

Configure LACP C885A-M8 NIC with Netplan

Contents

[Introduction](#)

[Background Information](#)

[Prerequisites](#)

[Components Used](#)

[Network Cards used](#)

[Verify NIC is not in DPU mode](#)

[Step 1: Identify PCIe Addresses](#)

[Step 2: Check NIC Configuration](#)

[Step 3: Convert to NIC Mode](#)

[Step 4: Reboot and Verify Configuration](#)

[Configuring LACP](#)

[Step 1: Create a New Netplan Configuration File](#)

[Step 2: Apply the Configuration](#)

[Verify](#)

[1. Check Network Interface Details](#)

[2. Confirm Bonding Details](#)

[Troubleshooting](#)

[Related Information](#)

Introduction

This document describes information on how to configure LACP with Netplan on the C885A-M8 Server.

Background Information

From BMC, is not possible to do any modification on the network side, all changes need to be done from OS Guest.

Prerequisites

Components Used

UCS C885A M8 Rack Server

PID: UCSC-885A-M8-H11

Firmware Version: 1.0.28

Nexus 9k Switches

Ubuntu 22.04.5 LTS

Network Cards used

BlueField-3 P-Series DPU 400GbE/NDR VPI dual-port

BlueField-3 E-series SuperNIC 400GbE/NDR single port

Intel(R) Ethernet Network Adapter X710-T2L for OCP 3.0

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

Verify NIC is not in DPU mode

To ensure proper functionality, verify that the NIC is not in DPU mode. Link Aggregation Control Protocol (LACP) is unsupported when the NIC is in DPU mode. If necessary, convert it to NIC mode.

Step 1: Identify PCIe Addresses

Run the command to list the PCIe addresses for the desired ports:

```
cisco-tac@C885A-M8:~$ sudo mst start
Starting MST (Mellanox Software Tools) driver set
Loading MST PCI module - Success
Loading MST PCI configuration module - Success
Create devices
Unloading MST PCI module (unused) - Success
```



Note: This is not required but can help in designating the correct device when making changes using mlxconfig

```
cisco-tac@C885A-M8:~$ sudo mst status -v
```

```
MST modules:
```

```
-----
```

```
MST PCI module is not loaded
```

```
MST PCI configuration module loaded
```

```
PCI devices:
```

```
-----
```

```
DEVICE_TYPE MST PCI RDMA NET NUMA
```

```
BlueField3(rev:1) /dev/mst/mt41692_pciconf9 f1:00.0 mlx5_6 net-ens205f0 1
BlueField3(rev:1) /dev/mst/mt41692_pciconf8 c5:00.0 mlx5_7 net-ens206f0 1
BlueField3(rev:1) /dev/mst/mt41692_pciconf7 a4:00.0 mlx5_11 net-ens208f0 1
BlueField3(rev:1) /dev/mst/mt41692_pciconf6 97:00.0 mlx5_10 net-ens207f0 1
BlueField3(rev:1) /dev/mst/mt41692_pciconf5.1 91:00.1 mlx5_9 net-ens214f1np1 1
BlueField3(rev:1) /dev/mst/mt41692_pciconf5 91:00.0 mlx5_8 net-ens214f0np0 1
BlueField3(rev:1) /dev/mst/mt41692_pciconf4 69:00.0 mlx5_0 net-ens202f0 0
BlueField3(rev:1) /dev/mst/mt41692_pciconf3 4b:00.0 mlx5_3 net-ens201f0 0
BlueField3(rev:1) /dev/mst/mt41692_pciconf2.1 45:00.1 mlx5_2 net-ens214f1np1 0
```

```
BlueField3(rev:1) /dev/mst/mt41692_pciconf2 45:00.0 mlx5_1 net-ens211f0np0 0
BlueField3(rev:1) /dev/mst/mt41692_pciconf1 2b:00.0 mlx5_5 net-ens203f0 0
BlueField3(rev:1) /dev/mst/mt41692_pciconf0 09:00.0 mlx5_4 net-ens204f0 0
```

This command provides a detailed list of devices, allowing you to identify the appropriate MST and PCI identifiers.

On this scenario NET **net-ens214f1np1** and NET **net-ens211f0np0** are used.

Step 2: Check NIC Configuration

Using the identified MST or PCI identifier, verify the current NIC configuration with one of the commands:

```
sudo mlxconfig -d 0000:<PCI_ID> -e q | grep 'INTERNAL_CPU_MODEL\|EXP_ROM_UEFI_ARM_ENABLE\|INTERNAL_CPU_
```

or

```
sudo mlxconfig -d /dev/<MST_DEVICE> -e q | grep 'INTERNAL_CPU_MODEL\|EXP_ROM_UEFI_ARM_ENABLE\|INTERNAL_
```

```
cisco-tac@C885A-M8:~$ sudo sudo mlxconfig -d dev/mst/mt41692_pciconf1 -e q | grep \ 'INTERNAL_CPU_MODEL
Configurations: Default Current Next Boot
INTERNAL_CPU_MODEL EMBEDDED_CPU(1) EMBEDDED_CPU(1) EMBEDDED_CPU(1)
INTERNAL_CPU_OFFLOAD_ENGINE ENABLED(0) ENABLED(0) ENABLED(0)
EXP_ROM_UEFI_ARM_ENABLE True(1) True(1) True(1)
```



Note: The CPU model and ARM UEFI enable variables confirm to us that the card is DPU mode capable

In the example above:

INTERNAL_CPU_OFFLOAD_ENGINE is set to **ENABLED(0\)** for the Current and Next Boot configurations.

This indicates that the NIC is currently operating in DPU mode and needs to be converted to NIC mode to proceed with LACP configuration.

Step 3: Convert to NIC Mode

To convert the NIC to NIC mode, issue the command:

```
sudo mlxconfig -d 0000:<PCI_ID> s INTERNAL_CPU_OFFLOAD_ENGINE=0
```

```
cisco-tac@C885A-M8:~$ sudo mlxconfig -d 0000:45:00.0 s INTERNAL_CPU_OFFLOAD_ENGINE=0
```

```
Device #1:
```

```
-----
```

```
Device type: BlueField3
```

```
Name: 900-9D3B6-00SN-A_Ax
```

```
Description: NVIDIA BlueField-3 B3240 P-Series Dual-slot FHHL DPU; 400GbE / NDR IB (default mode); Dual
```

```
Device: 0000:45:00.0
```

```
Configurations: Next Boot New
```

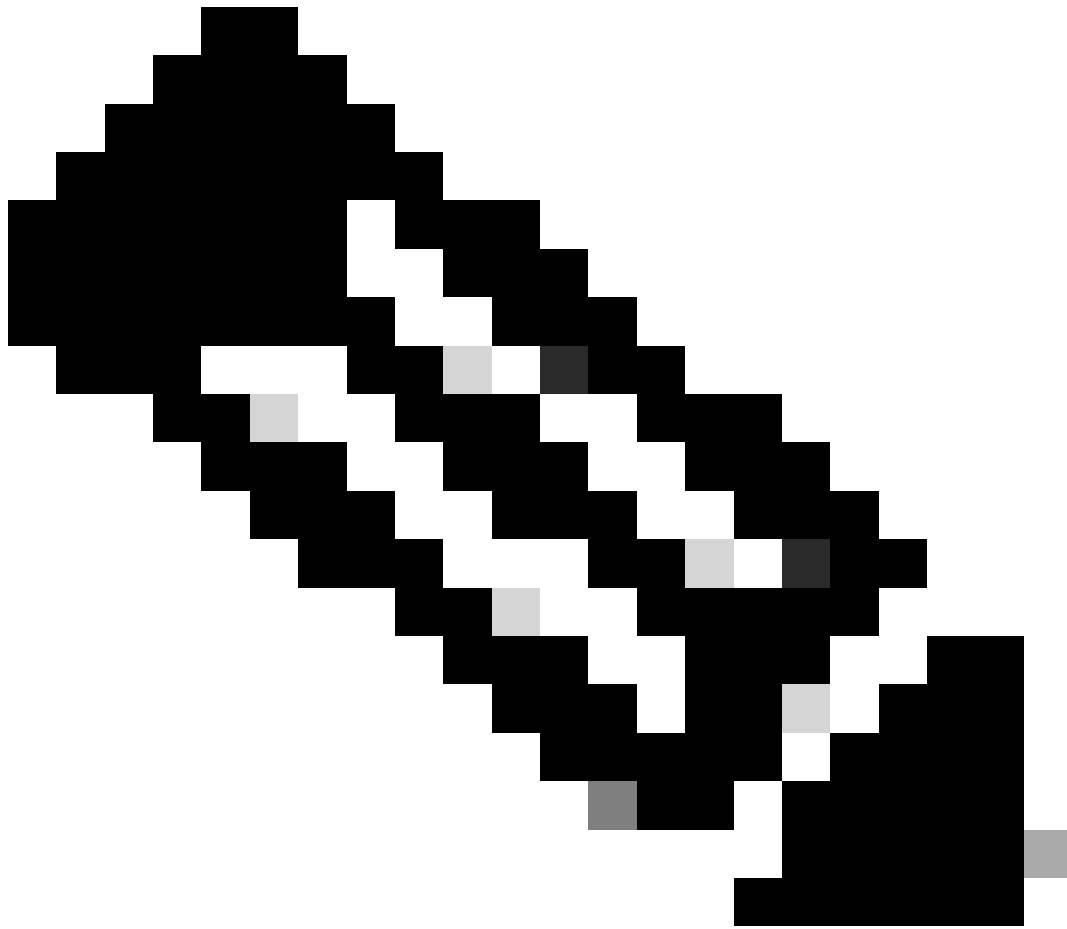
```
INTERNAL_CPU_OFFLOAD_ENGINE ENABLED(0) DISABLED(1)
```

```
Apply new Configuration? (y/n) [n] : y
```

```
Applying... Done!
```

```
-I- Please reboot machine to load new configurations.
```

```
cisco-tac@C885A-M8:~$ sudo init 0
```



Note: With **sudo init 0**, shuts down the host to ensure that the configuration get loaded appropriately.

Step 4: Reboot and Verify Configuration

After executing the `sudo init 0` command, the host system is going to shut down. Use the Baseboard Management Controller (BMC) to restart the host OS.

Once the system is back online, verify that the configuration changes have been successfully applied by running the same verification commands as shown earlier:

```
sudo mlxconfig -d 0000:<PCI_ID> -e q | grep 'INTERNAL_CPU_MODEL\|EXP_ROM_UEFI_ARM_ENABLE\|INTERNAL_CPU_
```

or

```
sudo mlxconfig -d /dev/<MST_DEVICE> -e q | grep 'INTERNAL_CPU_MODEL\|EXP_ROM_UEFI_ARM_ENABLE\|INTERNAL_
```

If the configuration has been successfully applied, you can proceed with creating the desired port-channel, bond, or LAG configuration.

Configuring LACP

Step 1: Create a New Netplan Configuration File

The current standard for configuring interfaces on Ubuntu is Netplan. To begin, check the existing Netplan YAML configuration files by running:

```
ls /etc/netplan/
```

By default, there is typically only one YAML file in the /etc/netplan/ directory. Netplan configuration files uses the naming convention `##-<relevant name>.yaml`, where `##` is a number between 01 and 99. Consider that the configuration files with lower numbers are overwritten by higher-numbered files if there are conflicts.

To ensure your changes are persistent and organized, it is recommended to create a new YAML file instead of editing the default one. If you modify the default file, there is a risk that changes do not persist after a reboot. For this guide, we are going to create a new file named

51-lACP-config.yaml.

To create the new file, use the command:

```
sudo nano /etc/netplan/51-lACP-config.yaml
```

Ensure that your file numbering does not conflict with existing files and uses the proper hierarchy to avoid unexpected behavior.

This is an example of a Netplan YAML configuration file (**51-lACP-config.yaml**) for setting up LACP with VLANs.

```
cisco-tac@C885A-M8:~$ cat /etc/netplan/51-lACP-config.yaml
network:
  ethernets:
    ens211f0np0:
      mtu: 9000
    ens211f1np1:
      mtu: 9000
```

```

bonds:
bond0:
mtu: 9000
dhcp4: false
dhcp6: false
interfaces:
- ens211f0np0
- ens211f1np1
parameters:
mode: 802.3ad
mii-monitor-interval: 100
vlans:
bond0.2001:
id: 2001
link: bond0

```

Value	Consideration	Remarks
MTU Settings	For ens211f0np0 and ens211f1np1, the MTU is set to 9000.	It is recommended to set the MTU to the highest value needed to avoid potential MTU issues in the future. Always specify the MTU whenever you mention an interface
LACP Mode	The mode parameter is set to 802.3ad	This is the required mode for LACP. Other bonding modes exist but are not suitable for LACP
VLAN Configuration	The bond0.2001 section creates a sub-interface for sending multiple VLANs through the bond. The id field specifies the VLAN ID (in this case, 2001). The link field associates the sub-interface with the bonded interface (bond0)	If you have multiple bonds in the same YAML file, this ensures the correct configuration is applied to the correct bond
IP Address and Routes	The addresses field specifies the IP address and subnet mask for the VLAN (example, 10.10.200.33/27)	The routes section allows you to define any routes that this interface handles. Although the example includes the route for 10.10.200.32/27, it is redundant in this lab setup but demonstrates how to add multiple routes
DNS Servers	The nameservers section allows you to specify DNS servers (addresses) and search domains (search)	Need to provide DNS server details based on work environment.
MTU for VLAN	The mtu field in the VLAN configuration is set to 9000	You can configure different MTU values on a per-VLAN basis if needed
Version Field	The version field at the end of the file can be included and set to 2	Ensure this is indented with two spaces to maintain YAML syntax

Step 2: Apply the Configuration

To apply the new configuration, run:

```
sudo netplan apply
```

If you encounter permission warnings, adjust the file permissions:

```
sudo chmod 600 /etc/netplan/51-lACP-config.yaml
sudo netplan apply
```

```
cisco-tac@C885A-M8:~$ sudo netplan apply
** (generate:140701): WARNING **: 17:10:58.313: Permissions for /etc/netplan/51-lACP-config.yaml are too
```

```
cisco-tac@C885A-M8:~$ ls -l /etc/netplan/
total 16
-rw----- 1 root root 869 Apr 24 14:26 50-cloud-init.yaml
-rw-r--r-- 1 root root 586 Apr 25 17:10 51-lACP-config.yaml
```

```
cisco-tac@C885A-M8:~$ sudo chmod 600 /etc/netplan/51-lACP-config.yaml
cisco-tac@C885A-M8:~$ ls -l /etc/netplan/
total 16
-rw----- 1 root root 869 Apr 24 14:26 50-cloud-init.yaml
-rw----- 1 root root 586 Apr 25 17:10 51-lACP-config.yaml
```

```
cisco-tac@C885A-M8:~$ sudo netplan apply
```

Verify

After applying the Netplan configuration and restarting the system, confirm that the LACP setup is functioning as expected using the commands.

1. Check Network Interface Details

Use this command to display the status of the network interfaces:

```
cisco-tac@C885A-M8:~$ ip link show
4: ens211f0np0: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 9000 qdisc mq master bond0 state UP mode DE
link/ether 86:a2:e3:01:3d:f2 brd ff:ff:ff:ff:ff:ff permaddr 5c:25:73:3a:9a:fc
altname enp69s0f0np0
5: ens211f1np1: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 9000 qdisc mq master bond0 state UP mode DE
link/ether 86:a2:e3:01:3d:f2 brd ff:ff:ff:ff:ff:ff permaddr 5c:25:73:3a:9a:fd
```

```
altname enp69s0f1np1
```

Ensure the interfaces (ens211f0np0 and ens211f1np1) show:

- Master set to bond0, confirming that they belong to the bonded interface.
- State as UP, confirming that the interfaces are active.
- Confirm the mtu value is set to 9000 as specified in the YAML file.
- Verify that both interfaces have valid hardware addresses (permaddr).

2. Confirm Bonding Details

Use this command to display the bonding configuration and ensure proper functionality of LACP:

```
cisco-tac@C885A-M8:~$cat /proc/net/bonding/bond0
Ethernet Channel Bonding Driver: v5.15.0-141-generic
```

```
Bonding Mode: IEEE 802.3ad Dynamic link aggregation
Transmit Hash Policy: layer2 (0)
MII Status: up
MII Polling Interval (ms): 100
Up Delay (ms): 0
Down Delay (ms): 0
Peer Notification Delay (ms): 0
```

```
802.3ad info
LACP active: on
LACP rate: slow
Min links: 0
Aggregator selection policy (ad_select): stable
```

```
Slave Interface: ens211f1np1
MII Status: up
Speed: 1000000000 Mbps
Duplex: full
Link Failure Count: 0
Permanent HW addr: 5c:25:73:3a:9a:fd
Slave queue ID: 0
Aggregator ID: 1
Actor Churn State: none
Partner Churn State: none
Actor Churned Count: 0
Partner Churned Count: 0
```

```
Slave Interface: ens211f0np0
MII Status: up
Speed: 1000000000 Mbps
Duplex: full
Link Failure Count: 0
Permanent HW addr: 5c:25:73:3a:9a:fc
Slave queue ID: 0
Aggregator ID: 1
Actor Churn State: none
Partner Churn State: none
```

Actor Churned Count: 0
Partner Churned Count: 0

Ensure the Bonding Mode is set to IEEE 802.3ad Dynamic link aggregation:

- LACP active: on indicates that LACP is enabled.
- LACP rate: slow reflects the configured LACP rate, though it can differ based on the setup.
- Verify the Slave Interfaces: Both ens211f0np0 and ens211f1np1 can appear as slave interfaces under the bond.
- MII Status: up confirms the links are active.
- The Permanent HW addr values must match the hardware addresses of the respective interfaces.
- Ensure both slave interfaces are part of the same Aggregator ID (for example., Aggregator ID: 1).

Troubleshooting

If the verification results do not match expectations:

- Check the Netplan YAML configuration file for errors or missing parameters.
- Ensure proper LACP support and configuration on the upstream switch.
- Confirm that the sudo netplan apply command completed successfully

Related Information

[TACDCN-2018](#)

[Cisco UCS C885A M8 Rack Server Data Sheet](#)

[UCS C885A M8 Rack Server Spec Sheet](#)