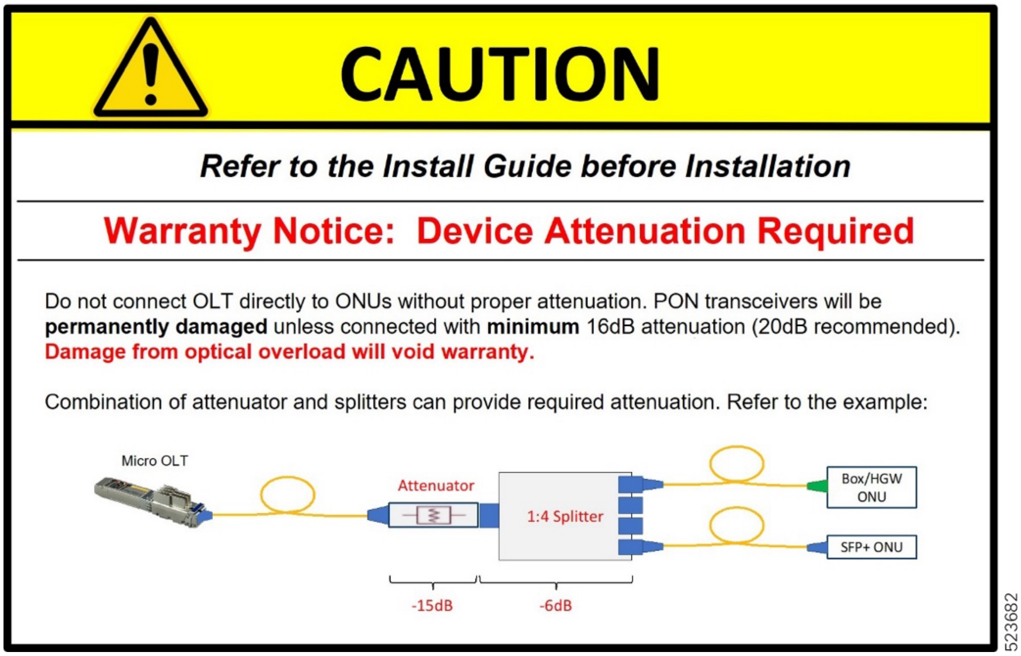
Cisco Routed PON Installation Guide



CAUTION

Refer to the Install Guide before Installation

Warranty Notice: Device Attenuation Required

Do not connect OLT directly to ONUs without proper attenuation. PON transceivers will be **permanently damaged** unless connected with **minimum** 16dB attenuation (20dB recommended). **Damage from optical overload will void warranty.**

Combination of attenuator and splitters can provide required attenuation. Refer to the example:

Table of Contents

List of Figures

Solution Overview

The Cisco Routed PON is Cisco's management solution for PON Networks. The Routed PON architecture is shown in Figure 1 and consists of the Routed PON Manager graphical user interface, Routed PON Netconf Server, and Routed PON Controller. Together these components provide a complete network management solution for provisioning and monitoring Routed PON OLT devices, as well as the subtended third-party ONU devices and third-party ONUs compliant with the 10G-EPON and XGS-PON standards.

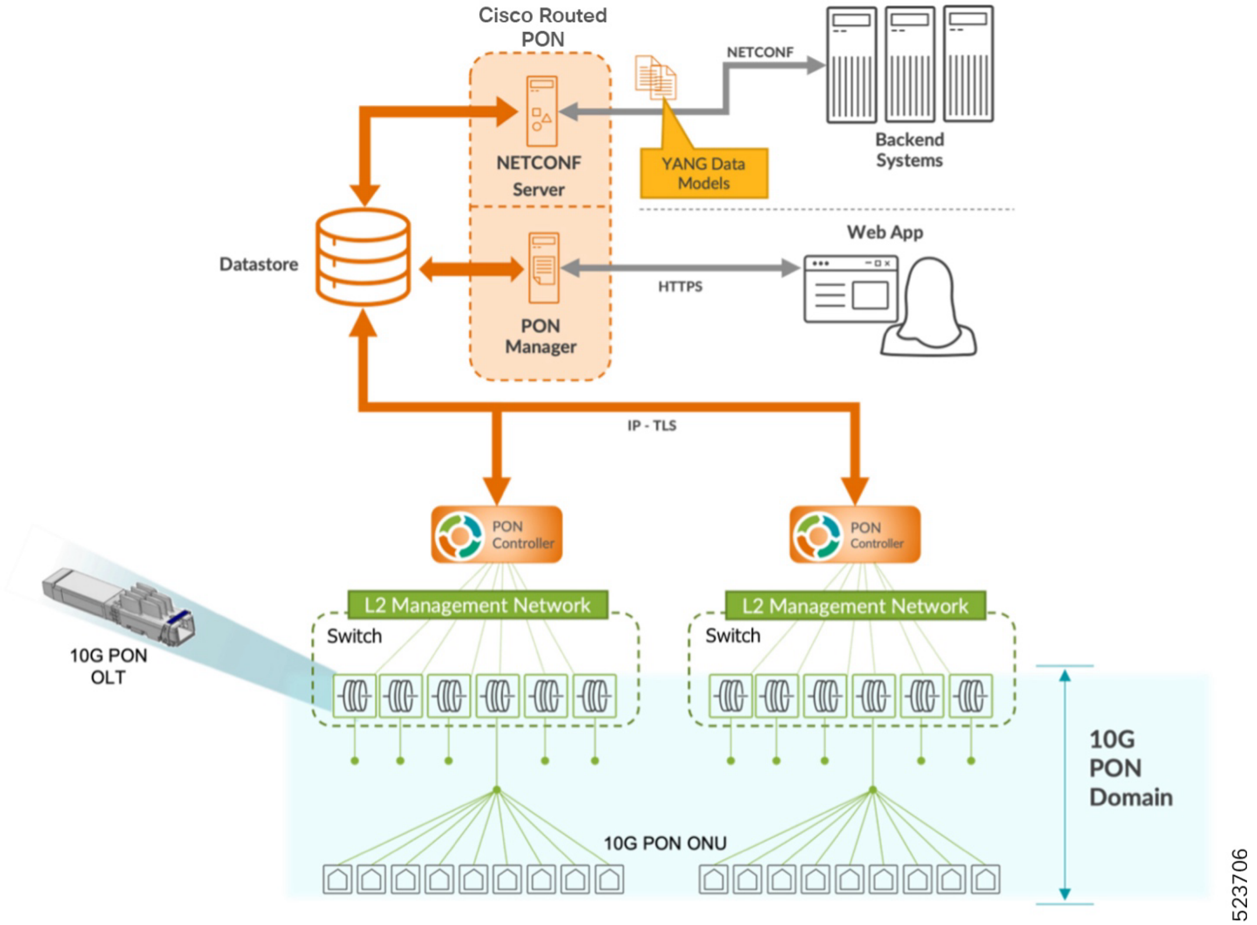


Figure 1 – Cisco Routed PON Management System Architecture

Cisco Routed PON Manager

The Cisco Routed PON Manager is a single-page web application (Web App) and an accompanying REST API that provides a graphical user interface for managing the PON Network. The Web App is built on the Angular web application framework, which provides an HTML and JavaScript front-end user interface. The REST API accompanies the Web App for the purposes of providing access to MongoDB for managing Routed PON Manager users and the PON Network.

The Routed PON Manager has the following features:

- Alarm management.
- Dashboard view with a summary of PON network conditions.
- Device monitoring and statistics.
- Device provisioning and management.
- Logging for diagnostics and troubleshooting.
- Routed PON Controller database management.
- Routed PON Manager user management.
- Polyglot graphical OMCI (and future 10G EPON OAM) service configuration tool.
- Service configuration, including VLANs, Service Level Agreements (SLAs), 802.1X Authentication, and DHCP Relay.

Note that the REST API should only be used with the Routed PON Manager. At this time, the REST API is not a published interface and will change in future releases. Future versions for Routed PON Manager may make the REST API available directly to customer applications.

Cisco Routed PON Controller

The Cisco Routed PON Controller is a stateless management controller and device driver application for configuring and monitoring the end points in a Cisco OLT PON network. MongoDB serves as the northbound application programming interface for the PON Controller. The PON Controller applies configuration to OLT and ONU devices from documents stored in MongoDB. The Routed PON Controller uses IEEE 1904.2 packets (Layer 2 Ethernet) to communicate with the OLT and ONU devices. The PON Controller also collects state information, statistics, and logs from devices and reports the information to higher layer applications through MongoDB. The PON Manager and Netconf interfaces manage the PON Controller through MongoDB\MongoDB Datastore

The Mongo database (MongoDB) provides the datastore for the Cisco Routed PON Management System. The MongoDB datastore contains all the configuration, state, statistics, alarms, and logging data for the devices in the PON network. Northbound interfaces, such as the Routed PON Manager, Routed PON Netconf Server, and customer applications interface with MongoDB to provision and retrieve monitoring information for devices in the PON network. MongoDB serves as the interface between the PON Manager, Netconf, and the PON Controller.

Provisioning data generally flows "downstream" through the management network. The PON Manager and Netconf interfaces write device configuration to MongoDB. The PON Controller reads the configuration data from MongoDB and programs the OLT and ONU devices accordingly.

Monitoring data, including device state, statistics, alarms, and logging, is collected and flows "upstream" through the management network. The PON Controller periodically collects state information from devices in the PON network and writes the monitoring data to MongoDB. The PON Manager reads the monitoring data from MongoDB for display in the Web App.

Server Sizing

Lab Testing

It is recommended to run all the Routed PON Management applications on a single VM (or single server).

- Management Applications
  - PON Manager
  - MongoDB
  - Netconf server
- Server Hardware Requirements
  - 2 vCPUs
  - 8 GB RAM
  - 20 GB disk space
- PON Size
  - 4 OLTs
  - 32 ONUs

Production Deployments

It is recommended to run each of the Routed PON Management applications on separate VMs (or servers). The following table provides guidelines for deployments.

Note: The sizing requirements are for the Application process only and does not include the base OS (Ubuntu 20.04).

Application	Subscriber Size	
	3072 Subscriber	**100K Subscribers**
PON Manager	2 vCPU 8 GB RAM 1 GB disk space	4 vCPU 16 GB RAM 1 GB disk space
Netconf	1 Instance: 2 vCPU 1 GB RAM 50 MB disk space	3 x Netconf Instances: 6 vCPU 3 GB RAM 150 MB disk space
MongoDB	2 vCPU 8 GB RAM 10 GB disk space	16 vCPU 64 GB RAM 100 GB disk space

[1] This is an estimate. Please contact Cisco Support for details

Installation for Lab Testing – Single VM

For initial lab and proof of concept testing, we recommend building an Ubuntu VM or system described in the Technical Note “Installation Guide for Ubuntu VM using VirtualBox on Windows”.

After the VM is created, we recommend connecting the PC and OLTs into a switch.

System Requirements

Before beginning the installation, ensure the system meets this list of requirements:

- Ubuntu 20.04
    - Verify system is running ubuntu version 20.04

      ```
      user@system:~$ lsb_release -a
      No LSB modules are available.
      Distributor ID: Ubuntu
      Description:    Ubuntu 20.04.6 LTS
      Release:        20.04
      Codename:       focol
      ```
- Two Ethernet interfaces
    - Ethernet interface for connectivity to router
    - Interface for Internet connectivity
        - Verify ubuntu system has connectivity to Internet
        - This is required to downloaded required python dependencies and the MongoDB Community Server.
- Firefox version 70.0.0 or higher
- MongoDB version v4.00 or higher is required. With Ubuntu 20.04, Mongo DB version v4.4 or higher is required.
    - Note: This will automatically be installed by the installation scripts

New Installation – Single VM

Note: If you are upgrading an existing installation, please skip to the next section with title “Upgrading Existing Installation” on the next page.

This section describes the installation for a new installation. If you are already running the Cisco PON Domain Manager, please skip to the next section, Upgrading Existing Installations.

For a new installation, run the installation script from the ubuntu command line:

```
user@system:<install_dir>/MCMS-R4.0.0-UB2004$ sudo ./install.sh -e <eth_interface>
```

This script will do the following:

- Install MongoDB (downloads from repo.mongo.org)
- Install Routed PON Manager
- Install Routed PON Netconf Server

The available options for the installation script (install.sh) are:

Required:

- -e <eth_interface> The ethernet vlan interface used by the Pon Controller and UMT relay

Optional:

- -d <database_ip> The database IP for MongoDB (default = 127.0.0.1)
- -n <database_name> The default database used by Pon Mgr (default = tibit_pon_controller)
- -m Install only the Pon Manager/MongoDB/Netconf Server

Creating Initial Account in PON Manager

This section describes how to create the initial login account in PON Manager. This account will have administrative privileges and supports the ability to create new accounts for access into the PON Manager.

- Connect to PON Manager using any of the IP addresses configured on the ubuntu system
- Create an Account. Ensure the password meets the Password Requirements listed.

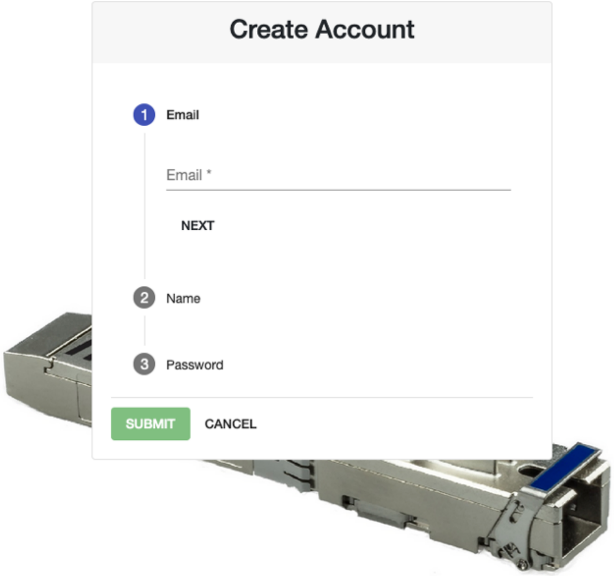
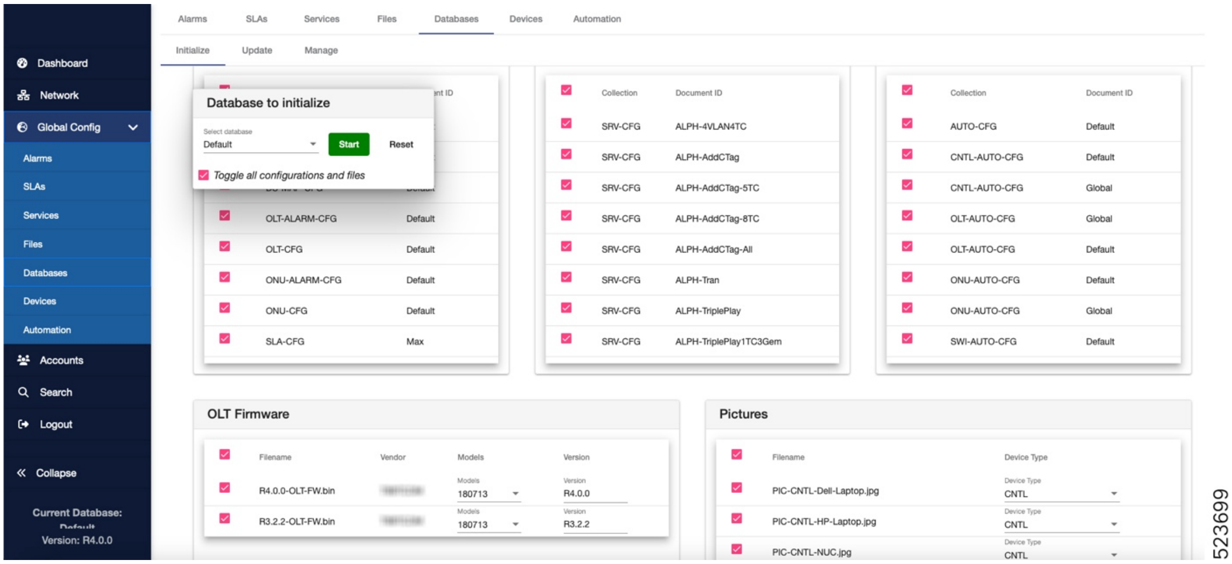


Figure 2 - Create Initial Login Account

Initialize Database

This section describes how to Initialize the Database after the upgrade process completes. Some items (specifically OLT Firmware) will need to be added into the MongoDB.

- Connect to the PON Manager
- Select the Global Config button on the main menu.
- Select Databases
- Then select the **Initialize** tab (see screenshot below)
- Select the Database to initialize (i.e. Default)
- Check (or uncheck) all the boxes for the information wanted to copy into the Database
- Click **Sart** in the **Database to initialize**



Database to initialize

Select database: Default

☑ Toggle all configurations and files

☑		…nt ID
☑	OLT-ALARM-CFG	Default
☑	OLT-CFG	Default
☑	ONU-ALARM-CFG	Default
☑	ONU-CFG	Default
☑	SLA-CFG	Max

☑	Collection	Document ID
☑	SRV-CFG	ALPH-4VLAN4TC
☑	SRV-CFG	ALPH-AddCTag
☑	SRV-CFG	ALPH-AddCTag-5TC
☑	SRV-CFG	ALPH-AddCTag-8TC
☑	SRV-CFG	ALPH-AddCTag-All
☑	SRV-CFG	ALPH-Tran
☑	SRV-CFG	ALPH-TriplePlay
☑	SRV-CFG	ALPH-TriplePlay1TC3Gem

☑	Collection	Document ID
☑	AUTO-CFG	Default
☑	CNTL-AUTO-CFG	Default
☑	CNTL-AUTO-CFG	Global
☑	OLT-AUTO-CFG	Global
☑	OLT-AUTO-CFG	Default
☑	ONU-AUTO-CFG	Default
☑	ONU-AUTO-CFG	Global
☑	SWI-AUTO-CFG	Default

OLT Firmware

☑	Filename	Vendor	Models	Version
☑	R4.0.0-OLT-FW.bin	[redacted]	180713	R4.0.0
☑	R3.2.2-OLT-FW.bin	[redacted]	180713	R3.2.2

Pictures

☑	Filename	Device Type
☑	PIC-CNTL-Dell-Laptop.jpg	CNTL
☑	PIC-CNTL-HP-Laptop.jpg	CNTL
☑	PIC-CNTL-NUC.jpg	CNTL

Figure 3: Initialize Database

Upgrading Existing Installations

Upgrade Single VM system with PON Manager and MongoDB

This section describes the installation for an existing installation of the PON Domain Manager. If this is a new installation, please follow the steps in the previous section, New Installation.

For upgrading to R4.0.0 (from any previous version), run the following upgrade script:

```
user@system:<install_dir>/R4.0.0-Cisco-UB200$ sudo ./upgrade.sh
```

This script will do the following:

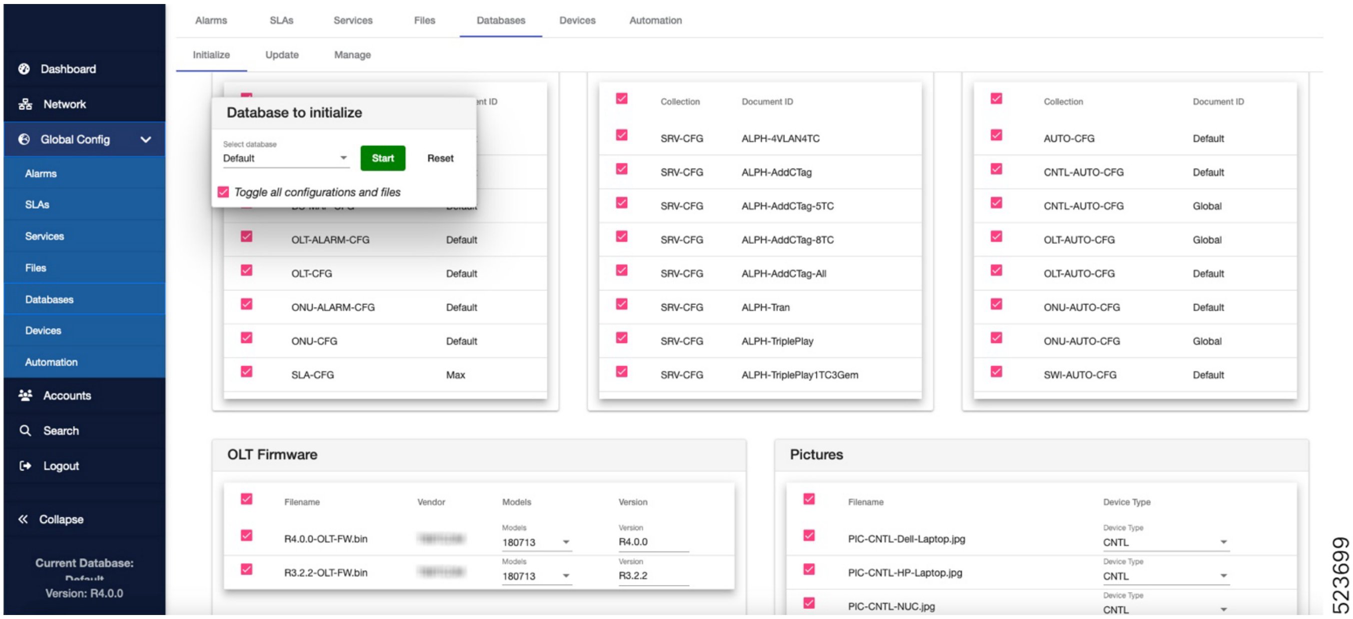
- Upgrade to R4.0.0 PON Manager
- Upgrade to R4.0.0 PON Controller
- Upgrade to R4.0.0 Netconf server (if installed)

Post Upgrade tasks

Initialize Database

This section describes how to Initialize the Database after the upgrade process completes. Some items (specifically OLT Firmware) will need to be added into the MongoDB.

- Connect to the PON Manager
- Select the Global Config button on the main menu.
- Select Databases
- Then select the **Initialize** tab (see screenshot below)
- Select the Database to initialize (i.e. Default)
- Check (or uncheck) all the boxes for the information wanted to copy into the Database
- Click **Sart** in the **Database to initialize**



Database to initialize

Select database: Default

Start Reset

☑ *Toggle all configurations and files*

☑		…nt ID
☑	OLT-ALARM-CFG	Default
☑	OLT-CFG	Default
☑	ONU-ALARM-CFG	Default
☑	ONU-CFG	Default
☑	SLA-CFG	Max

☑	Collection	Document ID
☑	SRV-CFG	ALPH-4VLAN4TC
☑	SRV-CFG	ALPH-AddCTag
☑	SRV-CFG	ALPH-AddCTag-5TC
☑	SRV-CFG	ALPH-AddCTag-8TC
☑	SRV-CFG	ALPH-AddCTag-All
☑	SRV-CFG	ALPH-Tran
☑	SRV-CFG	ALPH-TriplePlay
☑	SRV-CFG	ALPH-TriplePlay1TC3Gem

☑	Collection	Document ID
☑	AUTO-CFG	Default
☑	CNTL-AUTO-CFG	Default
☑	CNTL-AUTO-CFG	Global
☑	OLT-AUTO-CFG	Global
☑	OLT-AUTO-CFG	Default
☑	ONU-AUTO-CFG	Default
☑	ONU-AUTO-CFG	Global
☑	SWI-AUTO-CFG	Default

OLT Firmware

☑	Filename	Vendor	Models	Version
☑	R4.0.0-OLT-FW.bin	[redacted]	180713	R4.0.0
☑	R3.2.2-OLT-FW.bin	[redacted]	180713	R3.2.2

Pictures

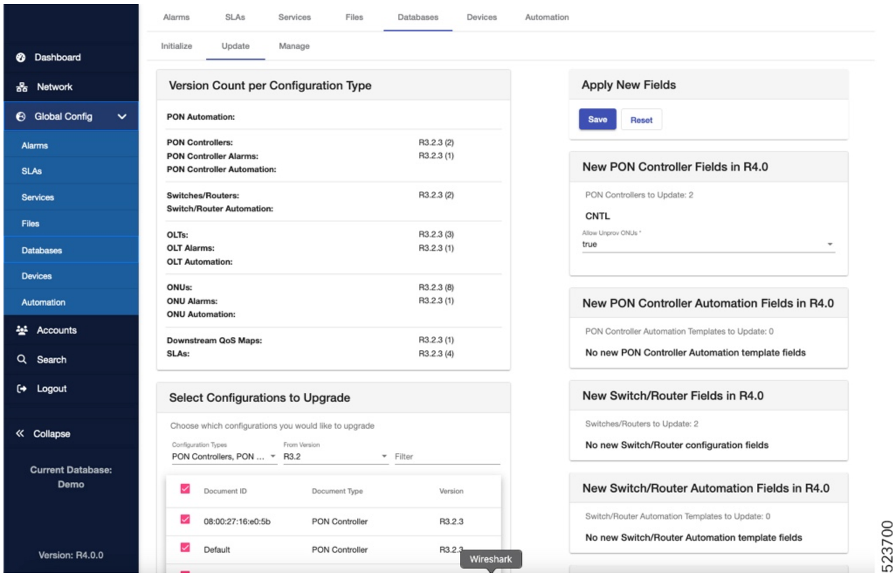
☑	Filename	Device Type
☑	PIC-CNTL-Dell-Laptop.jpg	CNTL
☑	PIC-CNTL-HP-Laptop.jpg	CNTL
☑	PIC-CNTL-NUC.jpg	CNTL

Figure 4: Initialize Database

Database Update

This section describes how to Update the Database after the upgrade process completes. An upgrade re-uses the existing previously installed Mongo DB. Some items will need updating after the upgrade.

- Connect to the PON Manager
- Select the Global Config button on the main menu.
- Select **Databases** and then select the **Update** tab (see screenshot below)
- Under **Select Configuration to Upgrade**, check all the boxes
- Under **From Version** select **R3.2**
- Click **Save** in the **Apply New Fields**



Version Count per Configuration Type

PON Automation:	
PON Controllers:	R3.2.3 (2)
PON Controller Alarms:	R3.2.3 (1)
PON Controller Automation:	
Switches/Routers:	R3.2.3 (2)
Switch/Router Automation:	
OLTs:	R3.2.3 (3)
OLT Alarms:	R3.2.3 (1)
OLT Automation:	
ONUs:	R3.2.3 (8)
ONU Alarms:	R3.2.3 (1)
ONU Automation:	
Downstream QoS Maps:	R3.2.3 (1)
SLAs:	R3.2.3 (4)

Select Configurations to Upgrade

Choose which configurations you would like to upgrade

Configuration Types: PON Controllers, PON …

From Version: R3.2

Filter

☑	Document ID	Document Type	Version
☑	08:00:27:16:e0:5b	PON Controller	R3.2.3
☑	Default	PON Controller	R3.2.3

Apply New Fields

Save Reset

New PON Controller Fields in R4.0

PON Controllers to Update: 2

CNTL

Allow Unprov ONUs *: true

New PON Controller Automation Fields in R4.0

PON Controller Automation Templates to Update: 0

No new PON Controller Automation template fields

New Switch/Router Fields in R4.0

Switches/Routers to Update: 2

No new Switch/Router configuration fields

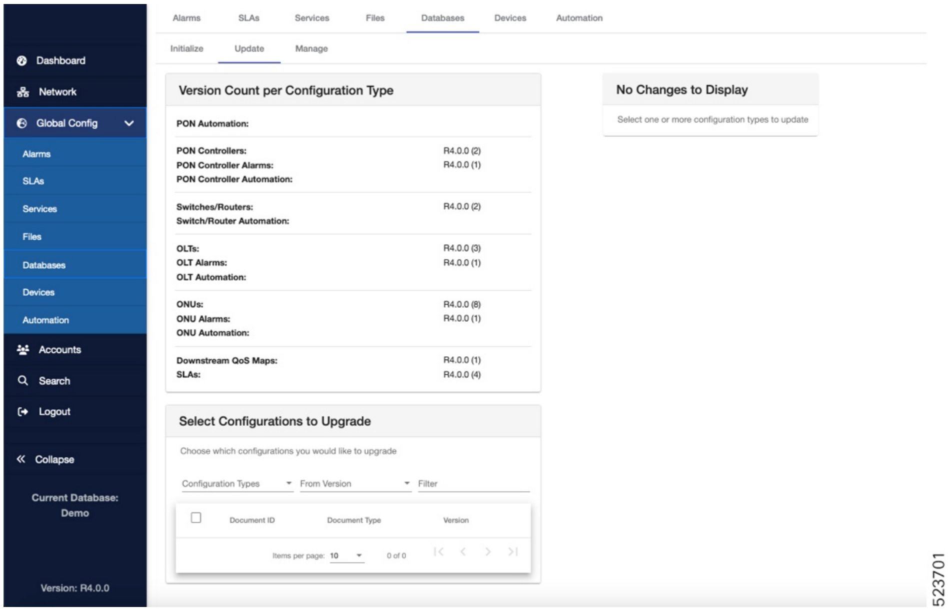
New Switch/Router Automation Fields in R4.0

Switch/Router Automation Templates to Update: 0

No new Switch/Router Automation template fields

Figure 5 – Apply New Fields

After selecting **Save** , the list of **Configuration Types** should update to R4.0.0



Version Count per Configuration Type

PON Automation:	
PON Controllers:	R4.0.0 (2)
PON Controller Alarms:	R4.0.0 (1)
PON Controller Automation:	
Switches/Routers:	R4.0.0 (2)
Switch/Router Automation:	
OLTs:	R4.0.0 (3)
OLT Alarms:	R4.0.0 (1)
OLT Automation:	
ONUs:	R4.0.0 (8)
ONU Alarms:	R4.0.0 (1)
ONU Automation:	
Downstream QoS Maps:	R4.0.0 (1)
SLAs:	R4.0.0 (4)

No Changes to Display

Select one or more configuration types to update

Select Configurations to Upgrade

Choose which configurations you would like to upgrade

Configuration Types | From Version | Filter

□	Document ID	Document Type	Version

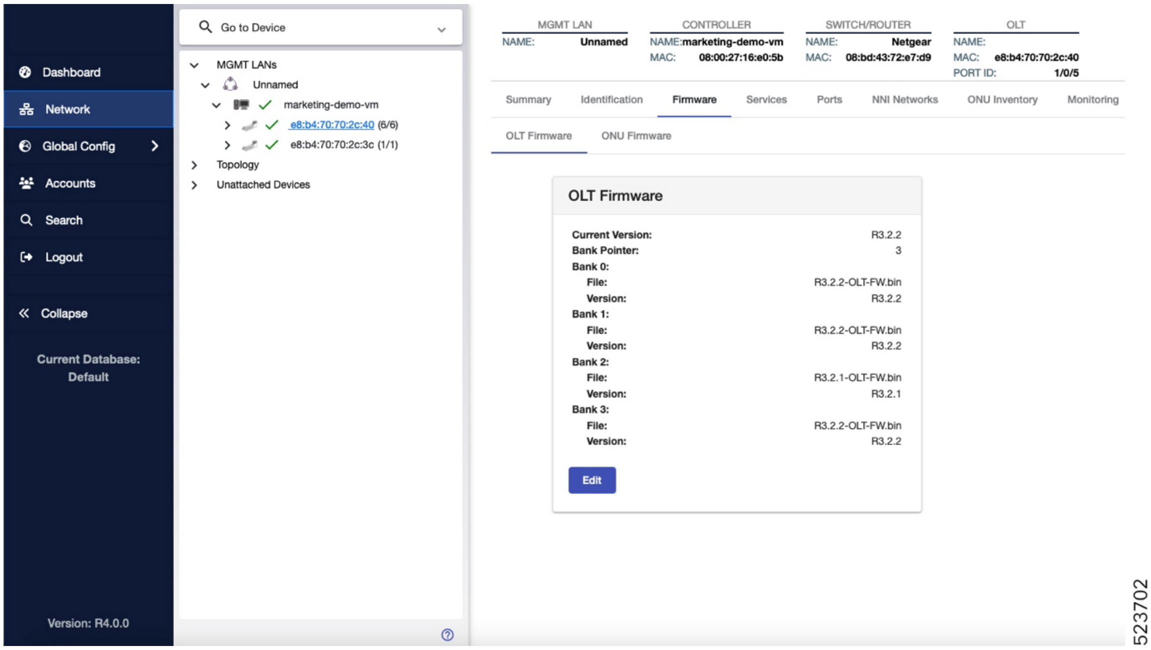
Items per page: 10 0 of 0

Figure 6 - Database Update

Upgrading firmware on OLT

This section describes how to upgrade the firmware on existing OLTs and the Default OLT configuration.

- Connect to the PON Manager
- Navigate to the **OLT Firmware** tab



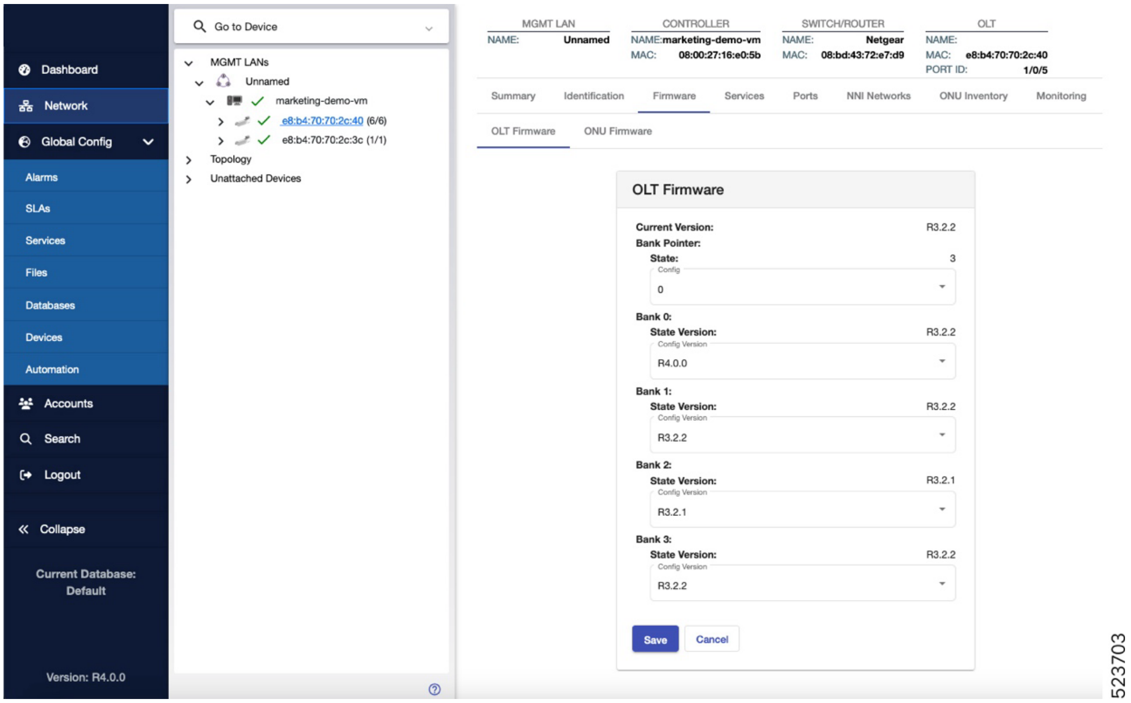
MGMT LAN		CONTROLLER		SWITCH/ROUTER		OLT	
NAME:	**Unnamed**	NAME:	**marketing-demo-vm**	NAME:	**Netgear**	NAME:	
		MAC:	**08:00:27:16:e0:5b**	MAC:	**08:bd:43:72:e7:d9**	MAC:	**e8:b4:70:70:2c:40**
						PORT ID:	**1/0/5**

OLT Firmware

Current Version:	R3.2.2
Bank Pointer:	3
Bank 0:	
File:	R3.2.2-OLT-FW.bin
Version:	R3.2.2
Bank 1:	
File:	R3.2.2-OLT-FW.bin
Version:	R3.2.2
Bank 2:	
File:	R3.2.1-OLT-FW.bin
Version:	R3.2.1
Bank 3:	
File:	R3.2.2-OLT-FW.bin
Version:	R3.2.2

Figure 7 - OLT Firmware

- Click the EDIT button



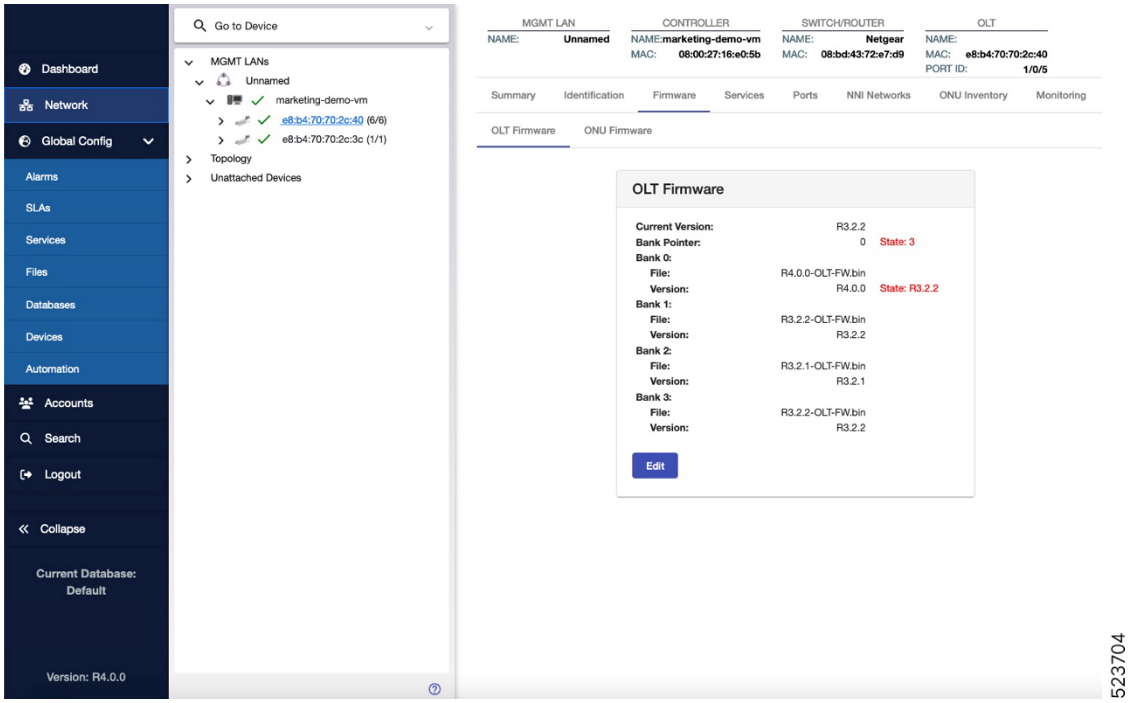
MGMT LAN		CONTROLLER		SWITCH/ROUTER		OLT	
NAME:	Unnamed	NAME:	marketing-demo-vm	NAME:	Netgear	NAME:	
		MAC:	08:00:27:16:e0:5b	MAC:	08:bd:43:72:e7:d9	MAC:	e8:b4:70:70:2c:40
						PORT ID:	1/0/5

OLT Firmware

Current Version: R3.2.2
Bank Pointer:
State: 3
Config: 0
Bank 0:
State Version: R3.2.2
Config Version: R4.0.0
Bank 1:
State Version: R3.2.2
Config Version: R3.2.2
Bank 2:
State Version: R3.2.1
Config Version: R3.2.1
Bank 3:
State Version: R3.2.2
Config Version: R3.2.2

Figure 8 - Edit OLT Firmware

- Select one of the firmware banks and select **R4.0.0**
- Click the SAVE button
- Wait for the PON Controller to transfer the firmware to OLT (~ 1 min.)
    - When the State: R3.2.2 disappears, the firmware is loaded.



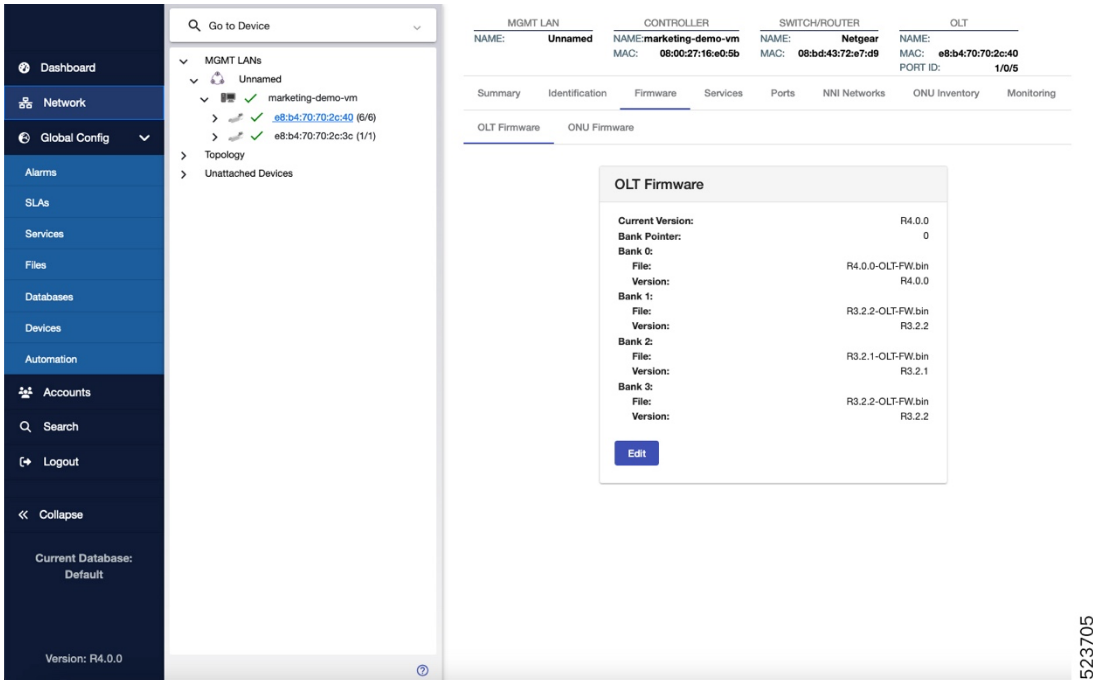
MGMT LAN		CONTROLLER		SWITCH/ROUTER		OLT	
NAME:	Unnamed	NAME:	marketing-demo-vm	NAME:	Netgear	NAME:	
		MAC:	08:00:27:16:e0:5b	MAC:	08:bd:43:72:e7:d9	MAC:	e8:b4:70:70:2c:40
						PORT ID:	1/0/5

OLT Firmware

Current Version:	R3.2.2	
Bank Pointer:	0	State: 3
Bank 0:		
File:	R4.0.0-OLT-FW.bin	
Version:	R4.0.0	State: R3.2.2
Bank 1:		
File:	R3.2.2-OLT-FW.bin	
Version:	R3.2.2	
Bank 2:		
File:	R3.2.1-OLT-FW.bin	
Version:	R3.2.1	
Bank 3:		
File:	R3.2.2-OLT-FW.bin	
Version:	R3.2.2	

Figure 9 - Upgrade Firmware: Load Firmware to OLT

- Wait for the PON Controller to reset OLT (~ 1 min.)



MGMT LAN
NAME: Unnamed

CONTROLLER
NAME: marketing-demo-vm
MAC: 08:00:27:16:e0:5b

SWITCH/ROUTER
NAME: Netgear
MAC: 08:bd:43:72:e7:d9

OLT
NAME:
MAC: e8:b4:70:70:2c:40
PORT ID: 1/0/5

OLT Firmware

Current Version: R4.0.0
Bank Pointer: 0
Bank 0:
File: R4.0.0-OLT-FW.bin
Version: R4.0.0
Bank 1:
File: R3.2.2-OLT-FW.bin
Version: R3.2.2
Bank 2:
File: R3.2.1-OLT-FW.bin
Version: R3.2.1
Bank 3:
File: R3.2.2-OLT-FW.bin
Version: R3.2.2

Figure 10 - OLT Firmware (after upgrade)

Post Installation – Verify Processes

This section describes how to verify the PON Manager, PON Controller and MongoDB processes are running.

1. Use the script `status.sh` provided with the release to check the status of the installation.

```
user@system:~/R4.0.0-Cisco-UB2004$ ./status.sh

● mongod.service - MongoDB Database Server
   Loaded: loaded (/lib/systemd/system/mongod.service; enabled; vendor preset: enabled)
   Active: active (running) since Tue 2021-11-30 14:56:17 PST; 4 months 16 days ago
     Docs: https://docs.mongodb.org/manual
 Main PID: 656 (mongod)
   CGroup: /system.slice/mongod.service
           └─656 /usr/bin/mongod --config /etc/mongod.conf

● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
  Drop-In: /lib/systemd/system/apache2.service.d
           └─apache2-systemd.conf
   Active: active (running) since Fri 2022-04-08 09:56:02 PDT; 1 weeks 2 days ago
  Process: 11716 ExecStop=/usr/sbin/apachectl stop (code=exited, status=0/SUCCESS)
  Process: 29373 ExecReload=/usr/sbin/apachectl graceful (code=exited, status=0/SUCCESS)
  Process: 11722 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
 Main PID: 11726 (apache2)
    Tasks: 151 (limit: 4915)
   CGroup: /system.slice/apache2.service

● tibit-netopeer2-server.service - Tibit Communications, Inc. Netopeer2 Server
   Loaded: loaded (/lib/systemd/system/tibit-netopeer2-server.service; enabled; vendor: enabled)
   Active: active (running) since Fri 2022-04-08 09:54:42 PDT; 1 weeks 2 days ago
 Main PID: 10819 (netopeer2-serve)
    Tasks: 7 (limit: 4915)
   CGroup: /system.slice/tibit-netopeer2-server.service
           └─10819 /opt/tibit/netconf/bin/netopeer2-server -v 1 -t 55

● tibit-netconf.service - Tibit Communications, Inc. NetCONF Server
   Loaded: loaded (/lib/systemd/system/tibit-netconf.service; enabled; vendor preset: enabled)
   Active: active (running) since Fri 2022-04-08 09:54:42 PDT; 1 weeks 2 days ago
 Main PID: 10812 (tibit-netconf)
    Tasks: 12 (limit: 4915)
   CGroup: /system.slice/tibit-netconf.service
           ├─10812 /opt/tibit/netconf/bin/tibit-netconf
           └─10821 /opt/tibit/netconf/bin/tibit-netconf
```

2. Verify MongoDB is running

```
user@system:~$ sudo systemctl status mongod.service

● mongod.service - MongoDB Database Server
   Loaded: loaded (/lib/systemd/system/mongod.service; enabled; vendor preset: enabled)
   Active: active (running) since Fri 2019-08-30 11:56:38 PDT; 3 days ago
     Docs: https://docs.mongodb.org/manual
 Main PID: 15035 (mongod)
   CGroup: /system.slice/mongod.service
           └─15035 /usr/bin/mongod --config /etc/mongod.con
```

3. Verify PON Manager Apache Web Server is running

```
user@system:~$ sudo systemctl status apache2.service

● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
  Drop-In: /lib/systemd/system/apache2.service.d
           └─apache2-systemd.conf
   Active: active (running) since Fri 2019-08-16 15:19:09 PDT; 1 weeks 2 days ago
  Process: 2981 ExecReload=/usr/sbin/apachectl graceful (code=exited, status=0/SUCCESS)
 Main PID: 8471 (apache2)
```

Installation for Production Deployments

For production deployments, Cisco recommends installing each of the Routed PON Management applications on separate VMs (or servers).

Note: Cisco PON Manager package is signed by Cisco. Refer to README document in the extracted Cisco PON Manager package for the instructions to verify the signing of the package:

```
├──── R4.0.0-Cisco-UB2004-sign.tar.gz
├──── README
└──── verify.tar.gz
```

MongoDB Installation

MongoDB serves the purpose of being the datastore for the Cisco Routed PON Management System. This includes all PON-specific information as well as user models.

For production deployments, Cisco recommends configuring multiple servers or VMs in a Replica Set using Authentication and TLS. Please reference the Cisco Tech Note, Mongo Replication with Authentication and TLS/SSL for configuration of each Mongo instance.

Installation

To install the MongoDB Community Edition on each instance,

1. Access the official MongoDB website.
2. Using the table of contents at the top of the page, select the option, “Uninstall MongoDB Community Edition”, and follow the instructions to uninstall MongoDB if installed.
3. Using the same table of contents at the top of the page, select the option; “Install MongoDB Community Edition using .deb Packages”, and follow the instructions to install the latest version of MongoDB.
    - Ensure that you are using the “Ubuntu 20.04” options.

MongoDB Scripts

Alternatively, these scripts can automate installing, starting, and uninstalling MongoDB.

Installation steps:

1. Transfer/download the .zip file to Ubuntu host or VM.
2. From a Linux shell, unpack the .zip file and change to the following directory:

```
unzip R4.0.0-Cisco-UB2004.zip
cd R4.0.0-Cisco-UB2004/R4.0.0-PonManager-UB2004
```

Note: Only the following installation scripts need to be transferred to the Ubuntu host or VM for each Mongo instance.

mongodb_install.sh

This script will uninstall the current version of MongoDB and then install and start the latest version of MongoDB server Community Edition.

- Supported Operating Systems:
    a. Ubuntu 20.04
- Dependencies:
    a. systemctl
    b. gpupg
    c. lsb-release
- Run the command:

```
sudo bash ./mongodb_install.sh
```

mongodb_start.sh

This script will start a forked instance of a MongoDB server

- Supported Operating Systems:
  a. Ubuntu 20.04
- Dependencies:
  a. systemctl
- Run the command

```
sudo bash ./mongodb_start.sh
```

Routed PON Manager Installation

The Routed PON Manager software is composed of a graphical user front-end web application (Web App) and a RESTful interface (REST API) that provides access to the MongoDB datastore.

Prerequisites:

- Ensure that you have an active internet connection.
- Install/upgrade existing python3 package to be equal to or greater than version 3.6.7.
- Install/upgrade existing pip3 package to be equal to or greater than version 9.0.0.
- Ensure that you have your MongoDB Server running that will manage user authentication and serve as your default database.
- Ensure you have the latest ubuntu updates

Installation steps:

1. Transfer/download the .zip file to Ubuntu host or VM.
2. From a Linux shell, unpack the .zip file and change to the following directory:

```
unzip R4.0.0-Cisco-UB2004.zip
cd R4.0.0-Cisco-UB2004/R4.0.0-PonManager-UB8104
```

3. *Optional* | If desired, you can configure the self-signed certificate created during installation by modifying the configuration file within this package directory; ‘tibit_ponmgr_selfsigned_cert_req.conf’.
   There are two sections that may be modified:

   ○ [req_distinguished_name]

countryName	Country code. | Ex: ‘US’ (United States), etc
ST	State or Province
localityName	City or Locality
organizationName	Name of organization

organizationalUnitName	Name of group/unit within organization
CN	Fully Qualified Domain Name of system

- [alt_names]

DNS.*number*	Name to access web server.
	Configure additional names by adding more ‘DNS.x’ lines with increasing numbers in place of ‘x’ in the example above.
	Ex.
	DNS.1 = tibitdev-web
	DNS.2 = tibitdev-api
	DNS.3 = www.MCMS.com

IP.*number*	IP of machine. Configure additional interfaces by adding more ‘IP.x’ lines with increasing numbers in place of ‘x’ in the example above.
	Ex.
	IP.1 = 127.0.0.1
	IP.2 = 10.1.10.225

3. Use apt-get to install the package.

Note: By default, Routed PON Manager will deploy over HTTPS utilizing a self-signed certificate created during installation. THIS IS NOT SAFE FOR A PRODUCTION ENVIRONMENT. If installing Routed PON Manager for any use other than evaluation, you should use legitimate certificates. Step 4 describes how to update the self-signed cerficate and key with valid certificate and key.

- Use the following command to install PON Manager with the default self-signed certificate:

```
sudo apt install ./tibit-ponmgr_R4.0.0_all.deb
```

- *Optional* | Use the following command to install PON Manager with the customized self-signed certificate configured in Step 2 above:

```
sudo TIBIT_PONMGR_CERT_REQ=/path/to/file.conf apt \
install ./tibit-ponmgr_R4.0.0_all.deb
```

Configuration steps:

Ensure that you have root/sudo access to modify files.

1. Navigate to the directory: '/etc/tibit/ponmgr/'
   - Within this directory, there is one PON Manager configuration file that will need to be edited with a text editor of your choice.
   - Update the PON Manager configuration files to match the settings for connectivity to MongoDB.
     - user_database.json – Stores configuration for database to be used for user authentication (see Table 1: PON Manager user_database.json file)
     - databases.json – Stores list of all available PON databases in PON Manager (see Table 2: PON Manager databases.json file)
     - recovery_email_configuration.json – Configures SMTP email server for sending user password recovery emails (see Table 3: PON Manager recovery_email_configuration.json file)

2. Restart Pon Controller process (tibit-poncntl):

```
sudo systemctl restart tibit-poncntl.service
```

Table 1: PON Manager user_database.json file

Key	Description
host	Hostname/IP Address of the MongoDB server hosting your user database.
name	Name of your user database. Default user database name is 'tibit_users'. *(If it doesn't exist, it will be created)*
port	MongoDB port number.
auth_enable	Boolean value determining if the MongoDB server at *host:port* is using authentication.
auth_db	Name of your MongoDB authentication database. *(Used when auth_enabled = true)*
username	The username of the MongoDB user to authenticate with. *(Used when auth_enabled = true)*
password	The password of the specified MongoDB user. *(Used when auth_enabled = true)*

Key	**Description**
tls_enable	Boolean value specifying whether the MongoDB server at *host:port* is using encryption.
ca_cert_path	The local path to the encryption certificate.
compression	Boolean value specifiything whether to enable snapy compression for MongoDB connections.
dns_srv	Boolean value specifying whether the MongoDB server is using a DNS seed list.
db_uri	Raw MongoDB connection URI. All Other fields are ignored if this is used.
replica_set_enable	Boolean value specifying if the MongoDB server is running as a replica set.
replica_set_name	The name of the MongoDB replica set. *(Used when replica_set_enabled = true)*
replica_set_hosts	List of hosts to be used as the MongoDB replica set. *(Used when replica_set_enabled = true)*
django_key	Cryptographic key used by django. The format of the field is a freeform string, however, Django recommends it be minimum 50 characters with minimum 5 unique characters. For more information on the Django Key, please refer to the Routed PON Manager User Guide. (Note: A randomized key will be generated during installation if the 'django_key' field is not present.)

Table 2: PON Manager databases.json file

Key	**Description**
host	Hostname/IP Address of your mongoDB server hosting your PON Controller database
name	Name of your PON Controller database. Default PON Controller database name is 'tibit_pon_controller'. *(If it doesn't exist, it will be created)*
port	MongoDB server port number.

Key	Description
auth_enable	Boolean value determining if the MongoDB server at *host:port* is using authentication.
auth_db	Name of your MongoDB authentication database. *(Used when auth_enabled = true)*
username	The username of the MongoDB user to authenticate with. *(Used when auth_enabled = true)*
password	The password of the specified MongoDB user. *(Used when auth_enabled = true)*
tls_enable	Boolean value specifying whether the MongoDB server at *host:port* is using encryption.
ca_cert_path	The local path to the encryption certificate.
dns_srv	Boolean value specifying whether the MongoDB server is using a DNS seed list.
db_uri	Raw MongoDB connection URI. All Other fields are ignored if this is used.
replica_set_enable	Boolean value specifying if the MongoDB server is running as a replica set.
replica_set_name	The name of the MongoDB replica set. *(Used when replica_set_enabled = true)*
replica_set_hosts	List of hosts to be used as the MongoDB replica set. *(Used when replica_set_enabled = true)*

Table 3: PON Manager recovery_email_configuration.json file

Key	Description
host	Host to use for sending email
port	Port to use for the SMTP server defined above *Type: String*
use_tls	Use TLS connection *'true' or 'false'* *Type: String*

Key	Description
user	Username for the SMTP server
password	Password for user

For more information on installing the Routed PON Manager, please reference the R4.0.0 Routed PON Manager User Guide. The section “Installation” on page 21 provides details on requirements and dependencies, package contents and installation methods.

Appendix A: Method for creating a permanent Network configuration (Netplan)

In the requirements section on Page 10, it shows you the commands to create temporary interfaces to be used by the Routed PON management system. The procedure below creates permanent interfaces.

Verify the ethernet interfaces are configured on the ubuntu system

1) Look for your Ethernet Interfaces: "ifconfig" or "ip a".
   <<< make note of interfaces listed >>>
2) There are multiple ways to configure interfaces on Ubuntu 20.04
3) This is the example for creating interfaces via Netplan
4) Netplan is located in the /etc/netplan directory
5) Located here will be a file similar in name to "01-network-manager-all.yaml"
6) Edit this file with your favorite editor such as "nano" or "vi"

```
user@system:~$ sudo nano /etc/netplan/<net-plan-name>.yaml
```

Sample Netplan text <<< Make sure indentation is consistent >>>

```
network:
  version: 2
  renderer: NetworkManager
  ethernets:
      eno1:                            <<< Routed PON IP Interface >>>
        dhcp4: no                      <<< No DHCP >>>
        dhcp6: no                      <<< No DHCP >>>
        addresses: [172.16.41.5/24]    <<< Static IPv4 >>>
        gateway4: 172.16.41.1          <<< IPv4 default Gateway >>>
        nameservers:                   <<< DNS Addresses >>>
               search: [local]         <<< Search on local machine first >>>
               addresses: [8.8.8.8, 8.8.4.4]  <<< external DNS servers >>>
      eno2:                            <<< PON Controller Interface >>>
         dhcp4: no                     <<< No DHCP >>>
         dhcp6: no                     <<< No DHCP >>>
  vlans:                               <<< Configured VLANs >>>
      vlan4090:                        <<< "l2EthInterfaceName" >>>
        id: 4090                       <<< VLAN number >>>
        link: eno2                     <<< PON Controller Interface >>>
```

After finished editing, Save and exit, then enter "sudo netplan apply" to enable new configuration.

1) Verify ubuntu system has connectivity to Internet.

2) Run “ip a” and look for the interfaces you’ve created. You should see an interface with the IP interface you configured, and an interface called “vlan4090” shown (or the name you chose to call the "l2EthInterfaceName").

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV Amsterdam,
The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at **https://www.cisco.com/go/offices**.