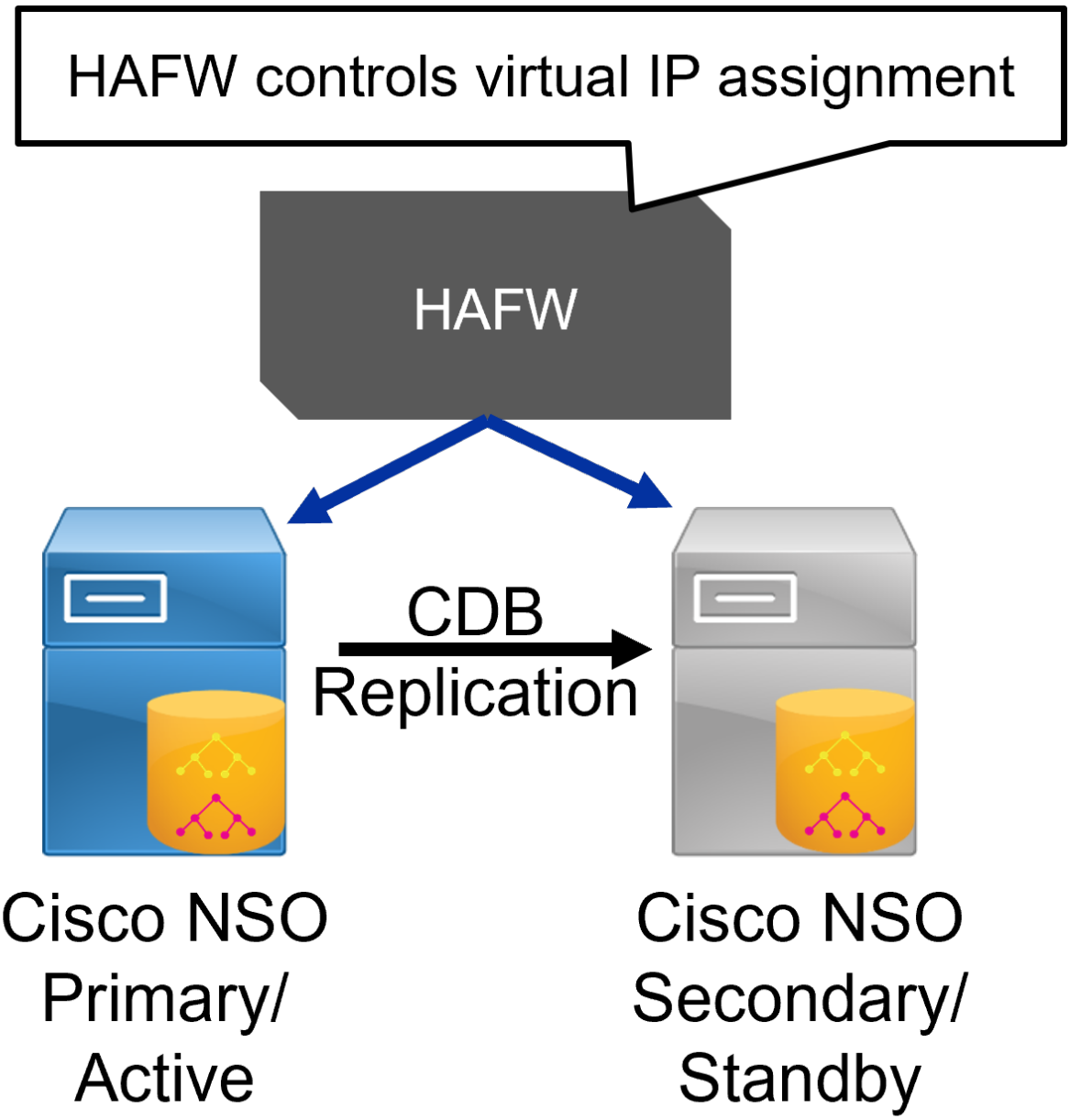
Both the HCC layer-2 VIP and layer-3 BGP functionality depend on iproute2 utilities and awk. An optional dependency is arping (either from iputils or Thomas Habets arping implementation), which allows HCC to announce the VIP to MAC mapping to all nodes in the network by sending gratuitous Address Resolution Protocol (ARP) requests.

The HCC layer 3 Border Gateway Protocol (BGP) functionality depends on the GoBGP daemon version 2.x being installed on each NSO host that is configured to run HCC in BGP mode. GoBGP is open-source software originally developed by NTT Communications and released under Apache License 2.0. GoBGP can be obtained directly from https://osrg.github.io/gobgp/ and is also packaged for mainstream Linux distributions.

Tailf-f HCC has two distinct scenarios for setting up HA:

- **Layer 2**: The purpose of the HCC layer 2 functionality is to ensure that the configured VIP addresses are bound in the Linux kernel of the NSO primary node only. This ensures that the primary node (and only the primary node) will accept traffic directed toward the VIP addresses.
- **Layer 3 BGP**: The purpose of the HCC layer 3 BGP functionality is to operate a BGP daemon on each NSO node and to ensure that routes for the VIP addresses are advertised by the BGP daemon on the primary node only.

The following figure illustrates how Tail-f HCC HAFW controls the virtual IP assignment between primary and secondary NSO nodes.



NSO Operating System Requirements

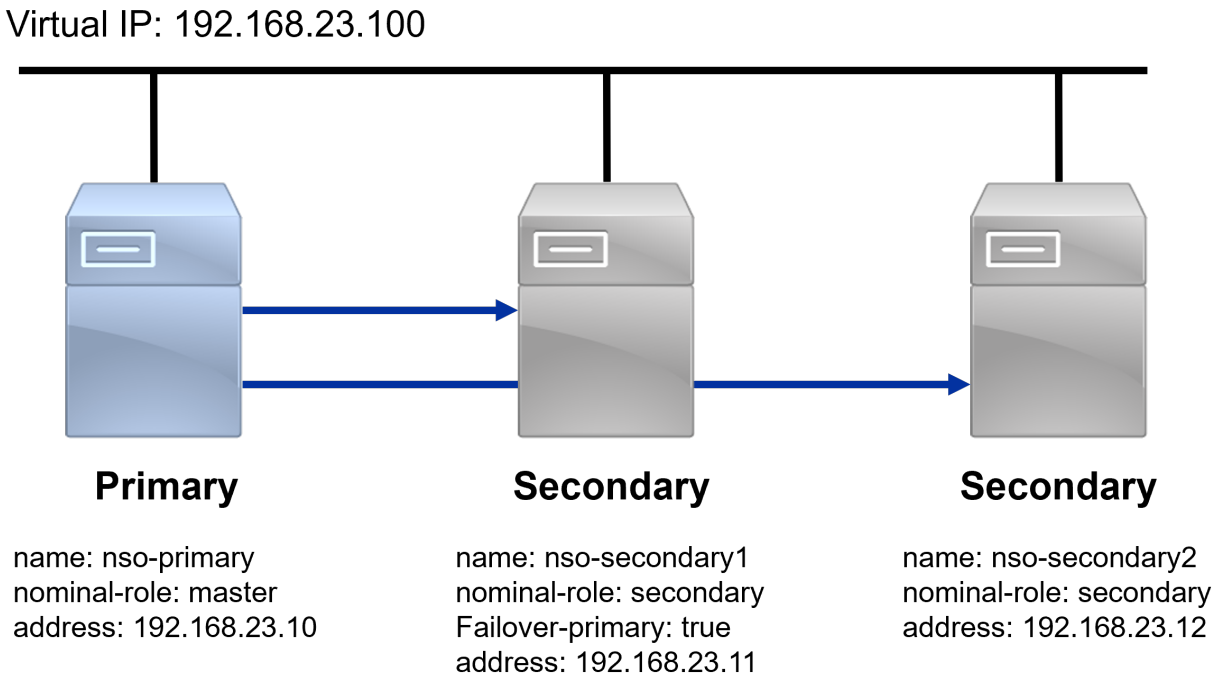
To enable HA, these NSO operating system requirements must be met:

- On all nodes, the operating system must be configured to allow for NSO high-availability operations (configure the firewall to allow the ports used by NSO for high availability).
- NSO high availability must be enabled in the ncs.conf configuration file.
- Encryption keys in ncs.conf must be the same on the primary and the secondary nodes.
- All secondary nodes must be reachable by the primary.

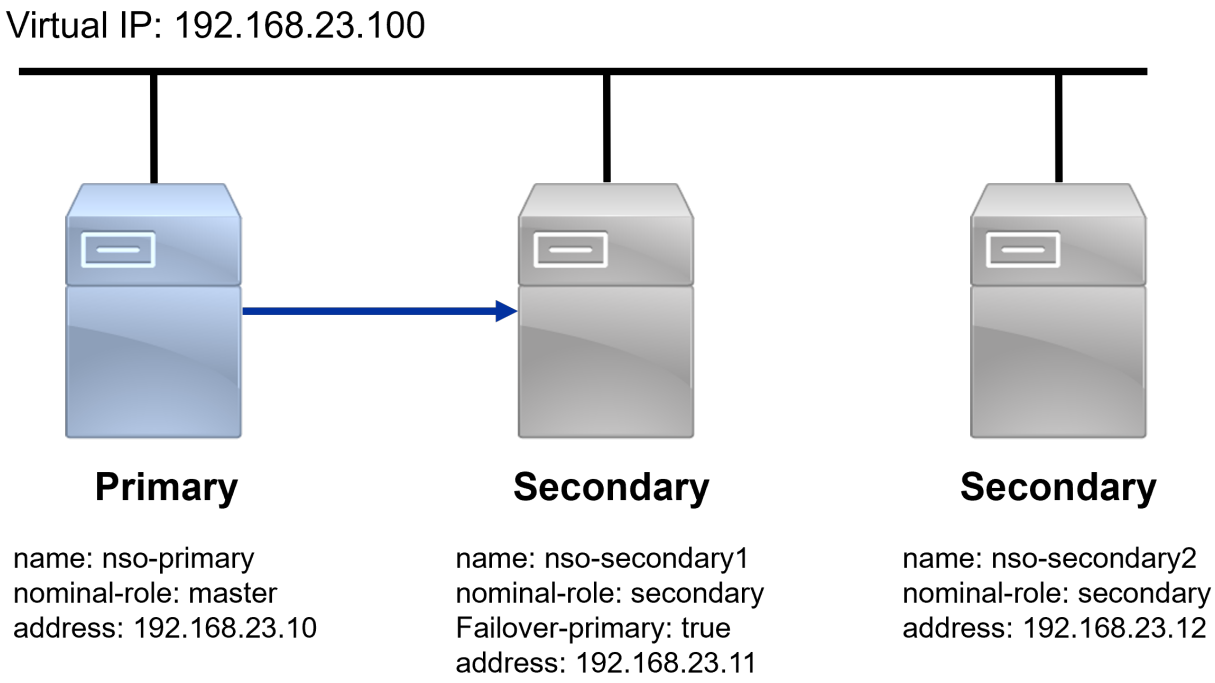
HCC Layer 2: Configuration Example

An HCC Layer 2 deployment would require you to configure virtual IP address management. Each NSO node will have to be configured with a VIP: if the node is

currently the primary node, it will bring up the virtual IP with the configured address on the interface that has the same private IP address of the node as configured in the settings (ha-node/address). The primary node has the interface with the virtual IP enabled, and all the traffic will arrive at the primary node, while the NSO HA will make sure that the data is properly replicated among secondaries.



A single node can be configured as the primary node (**nominal-role primary**), and other nodes are secondary nodes (**nominal-role secondary**). Also, only a single node can be configured with the failover-primary setting set to true, as shown in the following figure.



HCC-Basic: NSO Node Configuration

The next figure shows an output from the HA configuration that is configured on all NSO nodes.

```
admin@ncs# show running-config high-availability |
display curly-braces
token
$9$xoT+7ELWuNGrcmqz+csKFLzOk69xQiKh5Mk5gK6Hbbg=;
ha-node nso-secondary1 {
    address         192.168.23.11;
    nominal-role    secondary;
    failover-primary true;
}
ha-node nso-secondary2 {
    address      192.168.23.12;
    nominal-role primary;
}
ha-node nso-primary {
    address      192.168.23.10;
    nominal-role secondary;
}
…
```

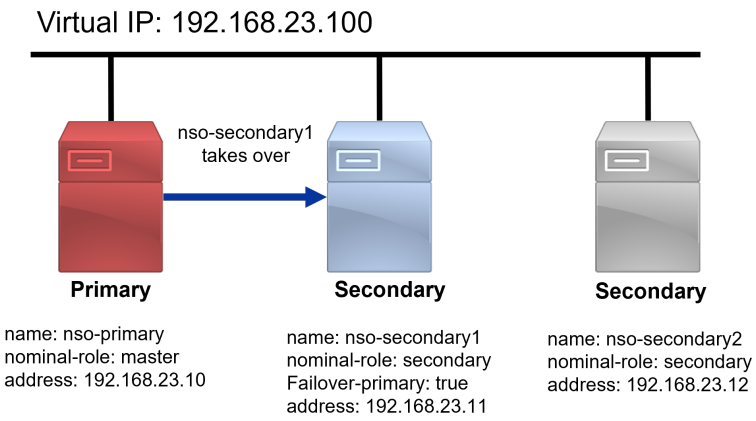
```
…
settings {
    enable-failover    true;
    start-up {
        assume-nominal-role true;
    }
    reconnect-interval 4;
    reconnect-attempts 10;
    reconnect-secondaries;
    consensus {
        enabled;
    }
}
admin@ncs# show running-config hcc
enabled
vip 192.168.23.100
```

A token is a shared secret, which is a mandatory property when setting HA between nodes. The *reconnect-interval* is a setting that sets the seconds on which the HA state is checked. The *reconnect-attempts* setting is the number of failed HA state checks before declaring a failure. The HA configuration in this example is configured with a virtual IP address, which will be used when bringing up a labeled interface on the NSO primary node. A list of members is provided, and each is configured with the management IP of the NSO node, default HA role (nominal-role), and a failover-primary setting. In case of a failure, a split-brain scenario may occur. A split-brain scenario is the result of two data nodes losing the ability to sync while potentially continuing to both act as a primary node. This may lead to data inconsistencies where each node has data that does not exist on the other side. To prevent split-brain to occur, NSO 5.7 or later comes with a rule-based algorithm. The algorithm is enabled by default, it can be disabled or changed from the *consensus* setting. To automatically enable HA upon start-up of the node, the assume-nominal-role setting must be enabled. HA Members assigned as secondaries, but neither failover-primaries nor set with the nominal-role primary may attempt to rejoin the HA group after losing connection to the primary. This is controlled with the *reconnect-secondaries* setting.

HCC Layer 2: HA Failover

To deactivate HA on a node, or to simulate that the primary node, is down use the following action command on that node.

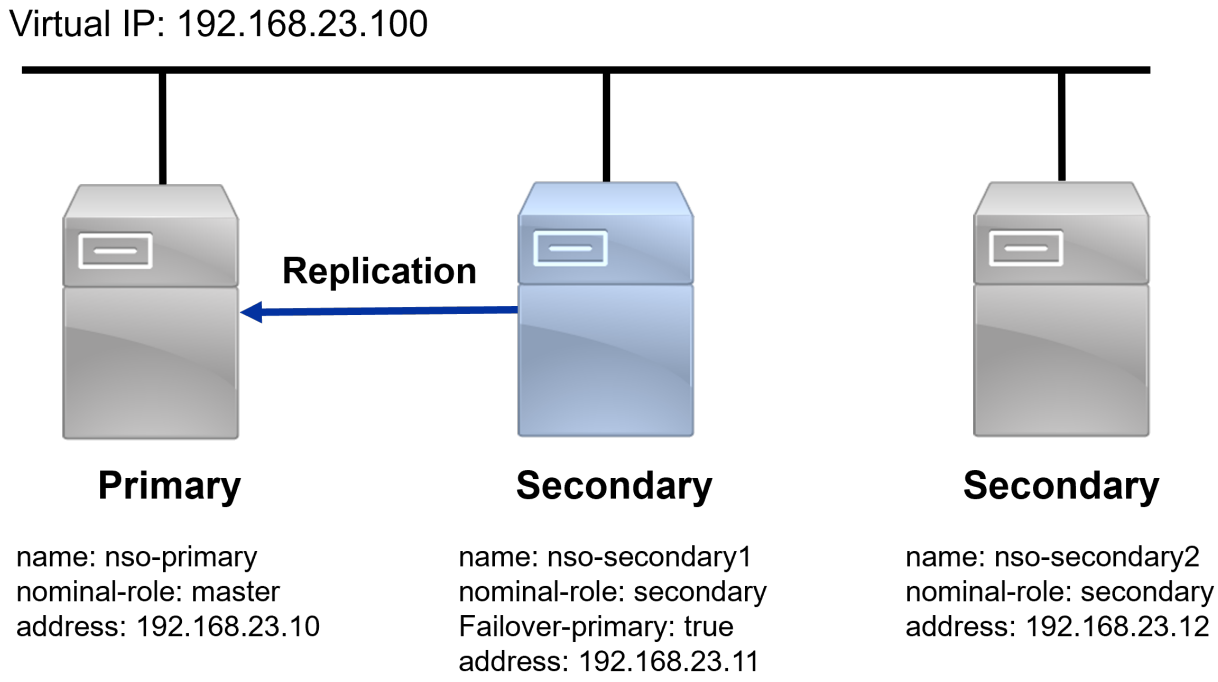
```
admin@ncs# high-availability disable
result NSO Built-in HA disabled
```



Failover will take (interval*failure limit) seconds before the secondary node that is configured as **failover-primary** confirms the loss of the primary node and initiates a failover. The secondary node configured as **failover-primary**, will take **ha-role primary** and become the primary node, as shown in the following figure.

HCC Layer 2: HA Failback

The failover happens automatically. Once the failed primary node has recovered, the node will search for a node with a primary role. If found, it will attempt to connect as secondary to that node.

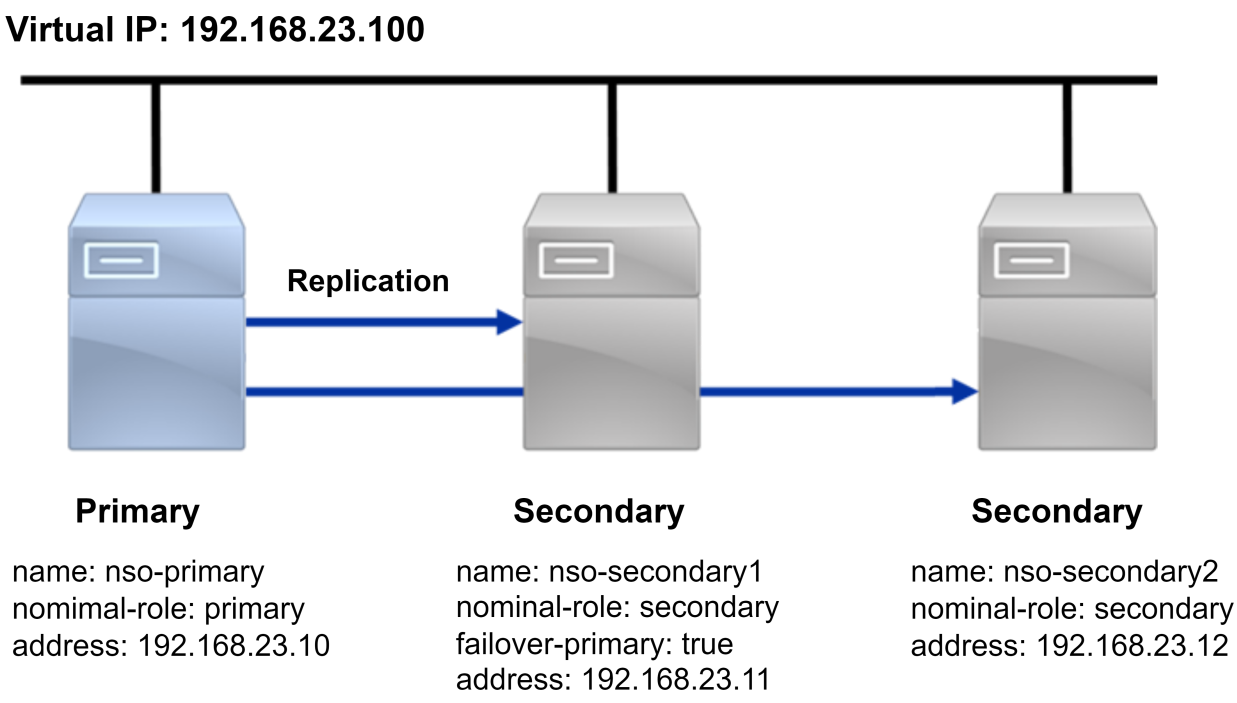


You can revert roles manually once you decide that the currently secondary node can take the primary role again. To restore the nominal roles in the HA cluster, you may need to manually invoke the /high-availability/be-secondary-to action.

```
admin@master# ha commands role-override role slave
override
admin@master# ha commands activate
activated
```

```
admin@ncs# high-availability be-secondary-to node nso-primary
```

As shown in the next figure, the previous primary will now reference the new primary node and receive the replicated CDB data.



HCC Layer 2: HA Failback (Role Revert)

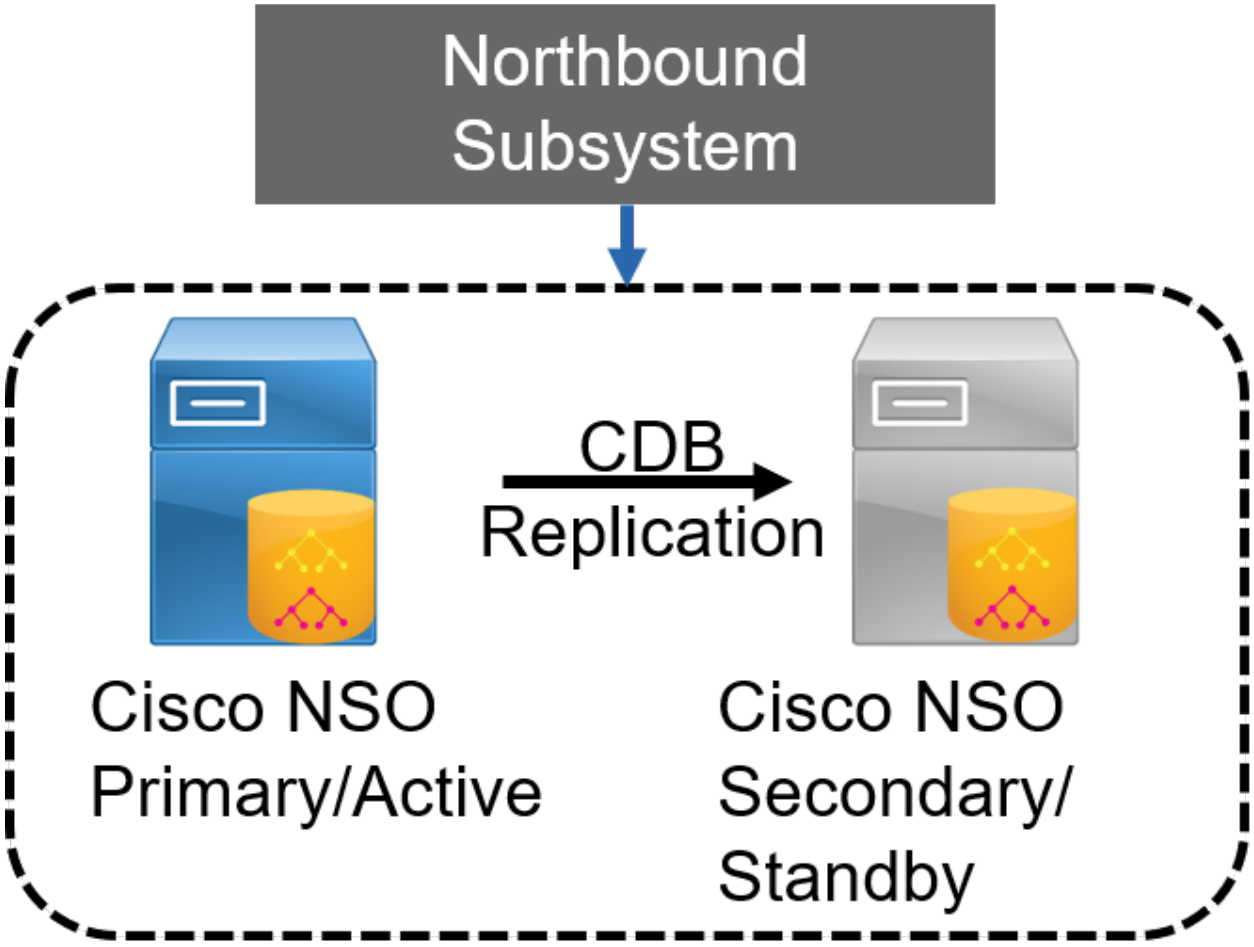
Northbound System Seamless Failover

To ensure seamless failover, simply implementing a switch between primary or secondary and a sync of CDB is not enough. You need to ensure a seamless switch from a northbound systems perspective.

You can use one of the following options:

- Layer 2 virtual IP
- Layer 3 BGP Anycast
- Load balancer

From the perspective of the Northbound systems integrated with NSO, a failure should not change how they connect to NSO, as illustrated in the following figure.



The example shows how a virtual IP can be configured so that your northbound systems can use a single IP address to access Cisco NSO even if behind that IP multiple failover scenarios happen.

High-availability Alarms

When any node loses connection, this can also be observed in high-availability alarms as either a ha-primary-down or a ha-secondary-down alarm.

```
alarms alarm-list alarm ncs ha-primary-down /high-availability/ha-
node[id='nso-primary']
 is-cleared              false
 last-status-change      2022-05-30T10:02:45.706947+00:00
 last-perceived-severity critical
 last-alarm-text         "Lost connection to primary due to: Primary
closed connection"
 status-change 2022-05-30T10:02:45.706947+00:00
  received-time      2022-05-30T10:02:45.706947+00:00
  perceived-severity critical
  alarm-text         "Lost connection to primary due to: Primary closed
connection"

alarms alarm-list alarm ncs ha-secondary-down /high-availability/ha-
node[id='nso-secondary1'] ""
```

```
is-cleared              false
last-status-change      2022-05-30T10:04:33.231808+00:00
last-perceived-severity critical
last-alarm-text         "Lost connection to secondary"
status-change 2022-05-30T10:04:33.231808+00:00
 received-time      2022-05-30T10:04:33.231808+00:00
 perceived-severity critical
 alarm-text         "Lost connection to secondary"
```