Hewlett Packard
Enterprise

HPE Cray Operating System Installation Guide for HPE Performance Cluster Manager (2.1.76) (S-8026)

Part Number: S-8026
Published: February 2022

HPE Cray Operating System Installation Guide for HPE Performance Cluster Manager

Contents

1 Copyright and Version

COS: 2.1.76-50

Doc git hash: de9559bafc9c9257ca8b099ccb392ffcbadc0c61

Generated: Mon Nov 22 2021

2 Overview

2.1 Installation Overview

This document describes how to install, upgrade, configure and boot the HPE Cray Operating System (COS) on a system managed by HPE Performance Cluster Manager (HPCM) software.

The installation process does not automatically affect the state of any other existing COS software on the system, for example any COS software that runs on compute nodes.

Multiple releases of COS can be installed on a system at the same time. The administrator can decide which release to use on compute nodes (or use multiple releases).

The same instructions are followed whether the administrator is installing COS for the first time or upgrading COS on a previously installed system.

2.2 COS Version Information

This document is specific to the COS 2.1 release. All commands and sample output refer to the release as 2.1.XX, where "XX" should be replaced with the version number in the release distribution file being installed. For example, if your release distribution file is cos-2.1.51-sle15sp2-x86_64.iso, you should substitute 2.1.51 for 2.1.XX when following the documentation.

2.3 Differences from the Previous Release

Significant changes from the previous release of COS are described in the following subsections.

2.3.1 New Features

- COS 2.1 is based on SLES 15 SP2. The previous release, COS 2.0, was based on SLES 15 SP1.

2.3.2 Deprecating Features

None

2.3.3 Deprecated Features

None

2.3.4 Other Changes

None

2.4 Prerequisites

- The slingshot-host-software product is installed. slingshot-host-software provides Slingshot kernel modules and related software compiled to match the COS kernel environment. The version of slingshot-host-software installed should correspond to the version of COS being installed. For example, if you are installing cos-2.1.51-sle15sp2-x86_64.iso, the slingshot-host-software product should include "cos-2.1.51" in its name.

3 Install COS 2.1

See the "COS Version Information" section of this document for details on how COS version numbers are referenced in the commands and output below.

3.1 Basic Installation

HPCM manages the installation process for COS similar to how it manages installation of other operating systems. Refer to the "Creating a COS image" section of the "HPE Performance Cluster Manager Administration Guide" for instructions on using cluster manager commands to

- import the COS ISO to a repository,
- create a repository group,
- build a RPM list, and
- create a COS compute image

4 Configure COS 2.1

4.1 Boot Parameters

COS includes a set of Linux kernel boot parameters that should be specified to ensure proper behavior. The boot parameters are included in the `cray-boot-parameters-mss-compute` RPM which can be found in the COS repository. The following steps describe how to query and specify the boot parameters.

1. Extract the `./boot/parameters-mss_c` file from the `cray-boot-parameters-mss-compute` RPM and examine its contents to determine the boot parameters to be set. For example:

   ```
   admin# cd /tmp
   admin# cp /opt/clmgr/repos/other/COS-2.1.XX-sle15sp2-x86_64/RPMS/cray-boot-parameters-mss-compute-* .
   admin# rpm2cpio cray-boot-parameters-mss-compute-* | cpio -id
   2 blocks
   admin# cat ./boot/parameters-mss_c
   bad_page=panic
   hugepagelist=2m-2g
   intel_iommu=off
   intel_pstate=disable
   iommu=pt
   numa_interleave_omit=headless
   oops=panic
   pageblock_order=14
   pcie_ports=native
   quiet
   turbo_boost_limit=999
   biosdevname=0
   ```

2. Use the following cluster manager command to associate the COS boot parameters from the previous step with the COS compute image. Substitute the appropriate COS image name for `IMAGE_NAME` and include the remaining parameters in place of `...` in this example.

   ```
   admin# cm image set -i IMAGE_NAME --kernel-extra-params "bad_page=panic hugepagelist=2m-2g ..."
   ```

3. Ensure the `crashkernel` boot parameter is set to at least 512M. Substitute the appropriate COS image name for `IMAGE_NAME` in this example. Additional information about querying and setting `crashkernel` values is provided in the "HPE Performance Cluster Manager Administration Guide".

   ```
   admin# cm image set -i IMAGE_NAME --crashkernel 512M
   ```

4.2 COS RPM List

The COS RPM list contains references to some RPMs that are not provided by the COS ISO. These RPMs are provided by the SUSE Linux Enterprise Server (SLES) distribution. In order to successfully create a COS compute image from the

COS RPM list, SLES content must be installed and part of the repository group used to create the COS compute image.

4.3 Related Products

COS compute image content is usually created in conjunction with other HPE products and third party software to provide a fully functional compute image. The products most likely to be of potential interest are

- HPE Slingshot
- HPE Cray Programming Environment (CPE), including Workload Manager (WLM) software such as Slurm and PBS Pro
- AMD ROCm
- SLES

Details on how to incorporate Slingshot and CPE product content in compute images can be found in the HPE documentation for those products. SLES content is incorporated into COS compute images as described in the "COS RPM list" section of this document.

5 Install AMD ROCM for GPU Support

1. Create ROCM Repo Directory

   ```
   mkdir /opt/clmgr/repos/other/rocm-4.0.1/
   ```

2. Download ROCM Content

   Download the 4.0.1 ROCM content and place in the newly created repo directory:

   ```
   cd /opt/clmgr/repos/other/rocm-4.0.1/
   wget -r -nH -np -nd https://repo.radeon.com/rocm/zyp/4.0.1/
   ```

3. Create ROCM Repo

   ```
   cm repo add --custom rocm-4.0.1 /opt/clmgr/repos/other/rocm-4.0.1
   cm repo refresh rocm-4.0.1
   ```

4. Create DKMS Repo Directory

   ```
   mkdir /opt/clmgr/repos/other/dkms/
   ```

5. Download DKMS Content

   Download the dkms rpm and place it in the newly created repo directory:

   ```
   cd /opt/clmgr/repos/other/dkms
   wget https://download.opensuse.org/repositories/home:/Ximi1970:/Dkms/openSUSE_Leap_15.2/noarch/ \
   dkms-2.5-lp152.5.1.noarch.rpm
   ```

6. Create DKMS Repo

   Note: This is a temporary solution. At a future date DKMS will be included by default.

   ```
   cm repo add --custom dkms /opt/clmgr/repos/other/dkms
   cm repo refresh dkms
   ```

7. Create ROCM Repo Group

   Create a repo group with everything we have so far, note this command has specific versions which may need to change:

   ```
   cm repo group add cos-hpcm-ss-rocm-4.0.1 --repos dkms Cluster-Manager-1.5-sles15sp2-x86_64 \
   SLE-15-SP2-Full-x86_64 Slingshot-mlx-0.9.1-x86_64 Slingshot-common-0.9.1-x86_64 COS-2.1.27-x86_64 \
   rocm-4.0.1
   ```

8. Create ROCM Image

   Create a new rocm image, cloned from the working Slingshot image:

   ```
   cm image copy -o cos-hpcm-ss -i cos-hpcm-ss-rocm -g cos-hpcm-ss-rocm-4.0.1
   ```

9. Install ROCM Into Image

   Install the ROCM and DKMS content into the existing image. Ignore the post install failure, we will address the issue manually:

   ```
   cm image zypper -i cos-hpcm-ss-rocm --repo-group cos-hpcm-ss-rocm-4.0.1 install dkms rocm-dkms \
   rocm-libs cray-rocm rocm-validation-suite
   ```

10. Manually Build ROCM Modules

    a. In order to get dkms to operate correctly in a chroot (the image), we need to create a proper chroot environment with proc, sys, dev.

    ```
    mount -B /proc/ /opt/clmgr/image/images/cos-hpcm-ss-rocm/proc
    mount -B /sys/ /opt/clmgr/image/images/cos-hpcm-ss-rocm/sys
    chroot /opt/clmgr/image/images/cos-hpcm-ss-rocm
    mount -t devtmpfs none /dev
    mount -o gid=5,mode=620,ptmxmode=0000 -t devpts devpts /dev/pts
    ```

    b. Then we can build and install the ROCM module using DKMS:

    NOTE: Some lines are specific to a particular COS version and will need to change.

    ```
    dkms status
    dkms build amdgpu/4.0-26.el7 -k 5.3.18-2.1_1.1__g982431a4ba-cray_shasta_c
    dkms install amdgpu/4.0-26.el7 -k 5.3.18-2.1_1.1__g982431a4ba-cray_shasta_c
    dkms status
    ```

    c. And tear down the chroot:

    ```
    umount /dev/pts
    umount /dev
    exit
    umount /opt/clmgr/image/images/cos-hpcm-ss-rocm/proc
    umount /opt/clmgr/image/images/cos-hpcm-ss-rocm/sys
    cm image revision commit -i cos-hpcm-ss-rocm -m "Rocm features to image. Rebuild dkms modules"
    ```

11. Test Boot

    a. Assign the image to a node:

    ```
    cm node set --image cos-hpcm-ss-rocm -n x3001c0s19b1n0
    ```

    b. Verify Assignment

    ```
    cm node show -I
    ```

    c. Reboot Node

    ```
    cm power reset -t node x3001c0s19b1n0
    ```

    d. Monitor Boot

    Console can be monitored here:

    ```
    tail -f /var/log/consoles/<nodename>
    ```

    or via the console command:

    ```
    console x3001c0s19b1n0
    ```

12. Verify ROCM Install

    SSH to the node and perform:

    ```
    lsmod | grep amd
    ```

    The following modules should be loaded: amdgpu amdkcl amd-sched amdttm

    Run the following command and verify the GPUs are found:

    ```
    /opt/rocm/bin/rocm-smi
    ```

6 Documentation Conventions

Several conventions have been used in the preparation of this documentation.

- Markdown Format
- File Formats
- Typographic Conventions
- Annotations for how we identify sections of the documentation that do not apply to all systems
- Command Prompt Conventions which describe the context for user, host, directory, chroot environment, or container environment

6.1 Markdown Format

This documentation is in Markdown format. Although much of it can be viewed with any text editor, a richer experience will come from using a tool which can render the Markdown to show different font sizes, the use of bold and italics formatting, inclusion of diagrams and screen shots as image files, and to follow navigational links within a topic file and to other files.

There are many tools which can render the Markdown format to get these advantages. Any Internet search for Markdown tools will provide a long list of these tools. Some of the tools are better than others at displaying the images and allowing you to follow the navigational links.

6.2 File Formats

Some of the installation instructions require updating files in JSON, YAML, or TOML format. These files should be updated with care since some file formats do not accept tab characters for indentation of lines. Only space characters are supported. Refer to online documentation to learn more about the syntax of JSON, YAML, and TOML files.

6.3 Typographic Conventions

`This style` indicates program code, reserved words, library functions, command-line prompts, screen output, file/path names, and other software constructs.

(backslash) At the end of a command line, indicates the Linux shell line continuation character (lines joined by a backslash are parsed as a single line).

6.4 Annotations

This repository may change annotations, for now, under the MarkDown governance these are the available annotations.

You must use these to denote the right steps to the right audience.

These are context clues for steps, if they contain these, and you are not in that context you ought to skip them.

`EXTERNAL USE`

This tag should be used to highlight anything that an HPE Cray internal user should ignore or skip.

`INTERNAL USE`

This tag is used before any block of instruction or text that is only usable or recommended for internal HPE Cray systems.

External (GitHub or customer) should disregard these annotated blocks - they maybe contain useful information as an example but are not intended for their use.

6.5 Command Prompt Conventions

6.5.0.1 Host name and account in command prompts

The host name in a command prompt indicates where the command must be run. The account that must run the command is also indicated in the prompt. - The root or super-user account always has the # character at the end of the prompt - Any non-root account is indicated with account@hostname>. A non-privileged account is referred to as user.

6.5.0.2 Node abbreviations

The following list contains abbreviations for nodes used below

- CN - compute Node
- NCN - Non Compute Node
- AN - Application Node (special type of NCN)
- UAN - User Access Node (special type of AN)
- PIT - Pre-Install Toolkit (initial node used as the inception node during software installation booted from the LiveCD)

Prompt	Description
ncn#	Run the command as root on any NCN, except an NCN which is functioning as an Application Node (AN), such as a UAN.
ncn-m#	Run the command as root on any NCN-M (NCN which is a Kubernetes master node).
ncn-m002#	Run the command as root on the specific NCN-M (NCN which is a Kubernetes master node) which has this hostname (ncn-m002).
ncn-w#	Run the command as root on any NCN-W (NCN which is a Kubernetes worker node).
ncn-w001#	Run the command as root on the specific NCN-W (NCN which is a Kubernetes master node) which has this hostname (ncn-w001).
ncn-s#	Run the command as root on any NCN-S (NCN which is a Utility Storage node).
ncn-s003#	Run the command as root on the specific NCN-S (NCN which is a Utility Storage node) which has this hostname (ncn-s003).
pit#	Run the command as root on the PIT node.
linux#	Run the command as root on a linux host.
uan#	Run the command as root on any UAN.
uan01#	Run the command as root on hostname uan01.
user@uan>	Run the command as any non-root user on any UAN.
cn#	Run the command as root on any CN. Note that a CN will have a hostname of the form nid124356, that is “nid” and a six digit, zero padded number.
hostname#	Run the command as root on the specified hostname.
user@hostname>	Run the command as any non-root user son the specified hostname.

6.5.0.3 Command prompt inside chroot

If the chroot command is used, the prompt changes to indicate that it is inside a chroot environment on the system.

```
hostname# chroot /path/to/chroot
chroot-hostname#
```

6.5.0.4 Command prompt inside Kubernetes pod

If executing a shell inside a container of a Kubernetes pod where the pod name is $podName, the prompt changes to indicate that it is inside the pod. Not all shells are available within every pod, this is an example using a commonly available shell.

```
ncn# kubectl exec -it $podName /bin/sh
pod#
```

6.5.0.5 Command prompt inside image customization session

If using ssh during an image customization session, the prompt changes to indicate that it is inside the image customization environment (pod). This example uses $PORT and $HOST as environment variables with specific settings. When using chroot in this context the prompt will be different than the above chroot example.

```
hostname# ssh -p $PORT root@$HOST
root@POD# chroot /mnt/image/image-root
:/#
```

6.5.0.6 Directory path in command prompt

Example prompts do not include the directory path, because long paths can reduce the clarity of examples. Most of the time, the command can be executed from any directory. When it matters which directory the command is invoked within, the cd command is used to change into the directory, and the directory is referenced with a period (.) to indicate the current directory

Examples of prompts as they appear on the system:

```
hostname:~ # cd /etc
hostname:/etc# cd /var/tmp
hostname:/var/tmp# ls ./file
hostname:/var/tmp# su - user
user@hostname:~> cd /usr/bin
user hostname:/usr/bin> ./command
```

Examples of prompts as they appear in this publication:

```
hostname # cd /etc
hostname # cd /var/tmp
hostname # ls ./file
hostname # su - user
user@hostname > cd /usr/bin
user@hostname > ./command
```

6.5.0.7 Command prompts for network switch configuration

The prompts when doing network switch configuration can vary widely depending on which vendor switch is being configured and the context of the item being configured on that switch. There may be two levels of user privilege which have different commands available and a special command to enter configuration mode.

Example of prompts as they appear in this publication:

Enter “setup” mode for the switch make and model, for example:

```
remote# ssh admin@sw-leaf-001
sw-leaf-001>  enable
sw-leaf-001#  configure terminal
sw-leaf-001(conf)#
```

Refer to the switch vendor OEM documentation for more information about configuring a specific switch.