



Troubleshooting Cisco Elastic Services Controller Post Install

- [Troubleshooting Cisco Elastic Services Controller Post Install, on page 1](#)

Troubleshooting Cisco Elastic Services Controller Post Install

Following are a few examples of the issues you may face:

Before you begin

Problem:

If there are communication issues between ESC and other components within the management and orchestration stack, there may be several reasons for this.

Procedure

Step 1 Client makes the request, and the request hangs after it makes a call to the server.

Step 2 Notifications are not getting transmitted successfully between the components. Following is an example which shows an extract from the mona.log, where a D-MONA agent is unable to send notifications to the central ESC:

```
2022-03-01 16:44:31.355 [QuartzScheduler_Worker-2] [ruleName] :  
rule-VM_ALIVE-1-esc-etsi-sol3-vnf-info-001__319bb3d8-71c4-4957-8fe3-34240296d8e9__0:PostVmEventAction  
  execution, Post to url:https://x.x.x.x:8443/ESCManager/dmona/api/events/notif,  
  payload:rule-VM_ALIVE-1-esc-etsi-sol3-vnf-info-001__319bb3d8-71c4-4957-8fe3-34240296d8e9__0  
2022-03-01 16:44:41.488 [QuartzScheduler_Worker-2] Error during esc post execution:I/O error  
  on POST request for "https://x.x.x.x:8443/ESCManager/dmona/api/events/notif": Read timed  
  out; nested exception is java.net.SocketTimeoutException: Read timed out
```

What to do next

Solution:

Issue a curl command from the client to the server to check routing issues. Send a small or empty payload and ensure that the server receives it. Once you know that there are no issues with the routing, try the actual call that the client is attempting and expect the call to get failed.

When a system tries to transmit data over an interface, the data gets fragmented into packets based on the Maximum Transmission Unit (MTU) size configured on the interface.

For example, D-MONA communicates to the VNF on the orchestration network, but ESC through the management network. If the management network does not receive the notifications successfully, check the MTU size.

Often, a larger MTU fails to work in the event. Hence, there are intermediate systems that transmit the packets towards their destination. The intermediate systems are configured with smaller MTUs (as the packets are transferred through their ingress and egress interfaces).

To check the MTU size on an interface:

```
$ ifconfig eth0
ether: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 9000
       inet 172.29.0.33 netmask 255.255.240.0 broadcast 172.29.15.255
```

And to update it to a safe value of 1500:

```
$ sudo ifconfig eth0 mtu 1500
```

Now, retry the actual call that the client was failing with and check that it is received successfully by the server.