Installing VTS in High Availability Mode

This chapter provides detailed information about installing VTS in high availability (HA) mode. It details the procedure to enable VTS L2 and VTS L3.

See Enabling VTS L2 High Availability, on page 1for the detailed procedure to enable VTS L2 HA.

See Enabling VTS L3 High Availability for for Underlay and Management Networksfor the detailed procedure to enable VTS L3 HA.

Important Notes regarding updating the cluster.conf file:

- master_name and slave_name can not be the same
- master_network_interface and slave_network_interface are interface names of VTC1 and VTC2 where the real IP resides. They should be the same.
- If you are using VTF's, fill in vip_private and private_network_interface fields. Otherwise, leave these two fields blank.
- Private_network_interface is the secondary interface names of VTC1 and VTC2 on the private network that VTF is also on.
- vip_private is the vip for the VTS master's private interface.
- private_gateway is the gateway for the private network of your vip_private.

This chapter has the following sections.

- Enabling VTS L2 High Availability, on page 1
- Enabling VTS L3 High Availability for for Underlay and Management Networks, on page 5
- Enabling VTS L3 High Availablity, on page 11
- Enabling VTS L3 High Availability for Management Network only, on page 12
- Switching Over Between Master and Slave Nodes, on page 27
- Uninstalling VTC High Availability, on page 29
- Troubleshooting Password Change Issues, on page 30
- Installing VTSR in High Availability Mode, on page 30
- Hight Availability Scenarios, on page 32

Enabling VTS L2 High Availability

To enable VTC L2 HA, VTC1 and VTC2 must be on the same subnet.

Spawn two VTC VMs. At a minimum, you would need to have 3 IP addresses for VTC. One for VTC1, One for VTC2, one for the public Virtual IP (VIP). If you are using VTFs, you will also need one for the private VIP, which other devices on the private network such as the VTF can reach.

Note Cisco VTS supports dual stack clusters for L2 HA. Have both the VTCs (vts01 and vts02) installed and configured with IPv6 & IPv4 address for dual stack to be supported. Both of the VTCs should be reachable by any means with IPv6 address or IPv4 address.

Note Before enabling HA, make sure that both VTC 1 and VTC 2 have the same password. If not, go to the VTC GUI and do a change password on newly brought up VTC, to make the password identical with that of the other VTC . When you upgrade a VTC / bring up a new VTC / do a hardware upgrade of VTC host, you should make sure that password is the same.

Enabling VTS L2 HA involves:

- Setting up the VTC Environment, on page 2
- Enabling VTC High Availability, on page 3
- Registering vCenter to VTC, on page 4
- Enabling VTSR High Availability, on page 5

Setting up the VTC Environment

You need to set up the VTC environment before you run the high availability script.

Step 1 Create a copy of cluster.conf file from cluster.conf.tmpl, which is under the /opt/vts/etc directory. For example:

```
admin@vts01:~$ cd /opt/vts/etc

admin@vts01:~$ sudo copy cluster.conf.tmpl cluster.conf
```

Step 2 Specify the VIP address and the details of the two nodes in cluster.conf file . For example:

```
admin@vts01:/var/# cd /opt/vts/etc/

admin@vts01/etc# sudo vi cluster.conf



###Virtual Ip of VTC Master on the public interface. Must fill in at least 1
vip_public=172.23.92.202
vip_public_ipv6=2001:420:10e:2015:c00::202



###VTC1 Information. Must fill in at least 1 ip address
master_name=vts01
master_ip=172.23.92.200
master_ipv6=2001:420:10e:2015:c00::200
```

```
###VTC2 Information. Must fill in at least 1 ip address
slave_name=vts02
slave_ip=172.23.92.201
slave_ipv6=2001:420:10e:2015:c00:201


###In the event that a network failure occurs evenly between the two routers, the cluster needs an
outside ip to determine where the failure lies
###This can be any external ip such as your vmm ip or a dns but it is recommended to be a stable ip
 within your environment
###Must fill in at least 1 ip address
external_ip=171.70.168.183
external_ipv6=2001:420:200:1::a
```

Note The two nodes communicate each other using VIP IP, and user can use VIP address to login to Cisco VTS UI. You will be directly logged in to the master node, when you use VIP IP address. Make sure that you specify the correct host name, IP Address, and interface type.

Enabling VTC High Availability

You must run the cluster_install.sh script on both VTCs to enable high availability.

Step 1 Run the cluster installer script /opt/vts/bin/cluster_install.sh on both VTC1 and VTC2 . For example:

```
admin@vts02:/opt/vts/etc$ sudo su  -

[sudo] password for admin:

root@vts02:/opt/vts/etc$ cd ../bin

root@vts02:/opt/vts/bin# ./cluster_install.sh
172.23.92.200 vts01
172.23.92.201 vts02
2001:420:10e:2015:c00::200 vts01
2001:420:10e:2015:c00::201 vts02

Change made to ncs.conf file. Need to restart ncs

Created symlink from /etc/systemd/system/multi-user.target.wants/pacemaker.service to
/lib/systemd/system/pacemaker.service.

Created symlink from /etc/systemd/system/multi-user.target.wants/corosync.service to
/lib/systemd/system/corosync.service.

Both nodes are online. Configuring master

Configuring Pacemaker resources

Master node configuration finished

HA cluster is installed
```

Step 2 Check the status on both the nodes to verify whether both nodes online, and node which got installed first is the master, and the other, slave. For example:

```
admin@vts02:/opt/vts/log/nso$ sudo crm status

[sudo] password for admin:

Last updated: Mon Apr 10 18:43:52 2017          Last change: Mon Apr 10 17:15:21 2017 by root via
crm_attribute on vts01

Stack: corosync

Current DC: vts01 (version 1.1.14-70404b0) - partition with quorum

2 nodes and 4 resources configured


Online: [ vts01 vts02 ]


Full list of resources:


Master/Slave Set: ms_vtc_ha [vtc_ha]

     Masters: [ vts02 ]

     Slaves: [ vts01 ]

ClusterIP      (ocf::heartbeat:IPaddr2):       Started vts02

ClusterIPV6    (ocf::heartbeat:IPaddr2):       Started vts02
```

Registering vCenter to VTC

To do this:

Step 1 Log in to VCSA.

Step 2 Go to **Networking > Distributed Virtual Switch > Manage > VTS**.

Note For vCenter 6.5, the VTS comes under Configure tab.

Step 3 Click on System Configuration

Step 4 Enter the following:

- VTS IP—This is the Virtual public IP address.
- VTS GUI Username
- VTS GUI Password

Step 5 Click Update.

Enabling VTSR High Availability

You need to enable VTSR HA, if you have VTFs in your setup. For information about enabling VTSR HA, see Installing VTSR in High Availability Mode, on page 30.

Enabling VTS L3 High Availability for for Underlay and Management Networks

This section describes the procedure to enable VTC L3 HA. In L3 HA environments, VTCs are located on separate overlay networks.

Before enabling VTC L3 HA, make sure that the following requirements are met:

Step 1 Both VTC 1 and VTC 2 must have the same password. If they are not the same, change the password via the VTC GUI to make it identical with that of the other VTC. When you upgrade a VTC, bring up a new VTC, or do a hardware upgrade of a VTC host, remember to make sure that the passwords match.

Step 2 VTC1 and VTC2 are deployed in two different networks VTC1 is able to ping VTC 2 and wise versa via both underlay and overlay networks.

We recommend that you add static routes in /etc/network/interfaces on the VTCs. For example:

```
post-up route add -net 44.44.44.0/24 gw 10.10.10.1 dev eth1
post-up route add -net 10.10.10.0/24 gw 44.44.44.1 dev eth1
```

Step 3 VTSR installation is mandatory for VTC L3 HA to work. Although the VTSR(s) will not be able to register to the master VTC's Virtual IP Address (VIP) initially, the master VTC must still be able to reach all VTSRs in order to configure them to support L3HA.

Step 4 A site-id is required and need to be updated in VTSR_template file before generating the VTSR iso file and is generated by either:

1. Creating a site directly from the VTC1 GUI.
2. Using the uuidgen linux command to generate a 32 character site id.

In both cases, note down the site id for use in the below section "Deploying VTSR VMs". In the second case the site id will also be required to update the VTC master site configuration and associate the UUID to the site after the HA process is complete.

The implementation of L3 HA is done for both management and underlay networks. For VIP ip reachability across networks the master VTC will configure BGP and route policies between the VTSRs and their respective directly connected TORs based on the user network settings provided in cluster.conf. Routing policy including tagging helps to migrate VIP IPs across the VTSRs during VTC HA.

Setting up the VTC Environment

You need to set up the VTC environment before you run the high availability script.

Step 1 Modify the /opt/vts/etc/cluster.conf_tmpl file on both the VTCs. And rename to cluster.conf .A sample modified file is given below:

Both the VTCs must have the identical information in the cluster.conf file.

```
###Virtual Ip of VTC Master on the public interface. Must fill in at least 1
vip_public= 192.168.10.254
vip_public_ipv6=
###VTC1 Information. Must fill in at least 1 ip address
master_name=Onion-VTC1
master_ip= 60.60.60.10
master_ipv6=
###VTC2 Information. Must fill in at least 1 ip address
slave_name= Onion-VTC2
slave_ip= 70.70.70.10
slave_ipv6=
###In the event that a network failure occurs evenly between the two routers, the cluster needs an
outside ip to determine where the failure lies
###This can be any external ip such as your vmm ip or a dns but it is recommended to be a stable ip
 within your environment
###Must fill in at least 1 ip address
external_ip= 81.81.81.1
external_ipv6=
###If you intend to use a virtual topology forwarder (VTF) in your environment, please fill in the
vip for the underlay as well as the underlay gateway. Otherwise leave blank.
###Virtual Ip of VTC Master on the private interface. You can fill in ipv4 configuration, ipv6, or
both if you use both
vip_private= 45.45.45.10
private_gateway=10.10.10.1
vip_private_ipv6=
private_gateway_ipv6=
###If you have your vtc's in different subnets, vtsr will need to be configured to route traffic and
 the below section needs to be filled in
###If you have your vtc's on the same subnet, the below section should be skipped
###Name of your vrf. Example: VTS_VIP
vrf_name= mgmt-vrf
###Ip of your first Vtsr. Example: 11.1.1.5
vtsr1_mgmt_ip=60.60.60.15
vtsr1_mgmt_ipv6=
###List of neighbors for vtsr1, separated by comma. Example: 11.1.1.1,11.1.1.2
vtsr1_bgp_neighbors=  11.11.11.11
vtsr1_bgp_neighbors_ipv6=
###Ip of your second Vtsr. Example: 12.1.1.5
vtsr2_mgmt_ip= 70.70.70.15
vtsr2_mgmt_ipv6=
###List of neighbors for vtsr2, separated by comma. Example: 12.1.1.1,12.1.1.2
vtsr2_bgp_neighbors= 12.12.12.12
vtsr2_bgp_neighbors_ipv6=
###Username for Vtsr
vtsr_user= admin
###Vtsr ASN information
remote_ASN= 6500
local_ASN= 6501
###Vtsr BGP information
bgp_keepalive= 10
bgp_hold= 30
###Update source for Vtsr1 (i.e. loopback)
vtsr1_update_source=loopback0
###Update source for Vtsr2 (i.e. loopback)
vtsr2_update_source=loopback0
###Router BGP Id for Vtsr1
vtsr1_router_id= 21.21.21.21
```

```
###Router BGP Id for Vtsr2
vtsr2_router_id= 31.31.31.31
###Ipv4 Route Distinguisher Loopback for IPv6 Vtsr1 (if VTSR has a defined IPv6 management address,
 an ipv4 loopback address will be needed for the route distinguisher)
vtsr1_rd_loopback_name=
vtsr1_rd_loopback_address=
###Ipv4 Route Distinguisher Loopback for IPv6 Vtsr2
vtsr2_rd_loopback_name=
vtsr2_rd_loopback_address=

###XRVR1 name
vtsr1_name= vtsr01
###XRVR2 name
vtsr2_name= vtsr02
###If you plan on having your VTC's on different subnets and intend to use a virtual topology forwarder
 (VTF) in your environment,
### please fill out the following fields. Otherwise, leave blank
###List of neighbors for vtsr1, separated by comma. Example: 2.2.2.2,2.2.2.3
vtsr1_underlay_neighbors= 2.2.2.2
vtsr1_underlay_neighbors_ipv6=
###List of neighbors for vtsr2, separated by comma. Example: 3.3.3.2,3.3.3.3
vtsr2_underlay_neighbors= 6.6.6.6,9.9.9.9
vtsr2_underlay_neighbors_ipv6=
###OSPF Parameters
ospf_id_v4=100
ospf_id_v6=
area=0.0.0.0
default_cost=10
###ISIS Parameters
isis_id=
is_type=
lsp_mtu=
key_chain_id=
key_id=
cryptographic_algorithm=
#Network name (consist of an even number of octets and be of the form 01.2345.6789.abcd.ef)
vtsr1_network_entity=
vtsr2_network_entity=

###Directly connected Tor information for Vtsr1
vtsr1_directly_connected_device_ip= 10.10.50.3
vtsr1_directly_connected_device_ipv6=
vtsr1_directly_connected_device_user= admin
vtsr1_directly_connected_device_neighbors= 21.21.21.21
vtsr1_directly_connected_device_neighbors_ipv6=
vtsr1_directly_connected_ospf=
vtsr1_directly_connected_router_id=2.2.2.2
vtsr1_directly_connected_update_source=loopback0
###Directly connected Tor information for Vtsr2
vtsr2_directly_connected_device_ip= 10.10.50.7
vtsr2_directly_connected_device_ipv6=
vtsr2_directly_connected_device_user=admin
vtsr2_directly_connected_device_neighbors=31.31.31.31
vtsr2_directly_connected_device_neighbors_ipv6=
vtsr2_directly_connected_ospf=
vtsr2_directly_connected_router_id=6.6.6.6
vtsr2_directly_connected_update_source=loopback0
###VPC Peer information if any. Otherwise leave blank
vtsr1_vpc_peer_ip=
vtsr1_vpc_peer_ipv6=
vtsr1_vpc_peer_user=
vtsr1_vpc_peer_ospf=
vtsr1_vpc_peer_router_id=
vtsr1_vpc_peer_update_source=
vtsr2_vpc_peer_ip=
```

```
vtsr2_vpc_peer_ipv6=
vtsr2_vpc_peer_user=
vtsr2_vpc_peer_ospf=
vtsr2_vpc_peer_router_id=
vtsr2_vpc_peer_update_source=
###VTC Underlay Addresses
vtc1_underlay= 10.10.10.10
vtc2_underlay= 44.44.44.44
vtc1_underlay_ipv6=
vtc2_underlay_ipv6=
##Gateway of secondary L3 underlay
vtc2_private_gateway=44.44.44.1
#vtc2_private_gateway_ipv6=
```

Step 2 Make sure VTSR configuration ISOs are up to date with above configurations. For example:

Note # The VTS_REGISTRATION_PASSWORD and VTS_SITE_UUID values are VTC UI password and SITE ID those are created on VTC above respectively

This is a sample VTSR configuration file

Copyright (c) 2015 cisco Systems

Please protect the generated ISO, as it contains authentication data

in plain text.

```
# VTS Registration Information:
#  VTS_ADDRESS should be the IP for VTS. The value must be either an ip or a mask.
#  VTS_ADDRESS is mandatory. If only the V4 version is specified,
#  The V4 management interface for the VTSR (NODE1_MGMT_NETWORK_IP_ADDRESS)
#  will be used. If the V6 version is specified, the V6 management interface
#  for the VTSR (NODE1_MGMT_NETWORK_IPV6_ADDRESS) must be specified and will be used.
VTS_ADDRESS="192.168.10.254"
VTS_IPV6_ADDRESS=
# VTS_REGISTRATION_USERNAME used to login to VTS.
VTS_REGISTRATION_USERNAME="admin"
# VTS_REGISTRATION_PASSWORD is in plaintext.
VTS_REGISTRATION_PASSWORD="Cisco123!"
# VTSR VM Admin user/password
USERNAME="admin"
PASSWORD="cisco123"

# Mandatory Management-VRF name for VTSR.
VTS_MANAGEMENT_VRF="mgmt-vrf"

# VTSR VM Network Configuration for Node 1:
#  NETWORK_IP_ADDRESS, NETWORK_IP_NETMASK, and NETWORK_IP_GATEWAY
#     are required to complete the setup. Netmask can be in the form of
#     "24" or "255.255.255.0"
#  The first network interface configured with the VTC VM will be used for
#  underlay connectivity; the second will be used for the management network.
#  For both the MGMT and UNDERLAY networks, a <net-name>_NETWORK_IP_GATEWAY
#  variable is mandatory; they are used for monitoring purposes.
#
#  V6 is only supported on the mgmt network and dual stack is
#  currently not supported, so if both are specified V6 will take priority (and
#  requires VTS_IPV6_ADDRESS to be set).
#  The *V6* parameters for the mgmt network are optional. Note that if V6 is used for mgmt
#  it must be V6 on both nodes. Netmask must be the prefix length for V6.
NODE1_MGMT_NETWORK_IP_ADDRESS="60.60.60.15"
NODE1_MGMT_NETWORK_IP_NETMASK="255.255.255.0"
NODE1_MGMT_NETWORK_IP_GATEWAY="60.60.60.1"
NODE1_MGMT_NETWORK_IPV6_ADDRESS=
NODE1_MGMT_NETWORK_IPV6_NETMASK=
```

```
NODE1_MGMT_NETWORK_IPV6_GATEWAY=
NODE1_UNDERLAY_NETWORK_IP_ADDRESS="10.10.10.33"
NODE1_UNDERLAY_NETWORK_IP_NETMASK="255.255.255.0"
NODE1_UNDERLAY_NETWORK_IP_GATEWAY="10.10.10.1"
# AUX network is optional
#NODE1_AUX_NETWORK_IP_ADDRESS="169.254.20.100"
#NODE1_AUX_NETWORK_IP_NETMASK="255.255.255.0"
#NODE1_AUX_NETWORK_IP_GATEWAY="169.254.20.1"
# XR Hostname
NODE1_XR_HOSTNAME="vtsr01"
# Loopback IP and netmask
NODE1_LOOPBACK_IP_ADDRESS="128.0.0.10"
NODE1_LOOPBACK_IP_NETMASK="255.255.255.255"

# Operational username and password - optional
# These need to be configured to start monit on VTSR

VTSR_OPER_USERNAME="admin"
# Password needs an encrypted value
# Example : "openssl passwd -1 -salt <salt-string> <password>"
VTSR_OPER_PASSWORD="$1$cisco$Qv2TLtPNI3jqwXMOA3M3f0/"

# VTSR monit interval - optional - default is 30 seconds
VTSR_MONIT_INTERVAL="30"

# VTSR VM Network Configuration for Node 2:
#  If there is no HA then the following Node 2 configurations will remain commented and
#  will not be used and Node 1 configurations alone will be applied
#  For HA , the following Node 2 configurations has to be uncommented
#  VTSR VM Network Configuration for Node 2
#  NETWORK_IP_ADDRESS, NETWORK_IP_NETMASK, and NETWORK_IP_GATEWAY
#     are required to complete the setup. Netmask can be in the form of
#     "24" or "255.255.255.0"
#  The first network interface configured with the VTC VM will be used for
#  underlay connectivity; the second will be used for the management network.
#  For both the MGMT and UNDERLAY networks, a <net-name>_NETWORK_IP_GATEWAY
#  variable is mandatory; they are used for monitoring purposes.
#
#  V6 is only supported on the mgmt network and dual stack is
#  currently not supported, so if both are specified V6 will take priority (and
#  requires VTS_IPV6_ADDRESS to be set).
#  The *V6* parameters for the mgmt network are optional. Note that if V6 is used for mgmt
#  it must be V6 on both nodes. Netmask must be the prefix length for V6.
NODE2_MGMT_NETWORK_IP_ADDRESS="70.70.70.15"
NODE2_MGMT_NETWORK_IP_NETMASK="255.255.255.0"
NODE2_MGMT_NETWORK_IP_GATEWAY="70.70.70.1"
NODE2_MGMT_NETWORK_IPV6_ADDRESS=
NODE2_MGMT_NETWORK_IPV6_NETMASK=
NODE2_MGMT_NETWORK_IPV6_GATEWAY=
NODE2_UNDERLAY_NETWORK_IP_ADDRESS="44.44.44.15"
NODE2_UNDERLAY_NETWORK_IP_NETMASK="255.255.255.0"
NODE2_UNDERLAY_NETWORK_IP_GATEWAY="44.44.44.1"
#  AUX network is optional
#  Although Aux network is optional it should be either present in both nodes
#  or not present in both nodes.
#  It cannot be present on Node1 and not present on Node2 and vice versa
#NODE2_AUX_NETWORK_IP_ADDRESS="179.254.20.200"
#NODE2_AUX_NETWORK_IP_NETMASK="255.255.255.0"
#NODE2_AUX_NETWORK_IP_GATEWAY="179.254.20.1"
# XR Hostname
NODE2_XR_HOSTNAME="vtsr02"
# Loopback IP and netmask
NODE2_LOOPBACK_IP_ADDRESS="130.0.0.1"
NODE2_LOOPBACK_IP_NETMASK="255.255.255.255"
```

```
# VTS site uuid
VTS_SITE_UUID="abcdefab-abcd-abcd-abcd-abcdefabcdef"
```

Enabling VTC High Availability

To enable VTC high availability, perform the following steps.

Note Step 1 to 3 have to be run on both VTCs. Step 4 must be run only on the node that you want to make the active VTC.

Step 1 SSH to VTC 1 and VTC 2

Step 2 Go to the following directory:

```
cd /opt/vts/etc/
```

Step 3 Copy the cluster.conf file to /opt/vts/etc on both VTC 1 and VTC 2.

Step 4 Go to the following directory:

```
cd /opt/vts/bin
```

a) Run the sudo ./cluster_install.sh command. For example:

```
admin@Onion-VTC1:/opt/vts/bin#sudo  ./cluster_install.sh
```

You will be asked to provide the vtsr password. vtsr password is the password for VTSR1 and VTSR2. In addition to this, you will also be asked for the passwords for the switches directly connected to VTSR1 and VTSR2. And you will be prompted to run cluster_install.sh on VTC2 as well A message similar to below:

```
Please run cluster_install.sh on Onion-VTC2. Will wait until finished ==> At this point on VTC2
run the cluster install script
```

b) Run the sudo ./cluster_install.sh command. For example:

```
admin@Onion-VTC2:/opt/vts/bin#sudo  ./cluster_install.sh
```

You will be asked to provide the vtsr password. vtsr password is the password for VTSR1 and VTSR2. In addition to this, you will also be asked for the passwords for the switches directly connected to VTSR1 and VTSR2.An output similar to what is given below is displayed:

```
Change made to ncs.conf file. Need to restart ncs
Finding running docker container ID
263f311dbdff
Created symlink from /etc/systemd/system/multi-user.target.wants/pacemaker.service to
/lib/systemd/system/pacemaker.service.
Created symlink from /etc/systemd/system/multi-user.target.wants/corosync.service to
/lib/systemd/system/corosync.service.
Retrieving and storing node certificates
Retrieving master certificate
Retrieving slave certificate
Importing master certificate
Importing slave certificate
HA cluster is installed
admin@Onion-VTC2:/opt/vts/bin#
```

c) Once this is over on VTC2 ,Automatically on VTC1 the process will resume and you will see an output similar to what is given below :

```
2018-08-25 19:33:31,358 - INFO - Setup finished at 19:33:31
Configuring Pacemaker resources
Master node configuration finished
Retrieving and storing node certificates
Retrieving master certificate
Retrieving slave certificate
Importing master certificate
Importing slave certificate
HA cluster is installed.
admin@Onion-VTC1:/opt/vts/bin#
```

Enabling IOS XRv High Availability

You need to enable IOS XRv HA, if you have VTFs in your setup. For information about enabling IOS XRv HA, see Installing VTSR in High Availability Mode, on page 30.

Registering vCenter to VTC

To do this:

Step 1 Log in to VCSA.

Step 2 Go to **Networking > Distributed Virtual Switch > Manage > VTS**.

Note For vCenter 6.5, the VTS comes under Configure tab.

Step 3 Click on System Configuration

Step 4 Enter the following:

- VTS IP—This is the Virtual public IP address.
- VTS GUI Username
- VTS GUI Password

Step 5 Click Update.

Enabling VTS L3 High Availablity

This section describes the procedure to enable VTC L3 HA. In L3 HA environments, VTCs are located on separate overlay networks.

Before enabling VTC L3 HA, make sure that the following requirements are met

:

Step 1 Both VTC 1 and VTC 2 must have the same password. If they are not the same, change the password via the VTC GUI to make it identical with that of the other VTC. When you upgrade a VTC, bring up a new VTC, or do a hardware upgrade of a VTC host, remember to make sure that the passwords match.

Step 2 VTC1 and VTC2 are deployed in two different networks VTC1 is able to ping VTC 2 and wise versa via both underlay and overlay networks.

We recommend that you add static routes in /etc/network/interfaces on the VTCs. For example:

```
sudo vi /etc/network/interfaces
post-up route add -net 44.44.44.0/24 gw 10.10.10.1 dev eth1
post-up route add -net 10.10.10.0/24 gw 44.44.44.1 dev eth1
```

Step 3 VTSR installation is mandatory for VTC L3 HA to work. Although the VTSR(s) will not be able to register to the master VTC's Virtual IP Address (VIP) initially, the master VTC must still be able to reach all VTSRs in order to configure them to support L3HA.

Step 4 A site-id is required and need to be updated in VTSR_template file before generating the VTSR iso file and is generated by either

1. Creating a site directly from the VTC1 GUI.
2. Using the uuidgen linux command to generate a 32 character site id.

   In both cases, note down the site id for use in the below section "Deploying VTSR VMs". In the second case the site id will also be required to update the VTC master site configuration and associate the UUID to the site after the HA process is complete.

   The implementation of L3 HA is done for both management and underlay networks. For VIP ip reachability across networks the master VTC will configure BGP and route policies between the VTSRs and their respective directly connected TORs based on the user network settings provided in cluster.conf. Routing policy including tagging helps to migrate VIP IPs across the VTSRs during VTC HA.

Enabling VTS L3 High Availability for Management Network only

Setting up the VTC Environment for L3 High Availability Management

Use tthis procedure only to set up the VTC environment for L3 Hig Availabitiy (HA) management.

Step 1 Copy the /opt/vts/etc/cluster.conf_tmpl file to /opt/vts/etc/cluster.conf on both VTCs. The cluster.conf files must be identical on both VTCs. A sample modified file is given below:

a)

```
###Virtual Ip of VTC Master on the public interface. Must fill in at least 1
vip_public= 192.168.10.254
vip_public_ipv6=

###VTC1 Information. Must fill in at least 1 ip address
```

```
master_name=Onion-VTC1
master_ip=60.60.60.10
master_ipv6=

###VTC2 Information. Must fill in at least 1 ip address
slave_name= Onion-VTC2
slave_ip=70.70.70.10
slave_ipv6=

###In the event that a network failure occurs evenly between the two routers, the cluster needs an
outside ip to determine where the failure lies
###This can be any external ip such as your vmm ip or a dns but it is recommended to be a stable ip
 within your environment
###Must fill in at least 1 ip address
external_ip=81.81.81.1
external_ipv6=

###If you intend to use a virtual topology forwarder (VTF) in your environment, please fill in the
vip for the underlay as well as the underlay gateway. Otherwise leave blank.
###Virtual Ip of VTC Master on the private interface. You can fill in ipv4 configuration, ipv6, or
both if you use both
vip_private=
private_gateway=

vip_private_ipv6=
private_gateway_ipv6
###If you have your vtc's in different subnets, vtsr will need to be configured to route traffic and
 the below section needs to be filled in
###If you have your vtc's on the same subnet, the below section should be skipped

###Name of your vrf. Example: VTS_VIP
vrf_name=mgmt-vrf

###Ip of your first Vtsr. Example: 11.1.1.5
vtsr1_mgmt_ip=60.60.60.15
vtsr1_mgmt_ipv6=

###List of neighbors for vtsr1, separated by comma. Example: 11.1.1.1,11.1.1.2
vtsr1_bgp_neighbors= 11.11.11.11
vtsr1_bgp_neighbors_ipv6=

###Ip of your second Vtsr. Example: 12.1.1.5
vtsr2_mgmt_ip=70.70.70.15
vtsr2_mgmt_ipv6=

###List of neighbors for vtsr2, separated by comma. Example: 12.1.1.1,12.1.1.2
vtsr2_bgp_neighbors= 12.12.12.12
vtsr2_bgp_neighbors_ipv6=

###Username for Vtsr
vtsr_user= admin

###Vtsr ASN information
remote_ASN= 6500
local_ASN= 6501

###Vtsr BGP information
bgp_keepalive= 10
bgp_hold= 30

###Update source for Vtsr1 (i.e. loopback)
vtsr1_update_source=loopback0

###Update source for Vtsr2 (i.e. loopback)
vtsr2_update_source=loopback0
```

```
###Router BGP Id for Vtsr1
vtsr1_router_id= 21.21.21.21

###Router BGP Id for Vtsr2
vtsr2_router_id= 31.31.31.31

###Ipv4 Route Distinguisher Loopback for IPv6 Vtsr1 (if VTSR has a defined IPv6 management address,
 an ipv4 loopback address will be needed for the route distinguisher)
vtsr1_rd_loopback_name=loopback5
vtsr1_rd_loopback_address=41.41.41.41

###Ipv4 Route Distinguisher Loopback for IPv6 Vtsr2
vtsr2_rd_loopback_name=loopback5
vtsr2_rd_loopback_address=51.51.51.51


###XRVR1 name
vtsr1_name= vtsr01

###XRVR2 name
vtsr2_name= vtsr02

###If you plan on having your VTC's on different subnets and intend to use a virtual topology forwarder
 (VTF) in your environment,
### please fill out the following fields. Otherwise, leave blank

###List of neighbors for vtsr1, separated by comma. Example: 2.2.2.2,2.2.2.3
vtsr1_underlay_neighbors=
vtsr1_underlay_neighbors_ipv6=

###List of neighbors for vtsr2, separated by comma. Example: 3.3.3.2,3.3.3.3
vtsr2_underlay_neighbors=
vtsr2_underlay_neighbors_ipv6=

###OSPF Parameters
ospf_id_v4=
ospf_id_v6=
area=
default_cost=

###ISIS Parameters
isis_id=
is_type=
lsp_mtu=
key_chain_id=
key_id=
cryptographic_algorithm=
Network name (consist of an even number of octets and be of the form 01.2345.6789.abcd.ef)
vtsr1_network_entity=
vtsr2_network_entity=

###Directly connected Tor information for Vtsr1
vtsr1_directly_connected_device_ip=
vtsr1_directly_connected_device_ipv6=
vtsr1_directly_connected_device_user=
vtsr1_directly_connected_device_neighbors=
vtsr1_directly_connected_device_neighbors_ipv6=
vtsr1_directly_connected_ospf=
vtsr1_directly_connected_router_id=
vtsr1_directly_connected_update_source=

###Directly connected Tor information for Vtsr2
vtsr2_directly_connected_device_ip=
vtsr2_directly_connected_device_ipv6=
```

```
vtsr2_directly_connected_device_user=
vtsr2_directly_connected_device_neighbors=
vtsr2_directly_connected_device_neighbors_ipv6=
vtsr2_directly_connected_ospf=
vtsr2_directly_connected_router_id=
vtsr2_directly_connected_update_source=

###VPC Peer information if any. Otherwise leave blank
vtsr1_vpc_peer_ip=
vtsr1_vpc_peer_ipv6=
vtsr1_vpc_peer_user=
vtsr1_vpc_peer_ospf=
vtsr1_vpc_peer_router_id=
vtsr1_vpc_peer_update_source=

vtsr2_vpc_peer_ip=
vtsr2_vpc_peer_ipv6=
vtsr2_vpc_peer_user=
vtsr2_vpc_peer_ospf=
vtsr2_vpc_peer_router_id=
vtsr2_vpc_peer_update_source=

###VTC Underlay Addresses
vtc1_underlay=
vtc2_underlay=
vtc1_underlay_ipv6=
vtc2_underlay_ipv6=

##Gateway of secondary L3 underlay
vtc2_private_gateway=
vtc2_private_gateway_ipv6=
```

Step 2 Make sure VTSR configuration ISOs are up to date with above configurations. For example:

Note The VTSR values VTS_REGISTRATION_PASSWORD and VTS_SITE_UUID map to the VTC UI password and SITE ID on the VTCs, respectively.

```
# This is a sample VTSR configuration file
# Copyright (c) 2015 cisco Systems

# Please protect the generated ISO, as it contains authentication data
# in plain text.

# VTS Registration Information:
#  VTS_ADDRESS should be the IP for VTS. The value must be either an ip or a mask.
#  VTS_ADDRESS is mandatory. If only the V4 version is specified,
#  The V4 management interface for the VTSR (NODE1_MGMT_NETWORK_IP_ADDRESS)
#  will be used. If the V6 version is specified, the V6 management interface
#  for the VTSR (NODE1_MGMT_NETWORK_IPV6_ADDRESS) must be specified and will be used.
VTS_ADDRESS="192.168.10.254"
VTS_IPV6_ADDRESS=
# VTS_REGISTRATION_USERNAME used to login to VTS.
VTS_REGISTRATION_USERNAME="admin"
# VTS_REGISTRATION_PASSWORD is in plaintext.
VTS_REGISTRATION_PASSWORD="Cisco123!"
# VTSR VM Admin user/password
USERNAME="admin"
PASSWORD="cisco123"

# Mandatory Management-VRF name for VTSR.
```

```
VTS_MANAGEMENT_VRF="mgmt-vrf"

# VTSR VM Network Configuration for Node 1:
#  NETWORK_IP_ADDRESS, NETWORK_IP_NETMASK, and NETWORK_IP_GATEWAY
#     are required to complete the setup. Netmask can be in the form of
#     "24" or "255.255.255.0"
#  The first network interface configured with the VTC VM will be used for
#  underlay connectivity; the second will be used for the management network.
#  For both the MGMT and UNDERLAY networks, a <net-name>_NETWORK_IP_GATEWAY
#  variable is mandatory; they are used for monitoring purposes.
#
#  V6 is only supported on the mgmt network and dual stack is
#  currently not supported, so if both are specified V6 will take priority (and
#  requires VTS_IPV6_ADDRESS to be set).
#  The *V6* parameters for the mgmt network are optional. Note that if V6 is used for mgmt
#  it must be V6 on both nodes. Netmask must be the prefix length for V6.
NODE1_MGMT_NETWORK_IP_ADDRESS="60.60.60.15"
NODE1_MGMT_NETWORK_IP_NETMASK="255.255.255.0"
NODE1_MGMT_NETWORK_IP_GATEWAY="60.60.60.1"
NODE1_MGMT_NETWORK_IPV6_ADDRESS=
NODE1_MGMT_NETWORK_IPV6_NETMASK=
NODE1_MGMT_NETWORK_IPV6_GATEWAY=
NODE1_UNDERLAY_NETWORK_IP_ADDRESS="10.10.10.33"
NODE1_UNDERLAY_NETWORK_IP_NETMASK="255.255.255.0"
NODE1_UNDERLAY_NETWORK_IP_GATEWAY="10.10.10.1"
# AUX network is optional
#NODE1_AUX_NETWORK_IP_ADDRESS="169.254.20.100"
#NODE1_AUX_NETWORK_IP_NETMASK="255.255.255.0"
#NODE1_AUX_NETWORK_IP_GATEWAY="169.254.20.1"
# XR Hostname
NODE1_XR_HOSTNAME="vtsr01"
# Loopback IP and netmask
NODE1_LOOPBACK_IP_ADDRESS="128.0.0.10"
NODE1_LOOPBACK_IP_NETMASK="255.255.255.255"

# Operational username and password - optional
# These need to be configured to start monit on VTSR

VTSR_OPER_USERNAME="admin"
# Password needs an encrypted value
# Example : "openssl passwd -1 -salt <salt-string> <password>"
VTSR_OPER_PASSWORD="$1$cisco$Qv2TLtPNI3jqwXMOA3M3f0/"

# VTSR monit interval - optional - default is 30 seconds
VTSR_MONIT_INTERVAL="30"

# VTSR VM Network Configuration for Node 2:
#  If there is no HA then the following Node 2 configurations will remain commented and
#  will not be used and Node 1 configurations alone will be applied
#  For HA , the following Node 2 configurations has to be uncommented
#  VTSR VM Network Configuration for Node 2
#  NETWORK_IP_ADDRESS, NETWORK_IP_NETMASK, and NETWORK_IP_GATEWAY
#     are required to complete the setup. Netmask can be in the form of
#     "24" or "255.255.255.0"
#  The first network interface configured with the VTC VM will be used for
#  underlay connectivity; the second will be used for the management network.
#  For both the MGMT and UNDERLAY networks, a <net-name>_NETWORK_IP_GATEWAY
```

```
#  variable is mandatory; they are used for monitoring purposes.
#
#  V6 is only supported on the mgmt network and dual stack is
#  currently not supported, so if both are specified V6 will take priority (and
#  requires VTS_IPV6_ADDRESS to be set).
#  The *V6* parameters for the mgmt network are optional. Note that if V6 is used for mgmt
#  it must be V6 on both nodes. Netmask must be the prefix length for V6.
NODE2_MGMT_NETWORK_IP_ADDRESS="70.70.70.15"
NODE2_MGMT_NETWORK_IP_NETMASK="255.255.255.0"
NODE2_MGMT_NETWORK_IP_GATEWAY="70.70.70.1"
NODE2_MGMT_NETWORK_IPV6_ADDRESS=
NODE2_MGMT_NETWORK_IPV6_NETMASK=
NODE2_MGMT_NETWORK_IPV6_GATEWAY=
NODE2_UNDERLAY_NETWORK_IP_ADDRESS="44.44.44.15"
NODE2_UNDERLAY_NETWORK_IP_NETMASK="255.255.255.0"
NODE2_UNDERLAY_NETWORK_IP_GATEWAY="44.44.44.1"
#  AUX network is optional
#  Although Aux network is optional it should be either present in both nodes
#  or not present in both nodes.
#  It cannot be present on Node1 and not present on Node2 and vice versa
#NODE2_AUX_NETWORK_IP_ADDRESS="179.254.20.200"
#NODE2_AUX_NETWORK_IP_NETMASK="255.255.255.0"
#NODE2_AUX_NETWORK_IP_GATEWAY="179.254.20.1"
# XR Hostname
NODE2_XR_HOSTNAME="vtsr02"
# Loopback IP and netmask
NODE2_LOOPBACK_IP_ADDRESS="130.0.0.1"
NODE2_LOOPBACK_IP_NETMASK="255.255.255.255"

# VTS site uuid
VTS_SITE_UUID="abcdefab-abcd-abcd-abcd-abcdefabcdef"
```

Deploying VTSR VMs

To deploy VTSR 1 and VTSR 2, follow the below steps:

Step 1 Update Links to point to VTSR bring up procedure including ISO generation.

Step 2 After a VTSR is up and running, ssh to it's management ip address and make sure that the routes are added and both VTC 1 and VTC 2 can reach VTSR 1 and VTSR 2, and vice versa.

Day Zero Configuration for High Availability

The following example shos Day zero configuration for L3 High Availability:

```
VTSR1 #

vrf mgmt-vrf
address-family ipv4 unicast
!
address-family ipv6 unicast
!
```

```
!
tpa
vrf default
  address-family ipv4
   update-source dataports GigabitEthernet0/0/0/0
  !
!
vrf mgmt-vrf
  address-family ipv4
   update-source dataports GigabitEthernet0/0/0/1
  !
  address-family ipv6
   update-source dataports GigabitEthernet0/0/0/1
  !
!
!
interface Loopback0
ipv4 address 128.0.0.10 255.255.255.255
!
interface Loopback1
ipv4 address 169.254.10.1 255.255.255.255
!
interface MgmtEth0/RP0/CPU0/0
shutdown
!
interface GigabitEthernet0/0/0/0
ipv4 address 10.10.10.33 255.255.255.0
!
interface GigabitEthernet0/0/0/1
vrf mgmt-vrf
ip address 60.60.60.15/24
!
interface GigabitEthernet0/0/0/2
shutdown
!
router static
address-family ipv4 unicast
   0.0.0.0/0  60.60.60.1
  !
!
!

VTSR2 #

vrf mgmt-vrf
address-family ipv4 unicast
!
address-family ipv6 unicast
!
!
tpa
vrf default
  address-family ipv4
   update-source dataports GigabitEthernet0/0/0/0
  !
!
vrf mgmt-vrf
  address-family ipv4
   update-source dataports GigabitEthernet0/0/0/1
  !
  address-family ipv6
   update-source dataports GigabitEthernet0/0/0/1
  !
!
```

```
!
call-home
service active
contact smart-licensing
profile CiscoTAC-1
  active
  destination transport-method http
!
!
interface Loopback0
ipv4 address 128.0.0.10 255.255.255.255
!
interface Loopback1
ipv4 address 169.254.10.1 255.255.255.255
!
interface MgmtEth0/RP0/CPU0/0
shutdown
!
interface GigabitEthernet0/0/0/0
ipv4 address 44.44.44.15 255.255.255.0
!
interface GigabitEthernet0/0/0/1
vrf mgmt-vrf
ipv4 address 70.70.70.15/24
!
interface GigabitEthernet0/0/0/2
shutdown
!
router static
address-family ipv4 unicast
  0.0.0.0/0 44.44.44.1
!
vrf mgmt-vrf
  address-family ipv4 unicast
   0.0.0.0/0 70.70.70.1
  !
!
!
```

Verifying VTSR High Availability Setup

Check the HA status with command sudo crm status . For example:

root@vtsr01:/opt/cisco/package# crm status

Last updated: Thu Aug 30 16:27:25 2018 Last change: Thu Aug 30 14:52:01 2018 by root via cibadmin on vtsr01

Stack: corosync

Current DC: vtsr01 (version 1.1.14-70404b0) - partition with quorum

2 nodes and 24 resources configured

Online: [vtsr01 vtsr02]

Full list of resources:

```
dl_server       (ocf::heartbeat:anything):      Started vtsr01
 redis_server       (ocf::heartbeat:anything):      Started vtsr01
 Clone Set: cfg_dl_clone [cfg_dl]
```

```
    Started: [ vtsr01 vtsr02 ]
 Clone Set: rc_clone [rc]
    Started: [ vtsr01 vtsr02 ]
 Clone Set: sm_clone [sm]
    Started: [ vtsr01 vtsr02 ]
 Clone Set: tunnel_clone [tunnel]
    Started: [ vtsr01 vtsr02 ]
 Clone Set: confd_clone [confd]
    Started: [ vtsr01 vtsr02 ]
 Clone Set: mping_clone [mgmt_ping]
    Started: [ vtsr01 vtsr02 ]
 Clone Set: uping_clone [underlay_ping]
    Started: [ vtsr01 vtsr02 ]
 Clone Set: monit_clone [monit]
    Started: [ vtsr01 vtsr02 ]
 Clone Set: socat_confd_clone [socat-confd]
    Started: [ vtsr01 vtsr02 ]
 Clone Set: socat_monit_clone [socat-monit]
    Started: [ vtsr01 vtsr02 ]
 Clone Set: mate_tunnel_clone [mate_tunnel]
    Started: [ vtsr01 vtsr02 ]
root@vtsr01:/opt/cisco/package#
```

root@vtsr02:/opt/cisco/package# crm status

Last updated: Thu Aug 30 16:32:06 2018 Last change: Thu Aug 30 14:52:01 2018 by root via cibadmin on vtsr01

Stack: corosync

Current DC: vtsr01 (version 1.1.14-70404b0) - partition with quorum

2 nodes and 24 resources configured

Online: [vtsr01 vtsr02]

Full list of resources:

```
dl_server       (ocf::heartbeat:anything):      Started vtsr01
redis_server       (ocf::heartbeat:anything):       Started vtsr01
Clone Set: cfg_dl_clone [cfg_dl]
    Started: [ vtsr01 vtsr02 ]
Clone Set: rc_clone [rc]
    Started: [ vtsr01 vtsr02 ]
Clone Set: sm_clone [sm]
    Started: [ vtsr01 vtsr02 ]
Clone Set: tunnel_clone [tunnel]
    Started: [ vtsr01 vtsr02 ]
Clone Set: confd_clone [confd]
    Started: [ vtsr01 vtsr02 ]
Clone Set: mping_clone [mgmt_ping]
    Started: [ vtsr01 vtsr02 ]
Clone Set: uping_clone [underlay_ping]
    Started: [ vtsr01 vtsr02 ]
Clone Set: monit_clone [monit]
    Started: [ vtsr01 vtsr02 ]
Clone Set: socat_confd_clone [socat-confd]
    Started: [ vtsr01 vtsr02 ]
```

```
 Clone Set: socat_monit_clone [socat-monit]
     Started: [ vtsr01 vtsr02 ]
 Clone Set: mate_tunnel_clone [mate_tunnel]
     Started: [ vtsr01 vtsr02 ]
root@vtsr02:/opt/cisco/package#
```

Enabling VTC High Availability

To enable VTC high availability, do the following steps:

Note Step 1 to 3 should be run on both VTCs. Step 4 must be run only on the node that you want to make the active VTC.

Step 1 SSH to VTC 1 and VTC 2 .

Step 2 Go to the following directory:

cd /opt/vts/etc/

Step 3 Copy the cluster.conf file to /opt/vts/etc on both VTC 1 and VTC 2.

Step 4 Go to the following directory::

cd /opt/vts/bin

Run the **sudo ./cluster_install.sh** command. For example: admin@Onion-VTC1:/opt/vts/bin#sudo ./cluster_install.sh You will be asked to provide the vtsr password. vtsr password is the password for VTSR1 and VTSR2. You will be prompted to run cluster_install.sh on VTC2 as well . A message is displyaed similar to below example:

Please run cluster_install.sh on Onion-VTC2. Will wait until finished ==> At this point on VTC2 run the cluster install script

Run the sudo ./cluster_install.sh command. For example:

admin@Onion-VTC2:/opt/vts/bin#sudo ./cluster_install.sh

You will be asked to provide the vtsr password. vtsr password is the password for VTSR1 and VTSR2.

An output similar to what is given below is displayed:

Change made to ncs.conf file. Need to restart ncs

Finding running docker container ID

263f311dbdff

Created symlink from /etc/systemd/system/multi-user.target.wants/pacemaker.service to /lib/systemd/system/pacemaker.service.

Created symlink from /etc/systemd/system/multi-user.target.wants/corosync.service to /lib/systemd/system/corosync.service.

Retrieving and storing node certificates

Retrieving master certificate

Retrieving slave certificate

Importing master certificate

Importing slave certificate

HA cluster is installed

admin@Onion-VTC2:/opt/vts/bin#

Once this is over on VTC2 ,Automatically on VTC1 the process will resume and you will see an output similar to what is given below :

```
2018-08-25 19:33:31,358 - INFO - Setup finished at 19:33:31
Configuring Pacemaker resources
Master node configuration finished
Retrieving and storing node certificates
Retrieving master certificate
Retrieving slave certificate
Importing master certificate
Importing slave certificate
HA cluster is installed.
admin@Onion-VTC1:/opt/vts/bin#
```

Verifying the VTC High Availabilty

To verify the

Step 1 Run the sudo crm status command. For example

admin@Onion-VTC1:~$ sudo crm status

[sudo] password for admin:

Last updated: Mon Aug 27 17:34:47 2018 Last change: Sat Aug 25 20:20:11 2018 by root via crm_attribute on Onion-VTC2

Stack: corosync

Current DC: Onion-VTC1 (version 1.1.14-70404b0) - partition with quorum

2 nodes and 5 resources configured

Online: [Onion-VTC1 Onion-VTC2]

Full list of resources:

Master/Slave Set: ms_vtc_ha [vtc_ha]

Masters: [Onion-VTC1]

Slaves: [Onion-VTC2]

ClusterIP (ocf::heartbeat:IPaddr2): Started Onion-VTC1

ClusterIP2 (ocf::heartbeat:IPaddr2): Started Onion-VTC1

admin@Onion-VTC1:~$

Step 2 Verify on which VTC the virtual IP is configured. For example:

root@Onion-VTC1:/home/admin# ip addr s

2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000

link/ether 52:54:00:c7:92:c2 brd ff:ff:ff:ff:ff:ff

inet 60.60.60.10/24 brd 60.60.60.255 scope global eth0

valid_lft forever preferred_lft forever

inet 192.168.10.254/32 scope global eth0

valid_lft forever preferred_lft forever

inet6 fe80::5054:ff:fec7:92c2/64 scope link

valid_lft forever preferred_lft forever

3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000

link/ether 52:54:00:0e:17:59 brd ff:ff:ff:ff:ff:ff

inet 10.10.10.10/24 brd 10.10.10.255 scope global eth1

valid_lft forever preferred_lft forever

inet6 fe80::5054:ff:fe0e:1759/64 scope link

valid_lft forever preferred_lft forever

Verifying VTSR High Availability

Before you begin

Step 1 Verify the following on VTSR 1 and VTSR 2.

```
!
route-policy BGP_HA_VTC1_MGMT_IPV4
  if destination in (192.168.10.254) and community matches-every (6501:1001) then
    delete community all
    set next-hop 60.60.60.10
  else
    drop
  endif
end-policy
!
route-policy BGP_HA_VTC2_MGMT_IPV4
  if destination in (192.168.10.254) and community matches-every (6501:1002) then
    delete community all
    set next-hop 70.70.70.10
  else
    drop
  endif
end-policy
!
route-policy REDISTRIBUTE_TO_BGP_HA_MGMT_IPV4
  if destination in (192.168.10.254) and tag is 1001 then
    set community (6501:1001)
    done
  elseif destination in (192.168.10.254) and tag is 1002 then
    set community (6501:1002)
```

```
    done
  else
    drop
  endif
end-policy
!
router static
address-family ipv4 unicast
  0.0.0.0/0 10.10.10.1
!
vrf mgmt-vrf
  address-family ipv4 unicast
   0.0.0.0/0 60.60.60.1
   192.168.10.254/32 60.60.60.10 tag 1002
  !
!
!
router ospf 100
router-id 21.21.21.21
address-family ipv4 unicast
area 0.0.0.0
  default-cost 10
  interface Loopback0
  !
  interface GigabitEthernet0/0/0/0
  !
!
!
router bgp 6501
bgp router-id 21.21.21.21
address-family ipv4 unicast
 !
 address-family vpnv4 unicast
!
 address-family ipv6 unicast
!
 address-family vpnv6 unicast
!
vrf mgmt-vrf
  rd 60.60.60.15:1
  bgp router-id 60.60.60.15
  address-family ipv4 unicast
  network 192.168.10.254/32 route-policy   REDISTRIBUTE_TO_BGP_HA_MGMT_IPV4
  !
  neighbor 11.11.11.11
   remote-as 6500
   ebgp-multihop 255
  timers 10 30
   description ***MGMT IPV4 Network Directly connected BGP Peer of Vtsr1
   update-source GigabitEthernet0/0/0/1
   address-family ipv4 unicast
    route-policy BGP_HA_VTC1_MGMT_IPV4 out
   !
  !
  neighbor 12.12.12.12
   remote-as 6500
   ebgp-multihop 255
   timers 10 30
   description ***MGMT IPV4 Network Directly connected BGP Peer of Vtsr2
   update-source GigabitEthernet0/0/0/1
   address-family ipv4 unicast
    route-policy BGP_HA_VTC2_MGMT_IPV4 out
   !
  !
```

```
!
!
 RP/0/RP0/CPU0:vtsr01#show bgp sessions
Wed Aug 29 10:06:15.706 UTC
```

Neighbor VRF Spk AS InQ OutQ NBRState NSRState

11.11.11.11 mgmt-vrf 0 6500 0 0 Established None

12.12.12.12 mgmt-vrf 0 6500 0 0 Established None

RP/0/RP0/CPU0:vtsr01#

RP/0/RP0/CPU0:vtsr01#show bgp all all

Thu Aug 30 15:56:22.350 UTC

Address Family: VPNv4 Unicast

Address Family: VPNv6 Unicast

BGP router identifier 21.21.21.21, local AS number 6501

BGP generic scan interval 60 secs

Non-stop routing is enabled

BGP table state: Active

Table ID: 0x0 RD version: 0

BGP main routing table version 4

BGP NSR Initial initsync version 3 (Not Reached)

BGP NSR/ISSU Sync-Group versions 0/0

BGP scan interval 60 secs

Status codes: s suppressed, d damped, h history, * valid, > best

i - internal, r RIB-failure, S stale, N Nexthop-discard

Origin codes: i - IGP, e - EGP, ? - incomplete

Network Next Hop Metric LocPrf Weight Path

Route Distinguisher: 41.41.41.41:1 (default for vrf mgmt-vrf)

*> 192.168.10.254/32

60.60.60.10 0 32768 i

Processed 1 prefixes, 1 paths

Address Family: IPv4 Unicast

Address Family: IPv6 Unicast

Step 2 Verify whether the configuration is pushed on VTSR 2.

```
route-policy BGP_HA_VTC1_MGMT_IPV4
  if destination in (192.168.10.254) and community matches-every (6501:1001) then
    delete community all
    set next-hop 60.60.60.10
  else
    drop
  endif
end-policy
!
route-policy BGP_HA_VTC2_MGMT_IPV4
  if destination in (192.168.10.254) and community matches-every (6501:1002) then
    delete community all
    set next-hop 70.70.70.10
  else
    drop
  endif
end-policy
!
route-policy REDISTRIBUTE_TO_BGP_HA_MGMT_IPV4
  if destination in (192.168.10.254) and tag is 1001 then
    set community (6501:1001)
    done
  elseif destination in (192.168.10.254) and tag is 1002 then
    set community (6501:1002)
    done
  else
    drop
  endif
end-policy
!
router static
address-family ipv4 unicast
  0.0.0.0/0 44.44.44.1
!
vrf mgmt-vrf
  address-family ipv4 unicast
   0.0.0.0/0 70.70.70.1
   192.168.10.254/32 60.60.60.10 tag 1002
  !
!
!
router ospf 100
router-id 31.31.31.31
address-family ipv4 unicast
area 0.0.0.0
  default-cost 10
  interface Loopback0
  !
  interface GigabitEthernet0/0/0/0
  !
!
!
router bgp 6501
bgp router-id 31.31.31.31
address-family ipv4 unicast
!
address-family vpnv4 unicast
!
address-family ipv6 unicast
!
address-family vpnv6 unicast
!
vrf mgmt-vrf
  rd 70.70.70.15:1
```

```
  bgp router-id 70.70.70.15
  address-family ipv4 unicast
  network 192.168.10.254/32 route-policy REDISTRIBUTE_TO_BGP_HA_MGMT_IPV4
  !
  neighbor 11.11.11.11
   remote-as 6500
   ebgp-multihop 255
   timers 10 30
   description ***MGMT IPV4 Network Directly connected BGP Peer of Vtsr1
   update-source GigabitEthernet0/0/0/1
   address-family ipv4 unicast
    route-policy BGP_HA_VTC1_MGMT_IPV4 out
   !
  !
  neighbor 12.12.12.12
   remote-as 6500
   ebgp-multihop 255
   timers 10 30
   description ***MGMT IPV4 Network Directly connected BGP Peer of Vtsr2
   update-source GigabitEthernet0/0/0/1
   address-family ipv4 unicast
    route-policy BGP_HA_VTC2_MGMT_IPV4 out
   !
  !
!
!
```

RP/0/RP0/CPU0:vtsr02#show bgp sessions

Wed Aug 29 10:13:34.909 UTC

Neighbor VRF Spk AS InQ OutQ NBRState NSRState

11.11.11.11 mgmt-vrf 0 6500 0 0 Established None

12.12.12.12 mgmt-vrf 0 6500 0 0 Established None

RP/0/RP0/CPU0:vtsr02#

Switching Over Between Master and Slave Nodes

There are two of ways to switch over from Master to Slave node.

- Restart the nso service on the Master. The switchover happens automatically. For example:

```
admin@vts02:/opt/vts/log/nso$ sudo service nso restart


admin@vts02:/opt/vts/log/nso$ sudo crm status

[sudo] password for admin:

Last updated: Mon Apr 10 18:43:52 2017          Last change: Mon Apr 10 17:15:21 2017
by root via crm_attribute on vts01

Stack: corosync

Current DC: vts01 (version 1.1.14-70404b0) - partition with quorum

2 nodes and 4 resources configured
```

```
Online: [ vts01 vts02 ]


Full list of resources:


Master/Slave Set: ms_vtc_ha [vtc_ha]

     Masters: [ vts01 ]

     Slaves: [ vts02 ]

ClusterIP      (ocf::heartbeat:IPaddr2):       Started vts01

ClusterIPV6    (ocf::heartbeat:IPaddr2):       Started vts01
```

Or,

- Set the Master node to standby, and then bring it online.

  In the below example, vts02 is initially the Master, which is then switched over to the Slave role.

```
admin@vts01:~$ sudo crm node standby
[sudo] password for admin:


admin@vts01:/opt/vts/log/nso$ sudo crm status

[sudo] password for admin:

Last updated: Mon Apr 10 18:43:52 2017          Last change: Mon Apr 10 17:15:21 2017
by root via crm_attribute on vts01

Stack: corosync

Current DC: vts01 (version 1.1.14-70404b0) - partition with quorum

2 nodes and 4 resources configured


Node vts01 standby
Online: [ vts02 ]


Full list of resources:


Master/Slave Set: ms_vtc_ha [vtc_ha]

     Masters: [ vts02 ]

     Stopped: [ vts01 ]
```

```
ClusterIP      (ocf::heartbeat:IPaddr2):       Started vts02

ClusterIPV6    (ocf::heartbeat:IPaddr2):       Started vts02


admin@vts01~$ sudo crm node online


admin@vts02:/opt/vts/log/nso$ sudo crm status

[sudo] password for admin:

Last updated: Mon Apr 10 18:43:52 2017          Last change: Mon Apr 10 17:15:21 2017
by root via crm_attribute on vts01

Stack: corosync

Current DC: vts01 (version 1.1.14-70404b0) - partition with quorum

2 nodes and 4 resources configured


Online: [ vts01 vts02 ]


Full list of resources:


Master/Slave Set: ms_vtc_ha [vtc_ha]

     Masters: [ vts02 ]

     Slaves: [ vts01 ]

ClusterIP      (ocf::heartbeat:IPaddr2):       Started vts02

ClusterIPV6    (ocf::heartbeat:IPaddr2):       Started vts02
```

Uninstalling VTC High Availability

To move VTC back to its pre-High Availability state, run the following script:

Note Make sure the ncs server is active/running. Then run this script on both the active and standby nodes.

```
root@vts02:/opt/vts/bin# ./cluster_uninstall.sh
This will move HA configuration on this system back to pre-installed state. Proceed?(y/n)
y
```

Troubleshooting Password Change Issues

If a password change is performed while the VTS Active and Standby were up, and the change does not get applied to the Standby, the changed password will not get updated in the /opt/vts/etc/credentials file on the Standby. Due to this, when VTS Standby VM is brought up, it cannot connect to NCS. CRM_MON shows the state as shutdown for Standby, and it does not come online.

To troubleshoot this:

Step 1 Copy the **/opt/vts/etc/credentials** file from the VTC Active to the same location (/opt/vts/etc/credentials) on the VTC Standby node.

Step 2 Run the crm node command on VTC Standby to bring it online.

```
crm node online VTC2
```

Step 3 Run the command crm status to show both VTC1 and VTC2 online.

```
crm status
```

Installing VTSR in High Availability Mode

VTSR high availability mode needs to be enabled before you install VTF(s) in your set up. The second VTSR will not get registered to the VTC if it starts up after VTF installation .

Enabling VTSR high availability involves:

- Generating two ISOs for the Master and the Slave VMs. See Generating an ISO for VTSR for details.
- Deploy the two VTSR VMs using the respective ISO files generated during the process. See Deploying VTSR on OpenStack or Deploying VTSR on VMware, based on your VMM type.

The system automatically detects which VM is the Master and which is the slave, based on the information you provide while generating the ISO files.

Verifying VTSR High Availability Setup

You can check the VTSR HA status using the sudo crm status. For example:

```
root@vtsr01:/opt/cisco/package# crm status
Last updated: Tue Aug 28 21:00:18 2018               Last change: Sat Aug 25 13:29:45 2018
 by root via cibadmin on vtsr01
Stack: corosync
Current DC: vtsr01 (version 1.1.14-70404b0) - partition with quorum
2 nodes and 24 resources configured

Online: [ vtsr01 vtsr02 ]

Full list of resources:

dl_server        (ocf::heartbeat:anything):        Started vtsr01
redis_server        (ocf::heartbeat:anything):        Started vtsr01
```

```
Clone Set: cfg_dl_clone [cfg_dl]
     Started: [ vtsr01 vtsr02 ]
Clone Set: rc_clone [rc]
     Started: [ vtsr01 vtsr02 ]
Clone Set: sm_clone [sm]
     Started: [ vtsr01 vtsr02 ]
Clone Set: tunnel_clone [tunnel]
     Started: [ vtsr01 vtsr02 ]
Clone Set: confd_clone [confd]
     Started: [ vtsr01 vtsr02 ]
Clone Set: mping_clone [mgmt_ping]
     Started: [ vtsr01 vtsr02 ]
Clone Set: uping_clone [underlay_ping]
     Started: [ vtsr01 vtsr02 ]
Clone Set: monit_clone [monit]
     Started: [ vtsr01 vtsr02 ]
Clone Set: socat_confd_clone [socat-confd]
     Started: [ vtsr01 vtsr02 ]
Clone Set: socat_monit_clone [socat-monit]
    Started: [ vtsr01 vtsr02 ]
Clone Set: mate_tunnel_clone [mate_tunnel]
     Started: [ vtsr01 vtsr02 ]
root@vtsr01:/opt/cisco/package#

root@vtsr02:/opt/cisco/package# crm status
Last updated: Tue Aug 28 21:04:38 2018               Last change: Sat Aug 25 13:29:45 2018
 by root via cibadmin on vtsr01
Stack: corosync
Current DC: vtsr01 (version 1.1.14-70404b0) - partition with quorum
2 nodes and 24 resources configured

Online: [ vtsr01 vtsr02 ]

Full list of resources:

dl_server        (ocf::heartbeat:anything):        Started vtsr01
redis_server        (ocf::heartbeat:anything):        Started vtsr01
Clone Set: cfg_dl_clone [cfg_dl]
     Started: [ vtsr01 vtsr02 ]
Clone Set: rc_clone [rc]
     Started: [ vtsr01 vtsr02 ]
Clone Set: sm_clone [sm]
     Started: [ vtsr01 vtsr02 ]
Clone Set: tunnel_clone [tunnel]
     Started: [ vtsr01 vtsr02 ]
Clone Set: confd_clone [confd]
     Started: [ vtsr01 vtsr02 ]
Clone Set: mping_clone [mgmt_ping]
     Started: [ vtsr01 vtsr02 ]
Clone Set: uping_clone [underlay_ping]
     Started: [ vtsr01 vtsr02 ]
Clone Set: monit_clone [monit]
     Started: [ vtsr01 vtsr02 ]
Clone Set: socat_confd_clone [socat-confd]
     Started: [ vtsr01 vtsr02 ]
Clone Set: socat_monit_clone [socat-monit]
    Started: [ vtsr01 vtsr02 ]
Clone Set: mate_tunnel_clone [mate_tunnel]
     Started: [ vtsr01 vtsr02 ]
root@vtsr02:/opt/cisco/package#
```

Hight Availability Scenarios

This section describes the various HA scenarios.

Manual Failover

To do a manual failover:

Step 1 Run **sudo crm node standby** on the current VTC Active to force a failover to the Standby node.

Step 2 Verify the other VTC to check whether it has taken over the Active role.

Step 3 On the earlier Active, run **crm node online** to bring it back to be part of the cluster again.

VTC Master Reboot

When the VTC Active reboots, much like a manual failover, the other VTC takes over as the Active. After coming up out of the reboot, the old Active VTC will automatically come up as the Standby.

Split Brain

When there is a network break and both VTCs are still up, VTC HA attempts to ascertain where the network break lies. During the network failure, the Active and Standby will lose connectivity with each other. At this point, the Active will attempt to contact the external ip (a parameter set during the initial configuration) to see if it still has outside connectivity.

If it cannot reach the external ip, VTC cannot know if the Standby node is down or if it has promoted itself to Active. As a result, it will shut itself down to avoid having two Active nodes.

The Standby, upon sensing the loss of connectivity with the Active, tries to promote itself to the Active mode. But first, it will check if it has external connectivity. If it does, it will become the Active node. However, if it also cannot reach the external ip (for instance if a common router is down), it will shut down.

At this point, the VTC that had the network break cannot tell if the other VTC is Active and receiving transactions. When the network break is resolved, it will be able to do the comparison and the VTC with the latest database will become Active.

If the other VTC also has a network break or is not available, the agent will not be able to do the comparison still, and it will wait. If the other VTC is not be available for some time, you may force the available VTC to be master:

```
admin@vtc1:/home/admin# sudo /opt/vts/bin/force_master.py
```

Double Failure

When both VTC are down at the same time, a double failure scenario has occurred. After a VTC has come up, it does not immediately know the state of the other VTC's database. Consequently, before HA is resumed, an agent runs in the background trying to compare the two databases. When both systems have recovered, it will be able to do the comparison and the VTC with latest database will become the Active.

If the other VTC is not be available for some time, you may force the available VTC to be master:

```
admin@vtc1:/home/admin# sudo /opt/vts/bin/force_master.py
```