Install Routed PON 24.1.2 - Single VM Lab

Contents

Introduction

Prerequisites

- Requirements
- Components Used

Configuration

- Virtual Machine
  - Tools
- Unpacking

Installation

- Netplan
- Package Install

Verification

- Service Status Check

Reference Documentation

Introduction

This document describes a walkthrough of the Cisco Routed PON Manager software installation in the local lab.

Prerequisites

Requirements

- Knowledge of the Linux Server Environment
- Knowledge of Linux Text Editors
- Linux Tools - openssh-server, net-tools, ntpd

Components Used

- Linux Virtual Machine (VM)
  - 2 vCPUs
  - 8GB RAM
  - 20GB Space (Minimum)
- Ubuntu 20.04.06 LTS

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.

Configuration

Virtual Machine

Tools

This document begins with the assumption that the VM has been configured for access, the **Ubuntu Operating System** (OS) has been installed and configured with network connectivity, the tools listed in Requirements have been installed, and the Routed PON zip file has been downloaded. For ease, the Ubuntu **apt** commands have been provided to download and install the recommended tools.

Note: A link to the download for Routed PON 24.1.2 is available here.

```
sudo apt install net-tools
sudo apt install openssh-server
sudo apt install ntpd
```

Unpacking

1) Create an installation directory to unzip the **Routed PON 24.1.2** zip file to.

<#root>

```
rpon@rpon-mgr:~$
```

mkdir Routed_PON_24_1_2

2) Unzip **Cisco_Routed_PON_24_1_2_Release.zip** into the assigned directory.

<#root>

```
rpon@rpon-mgr:~/PON_Mgr_24_1_2$
```

unzip Cisco_Routed_PON_24_1_2_Release.zip

```
Archive: Cisco_Routed_PON_24_1_2_Release.zip
inflating: PON_MANAGER_SIGNED_CCO/
```

R4.0.0-Cisco-UB2004-sign.tar.gz

```
inflating: PON_MANAGER_SIGNED_CCO/
```

README

```
inflating: PON_MANAGER_SIGNED_CCO/
```

verify.tar.gz

3) Change directory (cd) to the newly created **PON_MANAGER_SIGNED_CCO** folder and list (ls) the files.

<#root>

```
rpon@rpon-mgr:~/PON_Mgr_24_1_2/PON_MANAGER_SIGNED_CCO$
```

ls -la

```
total 29120
drwxrwxr-x 2 rpon rpon 4096     Jun 13 09:26 .
drwxrwxr-x 3 rpon rpon 4096     Jun 13 09:26 ..
-rw-rw-r-- 1 rpon rpon 29792662 Mar 15 05:21
```

R4.0.0-Cisco-UB2004-sign.tar.gz

```
-rw-rw-r-- 1 rpon rpon 1966     Mar 15 05:21
```

README

```
-rw-rw-r-- 1 rpon rpon 11502    Mar 15 05:21
```

verify.tar.gz



Note: The listed README file explains the file verification process. This step is not performed during this installation walkthrough.

---

4) Untar the **R4.0.0-Cisco-UB2004-sign.tar.gz** file using **tar -xvf R4.0.0-Cisco-UB2004-sign.tar.gz**.

<#root>

```
rpon@rpon-mgr:~/PON_Mgr_24_1_2/PON_MANAGER_SIGNED_CCO$
```

tar -xvf R4.0.0-Cisco-UB2004-sign.tar.gz

```
R4.0.0-Cisco-UB2004-sign/
R4.0.0-Cisco-UB2004-sign/
```

R4.0.0-Cisco-UB2004.tar.gz.signature

```
R4.0.0-Cisco-UB2004-sign/
```

R4.0.0-Cisco-UB2004.tar.gz

5) Cd to the **R4.0.0-Cisco-UB2004-sign** directory and list the files.

<#root>

```
rpon@rpon-mgr:~/PON_Mgr_24_1_2/PON_MANAGER_SIGNED_CCO$
```

cd R4.0.0-Cisco-UB2004-sign/

```
rpon@rpon-mgr:~/PON_Mgr_24_1_2/PON_MANAGER_SIGNED_CCO/R4.0.0-Cisco-UB2004-sign$
```

ls -la

```
total 29112
drwxr-xr-x 2 rpon rpon 4096     Mar 15 04:51 .
drwxrwxr-x 3 rpon rpon 4096     Jun 13 09:26 ..
-rw-r--r-- 1 rpon rpon 29796139 Mar 15 04:51
```

R4.0.0-Cisco-UB2004.tar.gz

```
-rw-r--r-- 1 rpon rpon 3546     Mar 15 04:51
```

R4.0.0-Cisco-UB2004.tar.gz.signature

6) Untar **R4.0.0-Cisco-UB2004-tar.gz** via **tar -xvf R4.0.0-Cisco-UB2004.tar.gz**.

<#root>

```
rpon@rpon-mgr:~/PON_Mgr_24_1_2/PON_MANAGER_SIGNED_CCO/R4.0.0-Cisco-UB2004-sign$
```

tar -xvf R4.0.0-Cisco-UB2004.tar.gz

```
---- snipped for brevity -----
```

7) A new directory is now created called **R4.0.0-Cisco-UB2004** CD to this directory.



Note: This is where the shell scripts to install, uninstall, check status, and additional PON Controller JSON file is located.

<#root>

```
rpon@rpon-mgr:~/PON_Mgr_24_1_2/PON_MANAGER_SIGNED_CCO/R4.0.0-Cisco-UB2004-sign/R4.0.0-Cisco-UB2004$
```

ls -la

```
total 116
drwxr-xr-x 7 rpon rpon 4096  Mar 14 11:11 .
drwxr-xr-x 3 rpon rpon 4096  Jun 13 09:26 ..
-rw-r--r-- 1 rpon rpon 8196  Mar 14 11:10
```

.DS_Store

-rwxr-xr-x 1 rpon rpon 13650 Mar 14 11:10
get-support-info.sh

drwxr-xr-x 3 rpon rpon 4096 Mar 14 11:10
grafana_dashboards

-rwxr-xr-x 1 rpon rpon 25392 Mar 14 11:10
install.sh

-rw-r--r-- 1 rpon rpon 1493 Mar 14 11:11
PonCntlInit.json

drwxr-xr-x 2 rpon rpon 4096 Mar 14 11:10
R4.0.0-Firmware

drwxr-xr-x 5 rpon rpon 4096 Mar 14 11:14
R4.0.0-Netconf-UB2004

drwxr-xr-x 6 rpon rpon 4096 Mar 14 11:14
R4.0.0-PonManager-UB2004

-rw-r--r-- 1 rpon rpon 7949 Mar 14 11:11
README.txt

-rwxr-xr-x 1 rpon rpon 2349 Mar 14 11:10
status.sh

drwxr-xr-x 2 rpon rpon 4096 Mar 14 11:10
tools

-rwxr-xr-x 1 rpon rpon 2245 Mar 14 11:10
uninstall.sh

-rwxr-xr-x 1 rpon rpon 8605 Mar 14 11:11
upgrade.sh

Installation

View the **README.txt** file for installation steps. This install is using **Option 2** for **New Installation**.

<#root>

rpon@rpon-mgr:~/PON_Mgr_24_1_2/PON_MANAGER_SIGNED_CCO/R4.0.0-Cisco-UB2004-sign/R4.0.0-Cisco-UB2004$

cat README.txt

```
--- snipped for brevity ---

Option 2: New Installation
==========================

Step 1 : Verify System Requirements:
------------------------------------

a) Verify system is running ubuntu version 20.04

user@system:~$ lsb_release -a
No LSB modules are available.
Distributor ID: Ubuntu
Description: Ubuntu 20.04.3 LTS
Release: 20.04
Codename: bionic

b) 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 18.04 and 20.04
3) The simplest way is via Netplan
4) Netplan is located in the /etc/netplan directory
5) There 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"

c) 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: <<< MCMS 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 >>>
 vlans: <<< Configured VLANs >>>
  vlan4090: <<< "l2EthInterfaceName" VLAN named vlan4090 assigned to PON Controller Interface >>>
  id: 4090 <<< VLAN number >>>
  link: eno2 <<< PON Controller Interface >>>

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

e) Verify ubuntu system has connectivity to Internet


Step 2 : Installation
---------------------
```

```
a) Run the installation script "install.sh":

user@system:<install_directory>/R4.0.0-UB2004$ sudo ./install.sh -e <l2EthInterfaceName>

This script will do the following:
· Install MongoDB
· Install MCMS PON Manager
· Install MCMS Netconf Server
· Install PON Controller and UMT Relay (using the Ethernet interface specificed)

Required:
-e <l2EthInterfaceName> interface name of L2 port

Optional:
-d <databaseIpAaddress> IP address of MongoDB database
-n <databaseName> MongoDB database name for PON Manager
-m Install only PonManager/MongoDB/NETCONF
-c Install only PonController

defaults:
-d <databaseIpAddress> = 127.0.0.1
-n <databaseName> = tibit_pon_controller


Informational: How to verify all processes are running
=======================================================

a) 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
Main PID: 15035 (mongod)
CGroup: /system.slice/mongod.service
└─15035 /usr/bin/mongod --config /etc/mongod.conf

b) Verify MCMS 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)


---- Removed addtional information regarding PON Controller as this is installed in the XR Router so the
```

**** ****

Netplan

Using a linux text file editor (nano, vi), edit the YAML file located in the **/etc/netplan/** directory using the template provided in the **README.txt** in the installation folder. Populate the IP information specific to the network and VM.

<#root>

```
rpon@rpon-mgr:~/PON-mgr-24.1.2/PON_MANAGER_SIGNED_CCO/R4.0.0-Cisco-UB2004-sign/R4.0.0-Cisco-UB2004$
```

sudo nano /etc/netplan/01-network-manager-all.yaml

```
network:
 version: 2
 renderer: NetworkManager
 network:
  ethernets:
   ens192:
```

<- This VM's network adapter is ens192. If the default is NOT ens192, change this value to the desired n

```
    dhcp4: no
    dhcp6: no
    addresses:
```

[IPv4 address and subnet]

```
    gateway4:
```

[V4Gateway]

```
    nameservers:
      addresses:
```

[DNS Server(s)]

```
vlans:
```

vlan.4090:

id: 4090

```
  link:
```

[VM network adapter name]

Note: Using nano to edit, once complete; press **Control + O** to save the file, then **Control X** to exit nano. In VIM, use **:wq!** to save and exit.

Note: Usage of **sudo netplan --debug apply** is useful when testing the netplan prior to application.

Verify the netplan configuration is correct by viewing the file via **cat**. This output is strictly a lab example, please utilize the IP addresses specific to the network. Once complete and exited from the text editor, run **sudo netplan apply**.

Lab Example:

<#root>

```
rpon@rpon-mgr:~/PON-mgr-24.1.2/PON_MANAGER_SIGNED_CCO/R4.0.0-Cisco-UB2004-sign/R4.0.0-Cisco-UB2004$
```

cat /etc/netplan/01-network-manager-all.yaml

```
# Let NetworkManager manage all devices on this system
network:
 version: 2
 renderer: NetworkManager
 ethernets:
  ens192:
    dhcp4: no
    dhcp6: no
    addresses: [10.122.140.232/28]
    gateway4: 10.122.140.225
    nameservers:
      addresses: [172.18.108.43,172.18.108.34]
 vlans:
   vlan4090:
     id: 4090
     link: ens192
```

```
rpon@rpon-mgr:~/PON-mgr-24.1.2/PON_MANAGER_SIGNED_CCO/R4.0.0-Cisco-UB2004-sign/R4.0.0-Cisco-UB2004$
```

sudo netplan apply

Package Install

Perform the install using the chosen arguments. For this install, using -e, -d, and -m. As per the README.txt, -e tells the installer which ethernet interface on the VM to use, -d sets the IP that is applied to the **mongo.conf** file for the MongoDB to use, and -m installs PON Manager, MongoDB, and NETCONF.

Example:
sudo ./install.sh -e ens192 -d [IPaddr] -m



Note: If this is a new VM, install times can vary up to 5 minutes while the dependencies are added and updated. Once the install is complete, a log message is generated.

<#root>

```
rpon@rpon-mgr:~/PON_MANAGER_SIGNED_CCO/R4.0.0-Cisco-UB2004-sign/R4.0.0-Cisco-UB2004$
```

sudo ./install.sh -e ens192 -d 10.122.140.232 -m

```
--- Installation snipped for brevity ---
```

```
Installation complete!

MCMS Component Versions:
```

PON Manager: R4.0.0

PON NETCONF: R4.0.0

PON Controller: Not Installed



Note: The PON Controller is hosted on the XR Router, therefore an installation on the VM is unnecessary.

Verification

Service Status Check

Perform a status check on the installed services to verify they are up and running via the **status.sh** script, located in the same install directory.



Note: If a full installation was performed as indicated by -m, verify the listed services are up and in a running state.

- **mongod.service**
- **apache2.service**
- **tibit-netopeer2-server.service**
- **tibit-netconf.service**

Example:

<#root>

```
rpon@rpon-mgr:~/PON-mgr-24.1.2/PON_MANAGER_SIGNED_CCO/R4.0.0-Cisco-UB2004-sign/R4.0.0-Cisco-UB2004$
```

sudo ./status.sh

```
MCMS Component Versions:
PON Manager: R4.0.0
PON NETCONF: R4.0.0
PON Controller: Not Installed
```

●

mongod.service

 - MongoDB Database Server
Loaded: loaded (/lib/systemd/system/mongod.service; enabled; vendor preset: enabled)

Active: active (running) since Wed 2024-06-12 19:45:37 EDT; 2min 49s ago

Main PID: 54731 (mongod)
Memory: 74.7M
CGroup: /system.slice/mongod.service
└─54731 /usr/bin/mongod --config /etc/mongod.conf

● apache2.service - The Apache HTTP Server
Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)

Active: active (running) since Wed 2024-06-12 19:46:44 EDT; 1min 42s ago

Main PID: 62165 (apache2)
Tasks: 123 (limit: 9419)
Memory: 18.6M
CGroup: /system.slice/apache2.service
├─62165 /usr/sbin/apache2 -k start
├─62167 /usr/sbin/apache2 -k start
├─62168 /usr/sbin/apache2 -k start
└─62169 /usr/sbin/apache2 -k start

●

tibit-netopeer2-server.service

 - Tibit Communications, Inc. Netopeer2 Server
Loaded: loaded (/lib/systemd/system/tibit-netopeer2-server.service; enabled; vendor preset: enabled)
Active: active (running) since Wed 2024-06-12 19:47:04 EDT; 1min 21s ago
Process: 63029 ExecStart=/opt/tibit/netconf/bin/start_netopeer2_server.sh (code=exited, status=0/SUCCES
Main PID: 63035 (netopeer2-serve)
Tasks: 7 (limit: 9419)
Memory: 5.4M
CGroup: /system.slice/tibit-netopeer2-server.service
└─63035 /opt/tibit/netconf/bin/netopeer2-server -v 1 -t 55

●

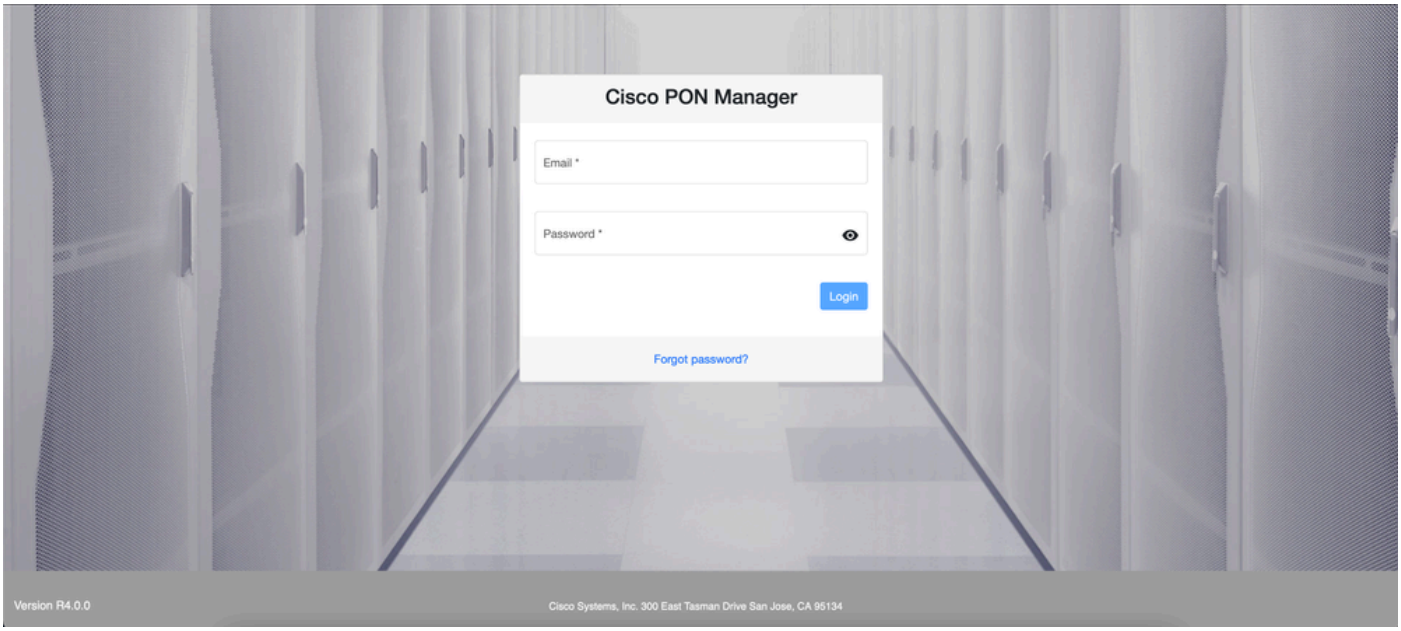
tibit-netconf.service

 - Tibit Communications, Inc. NetCONF Server
Loaded: loaded (/lib/systemd/system/tibit-netconf.service; enabled; vendor preset: enabled)

Active: active (running) since Wed 2024-06-12 19:47:04 EDT; 1min 21s ago

Process: 63023 ExecStartPre=/opt/tibit/netconf/bin/shm_clean.sh (code=exited, status=0/SUCCESS)
Process: 63027 ExecStartPre=/opt/tibit/netconf/bin/sysrepocfg --copy-from startup -d running (code=exit
Main PID: 63028 (tibit-netconf)
Tasks: 17 (limit: 9419)
Memory: 49.4M
CGroup: /system.slice/tibit-netconf.service
├─63028 /opt/tibit/netconf/bin/tibit-netconf
└─63037 /opt/tibit/netconf/bin/tibit-netconf

11. Open an internet browser and input the IP of the VM.



PON Manager Login Screen

Reference Documentation

- Cisco Support and Downloads Page
- Cisco Routed PON Solution Page
- Cisco Routed PON Installation Guide
- Cisco Routed PON Deployment Guide
- Release Notes for Cisco Routed PON, Cisco IOS® XR Release 24.1.1 and 24.1.2