# Troubleshooting Cisco Elastic Services Controller High Availability



# Troubleshooting Cisco Elastic Services Controller High Availability

ESC HA consists of many components/services and keeps monitoring the self health checking. Any failure of ESC micro services causes HA synchronization and other related issues.

# Cisco Elastic Services Controller High Availability Troubleshooting Overview

Following are some generic troubleshooting items for ESC HA:

**Problem :** Network Problem

**Solution:** If you have a networking problem, check for the following items:

- The static IP addresses for both ESC nodes are correct based on the OpenStack configuration and each node is able to access the other node.
- The gateway for each network interface is accessible from each instance.
- Virtual ipaddress (kad_vip) is pingable from master node. (to find kad_vip, run: "sed -n '/virtual_ipaddress/{n;p;}' /etc/keepalived/keepalived.conf").

**Checking the logs:**

Following are some logs and their locations to check ESC HA troubleshooting:

- The ESC manager log, located at `/var/log/esc/escmanager.log`.
- The ESC HA log about esc service startup/stop, located at /var/log/esc/esc_haagent.log (ESC 2.X) and /var/log/esc/escadm.log (ESC 3.X).
- The exabgp log, located at `/var/log/exabgp.log`.

**Configuration and log check for Keepalived:**

Verify the keepalived configuration in the following path:

- You can check the configuration file at `/etc/keepalived/keepalived.conf` to verify the keepalived configuration .
- The keepalived log is located at `/var/log/messages` by grep keepalived or vrrp.

**Configuration and log check for DRBD:**

Verify the DRBD configuration in the following path:

- To verify the DRBD configuration, check the file at `/etc/drbd.d/esc.res` .
- The DRBD log is located at `/var/log/messages` by grep drbd.

**Configuration check for BGP:**

Verify the BGP configuration:

- The BGP configuration must be the same as installation arguments and ASR configuration.
- The BGP configuration can be verified by checking the file at /opt/cisco/esc/esc-scripts/bgp-sa/exabgp/neighbor_init.conf.

# High Availability Active Node Stays in the Switching-to-Active State

The ESC High Availability (HA) cluster might have some issues at the startup. The following are the possible issues listed:

**Problem:**

- ESC HA node cannot reach its peer during the initial installation. Verify that ESC HA is able to reach its peer when switching to Active for the first time.
- ESC service (tomcat/escmanager) cannot start properly due to database problems (etc. database migration, database file corruption).
- Confd cannot start due to the CDB file corruption.
- Postgresql cannot start or init due to issues in the file system (disk space is 100% full).
- The connection between ESC nodes is too slow (MTU issue).

**Verification:**

Verify the following are the items to troublehsoot the previous problems:

- The connectivity between ESC Active node and standby node. For initial installation, ESC active (escadm) service will not be up if it cannot reach the standby node. Ensure that you have both ESC nodes successfully deployed and they can reach each other.
- Check ESC logs at /var/log/esc/esc_haagent.log (ESC 2.X), or /var/log/esc/escadm.log (ESC 3.X and up). In most of the cases, it displays why ESC service gets blocked and which step/service startup did not work well.
- If esc_service/escadm and postgresql have started, check the log at /var/log/esc/escmanager.log for more information about the error messages.

# Keepalived Service State on Both HA VM Instances Stay in Backup State

**Problem :**

ESC HA has four different states: Active, Backup, Fault, and Stop. The Backup state is a transit state between Stop to Active, or Fault to Active. It is probable that both ESC VMs stick to the Backup state but usually do not last for a longer period. If you observe that the keepalived state on both ESC HA VMs is in a Backup state for more than two minutes, it could be a problem. However, there is a possibility of VRRP broadcast interference in your network.

**Solution :**

Run the following commands in any of your ESC VM to diagnose this problem:

```
$ sudo tcpdump -vvv -n -i ethX host ff02::12 (for IPv6 network)
$ sudo tcpdump -vvv -n -i ethX host 224.0.0.18 (for IPv4 Network)
```

The previous tcpdump commands listens to the VRRP broadcast packets in your ESC's heartbeat network. Use your heartbeat network interface to replace the ethX in the previous commands. For example, eth0. It provides you the information that whether your ESC VM is able to listen to the VRRP broadcast generated by any node in the subnet and you will find out who is doing the VRRP broadcasting in your network. For example:

```
# sudo tcpdump -vvv -n -i eth0 host 224.0.0.18
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
21:40:37.269728 IP (tos 0xc0, ttl 255, id 16606, offset 0, flags [none], proto VRRP (112),
 length 40)
    152.16.3.76 > 224.0.0.18: vrrp 152.16.3.76 > 224.0.0.18: VRRPv2, Advertisement, vrid
78, prio 101, authtype none, intvl 5s, length 20, addrs: 152.16.3.78
21:40:37.271332 IP (tos 0xc0, ttl 255, id 63866, offset 0, flags [none], proto VRRP (112),
 length 40)
    152.16.7.228 > 224.0.0.18: vrrp 152.16.7.228 > 224.0.0.18: VRRPv2, Advertisement, vrid
 230, prio 101, authtype none, intvl 5s, length 20, addrs: 152.16.7.230
21:40:38.269976 IP (tos 0xc0, ttl 255, id 49799, offset 0, flags [none], proto VRRP (112),
 length 40)
    152.16.3.61 > 224.0.0.18: vrrp 152.16.3.61 > 224.0.0.18: VRRPv2, Advertisement, vrid
74, prio 101, authtype none, intvl 5s, length 20, addrs: 152.16.3.74
21:40:39.271020 IP (tos 0xc0, ttl 255, id 20946, offset 0, flags [none], proto VRRP (112),
 length 40)
    152.16.1.195 > 224.0.0.18: vrrp 152.16.1.195 > 224.0.0.18: VRRPv2, Advertisement, vrid
 193, prio 101, authtype none, intvl 5s, length 20, addrs: 152.16.1.193
21:40:42.270541 IP (tos 0xc0, ttl 255, id 16607, offset 0, flags [none], proto VRRP (112),
 length 40)
```

**Solution :**

Ensure that no other VM or machine is doing the broadcasting with the same VRID as your ESC HA configuration. Otherwise, it will cause the interferences to your ESC HA heartbeat thereby causing both the ESC HA VMs to stay in Backup state. Run the following command to find the VRID value of your ESC HA:

```
$ cat /etc/keepalived/keepalived.conf | grep virtual_router_id
```

If you find that the VRID of your ESC HA is used by other systems in your subnet, specify a value of `--kad_vri` in your bootvm.py argument.

# Cisco Elastic Services Controller HA is Running Slow

**Problem :**

In some OpenStack environment, the neutron configuration is different, the network throughput is extremely slow. In such cases, ESC VM needs to reduce the MTU for network interfaces from 1500 to 1450.

**Note** The MTU value for ESC's network interface should match the MTU for other network interfaces for VMs which manage components ESC directly communicates with, such as a VIM, an NFVO or an administrative jump box.

**Solution :**

Use the following steps to reduce the MTU value:

- Identify the interface interface you want to change and then go to the `/etc/sysconfig/network-scripts/ifcfg-ethX`. X represents the interface number you want to change.
- Use a text editor like VIM to add or edit the MTU items.

  ```
  mtu=1450
  ```

- Use the following command to restart the network interface:

  ```
  # network service restart
  i.e: sudo ifdown eth0 && sudo ifup eth0
  ```

# Unable to Access Cisco Elastic Services Controller HA with the VIP

Ensure that your VIP is in allowed_address_pairs of the ports of ESC instances.

### Before you begin

**Problem 1:**

Unable to access ESC HA with the VIP

ESC VIP floats across ESC HA instances and it redirects the connection to ESC Master.

**Verification and Troubleshooting:**

Check the following two items if your VIP does not work in the OpenStack environment:

- You must assign the VIP as allowed address pair to the original interface of ESC instances.
- Check the port of your ESC's interfaces and ensure that the allowed address pair configurations are correct.

### Procedure

**Step 1** Find the port UUID of your ESC interface for VIP Failover. In the following example, `152.16.3.76` is the IP:

```
$ neutron port-list | grep 152.16.3.76
| 80d7e031-04cd-4fb7-8f48-dcbcdbbd8685 |     | fa:16:3e:87:c9:e5 | {"subnet_id":
"7b2ce63b-eb20-4ff8-8d49-e46ee8dde0f5", "ip_address": "152.16.3.76"}
```

**Step 2** Check the allowed address pairs for the port and add the VIP to the allowed address pair of the port.

For example:

```
$ neutron port-show 80d7e031-04cd-4fb7-8f48-dcbcdbbd8685
+----------------------+-------------------------------------------------------------------------------------+
| Field                | Value
                    |
+----------------------+-------------------------------------------------------------------------------------+
| admin_state_up       | True
                    |
| allowed_address_pairs |
                    |
| binding:host_id      | my-ucs-64
                    |
| binding:profile      | {}
                    |
| binding:vif_details  | {"port_filter": true, "ovs_hybrid_plug": false}
                    |
| binding:vif_type     | ovs
                    |
| binding:vnic_type    | normal
                    |
| created_at           | 2017-12-13T21:16:56
                    |
| description          |
                    |
| device_id            | b895cd19-2491-4ac0-b4b5-087a4f76b701
                    |
| device_owner         | compute:None
                    |
| extra_dhcp_opts      |
                    |
| fixed_ips            | {"subnet_id": "7b2ce63b-eb20-4ff8-8d49-e46ee8dde0f5", "ip_address":
 "152.16.3.76"} |
| id                   | 80d7e031-04cd-4fb7-8f48-dcbcdbbd8685
                    |
| mac_address          | fa:16:3e:87:c9:e5
                    |
| name                 |
                    |
| network_id           | c7fafeca-aa53-4349-9b60-1f4b92605420
                    |
| port_security_enabled | True
                    |
```

```
| security_groups       | e8e9e10c-0e73-4e01-b364-115f785f787d
                  |
| status                | ACTIVE
                  |
| tenant_id             | d972982b511d4caa973f2ab71b58c2fe
                  |
| updated_at            | 2017-12-13T21:17:20
                  |
+---------------------+-------------------------------------------------------------------------------+


$ neutron port-update <your_esc_port_id> --allowed-address-pairs type=dict list=true
ip_address=<your_vip_address>
For Example:
$ neutron port-update 80d7e031-04cd-4fb7-8f48-dcbcdbbd8685 --allowed-address-pairs type=dict
 list=true ip_address=152.16.3.78
Updated port: 80d7e031-04cd-4fb7-8f48-dcbcdbbd8685

$ neutron port-show 80d7e031-04cd-4fb7-8f48-dcbcdbbd8685
+---------------------+-------------------------------------------------------------------------------+
| Field                 | Value
                  |
+---------------------+-------------------------------------------------------------------------------+
| admin_state_up        | True
                  |
| allowed_address_pairs | {"ip_address": "152.16.3.78", "mac_address": "fa:16:3e:87:c9:e5"}
                |
| binding:host_id       | my-ucs-64
                  |
| binding:profile       | {}
                  |
| binding:vif_details   | {"port_filter": true, "ovs_hybrid_plug": false}
                  |
| binding:vif_type      | ovs
                  |
| binding:vnic_type     | normal
                  |
| created_at            | 2017-12-13T21:16:56
                  |
| description           |
                  |
| device_id             | b895cd19-2491-4ac0-b4b5-087a4f76b701
                  |
| device_owner          | compute:None
                  |
| extra_dhcp_opts       |
                  |
| fixed_ips           | {"subnet_id": "7b2ce63b-eb20-4ff8-8d49-e46ee8dde0f5", "ip_address":
 "152.16.3.76"} |
| id                    | 80d7e031-04cd-4fb7-8f48-dcbcdbbd8685
                  |
| mac_address           | fa:16:3e:87:c9:e5
                  |
| name                  |
                  |
| network_id            | c7fafeca-aa53-4349-9b60-1f4b92605420
                  |
| port_security_enabled | True
                  |
| security_groups       | e8e9e10c-0e73-4e01-b364-115f785f787d
                  |
| status                | ACTIVE
                  |
| tenant_id             | d972982b511d4caa973f2ab71b58c2fe
```

```
                    |
| updated_at            | 2018-01-29T21:35:17
                    |
+---------------------+-------------------------------------------------------------------------------------+
```

### What to do next

**Other VM takes over the VIP IP address:**

In such scenarios, you must find out who took the VIP IP address. Once you know that, you release the IP address or select another IP address for your HA VIP. To ensure that the VIP you are using is safe and no one takes over it, you can create a port to occupy the VIP. To reserve the VIP address, run the following command:

```
$ neutron port-create <network_name> --fixed-ip ip_address=<your_vip_address> --name kad-vip


For example:

$ neutron port-create esc-net --fixed-ip ip_address=152.16.3.78 --name kad-vip
Created a new port:
+---------------------+-------------------------------------------------------------------------------------+
| Field                 | Value
                    |
+---------------------+-------------------------------------------------------------------------------------+
| admin_state_up        | True
                    |
| allowed_address_pairs |
                    |
| binding:host_id       |
                    |
| binding:profile       | {}
                    |
| binding:vif_details   | {}
                    |
| binding:vif_type      | unbound
                    |
| binding:vnic_type     | normal
                    |
| created_at            | 2018-01-29T21:53:33
                    |
| description           |
                    |
| device_id             |
                    |
| device_owner          |
                    |
| extra_dhcp_opts       |
                    |
| fixed_ips           | {"subnet_id": "7b2ce63b-eb20-4ff8-8d49-e46ee8dde0f5", "ip_address":
 "152.16.3.78"} |
| id                    | 3c037a4b-4245-4554-adf5-56ca6bbffa98
                    |
| mac_address           | fa:16:3e:4e:f2:96
                    |
| name                  | kad-vip
                      |
| network_id            | c7fafeca-aa53-4349-9b60-1f4b92605420
                    |
| port_security_enabled | True
                    |
| security_groups       | e8e9e10c-0e73-4e01-b364-115f785f787d
```

```
                     |
| status               | DOWN
                     |
| tenant_id            | d972982b511d4caa973f2ab71b58c2fe
                     |
| updated_at           | 2018-01-29T21:53:33
                     |
+--------------------+-------------------------------------------------------------------------------+
```

**VIP is in a different network than the management network:**

ESC HA configuration provides the following three configuration parameters (bootvm.py arguments):

- **`--ha_node_list`**: The list of IP addresses for HA nodes in the Active/Standby cluster. For ESC nodes with multiple network interfaces, these IPs should be the addresses in the network used for data synchronization. This argument is utilized for replication-based HA solution only. For example:

  ```
  --ha_node_list 192.168.0.12 192.168.0.22
  ```

- **`--kad_vip`** : The IP address for keepalived VIP (virtual IP) and the interface for keepalived VIP (ESC 2.2). For example:

  ```
  -kad_vip 10.20.0.194
  ```

  From ESC 2.2, the interface of VIP is specified in the following format:

  ```
  --kad_vip 10.20.0.194:eth2 or --kad_vip [2001:cc0:2020::fc]:eth2;
  ```

- **`--kad_vif`**:
  - The interface for keepalived VRRP and VIP (ESC 1.0 ~ ESC 2.1).
  - The interface for keepalived VRRP only if the VIP interface is specified in kad_vip (ESC 2.2). For example:

    ```
    --kad_vif eth0
    ```

    Use the VIP in a different interface than where network/interface of synchronization interface (kad_vif), --ha_node_list, and --kad_vif should be configured in one network/interface (eth1) and the --kad_vip in another network/interface (eth0).

    For example, for following bootvm.py commands, ESC HA uses eth1 (192.168.0.0/24) for data synchronization and heartbeat and uses eth0 (192.168.5.0/24) for VIP access. The VIP 192.168.5.200 floats between ESC nodes in the network (192.168.5.0/24).

    ```
    ./bootvm.py esc-ha-1 --image ESC-2_2_8_106 --net lab-net-0 esc-net --gateway_ip
    192.168.0.1 --ipaddr 192.168.5.239 192.168.0.239 --ha_node_list 192.168.0.239
    192.168.0.243 --kad_vip 192.168.5.200/24:eth0 --kad_vif eth1 --ha_mode drbd --route
     10.85.103.0/24:192.168.0.1:eth1 --avail_zone nova:my-ucs-26
    ./bootvm.py esc-ha-0 --image ESC-2_2_8_106 --net lab-net-0 esc-net --gateway_ip
    192.168.0.1 --ipaddr 192.168.5.243 192.168.0.243 --ha_node_list 192.168.0.239
    192.168.0.243 --kad_vip 192.168.5.200/24:eth0 --kad_vif eth1 --ha_mode drbd --route
     10.85.103.0/24:192.168.0.1:eth1 --avail_zone nova:my-ucs-27
    ```

# Status Check in Active VM Not Displaying the Status of BACKUP VM

The heartbeat of ESC HA is based on VRRP protocol. Based on VRRP protocol, ESC Active VM does not know the status of Backup VM instance. Hence, the status check also does not include the Backup VM status because the ESC service works fine as long as Active VM is working.

If you want to check the status of Backup VM, in ESC Active VM, run the following command:

```
$ sudo cat /proc/drbd
version: 8.4.10-1 (api:1/proto:86-101)
GIT-hash: a4d5de01fffd7e4cde48a080e2c686f9e8cebf4c build by abcbuild@, 2017-09-15 14:23:22
 1: cs:Connected ro:Primary/Secondary ds:UpToDate/UpToDate C r-----
    ns:5883476 nr:3012 dw:5886500 dr:378689 al:26 bm:0 lo:0 pe:0 ua:0 ap:0 ep:1 wo:f oos:0
```

Ensure that the `ro` shows `Primary/Secondary` and the `ds` shows `UpToDate/UpToDate`. It means that your Backup is connected to the Active VM and synchronization between Active and Backup is good. The following example shows when your Backup VM gets disconnected:

```
$ sudo cat /proc/drbd
version: 8.4.10-1 (api:1/proto:86-101)
GIT-hash: a4d5de01fffd7e4cde48a080e2c686f9e8cebf4c build by abcbuild@, 2017-09-15 14:23:22
 1: cs:WFConnection ro:Primary/Unknown ds:UpToDate/DUnknown C r-----
    ns:5888880 nr:3012 dw:5891912 dr:378689 al:26 bm:0 lo:0 pe:0 ua:0 ap:0 ep:1 wo:f oos:84
```