USER GUIDE

Intel Corporation

Network and Edge Container Bare Metal Reference System Architecture Release v23.02

Authors

Aparna Balachandran

Octavia Carotti

Calin Gherghe

Veronika Karpenko

Dana Nehama

Seungweon Park

Abhijit Sinha

Daniel Ugarte

1 Introduction

1.1 Purpose and Scope

The **Container Bare Metal Reference Architecture (BMRA)** is part of the Network and Edge Reference System Architectures Portfolio. The BMRA is a cloud-native, forward-looking common template platform for network implementations. It addresses the need to deploy cloud-native Kubernetes clusters optimized with Intel hardware and software innovation for diverse workloads across network locations.

Services delivered across the network require deployment of different hardware, software, and configuration specifications due to varying cost, density, and performance requirements. The BMRA common platform allows support for those diverse deployment needs using Network Location Configuration Profiles. Ansible playbooks implement the Configuration Profiles for fast, automatic deployment. The result is an installed optimized BMRA Flavor as specified by the Network Location Configuration Profile.

This user guide covers implementation of BMRA using the Network Location Configuration Profiles as well as deployment using generic Configuration Profiles designed for flexible, a-la-cart deployments.

Network-Location Configuration Profiles covered in this document include:

- **On-Premises Edge Configuration Profile** – Typical customer premises deployment supporting, for example, Video Structuring Server (VSS).
- **Access Edge Configuration Profile** – Far edge wireless-access network deployments. Tuned to support virtual radio access network (vRAN) and FlexRAN™ solution deployments, which require high throughput, low latency, security, and power management control.
- **Remote Central Office-Forwarding Configuration Profile** – Near edge deployments supporting fast packet-forwarding workloads such as cable modem termination system (CMTS), user plane function (UPF), and application gateway function (AGF).
- **Regional Data Center Configuration Profile** – Central-office location typical Configuration Profile. Tailored for video production, visual processing workloads such as CDN transcoding.

Generic Configuration Profiles enable flexible deployments and include the following:

- **Basic Configuration Profile** – A generic minimum BMRA Kubernetes cluster setup.
- **Build-Your-Own Configuration Profile** – A BMRA Kubernetes cluster setup allowing you to select your preferred options.

More information on Configuration Profiles is provided later in this document.

1.2 User Guide Information

This document contains step-by-step instructions on installation, configuration, and use of networking and device plug-in features for deploying the BMRA Release v23.02 per the above Configuration Profiles. Validated, open source Ansible playbooks automatically provision a Kubernetes cluster for the selected Configuration Profile, enabling users to quickly implement predictable deployments.

By following this document, it is possible to set up a Kubernetes cluster and automatically configure it using the Ansible playbooks.

The document provides the following information:

- Part 1 (Sections 2 – 5): Requirements for hardware and software components setup.
- Part 2 (Sections 6 – 12): Step-by-step instructions on how to build your BMRA Flavor using the Ansible scripts. **If you wish to start building the BMRA right away, you may directly go to these sections and start automatically provisioning the BMRA Flavor of your choice**.
- Part 3 (Section 13): BMRA application examples.
- Part 4 (Appendix A): The BMRA Release Notes.
- Part 5 (Appendix B): Abbreviations

1.3 Version 23.02 Release Information

BMRA v23.02 common platform is based on 3rd and 4th Gen Intel® Xeon® Scalable processors and Intel® Xeon® D processors, and the 4th Gen Intel® Xeon® Scalable processor with Intel® vRAN Boost used for vRAN use case support. Other advanced Intel hardware technologies supported include the Intel® Ethernet Controller, Intel® QuickAssist Technology (Intel® QAT), Intel® Server GPU, and Intel® Data Center GPU Flex Series (formerly Artic Sound-M).

The supported software components comprise open-source cloud-native software delivered by Intel, partners, and open-source communities (for example, Kubernetes, Telegraf, Istio, FD.io).

Release v23.02 builds upon release v22.11. The following are the key release updates:

Use Cases Updates:

- Introduced the Edge Video Structuring Server (VSS) use case.
- Introduced support for FlexRAN™ software running as a Docker container on the 3rd Gen Intel Xeon platform. In addition, updated to the latest FlexRAN software release for 3rd and 4th Gen Intel Xeon Scalable processors with Intel® vRAN Boost.
- Upgraded Local Storage and Remote Storage support options by using diverse implementations for Object Storage and Block/File Storage per need (for example, MinIO, Local Persistent Volume Static Provisioner (LPVSP), Rook/Ceph).

  Note: Implemented MinIO Object Storage and Block/file storage across multiple BMRA configuration profiles and eliminated the dedicated BMRA Configuration Profile for Object Storage.

Software Updates:

- Included Media Analytics Libraries in support of the VSS use case. This includes: OpenVINO™ toolkit, Intel® Deep Learning Streamer (Intel® DL Streamer), GStreamer, OpenCL™ software, Level zero GPU, DPC++, and VAAPI from the Intel® GPU toolkit.
- Supported Rook/Ceph for storage support.
- Added Rocky Linux 9.1 as base operating system.
- Support of geo-specific mirrors for Kubespray (for example, in the People's Republic of China).
- Software version upgraded for the majority of RA components (See elsewhere in this document for complete BOM and versions).

Under NDA:

- Select capabilities available under NDA are integrated and validated with the BMRA. Contact your Intel representative for access to the following NDA material: Intel® FlexRAN™ software 22.11.

For additional details, refer to the RA Release Notes.

Experience Kits, the collaterals that explain in detail the technologies enabled in BMRA release 23.02, including benchmark information, are available on Intel Network Builder at Network & Edge Platform Experience Kits.

Table of Contents

Figures

Tables

1.4 Document Revision History

Three previous editions of the BMRA document were released, starting April 2019.

- Covered 2nd Gen Intel® Xeon® Scalable processors
- Covered 2nd and 3rd Gen Intel® Xeon® Scalable processors
- Covered 2nd and 3rd Gen Intel® Xeon® Scalable processors and Intel® Xeon® D processor

REVISION	DATE	DESCRIPTION
001	February 2022	Initial release.
002	March 2022	Updated a few URLs.
003	June 2022	Covers the 4th Gen Intel® Xeon® Scalable processor (formerly codenamed Sapphire Rapids). See "Version 22.05 Release Information" for details.
004	June 2022	Changes include updates to the discussion of the BMRA for Storage Deployment Model.
005	July 2022	Added NDA support for FlexRAN™ software, updated Istio and service mesh features.
006	August 2022	The changes include updates to the discussions of the Access Edge Configuration Profile and Intel® Ethernet Operator.
007	October 2022	Updated for BMRA Release 22.08; added information about the new Cloud Reference System Architecture (Cloud RA) deployment model.
008	December 2022	Updated for BMRA Release 22.11; includes improvements and updates on RA in alignment with the launch of the 4th Gen Intel® Xeon® Scalable processor.
009	March 2023	Updated for BMRA Release 23.02; includes improvements to run FlexRAN™ software in a container and addition of Media Analytics Libraries.

1.5 Key Terms

Table 1 lists the key terms used throughout the portfolio. These terms are specific to Network and Edge Reference System Architectures Portfolio deployments.

Table 1. Terms Used

TERM	DESCRIPTION
Experience Kits	Guidelines delivered in the form of—manuals, user guides, application notes, solution briefs, training videos—for best-practice implementation of cloud native and Kubernetes technologies to ease developments and deployments.
Network and Edge Reference System Architectures Portfolio	A templated system-level blueprint for a range of locations in enterprise and cloud infrastructure with automated deployment tools. The portfolio integrates the latest Intel platforms and cloud-native technologies for multiple deployment models to simplify and accelerate deployments of key workloads across a service infrastructure.
Deployment Model	Provides flexibility to deploy solutions according to IT needs. The portfolio offers three deployment models: • **Container Bare Metal Reference System Architecture (BMRA)** – A deployment model of a Kubernetes cluster with containers on a bare metal platform. • **Virtual Machine Reference System Architecture (VMRA)** – A deployment model of a virtual cluster on a physical node. The virtual cluster can be a Kubernetes containers-based cluster. • **Cloud Reference System Architecture (Cloud RA)** – A deployment model that uses CSP's Intel-based instances for running cloud-native applications in the Cloud. The worker instances are provided based on the Configuration Profile that workload demands.
Configuration Profiles	A prescribed set of components—hardware, software modules, hardware/software configuration specifications—designed for a deployment for specific workloads at a network location (such as Access Edge). Configuration Profiles define the components for optimized performance, usability, and cost per network location and workload needs.[1] In addition, generic Configuration Profiles are available for developers' flexible deployments.
Reference Architecture Flavor	A Reference Architecture deployment using a Configuration Profile.
Ansible Playbook	A set of validated scripts that prepare, configure, and deploy a Reference Architecture Flavor by implementing a Configuration Profile.
Configuration Profile Ansible Scripts	Automates quick, repeatable, and predictive deployments using Ansible playbooks. Various Configuration Profiles and Ansible scripts allow automated installations that are application-ready, depending on the workload and network location.
Kubernetes Cluster	A deployment that installs at least one worker node running containerized applications. Pods are the components of the application workload that are hosted on worker nodes. Control nodes manage the pods and worker nodes.
Intel Platforms	Prescribes Intel platforms for optimized operations. The platforms are based on 3rd and 4th Gen Intel® Xeon® Scalable processors plus the Intel® Xeon® D processor. These platforms include the Taylors Falls Reference Design. The platforms integrate Intel® Ethernet Controller 700 Series and 800 Series, Intel® QuickAssist Technology (Intel® QAT), Intel® Server GPU (graphics processing unit), Intel® Optane™ technology, and more. ***Note:*** This release of VMRA does not support the Intel Xeon D processor.

In addition to key terms, portfolio deployment procedures follow a hardware and software configuration taxonomy. Table 2 describes the taxonomy used throughout this document.

[1] Workloads and configurations. Results may vary.

Table 2. Hardware and Software Configuration Taxonomy

TERM	DESCRIPTION
Hardware Taxonomy	
ENABLED	Setting must be enabled in the BIOS (configured as Enabled, Yes, True, or similar value)
DISABLED	Setting must be disabled in the BIOS (configured as Disabled, No, False, or any other value with this meaning.)
OPTIONAL	Setting can be either disabled or enabled, depending on workload. Setting does not affect the Configuration Profile or platform deployment
Software Taxonomy	
TRUE	Feature is included and enabled by default
FALSE	Feature is included but disabled by default - can be enabled and configured by user
N/A	Feature is not included and cannot be enabled or configured

1.6 Intel Investments of Capabilities

Intel investments in networking solutions are designed to help IT centers accelerate deployments, improve operational efficiencies, and lower costs. Table 3 highlights Intel investments in the portfolio and their benefits.

Table 3. Intel Capabilities Investments and Benefits

CAPABILITY	BENEFIT
Performance	Intel® platform innovation and accelerators, combined with packet processing innovation for cloud-native environments, deliver superior and predictive application and network performance.
Orchestration and Automation	Implementing Kubernetes containers orchestration, including Kubernetes Operators, simplifies and manages deployments and removes barriers in Kubernetes to support networking functionality.
Observability	Collecting platform metrics by using, as an example, the collectd daemon and Telegraf server agent, publishing the data, and generating reports, enables high visibility of platform status and health.
Power Management	Leveraging Intel platform innovation, such as Intel® Speed Select Technology (Intel® SST), supports optimized platform power utilization.
Security	Intel security technologies help ensure platform and transport security. These technologies include the following: • Intel® Security Libraries for Data Center (Intel® SecL - DC) • Intel® QuickAssist Technology Engine for OpenSSL (Intel® QAT Engine for OpenSSL) • Intel® Software Guard Extensions (Intel® SGX) • Key Management Reference Application (KMRA) implementation
Storage	Creating a high-performance, scalable local-storage or remote-storage platform using diverse storage technologies (Object Storage; File/Block storage) and implementations. For example, MinIO implementation for remote Object Storage supports data-intensive applications, such as media streaming, big data analytics, AI, and machine learning.
Service Mesh	Implementing a service mesh architecture using Istio allows application services that can be added, connected, monitored, more secure, and load-balanced with few or no code changes. Service mesh is integrated with the Trusted Certificate Service for Kubernetes* platform, providing more secure key management.

1.7 Reference Documentation

The Network and Edge Reference System Architectures Portfolio User Manual contains a complete list of reference documents. A virtual machine-based reference architecture (VMRA) deployment allows creation of a Kubernetes cluster for a Configuration Profile on a virtualized infrastructure. The Network and Edge Virtual Machine Reference System Architecture User Guide provides information and installation instructions for a VMRA. The Cloud Reference System Architecture (Cloud RA) provides the means to develop and deploy cloud-native applications in a CSP environment and still experience Intel® technology benefits. Find more details in the Network and Edge Cloud Reference System Architecture User Guide.

Other collaterals, including technical guides and solution briefs that explain in detail the technologies enabled in this BMRA release, are available in the following location: Network & Edge Platform Experience Kits.

Part 1:

Reference Architecture Components and Deployment Guidelines:

Ansible Playbooks

Hardware Components

Software Ingredients

Recommended Configurations

2 Reference Architecture Deployment

This chapter explains how a BMRA Flavor is generated and deployed. The process includes installation of the hardware setup followed by system provisioning.

2.1 BMRA Architecture

The BMRA is a Kubernetes cluster that can be configured to support a flexible number of Kubernetes control nodes and worker nodes (see Figure 1). To deploy the BMRA, you deploy and configure the following elements:

- **Hardware Components**: Multiple platform hardware options are available, including a variety of 4th and 3rd Gen Intel® Xeon® Scalable processor SKUs, Intel® Xeon® D processor SKUs, Intel® Ethernet Network Adapters, Intel® QAT, and Intel® Server GPU. BIOS options are listed elsewhere in this guide. Deployment engineers should refer to Section 3.6 during deployment to select and configure optimal BIOS values before cluster provisioning.
- **Software Capabilities**: The software capabilities are based on open-source software delivered by cloud-native and CNCF communities driving Kubernetes, Istio, observability, DPDK, FD.io. OVS, OVS-DPDK, and through Intel GitHub. Options for RHEL and Ubuntu Linux operating systems are available. The container environment is based on Docker, containerd, or CRI-O container runtimes.
- **Configuration Profiles**: Specific hardware and software configurations are provided in the Configuration Profiles based on Intel assessment and verification. Hardware configurations address two performance capabilities: base and plus.
- **Installation Playbooks**: Ansible playbooks implement the Configuration Profiles for best-practice, reliable, and accelerated BMRA Flavor deployment.

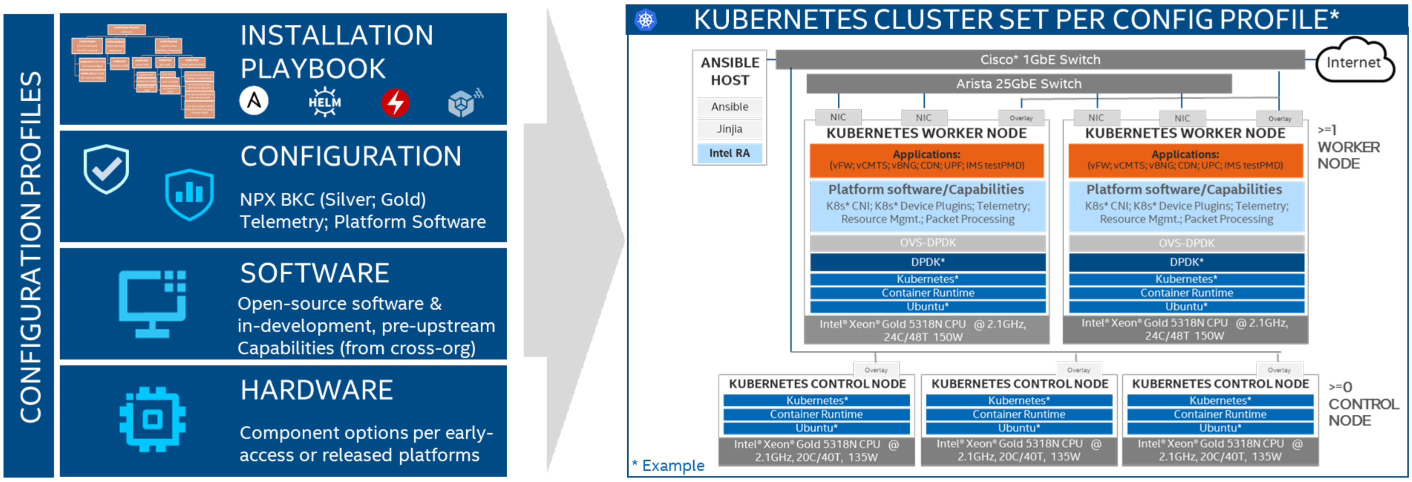

CONFIGURATION PROFILES
INSTALLATION PLAYBOOK
CONFIGURATION
NPX BKC (Silver; Gold)
Telemetry; Platform Software
SOFTWARE
Open-source software &
in-development, pre-upstream
Capabilities (from cross-org)
HARDWARE
Component options per early-access or released platforms
KUBERNETES CLUSTER SET PER CONFIG PROFILE*
ANSIBLE HOST
Ansible
Jinjia
Intel RA
Cisco* 1GbE Switch
Internet
Arista 25GbE Switch
NIC
NIC
Overlay
KUBERNETES WORKER NODE
Applications:
(vFW; vCMTS; vBNG; CDN; UPF; IMS testPMD)
Platform software/Capabilities
K8s* CNI; K8s* Device Plugins; Telemetry; Resource Mgmt.; Packet Processing
OVS-DPDK
DPDK*
Kubernetes*
Container Runtime
Ubuntu*
Intel® Xeon® Gold 5318N CPU @ 2.1GHz, 24C/48T 150W
NIC
NIC
Overlay
KUBERNETES WORKER NODE
Applications:
(vFW; vCMTS; vBNG; CDN; UPC; IMS testPMD)
Platform software/Capabilities
K8s* CNI; K8s* Device Plugins; Telemetry; Resource Mgmt.; Packet Processing
OVS-DPDK
DPDK*
Kubernetes*
Container Runtime
Ubuntu*
Intel® Xeon® Gold 5318N CPU @ 2.1GHz, 24C/48T 150W
>=1 WORKER NODE
Overlay
KUBERNETES CONTROL NODE
Kubernetes*
Container Runtime
Ubuntu*
Intel® Xeon® Gold 5318N CPU @ 2.1GHz, 20C/40T, 135W
Overlay
KUBERNETES CONTROL NODE
Kubernetes*
Container Runtime
Ubuntu*
Intel® Xeon® Gold 5318N CPU @ 2.1GHz, 20C/40T, 135W
Overlay
KUBERNETES CONTROL NODE
Kubernetes*
Container Runtime
Ubuntu*
Intel® Xeon® Gold 5318N CPU @ 2.1GHz, 20C/40T, 135W
>=0 CONTROL NODE
* Example

Figure 1. BMRA Illustration and Applicable Elements

2.2 Configuration Profiles

A Configuration Profile describes specific hardware and software bills of material (BOM) and configurations, applicable for a specific deployment. Configuration Profiles consider the best-known configuration (BKC) validated by Intel for optimized performance.[2]

Installation scripts are available to deploy the required components for a BMRA Flavor. Each BMRA is built on the following:

- **Intel Platform foundation** with Intel processors and technologies.
- **Hardware BOM** optimized for delivering an application at a specific location using a deployment model. For example, to support a UPF workload at the Remote CO, the BMRA deployment is populated with the maximum available Ethernet adapters or network interface cards (NICs).
- **Software BOM** leverages the Intel platform and enables cloud-native adoption.
- **Installation (Ansible) Playbook** automates the installation of a BMRA Flavor per a Configuration Profile specification.

The following Reference Architecture Configuration Profiles are network location-specific:

- **On-Premises Edge Configuration Profile** – Small cluster of stationary or mobile server platforms, ranging from one to four servers. Usage scenarios include data collection from sensors, local (edge) processing, and upstream data transmission. Sample locations are hospitals, factory floors, law enforcement, media, cargo transportation, and power utilities. This Configuration Profile recommends a Kubernetes cluster hardware configuration, software capabilities, and specific hardware and software configurations that typically support enterprise edge workloads used in Smart City deployments and Ad-insertion.
- **Access Edge Configuration Profile** – A small cluster designed to support cellular access network deployments, typically in an

[2] Workloads and configurations. Results may vary.

outside plant in harsh, minimally controlled temperature cabinets. Targeted use cases are 5G Virtual Radio Access Networks (vRAN) and FlexRAN™ 5G solutions that require high throughput, low latency, security, and power management control.
- **Remote Central Office-Forwarding Configuration Profile** – Clusters ranging from a half rack to a few racks of servers, typically in a pre-existing, repurposed, unmanned structure. The usage scenarios include running latency-sensitive applications near the user (for example, real-time gaming, stock trading, video conferencing). This Configuration Profile addresses a Kubernetes cluster hardware, software capabilities, and configurations that enable high performance for packet forwarding packets. In this category, you can find workloads such as UPF, vBNG, vCMTS, and vCDN.
- **Regional Data Center Configuration Profile** – The Regional Data Center consists of a management domain with many racks of servers, typically managed and orchestrated by a single instance of resource orchestration. Usage scenarios include services such as content delivery, media, mobile connectivity, and cloud services. This Configuration Profile is tailored exclusively and defined for Media Visual Processing workloads such as CDN Transcoding.

Additional Reference Architecture Configuration Profiles are not location-specific and enable flexible deployments per need:
- **Basic Configuration Profile** – A minimum set of software features where network acceleration is the only concern.
- **Build-Your-Own Configuration Profile** – A complete set of all available software features targeted at developers and deployers that are looking to evaluate, control, and configure all of the software and hardware ingredients and dependencies individually.

2.3 Reference Architecture Installation Prerequisites

This section helps you get ready to run the Ansible scripts. Before the Ansible playbook can begin, you must identify the required hardware components, ensure hardware connectivity, and complete the initial configuration, for example BIOS setup. This section describes the minimal system prerequisites needed for the Ansible host and Kubernetes control and worker nodes. It also lists the steps required to prepare hosts for successful deployment. Detailed instructions are provided in relative sections, which are referred to in this section. Steps include:
- Hardware BOM selection and setup
- Required BIOS/UEFI configuration, including virtualization and hyper-threading settings
- Network topology requirements – a list of necessary network connections between the nodes
- Installation of software dependencies needed to execute Ansible playbooks
- Generation and distribution of SSH keys that are used for authentication between the Ansible host and Kubernetes cluster target servers

After satisfying these prerequisites, Ansible playbooks for 3rd and 4th Gen Intel Xeon Scalable processors and Intel Xeon D processors can be downloaded directly from the dedicated GitHub page (Container Experience Kits Releases) or cloned using the Git. Request access to the NDA Ansible playbooks for 4th Gen Intel Xeon Scalable processors from your regional Intel representative.

2.3.1 Hardware BOM Selection and Setup for Control and Worker Nodes

Before software deployment and configuration, deploy the physical hardware infrastructure for the site. To obtain ideal performance and latency characteristics for a given network location, Intel recommends the hardware BOMs and configurations described in the following sections:

- Control nodes - Review Section 3.1 for recommended control node assembly.
- Worker nodes – Refer to the following sections for recommended worker node assembly:
  - Base worker node – Review Section 3.2 to satisfy base performance characteristics.
  - Plus worker node – Review Section 3.3 to satisfy plus performance characteristics.
- Configuration Profile BOM – See Sections 7 through 12 for details about hardware BOM selection and setup for your chosen Configuration Profile.

2.3.2 BIOS Selection for Control and Worker Nodes

Enter the UEFI or BIOS menu and update the configuration as listed in Section 6 and in the tables in Section 3.6, which describe the BIOS selection in detail.

2.3.3 Operating System Selection for Ansible Host and Control and Worker Nodes

The following Linux operating systems are supported for Control and Worker Nodes:
- RHEL for x86_64 Version 9 (9.0)
- RHEL 8.6 RT
- Rocky Linux 9.1
- Ubuntu 22.04
- Ubuntu 22.04 RT

For all supported distributions, the base operating system installation images are sufficient when built using the "Minimal" option during installation. In addition, the following must be met:

- The Control and Worker Nodes must have network connectivity to the Ansible host.
- All systems must have public internet connectivity.
- SSH connections are required. If needed on Ubuntu, install SSH Server with the following commands (internet access is required):

```
# sudo apt update
# sudo apt install openssh-server
```

2.3.4 Network Interface Requirements for Control and Worker Nodes

The following list provides a brief description of different networks and network interfaces needed for deployment.

- Internet network
  – Ansible host accessible
  – Capable of downloading packages from the internet
  – Can be configured for Dynamic Host Configuration Protocol (DHCP) or with static IP address
- Management network and Calico pod network interface (This can be a shared interface with the internet network)
  – Kubernetes control and worker node inter-node communications
  – Calico pod network runs over this network
  – Configured to use a private static address
- Tenant data networks
  – Dedicated networks for traffic
  – Single Root Input/Output Virtualization (SR-IOV) enabled
  – Virtual function (VF) can be DPDK bound in pod

2.4 Ansible Playbook

This section describes how the Ansible playbooks allow for an automated deployment of a fully functional BMRA cluster, including initial system configuration, Kubernetes deployment, and setup of capabilities as described in Section 2.5.

2.4.1 Ansible Playbook Building Blocks

The following components make up the BMRA Ansible playbooks.

Note: Ansible playbooks for 3rd and 4th Gen Intel Xeon Scalable processors and Intel® Xeon® D processors are open source and available here.

Configuration Files provide examples of cluster-wide and host-specific configuration options for each of the Configuration Profiles. With minimal changes, these configuration files can be used directly with their corresponding playbooks. The path to these configuration files is:

- `inventory.ini`
- `group_vars/all.yml`
- `host_vars/node1.yml`

For default values in these files, refer to the Configuration Profile-specific sections for BMRA installations:

Section 7, BMRA Basic Configuration Profile Setup
Section 8, BMRA Build-Your-Own Configuration Profile Setup
Section 9, BMRA On-Premises Edge Configuration Profile Setup
Section 10, BMRA Access Edge Configuration Profile Setup
Section 11, BMRA Remote Central Office-Forwarding Configuration Profile Setup
Section 12, BMRA Regional Data Center Configuration Profile Setup

2.4.2 Ansible Playbook Phases

Regardless of the selected Configuration Profile, the installation process always consists of three main phases:

1. **Infrastructure Setup** (sub-playbooks in `playbooks/infra/` directory)
   These playbooks modify kernel boot parameters and apply the initial system configuration for the cluster nodes. Depending on the selected Configuration Profile, Infrastructure Setup includes:
   – Generic host OS preparation, e.g., installation of required packages, Linux kernel configuration, proxy and DNS configuration, and modification of SELinux policies and firewall rules.
   – Configuration of the kernel boot parameters according to the user-provided configuration in order to configure CPU isolation, SR-IOV related settings such as IOMMU, hugepages, or explicitly enable/disable Intel P-state technology.
   – Configuration of SR-IOV capable network cards and QAT devices. This includes the creation of virtual functions and binding to appropriate Linux kernel modules.

- Network Adapter drivers and firmware updates, which help ensure that all latest capabilities such as Dynamic Device Personalization (DDP) profiles are enabled.
- Intel® Speed Select Technology (Intel® SST) configuration, which provides control over base frequency.
- Installation of DDP profiles, which can increase packet throughput, help reduce latency, and lower CPU usage by offloading packet classification and load balancing to the network adapter.

2. **Kubernetes Setup** (in `playbooks/k8s/` directory)
   This playbook deploys a high availability (HA) Kubernetes (K8s) cluster using Kubespray, which is a project under the Kubernetes community that deploys production-ready Kubernetes clusters. The Multus container network interface (CNI) plugin, which is specifically designed to support multiple networking interfaces in a Kubernetes environment, is deployed by Kubespray along with Calico and Helm. Preferred security practices are used in the default configuration. On top of Kubespray, there is also a container registry instance deployed to store images of various control-plane Kubernetes applications, such as Telemetry Aware Scheduling (TAS), CPU Manager for Kubernetes (CMK), or device plugins.
3. **BMRA System Capabilities Setup** (sub-playbooks in the `playbooks/intel` directory)
   Advanced networking technologies, enhanced platform awareness, and device plugin features are deployed by this playbook using operators or Helm charts as part of the BMRA. The following capabilities are deployed:
   - Device plugins that allow using, for example, SR-IOV, QAT, and GPU devices in workloads running on top of Kubernetes.
   - SR-IOV CNI plugin, Bond CNI plugin, and Userspace CNI plugin, which allow Kubernetes pods to be attached directly to accelerated and highly available hardware and software network interfaces.
   - Native CPU Manager for Kubernetes (replacement for CMK), which performs a variety of operations to enable core pinning and isolation on a container or a thread level.
   - Node Feature Discovery (NFD), which is a Kubernetes add-on to detect and advertise hardware and software capabilities of a platform that can, in turn, be used to facilitate intelligent scheduling of a workload.
   - Telemetry Aware Scheduling (TAS), which allows scheduling workloads based on telemetry data.
   - Full Telemetry Stack consisting of collectd, Kube-Prometheus, Jaeger, OpenTelemetry and Grafana, which provides cluster and workload monitoring capabilities and acts as a source of metrics that can be used in TAS to orchestrate scheduling decisions.
   - MinIO operator/console, which supports deploying MinIO tenants onto private and public cloud infrastructures ("Hybrid" Cloud).

2.5 Deployment Using Ansible Playbook

This section describes common steps to obtain the BMRA Ansible Playbooks source code, prepare target servers, configure inventory and variable files, and deploy the BMRA Kubernetes cluster.

2.5.1 Prepare Target Servers

For each target server that will act as a control or worker node, you must make sure that it meets the following requirements:

- Install Python 3. The following example assumes that the host is running RHEL. Other operating systems may have slightly different installation steps:

  ```
  yum install python3
  ```
- Internet access on all target servers is mandatory. Proxies are supported and can be configured in the Ansible vars.
- BIOS configuration matching the desired state is applied. For details, refer to the specific Configuration Profile section below for your profile:
  Section 7, BMRA Basic Configuration Profile Setup
  Section 8, BMRA Build-Your-Own Configuration Profile Setup
  Section 9, BMRA On-Premises Edge Configuration Profile Setup
  Section 10, BMRA Access Edge Configuration Profile Setup
  Section 11, BMRA Remote Central Office-Forwarding Configuration Profile Setup
  Section 12, BMRA Regional Data Center Configuration Profile Setup

For detailed steps on how to build the Ansible host, refer to Section 6.1.

2.5.2 Prepare Ansible Host and Configuration Templates

Perform the following steps:

1. Log in to your Ansible host (the one that you will run these Ansible playbooks from).
2. Install packages on Ansible host. The following example assumes that the host is running RHEL. Other operating systems may have slightly different installation steps:
   ```
   yum install python3
   pip3 install --upgrade pip
   ```
3. Enable passwordless login between all nodes in the cluster.
   Create authentication SSH-Keygen keys on Ansible host:
   ```
   ssh-keygen
   ```

4. SSH is used by the Ansible host to communicate with each target node. Configure the same SSH keys on each machine. Copy your generated public keys to all the nodes from the Ansible host:
   ```
   ssh-copy-id root@<target_server_address>
   ```
5. Clone the source code and change working directory.
   ```
   git clone https://github.com/intel/container-experience-kits/
   cd container-experience-kits
   ```
   Check out the latest version of the playbooks – using the tag from Table 33, for example:
   ```
   git checkout v23.02
   ```
   Note: Alternatively go to Container Experience Kits Releases, download the latest release tarball, and unarchive it:
   ```
   wget https://github.com/intel/container-experience-kits/archive/v23.02.tar.gz
   tar xzf v23.02.tar.gz
   cd container-experience-kits-23.02
   ```
6. Initialize Git submodules to download Kubespray code.
   ```
   git submodule update --init
   ```
7. Decide which Configuration Profile that you want to deploy and export the environmental variable.

   For Kubernetes **Basic** Configuration Profile deployment:
   ```
   export PROFILE=basic
   ```
   For Kubernetes **Build-Your-Own** Configuration Profile deployment:
   ```
   export PROFILE=build_your_own
   ```
   For Kubernetes **On-Premises Edge** Configuration Profile deployment:
   ```
   export PROFILE=on_prem
   ```
   For Kubernetes **Access Edge** Configuration Profile deployment:
   ```
   export PROFILE=access
   ```
   For Kubernetes **Remote Central Office-Forwarding** Configuration Profile deployment:
   ```
   export PROFILE=remote_fp
   ```
   For Kubernetes **Regional Data Center** Configuration Profile deployment:
   ```
   export PROFILE=regional_dc
   ```
8. Install requirements needed by deployment scripts.
   ```
   pip3 install -r requirements.txt
   ```
9. Generate example profiles. Be aware of the machine's architecture and data plane network before generating profiles. Example machine architectures (ARCH) are `spr`, `icx`, and `clx` and data plane networks network adapters are `fvl` and `cvl`.
   ```
   make k8s-profile PROFILE=$PROFILE ARCH=spr NIC=cvl
   ```

2.5.3 Update Ansible Inventory File

Perform the following steps:

1. Edit the `inventory.ini` file generated in the previous steps.
2. In section `[all]`, specify all your target servers. Use their actual hostnames and Management IP addresses. Also update `ansible_user` and `ansible_password` to match the SSH configuration of the target servers. If any of the servers are configured with passwordless SSH, the `ansible_password` host variable can be removed.
   ```
   [all]
   controller1 ansible_host=10.0.0.1 ip=10.0.0.1 ansible_user=USER ansible_password=XXXX
   controller2 ansible_host=10.0.0.2 ip=10.0.0.2 ansible_user=USER ansible_password=XXXX
   controller3 ansible_host=10.0.0.3 ip=10.0.0.3 ansible_user=USER ansible_password=XXXX
   node1   ansible_host=10.0.0.4 ip=10.0.0.4 ansible_user=USER ansible_password=XXXX
   node2   ansible_host=10.0.0.5 ip=10.0.0.5 ansible_user=USER ansible_password=XXXX
   localhost ansible_connection=local ansible_python_interpreter=/usr/bin/python3
   [vm_host]
   [kube_control_plane]
   controller1
   controller2
   controller3
   [etcd]
   controller1
   controller2
   controller3
   [kube_node]
   node1
   node2
   [k8s_cluster:children]
   kube_control_plane
   kube_node
   ```

```
[all:vars]
ansible_python_interpreter=/usr/bin/python3
```

2.5.4 Update Ansible Host and Group Variables

Perform the following steps.

1. Create `host_vars/<hostname>.yml` files for all worker nodes, matching their hostnames from the inventory file. The provided `host_vars/node1.yml` file can be used as a template.
2. Edit all `host_vars/<hostname>.yml` and `group_vars/all.yml` files to match your desired configuration. Each Configuration Profile uses its own set of variables. Refer to the specific Configuration Profile section for your profile to get a full list of variables and their documentation:
   Section 7, BMRA Basic Configuration Profile Setup
   Section 8, BMRA Build-Your-Own Configuration Profile Setup
   Section 9, BMRA On-Premises Edge Configuration Profile Setup
   Section 10, BMRA Access Edge Configuration Profile Setup
   Section 11, BMRA Remote Central Office-Forwarding Configuration Profile Setup
   Section 12, BMRA Regional Data Center Configuration Profile Setup

2.5.5 Run Ansible Cluster Deployment Playbook

After the inventory and vars are configured, you can run the provided playbooks from the root directory of the project.

It is recommended that you check dependencies of components enabled in `group_vars` and `host_vars` with the packaged dependency checker:

```
ansible-playbook -i inventory.ini playbooks/preflight.yml
```

If you are deploying an RHEL 8 cluster, you need to patch Kubespray:

```
ansible-playbook -i inventory.ini playbooks/k8s/patch_kubespray.yml
```

Otherwise, you can skip directly to your chosen Configuration Profile playbook:

```
ansible-playbook -i inventory.ini playbooks/${PROFILE}.yml
```

Pay attention to logs and messages displayed on the screen. Depending on the selected Configuration Profile, network bandwidth, storage speed, and other similar factors, the execution may take up to 30-40 minutes.

After the playbook finishes without any “Failed” tasks, you can proceed with the deployment validation described in Section 5.

Note: Additional information can be found in the Ansible project root directory readme.

2.5.6 Run Ansible Cluster Removal Playbook

If the playbook fails or if you want to clean up the environment to run a new deployment, you can optionally use the provided Cluster Removal Playbook (`redeploy_cleanup.yml`) to remove any previously installed Kubernetes and related plugins.

```
ansible-playbook -i inventory.ini playbooks/redeploy_cleanup.yml
```

After successful removal of Kubernetes components, you can repeat Section 2.5.5.

Note: Any OS and/or hardware configurations (for example, proxies, drivers, kernel parameters) are not reset by the cleanup playbook.

3 Reference Architecture Hardware Components and BIOS

For all Configuration Profiles, this section provides a menu of all possible hardware components for control node and worker node as well as the BIOS components available.

3.1 Hardware Components List for Control Node

The following tables list the hardware options for control nodes.

Table 4. Hardware Options for Control Node – 3rd Gen Intel Xeon Scalable Processor

INGREDIENT	REQUIREMENT	REQUIRED/ RECOMMENDED
3rd Gen Intel Xeon Scalable processors	Intel® Xeon® Gold 5318N processor at 2.1 GHz, 20 C/40 T, 135 W, or higher number Intel® Xeon® Gold or Platinum CPU SKU	Required
Memory	256 GB DRAM (16 x 16 GB DDR4, 2666 MHz)	Required
Network Adapter	Dual Port 100 GbE Intel® Ethernet Network Adapter E810-CQDA2 QSFP28	Required
Intel® QAT	Intel® QuickAssist Adapter 8960 or 8970 (PCIe*) AIC or equivalent third-party Intel® C620 Series Chipset	Recommended
Storage (Boot Drive)	Intel® SATA Solid State Drive D3 S4510 at 480 GB or equivalent boot drive	Required
Storage (Capacity)	Intel® SSD D7-P5510 Series at 3.84 TB or equivalent drive (recommended NUMA aligned)	Recommended
LAN on Motherboard (LOM)	10 Gbps or 25 Gbps port for Preboot Execution Environment (PXE) and Operation, Administration, and Management (OAM)	Required
	1/10 Gbps port for Management Network Adapter	Required
Additional Plug-in cards	N/A	

Table 5. Hardware Options for Control Node – 4th Gen Intel Xeon Scalable Processor

INGREDIENT	REQUIREMENT	REQUIRED/ RECOMMENDED
4th Gen Intel Xeon Scalable processors	Intel® Xeon® Gold 5418N processor at 2.0 GHz, 24 C/48 T, 165 W	Required
Memory	DRAM only configuration: 256 GB DRAM (16 x 16 GB DDR5)	Required
Network Adapter	Intel® Ethernet Network Adapter E810-CQDA2 or E810-XXVDA2	Required
Intel® QAT	Integrated in the processor	
Storage (Boot Drive)	Intel® SATA Solid State Drive D3 S4510 at 480 GB or equivalent boot drive	Required
Storage (Capacity)	Intel® SSD D7-P5510 Series at 3.84 TB or equivalent drive (recommended NUMA aligned)	Recommended
LAN on Motherboard (LOM)	10 Gbps or 25 Gbps port for Preboot Execution Environment (PXE) and Operation, Administration, and Management (OAM)	Required
	1/10 Gbps port for Management Network Adapter	Required
Additional Plug-in cards	N/A	

Table 6. Hardware Options for Control Node – Intel® Xeon® D Processor

INGREDIENT	REQUIREMENT	REQUIRED/ RECOMMENDED
Intel® Xeon® D processors	Intel® Xeon® D-1700 processor, 4 core LCC, 45 W, or higher	Required
Memory	DRAM only configuration: 16 GB DDR4 2933 MHz	Required
Network Adapter	2 x 10 GbE integrated Ethernet ports	Required
Intel® QAT	20 G Intel® QAT	Recommended
Storage (Boot Drive)	Intel® SSD 256 GB 2.5" internal SSD/M.2	Required
Additional Plug-in cards	N/A	

3.2 Hardware Components List for Worker Node Base

The following tables list the hardware options for worker nodes in the “base” configuration. If your configuration needs improved processing, you may choose to use the “plus” configuration instead (Section 3.3).

Table 7. Hardware Components for Worker Node Base – 3rd Gen Intel Xeon Scalable Processor

INGREDIENT	REQUIREMENT	REQUIRED/ RECOMMENDED
3rd Gen Intel Xeon Scalable processors	Intel® Xeon® Gold 5318N processor at 2.1 GHz, 24 C/48 T, 150 W, or higher number Intel® Xeon® Gold or Platinum CPU SKU	Required
Memory	Option 1: DRAM only configuration: 256 GB (8 x 32 GB DDR4, 2666 MHz)	Required
	Option 2: DRAM only configuration: 256 GB (16 x 16 GB DDR4, 2666 MHz)	
Intel® Optane™ Persistent Memory	512 GB (4x 128 GB Intel® Optane™ persistent memory in 2-1-1 Topology)	Recommended
Network Adapter	Option 1: Intel® Ethernet Network Adapter E810-CQDA2	Required
	Option 2: Intel® Ethernet Network Adapter E810-XXVDA2	
Intel® QAT	Intel® QuickAssist Adapter 8960 or 8970 (PCIe*) AIC or equivalent third-party Intel® C620 Series Chipset	Required
Storage (Boot Drive)	Intel® SATA Solid State Drive D3 S4510 at 480 GB or equivalent boot drive	Required
Storage (Capacity)	Intel® SSD D7-P5510 Series at 3.84 TB or equivalent drive (recommended NUMA aligned)	Required
LAN on Motherboard (LOM)	10 Gbps or 25 Gbps port for Preboot Execution Environment (PXE) and Operation, Administration, and Management (OAM)	Required
	1/10 Gbps port for Management Network Adapter	Required
Additional Plug-in cards	N/A	

Table 8. Hardware Components for Worker Node Base – 4th Gen Intel Xeon Scalable Processor

INGREDIENT	REQUIREMENT	REQUIRED/ RECOMMENDED
4th Gen Intel Xeon Scalable processors	Intel® Xeon® Gold 5418N processor at 2.0 GHz, 24 C/48 T, 165 W	Required
Memory	DRAM only configuration: 256 GB DRAM (16 x 16 GB DDR5)	Required
Intel® Optane™ Persistent Memory	512 GB (4 x 128 GB Intel® Optane™ persistent memory in 2-1-1 topology)	Recommended
Network Adapter	Option 1: Intel® Ethernet Network Adapter E810-CQDA2	Required
	Option 2: Intel® Ethernet Network Adapter E810-2CQDA2	
Intel® QAT	Integrated in the processor	
Storage (Boot Drive)	Intel® SATA Solid State Drive D3 S4510 at 480 GB or equivalent boot drive	Required
Storage (Capacity)	Intel® SSD D7-P5510 Series at 3.84 TB or equivalent drive (recommended NUMA aligned)	Required
LAN on Motherboard (LOM)	10 Gbps or 25 Gbps port for Preboot Execution Environment (PXE) and Operation, Administration, and Management (OAM)	Required
	1/10 Gbps port for Management Network Adapter	Required
Additional Plug-in cards	N/A	

Table 9. Hardware Components for Worker Node Base (Access Edge - vRAN) – 4th Gen Intel Xeon Scalable Processor

INGREDIENT	REQUIREMENT	REQUIRED/ RECOMMENDED
4th Gen Intel Xeon Scalable processors	Intel® Xeon®-SP 5411N 24 C/48 T 1.9 GHz 165 W	Required
Memory	DRAM only configuration: 128 GB DRAM (8 x 16 GB DDR5)	Required
Intel® Optane™ Persistent Memory	512 GB (4 x 128 GB Intel® Optane™ persistent memory in 2-1-1 topology)	Recommended
Network Adapter	Option 1: Intel® Ethernet Network Adapter E810-CQDA2	Required
	Option 2: Intel® Ethernet Network Adapter E810-XXVDA4	
Intel® QAT	Integrated in the processor	
Storage (Boot Drive)	Intel® SATA Solid State Drive D3 S4510 at 480 GB or equivalent boot drive	Required
Storage (Capacity)	Intel® SSD D7-P5510 Series at 3.84 TB or equivalent drive (recommended NUMA aligned)	Required
LAN on Motherboard (LOM)	10 Gbps or 25 Gbps port for Preboot Execution Environment (PXE) and Operation, Administration, and Management (OAM)	Required
	1/10 Gbps port for Management Network Adapter	Required
Additional Plug-in cards	Intel® vRAN Accelerator ACC100 Adapter	Required

Table 10. Hardware Components for Worker Node Base (Access Edge - vRAN) – 4th Gen Intel Xeon Scalable Processor with integrated vRAN Boost

INGREDIENT	REQUIREMENT	REQUIRED/ RECOMMENDED
4th Gen Intel Xeon Scalable processors	Intel® Xeon® SPR-EE LCC 20 core CPU SKU with integrated vRAN Boost accelerator	Required
Memory	DRAM only configuration: 128 GB DRAM (8 x 16 GB DDR5)	Required
Intel® Optane™ Persistent Memory	512 GB (4 x 128 GB Intel® Optane™ persistent memory in 2-1-1 topology)	Recommended
Network Adapter	Option 1: Intel® Ethernet Network Adapter E810-CQDA2	Required
	Option 2: Intel® Ethernet Network Adapter E810-XXVDA4	
Intel® QAT	Integrated in the processor	
Storage (Boot Drive)	Intel® SATA Solid State Drive D3 S4510 at 480 GB or equivalent boot drive	Required
Storage (Capacity)	Intel® SSD D7-P5510 Series at 3.84 TB or equivalent drive (recommended NUMA aligned)	Required
LAN on Motherboard (LOM)	10 Gbps or 25 Gbps port for Preboot Execution Environment (PXE) and Operation, Administration, and Management (OAM)	Required
	1/10 Gbps port for Management Network Adapter	Required

Table 11. Hardware Components for Worker Node Base – Intel® Xeon® D Processor

INGREDIENT	REQUIREMENT	REQUIRED/ RECOMMENDED
Intel® Xeon® D processors	Intel® Xeon® D-1700 processor, 4 core LCC, 45 W, or Intel® Xeon® D-1700 processor, 10 core LCC, or Intel Xeon D-2733 NT processor, 8 cores HCC, 80 W	Required
Memory	DRAM only configuration: 32 GB DDR4 2667 MHz	Required
Network Adapter	2 x 10/25 GbE integrated Ethernet ports OR Intel® Ethernet Network Adapter E810-CQDA2	Required
Intel® QAT	Intel® QuickAssist Adapter 8960 or 8970 (PCIe*) AIC or equivalent third-party Intel® C620 Series Chipset	Recommended
Storage (Boot Drive)	Intel® SSD 256 GB 2.5" internal SSD/M.2	Required
Additional Plug-in cards	N/A	

3.3 Hardware Components List for Worker Node Plus

The following tables list the hardware options for worker nodes in the “plus” configuration, which helps improve the processing capability due to more powerful CPU, more memory, more disk space, and a faster network.

Table 12. Hardware Components for Worker Node Plus – 3rd Gen Intel Xeon Scalable Processor

INGREDIENT	REQUIREMENT	REQUIRED/ RECOMMENDED
3rd Gen Intel Xeon Scalable processors	Intel® Xeon® Gold 6338N CPU @ 2.2 GHz 32 C/64 T, 185 W, or higher number Intel® Xeon® Gold or Platinum CPU SKU	Required
Memory	Option 1: DRAM only configuration: 512 GB (16 x 32 GB DDR4, 2666 MHz)	Required
	Option 2: DRAM only configuration: 512 GB (3 2x 16 GB DDR4, 2666 MHz)	
Intel® QAT	Intel® C620 Series Chipset integrated on base board Intel® C627/C628 Chipset, integrated with NUMA connectivity to each CPU or minimum 16 Peripheral Component Interconnect Express (PCIe) lane connectivity to one CPU	Required
Intel® Optane™ Persistent Memory	Option 1: 1 TB (8 x 128 GB Intel® Optane™ persistent memory in 8+4 Topology)	Recommended
	Option 2: 2 TB (16 x 128 GB Intel® Optane™ persistent memory in 8+8 Topology)	
Network Adapter	Option 1: Intel® Ethernet Network Adapter E810-CQDA2	Required
	Option 2: Intel® Ethernet Network Adapter E810-2CQDA2	
Storage (Boot Drive)	Intel® SATA Solid State Drive D3 S4510 at 480 GB or equivalent boot drive	Required
Storage (Capacity)	Intel® SSD D7-P5510 Series at 4 TB or equivalent drive (recommended NUMA aligned)	Recommended
LAN on Motherboard (LOM)	10 Gbps or 25 Gbps port for Preboot Execution Environment (PXE) and Operation, Administration, and Management (OAM)	Required
	1/10 Gbps port for Management Network Adapter	Required

INGREDIENT	REQUIREMENT	REQUIRED/ RECOMMENDED
Additional Plug-in cards	Intel® Data Center GPU Flex Series	Optional

Table 13. Hardware Components for Worker Node Plus – 4th Gen Intel Xeon Scalable Processor

INGREDIENT	REQUIREMENT	REQUIRED/ RECOMMENDED
4th Gen Intel Xeon Scalable processors	Intel® Xeon® Gold 6438N processor at 1.8GHz, 32 C/64 T, 205 W	Required
Memory	Option 1: DRAM only configuration: 512 GB (16 x 32 GB DDR5)	Required
	Option 2: DRAM only configuration: 512 GB (32 x 16 GB DDR5)	
Intel® QAT	Integrated in the processor	Required
Intel® Optane™ Persistent Memory	Option 1: 1 TB (8 x 128 GB Intel® Optane™ persistent memory in 8+4 Topology)	Recommended
	Option 2: 2 TB (16 x 128 GB Intel® Optane™ persistent memory in 8+8 Topology)	
Network Adapter	Option 1: Intel® Ethernet Network Adapter E810-CQDA2	Required
	Option 2: Intel® Ethernet Network Adapter E810-2CQDA2	
Storage (Boot Drive)	Intel® SATA Solid State Drive D3 S4510 at 480 GB or equivalent boot drive	Required
Storage (Capacity)	Intel® SSD D7-P5510 Series at 4 TB or equivalent drive (recommended NUMA aligned)	Recommended
LAN on Motherboard (LOM)	10 Gbps or 25 Gbps port for Preboot Execution Environment (PXE) and Operation, Administration, and Management (OAM)	Required
	1/10 Gbps port for Management Network Adapter	Required
Additional Plug-in cards	Intel® Data Center GPU Flex Series	Optional

Table 14. Hardware Components for Worker Node Plus (Access Edge - vRAN) – 4th Gen Intel Xeon Scalable Processor

INGREDIENT	REQUIREMENT	REQUIRED/ RECOMMENDED
4th Gen Intel Xeon Scalable processors	Intel® Xeon®-SP Gold 6421N 32 C/ 64 T 1.8 GHz 185 W	Required
Memory	DRAM only configuration: 128 GB DRAM (8 x 16 GB DDR5)	Required
Intel® Optane™ Persistent Memory	1 TB (8 x 128 GB Intel® Optane™ persistent memory in 2-1-1 topology)	Recommended
Network Adapter	Option 1: Intel® Ethernet Network Adapter E810-CQDA2	Required
	Option 2: Intel® Ethernet Network Adapter E810-2CQDA2	
	Option 3: Intel® Ethernet Network Adapter E810-XXVDA4	
Storage (Boot Drive)	Intel® SATA Solid State Drive D3 S4510 at 480 GB or equivalent boot drive	Required
Storage (Capacity)	Intel® SSD D7-P5510 Series at 3.84 TB or equivalent drive (recommended NUMA aligned)	Required
LAN on Motherboard (LOM)	10 Gbps or 25 Gbps port for Preboot Execution Environment (PXE) and Operation, Administration, and Management (OAM)	Required
	1/10 Gbps port for Management Network Adapter	Required
Additional Plug-in cards	Intel® vRAN Accelerator ACC100 Adapter	Required

Table 15. Hardware Components for Worker Node Plus (Access Edge - vRAN) – 4th Gen Intel Xeon Scalable Processor with integrated vRAN Boost

INGREDIENT	REQUIREMENT	REQUIRED/ RECOMMENDED
4th Gen Intel Xeon Scalable processors	Intel® Xeon® SPR-EE MCC 20 core CPU SKU with integrated vRAN Boost accelerator	Required
Memory	DRAM only configuration: 128 GB DRAM (8 x 16 GB DDR5)	Required
Intel® Optane™ Persistent Memory	512 GB (4 x 128 GB Intel® Optane™ persistent memory in 2-1-1 topology)	Recommended
Network Adapter	Option 1: Intel® Ethernet Network Adapter E810-CQDA2	Required
	Option 2: Intel® Ethernet Network Adapter E810-XXVDA4	
Intel® QAT	Integrated in the processor	
Storage (Boot Drive)	Intel® SATA Solid State Drive D3 S4510 at 480 GB or equivalent boot drive	Required

INGREDIENT	REQUIREMENT	REQUIRED/ RECOMMENDED
Storage (Capacity)	Intel® SSD D7-P5510 Series at 3.84 TB or equivalent drive (recommended NUMA aligned)	Required
LAN on Motherboard (LOM)	10 Gbps or 25 Gbps port for Preboot Execution Environment (PXE) and Operation, Administration, and Management (OAM)	Required
	1/10 Gbps port for Management Network Adapter	Required

Table 16. Hardware Components for Worker Node Plus – Intel® Xeon® D Processor

INGREDIENT	REQUIREMENT	REQUIRED/ RECOMMENDED
Intel® Xeon® D processors	Intel® Xeon® D-2766NT processor 2.1 GHz, 14 core HCC, 97 W, or higher	Required
Memory	DRAM only configuration: 64 GB DDR4 2667 MHz	Required
Network Adapter	4 x 10/25 GbE integrated Ethernet ports Intel® Ethernet Network Adapter E810-CQDA2	Required
Intel® QAT	Intel® QuickAssist Adapter 8960 or 8970 (PCIe*) AIC or equivalent third-party Intel® C620 Series Chipset	Recommended
Storage (Boot Drive)	Intel® SSD 512 GB 2.5" internal SSD/M.2	Required
Additional Plug-in cards	N/A	

3.4 Hardware Components List for Storage Node

Table 17. Hardware Components for Storage Node – 3rd Gen Intel Xeon Scalable Processor

INGREDIENT	REQUIREMENT	REQUIRED/ RECOMMENDED
3rd Gen Intel Xeon Scalable processors	Intel® Xeon® Gold 6338N CPU @ 2.2 GHz 32 C/64 T, 185 W, or higher number Intel® Xeon® Gold or Platinum CPU SKU	Required
Memory	Option 1: DRAM only configuration: 512 GB (16 x 32 GB DDR4, 2666 MHz)	Required
	Option 2: DRAM only configuration: 512 GB (32 x 16 GB DDR4, 2666 MHz)	
Intel® QAT	Intel® C620 Series Chipset integrated on base board Intel® C627/C628 Chipset, integrated with NUMA connectivity to each CPU or minimum 16 Peripheral Component Interconnect Express (PCIe) lane connectivity to one CPU	Required
Intel® Optane™ Persistent Memory	512 GB (4 x 128 GB Intel® Optane™ persistent memory in 2-1-1 topology)	Recommended
Network Adapter	Option 1: Intel® Ethernet Network Adapter E810-CQDA2	Required
	Option 2: Intel® Ethernet Network Adapter E810-2CQDA2	
Storage (Boot Drive)	Intel® SATA Solid State Drive D3 S4510 at 480 GB or equivalent boot drive	Required
Storage (Capacity)	Kioxia CM6 3.2 TB NVMePCIe4x4 2.5"15mm SIE 3DWPD - KCM6XVUL3T20	Required
LAN on Motherboard (LOM)	10 Gbps or 25 Gbps port for Preboot Execution Environment (PXE) and Operation, Administration, and Management (OAM)	Required
	1/10 Gbps port for Management Network Adapter	Required

3.5 Hardware BOMs Supporting All BMRA Configuration Profiles

The following tables list the hardware BOMs for control nodes, worker node base, and worker node plus.

Choose your controller profile from the three available profiles (Controller_xGen_1, Controller_xGen_2, or Controller_xGen_3) based on your BIOS profile (Profiles available: Energy Balance, Deterministic, or Max Performance, respectively).

The profiles for Worker Nodes vary with respect to network interface card, Intel® QuickAssist Technology, and BIOS profiles. You may choose based on the requirements for the workloads to be run on the worker nodes.

Table 18. Control Node Hardware Setup for all Configuration Profiles – 3rd Gen Intel Xeon Scalable Processor

NAME	Controller_3rdGen_1	Controller_3rdGen_2	Controller_3rdGen_3
Platform	M50CYP	M50CYP	M50CYP
CPU/node	2x 5318N 20c	2x 5318N 20c	2x 5318N 20c

Mem	256 GB	256 GB	256 GB
Intel Optane Persistent Memory	Recommended	Recommended	Recommended
Network Adapter	2x E810-CQDA2	2x E810-CQDA2	2x E810-CQDA2
Storage (Boot Media)	Required - 2x	Required - 2x	Required - 2x
Storage (Capacity)	Recommended - 2x (1 per NUMA)	Recommended - 2x (1 per NUMA)	Recommended - 2x (1 per NUMA)
LOM	No	No	No
Intel® QAT	Recommended	N/A	N/A
BIOS Configuration			
Intel® HT Technology enabled	Yes	Yes	Yes
Intel® VT-x enabled	No	Yes	Yes
Intel® VT-d enabled	No	Yes	Yes
BIOS Profile	Energy Balance	Deterministic	Max Performance
Virtualization enabled	No	Yes	Yes

Table 19. Control Node Hardware Setup for all Configuration Profiles – 4th Gen Intel Xeon Scalable Processor

NAME	Controller_4thGen_1	Controller_4thGen_2	Controller_4thGen_3
Platform	Archer City / Quanta - S6Q	Archer City / Quanta - S6Q	Archer City / Quanta - S6Q
CPU/node	2x 5418N	2x 5418N	2x 6438N
Mem	256 GB	256 GB	512 GB
Intel Optane Persistent Memory	Recommended	Recommended	Recommended – 2 TB
Network Adapter	2x E810-CQDA2 or E810-XXVDA2	2x E810-CQDA2 or E810-XXVDA2	2x E810-2CQDA2 or 4x E810-CQDA2
Storage (Boot Media)	Required - 2x	Required - 2x	Required - 2x
Storage (Capacity)	Recommended - 2x (1 per NUMA)	Recommended - 2x (1 per NUMA)	Required- 4x (2 per NUMA)
LOM	No	No	Yes
Intel® QAT	Integrated in the processor	Integrated in the processor	Integrated in the processor
Intel® HT Technology enabled	Yes	Yes	Yes
Intel® VT-x enabled	No	Yes	Yes
Intel® VT-d enabled	No	Yes	Yes
BIOS Profile	Energy Efficiency Turbo	Energy Efficiency Turbo	Energy Balance Turbo

Virtualization enabled	No	Yes	Yes

Table 20. Control Node Hardware Setup for all Configuration Profiles – Intel® Xeon® D Processor

NAME	Controller_Xeon_D_1	Controller_Xeon_D_2	Controller_Xeon_D_3
Platform	Intel® SoC Based Server Reference Platform Board (Codenamed Brighton City) K97971-101	Intel® SoC Based Server Reference Platform Board (Codenamed Brighton City) K97971-101	Intel® SoC Based Server Reference Platform Board (Codenamed Brighton City) K97971-101
CPU/node	Intel® Xeon® D-1700 processor, 4 core LCC, 45 W, or higher	Intel® Xeon® D-1700 processor, 4 core LCC, 45 W, or higher	Intel® Xeon® D-1700 processor, 4 core LCC, 45 W, or higher
Mem	16 GB DDR4 2933 MHz	16 GB DDR4 2933 MHz	16 GB DDR4 2933 MHz
Network Adapter	2 x 10 GbE integrated Ethernet ports	2 x 10 GbE integrated Ethernet ports	2 x 10 GbE integrated Ethernet ports
Storage (Boot Media)	Intel® SSD 256 GB 2.5" internal SSD/M.2	Intel® SSD 256 GB 2.5" internal SSD/M.2	Intel® SSD 256 GB 2.5" internal SSD/M.2
LOM	No	No	No
Intel® QAT AIC	Recommended	N/A	N/A
BIOS Configuration			
Intel® HT Technology enabled	Yes	Yes	Yes
Intel® VT-x enabled	No	Yes	Yes
Intel® VT-d enabled	No	Yes	Yes
BIOS Profile	Energy Balance	Deterministic	Max Performance
Virtualization enabled	No	Yes	Yes

Table 21. Worker Node Base Hardware Setup for all Configuration Profiles – 3rd Gen Intel Xeon Scalable Processor

NAME	Worker_3rdGen_Base_1	Worker_3rdGen_Base_2	Worker_3rdGen_Base_3
Platform	M50CYP	M50CYP	M50CYP
CPU/node	2x 5318N 24c	2x 5318N 24c	2x 5318N 24c
Mem	512 GB	512 GB	512 GB
Intel Optane Persistent Memory	Recommended – 512 GB	Recommended – 512 GB	Recommended – 512 GB
Network Adapter	2x E810-CQDA2	2x E810-CQDA2	2x E810-2CQDA2 or 4x E810-CQDA2
Storage (Boot Media)	Required - 2x	Required - 2x	Required - 2x
Storage (Capacity)	Required- 2x (1 per NUMA)	Required- 2x (1 per NUMA)	Required- 2x (1 per NUMA)
LOM	No	Yes	No

Intel® QAT	No	Yes	Optional
Additional Plug-in cards	No	No	No
BIOS Configuration			
Intel® HT Technology enabled	Yes	Yes	Yes
Intel® VT-x enabled	Yes	Yes	Yes
Intel® VT-d enabled	Yes	Yes	Yes
BIOS Profile	Energy Balance	Max Performance	Deterministic
Virtualization enabled	No	Yes	Yes

Table 22. Worker Node Plus and Storage Node Hardware Setup for all Configuration Profiles – 3rd Gen Intel Xeon Scalable Processor

NAME	Worker_3rdGen_Plus_1	Worker_3rdGen_Plus_2	Worker_3rdGen_Plus_3	Storage_3rdGen_1
Platform	M50CYP	M50CYP	M50CYP	M50CYP
CPU/node	2x 6338N 32c	2x 6338N 32c	2x 6338N 32c	2x 6338N 32c
Mem	512 GB	512 GB	512 GB	512 GB
Intel Optane Persistent Memory	Recommended – 512 GB	Recommended – 512 GB	Recommended – 512 GB	Recommended – 512 GB
Network Adapter	2x E810-2CQDA2 or 4x E810-CQDA2	2x E810-2CQDA2	2x E810-2CQDA2 or 4x E810-CQDA2	2x E810-2CQDA2 or 4x E810-CQDA2
Storage (Boot Media)	Required - 2x	Required - 2x	Required - 2x	Required - 2x
Storage (Capacity)	Required- 4x (2 per NUMA)	Required- 4x (2 per NUMA)	Required- 4x (2 per NUMA)	Required Kioxia 3.2TB - 8x
LOM	Yes	Yes	No	Yes
Intel® QAT	Yes	No	Optional	Yes
Additional Plug-in cards	No	Intel Server GPU	No	No
BIOS Configuration				
Intel® HT Technology enabled	Yes	Yes	Yes	Yes
Intel® VT-x enabled	Yes	Yes	Yes	Yes
Intel® VT-d enabled	Yes	Yes	Yes	Yes
BIOS Profile	Max Performance	Max Performance	Deterministic	Max Performance
Virtualization enabled	Yes	Yes	Yes	Yes

Table 23. Worker Node Base Hardware Setup for all Configuration Profiles – 4th Gen Intel Xeon Scalable Processor

NAME	Worker_4thGen_Base_1	Worker_4thGen_Base_2	Worker_4thGen_Base_3 (Access Edge - vRAN)
Platform	Archer City / Quanta - S6Q	Archer City / Quanta - S6Q	Archer City / Quanta - S6Q / Ruby Pass
CPU/node	2x 5418N	2x 5418N	1x 5411N
Mem	512 GB	512 GB	128 GB

Intel Optane Persistent Memory	Recommended – 512 GB	Recommended – 512 GB	Recommended – 512 GB
Network Adapter	2x E810-CQDA2	2x E810-CQDA2	2x E810-CQDA2 or 8x E810-XXVAM-DA4
Storage (Boot Media)	Required - 2x	Required - 2x	Required - 2x
Storage (Capacity)	Required- 2x (1 per NUMA)	Required- 2x (1 per NUMA)	Required- 2x (1 per NUMA)
LOM	No	Yes	Yes
Intel® QAT	Integrated in the processor	Integrated in the processor	Integrated in the processor
Additional Plug-in cards	No	No	Intel® vRAN Accelerator ACC100 Adapter
BIOS Configuration			
Intel® HT Technology enabled	Yes	Yes	Yes
Intel® VT-x enabled	Yes	Yes	Yes
Intel® VT-d enabled	Yes	Yes	Yes
BIOS Profile	Energy Efficiency Turbo	Max Performance Turbo	Low Latency
Virtualization Enable	No	Yes	Yes

Table 24. Worker Node Plus Hardware Setup for all Configuration Profiles – 4th Gen Intel Xeon Scalable Processor

NAME	Worker_4thGen_Plus_1	Worker_4thGen_Plus_2	Worker_4thGen_Plus_3	Worker_4thGen_Plus_4 (Access Edge - vRAN)
Platform	Archer City / Quanta - S6Q	Archer City / Quanta - S6Q	Archer City / Quanta - S6Q	Archer City / Quanta - S6Q / Ruby Pass
CPU/node	2x 6438N	2x 6438N	2x 6438N	1x 6421N, 1x SPR-EE-LCC/MCC
Mem	512 GB	512 GB	512 GB	128 GB
Intel Optane Persistent Memory	Recommended – 2 TB	Recommended – 1 TB	Recommended – 2 TB	Recommended – 1 TB
Network Adapter	2x E810-2CQDA2 or 4x E810-CQDA2	2x E810-2CQDA2 or 8x E810 XXVAM-DA4	2x E810-2CQDA2 or 4x E810-CQDA2	1x E810-2CQDA2 or 2x E810-CQDA2 8x E810-XXVAM-DA4
Storage (Boot Media)	Required - 2x	Required - 2x	Required - 2x	Required - 2x
Storage (Capacity)	Required- 4x (2 per NUMA)	Required- 4x (2 per NUMA)	Required- 4x (2 per NUMA)	Required- 4x (1 per NUMA)
LOM	Yes	Yes	No	Yes
Intel® QAT	Integrated in the processor	Integrated in the processor	Integrated in the processor	Integrated in the processor
Additional Plug-in cards	No	Intel Server GPU	No	Intel® vRAN Accelerator ACC100 Adapter
BIOS Configuration				
Intel® HT Technology enabled	Yes	Yes	Yes	Yes
Intel® VT-x enabled	Yes	Yes	Yes	Yes

NAME	Worker_4thGen_Plus_1	Worker_4thGen_Plus_2	Worker_4thGen_Plus_3	Worker_4thGen_Plus_4 (Access Edge - vRAN)
Intel® VT-d enabled	Yes	Yes	Yes	Yes
BIOS Profile	Energy Balance Turbo	Energy Balance Turbo	Max Performance Turbo	Low Latency
Virtualization enabled	Yes	Yes	Yes	Yes

Table 25. Worker Node Base Hardware Setup for all Configuration Profiles – Intel® Xeon® D Processor

NAME	Worker_Xeon_D_Base_1	Worker_Xeon_D_Base_2		Worker_Xeon_D_Base_3
Platform	Intel® SoC Based Server Reference Platform Board (Codenamed Brighton City) K97971-101 or Taylors Falls Reference Design	Intel® SoC Based Server Reference Platform Board (Codenamed Brighton City) K97971-101 or Taylors Falls Reference Design		Intel® SoC Based Server Reference Platform Board (Codenamed Brighton City) K97971-101 or Taylors Falls Reference Design
CPU/node	Intel® Xeon® D-1700 processor, 4 core LCC, 45 W, or higher	Intel® Xeon® D-1700 processor, 4 core LCC, 45 W, or higher		Intel® Xeon® D-1700 processor, 4 core LCC, 45 W, or higher
Mem	16 GB DDR4 2933 MHz	16 GB DDR4 2933 MHz		16 GB DDR4 2933 MHz
Network Adapter	Intel® Ethernet Network Adapter E810-CQDA2	Intel® Ethernet Network Adapter E810-CQDA2	Intel® Ethernet Network Adapter E810-CQDA2	
Storage (Boot Media)	Required – 256 GB	Required – 256 GB	Required – 256 GB	
LOM	No	Yes	No	
Intel® QAT	No	Yes	Optional	
Additional Plug-in cards	No	No	No	
BIOS Configuration				
Intel® HT Technology enabled	Yes	Yes	Yes	
Intel® VT-x enabled	Yes	Yes	Yes	
Intel® VT-d enabled	Yes	Yes	Yes	
BIOS Profile	Max Performance	Deterministic	Max Performance	
Virtualization enabled	Yes	Yes	Yes	

Table 26. Worker Node Plus Hardware Setup for all Configuration Profiles – Intel® Xeon® D Processor

NAME	Worker_Xeon_D_Plus_1	Worker_Xeon_D_Plus_2
Platform	Intel® SoC Based Server Reference Platform Board (Codenamed Moro City)	Intel® SoC Based Server Reference Platform Board(Codenamed Moro City)
CPU/node	Intel Xeon D-2700 processor, 16 core HCC, 105 W, or higher	Intel Xeon D-2700 processor, 16 core HCC, 105 W, or higher
Mem	64 GB DDR4 2933 MHz	64 GB DDR4 2933 MHz
Network Adapter	Intel® Ethernet Network Adapter E810-CQDA2	Intel® Ethernet Network Adapter E810-CQDA2
Storage (Boot Media)	Required – 512 GB	Required – 512 GB

NAME	Worker_Xeon_D_Plus_1	Worker_Xeon_D_Plus_2
LOM	Yes	No
Intel® QAT	Yes	Optional
Additional Plug-in cards	No	No
BIOS Configuration		
Intel® HT Technology enabled	Yes	Yes
Intel® VT-x enabled	Yes	Yes
Intel® VT-d enabled	Yes	Yes
BIOS Profile	Max Performance	Deterministic
Virtualization enabled	Yes	Yes

3.6 Platform BIOS

This section provides BIOS Configuration Profiles for each of the BMRA Configuration Profiles. For details on how the BIOS configuration should be set per each Configuration Profile, see the tables in Section 3.5.

For more information about BIOS settings, visit the Intel BIOS Setup Utility User Guide.

Table 27. Platform BIOS Settings for 3rd Gen Intel® Xeon® Scalable Processor

MENU (ADVANCED)	PATH TO BIOS SETTING	BIOS SETTING	ENERGY BALANCE	MAX PERFORMANCE WITH TURBO	DETERMINISTIC
Socket Configuration	Processor Configuration	Hyper-Threading	Enable	Enable	Enable
		XAPIC	Enable	Enable	Enable
		VMX	Enable	Enable	Enable
		Uncore frequency scaling	Enable	Enable	Disable
		Uncore frequency	800-2400	1.8MHz (hex 0x12)	2400
Power Configuration	Power and Performance	CPU Power and Performance Policy	Balance Performance	Performance	Performance
		Workload Configuration	I/O sensitive	I/O sensitive	I/O sensitive
	CPU P-state Control	EIST PSD Function	HW_ALL	HW_ALL	HW_ALL
		Boot Performance Mode	Max. Performance	Max. Performance	Max. Performance
		AVX License Pre-Grant	Disable	Disable	Disable
		AVX ICCP Pre Grant Level	NA	NA	NA
		AVX P1	Nominal	Nominal	Nominal
		Energy Efficient Turbo	Enable	Enable	Disable
		WFR Uncore GV rate Reduction	Enable	Enable	Enable
		GPSS timer	500us	0us	0us

		Intel Turbo Boost Technology	Enable	Enable	Disable
		Intel SpeedStep® Technology (P-states)	Enable	Enable	Disable
	Frequency Prioritization	RAPL Prioritization	Enable	Disable	Disable
	Hardware PM State Control	Hardware P-states	Native Mode with no legacy Support	Native Mode with no legacy Support	Disable
		EPP enable	Enable	Disable	Disable
	CPU C-state Control	Enable Monitor Mwait	Enable	Enable	Enable
		CPU C1 Auto Demotion	Enable	Disable	Disable
		CPU C1 Auto unDemotion	Enable	Disable	Disable
		CPU C6 Report	Enable	Enable	Disable
		Processor C6	Enable	Enable	Disable
		Enhanced Halt State (C1E)	Enable	Enable	Disable
		OS ACPI Cx	ACPI C2	ACPI C2	ACPI C2
	Energy Performance Bias	Power Performance Tuning	OS Controls EPB	OS Controls EPB	OS Controls EPB
		ENERGY_PERF_BIAS_CFG mode	Performance	Performance	Performance
		Workload Configuration	I/O Sensitive	I/O Sensitive	I/O Sensitive
	Package C-state Control	Package C-state	C6 Retention	C0/C1 State	C0/C1 State
		Dynamic L1	Enable	Disable	Disable
		Package C-state Latency Negotiation	Disable	Disable	Disable
		PKGC_SA_PS_CRITERIA	Disable	Disable	Disable
Memory Configuration		Memory Configuration	2-way interleave	2-way interleave	2-way interleave
		Enforce POR	Enable	Enable	Enable
Platform Configuration	Miscellaneous Configuration	Serial Debug Message Level	Minimum	Minimum	Minimum
	PCI Express* Configuration	PCIe* ASPM Support	Per Port	Per Port	Per Port
	PCI Express* Configuration	PCIe* ASPM	Enable	Disable	Disable

	PCI Express* Configuration	ECRC generation and checking	Enable	Enable	Enable
Server Management		Resume on AC Power Loss	Power On	Power On	Power On
System Acoustic and Performance Configuration		Set Fan Profile	Acoustic	Performance	Performance

Table 28. Platform BIOS Settings for 4th Gen Intel® Xeon® Scalable Processor

MENU (ADVANCED)	PATH TO BIOS SETTING	BIOS SETTING	LOW LATENCY	MAX PERFORMANCE WITH TURBO	ENERGY BALANCE TURBO
Socket Configuration	Processor Configuration	Hyper-Threading	Enable	Enable	Enable
		X2APIC	Enable	Enable	Enable
		VMX	Enable	Enable	Enable
		Homeless Prefetch	Enable	Disable (default)	Disable (default)
		LLC Prefetch	Disable	Enable	Enable
		SNC	Disable	Disable	Disable
		Uncore RAPL	Disable	Disable	Enable
		Uncore frequency scaling	Disable	Disable	Enable
		Uncore frequency	1.8GHz (hex 0x12)	1.6MHz (hex 0x10)	800MHz to 2.5GHz
Power Configuration	CPU P-state Control	EIST PSD Function	HW_ALL	HW_ALL	HW_ALL
		Boot Performance Mode	Max. Performance	Max. Performance	Max. Performance
		AVX License Pre-Grant	Enable	Disable	Disable
		AVX ICCP Pre Grant Level	Level 5	NA	NA
		AVX P1 (ConfigTDP)	Level 2	Nominal (default)	Nominal
		Energy Efficient Turbo	Disable	Disable	Enable
		GPSS timer	0us	0us	0us
		Turbo	Enable	Enable	Enable
		Intel® Speed Step® Technology	Enable	Enable	Enable
	Frequency Prioritization	RAPL Prioritization	Disable	Disable	Disable
	Common Ref Code	UMA-Based Clustering	Quadrant	Quadrant	Quadrant
	Hardware PM State Control	Hardware P-states	Native with no Legacy Support	Native with no Legacy Support	Native with no Legacy Support
		EPP enable	Disable	Disable	Disable
	CPU C-state Control	Enable Monitor Mwait	Enable	Enable	Enable

		CPU C1 Auto Demotion	Disable	Disable	Disable
		CPU C1 Auto unDemotion	Disable	Disable	Disable
		Processor C6 or CPU C6 Report	Enable	Enable	Enable
		Enhanced Halt State (C1E)	Enable (per Core Level)	Enable	Enable
		OS ACPI Cx	ACPI C2	ACPI C2	ACPI C2
	Energy Performance Bias	Power Performance Tuning	OS Control EPB	OS Controls EPB	OS Controls EPB
		Workload Configuration	I/O Sensitive	I/O Sensitive	Balanced
	Package C-state Control	Package C-state	C6 Retention	C0/C1 State	C0/C1 State
		Dynamic L1	Enable	Disable	Disable
Memory Configuration		Memory Configuration	8-way interleave	8-way interleave	8-way interleave
		Enforce POR / Memory Patrol Scrub	Enable/Disable	Enable/Enable	Enable/Enable
		Memory DIMM Refresh Rate	1x	1x	2x
Platform Configuration	Miscellaneous Configuration	Serial Debug Message Level	Minimum	Minimum	Minimum
	PCI Express* Configuration	PCIe* ASPM	Disable	Enable	Enable
		ECRC generation and checking	Disable	Enable	Enable
Server Management		Resume on AC Power Loss	Power On	Power On	Power On
System Acoustic and Performance Configuration		Set Fan Profile	Performance	Acoustic	Acoustic

Table 29. Platform BIOS Settings for Intel® Xeon® D Processor

MENU (ADVANCED)	PATH TO BIOS SETTING	BIOS SETTINGS	ENERGY BALANCE	MAX PERFORMANCE	DETERMINISTIC
Power Configuration	Power and Performance	CPU Power and Performance Policy	Balanced Performance	Performance	Performance
		Workload Configuration	I/O sensitive	I/O sensitive	I/O sensitive
		Turbo	Disabled	Enabled	Disabled
	CPU P-state control	Enhanced Intel SpeedStep® Technology	Enabled	Enabled	Disabled
		GPSS timer	500 µs	0 µs	0 µs
	Hardware P-states	Hardware P-states	Native Mode with no legacy Support	Disabled	Disabled
	CPU C-state Control	Package C-state	C6 Retention	C6 Retention	C0/C1 State
		C1E	Enabled	Enabled	Disabled
		Processor C6	Enabled	Enabled	Disabled
	Uncore Power Management	Uncore Frequency scaling	Enabled	Disabled	Disabled
		Performance P-limit	Enabled	Disabled	Disabled
Memory Configuration	Memory Configuration	IMC Interleaving	2-way interleave	2-way interleave	2-way interleave
Thermal Configuration	System Acoustic and Performance Configuration	Set Fan Profile	Acoustic	Performance	Performance
GPU	GPU Fz	Lock 900 MHz	Optional	Optional	Optional

Use the following table to configure the BIOS settings to use Intel SST-BF, Intel SST-TF, and Intel SST-PP in 3rd and 4th Gen Intel Xeon Scalable processor systems.

Table 30. BIOS Settings to Enable Intel SST-BF, Intel SST-TF, and Intel SST-PP

BIOS SETTING	STATUS
Hardware PM State Control	
Scalability	Disable
Hardware PM Interrupt	Disable
CPU P-state	
Dynamic SST-PP	Enable
Speed Step (P-states)	Enable
Activate SST-BF	Enable
Configure SST-BF	Enable
EIST PSD Function	HW_All
Turbo	Enable
Energy Efficient Turbo	Enable
Boot Performance	Max

BIOS SETTING	STATUS
Freq: Prioritization AC	
SST-CP	Enable

In BIOS, the configuration paths might be slightly different, depending on platform, but the key settings are as follows and must be performed in order.

Table 31. BIOS Settings to Enable Intel® SGX on 3rd Gen Intel Xeon Scalable Processor

BIOS SETTING	STATUS
Socket Configuration > Processor Configuration > Total Memory Encryption (TME)	Enable
Socket Configuration > Common RefCode Configuration > UMA-Based Clustering	Disable (All2All)
Socket Configuration > Processor Configuration > SW Guard Extensions (SGX)	Enable
Socket Configuration > Processor Configuration > Enable/Disable SGX Auto MP Registration Agent	Enable

Table 32. BIOS Settings to Enable Intel® SGX on 4th Gen Intel Xeon Scalable Processor

BIOS SETTING	STATUS
Advanced > Processor Configuration > Total Memory Encryption (TME)	Enable
Advanced > Memory Configuration > Memory RAS and Performance Configuration > UMA-Based Clustering	Disable (All2All)
Advanced > Processor Configuration > SW Guard Extensions (SGX)	Enable
Advanced > Processor Configuration > Enable/Disable SGX Auto MP Registration Agent	Enable

4 Reference Architecture Software Components

4.1 Software Components Supported

Table 33 lists the software components automatically deployed per Configuration Profile in a BMRA and their sources.

Table 33. Software Components

SOFTWARE FUNCTION	SOFTWARE COMPONENT	LOCATION
OS	Ubuntu 22.04 Ubuntu 22.04 RT	https://www.ubuntu.com
OS	RHEL 9.0	https://www.redhat.com/
OS	RHEL 8.6	https://www.redhat.com/
OS	Rocky 9.1	https://rockylinux.org/
Data Plane Development Kit	DPDK 22.11.1	https://core.dpdk.org/download/
Open vSwitch with DPDK	OVS-DPDK v3.0.3	https://github.com/openvswitch/ovs
Vector Packet Processing	VPP 23.02	https://packagecloud.io/fdio/
Telegraf	1.2	https://github.com/intel/observability-telegraf
Collectd	v1.0	https://github.com/intel/observability-collectd/releases/
Grafana	9.3.6	https://www.grafana.com/
Prometheus	2.42.0	https://quay.io/repository/prometheus/prometheus?tab=tags
Prometheus nginx image	1.23.2-alpine	docker.io/library/nginx:1.23.2-alpine
Ansible	4.10.0	https://www.ansible.com/
BMRA Ansible Playbook	v23.02	https://github.com/intel/container-experience-kits
Python	Python 3.6.x for RHEL 8/9	https://www.python.org/
Kubespray	2023/02/7 commit	https://github.com/kubernetes-sigs/kubespray
Docker	20.10.20	https://www.docker.com/
containerd	1.6.16	https://github.com/containerd/containerd/tags
CRI-O	1.26.0	https://github.com/cri-o/cri-o/tags
crictl	1.26.0	https://github.com/kubernetes-sigs/cri-tools/releases
Container orchestration engine	Kubernetes v1.26.1	https://github.com/kubernetes/kubernetes
CPU Manager (native to Kubernetes)	Available natively in Kubernetes	N/A
etcd	v3.5.6	https://github.com/etcd-io/etcd/tags
cri-dockerd	0.3.0	https://github.com/Mirantis/cri-dockerd/releases
runc	1.1.4	https://github.com/opencontainers/runc/releases
Platform Aware Scheduling (TAS)	TAS 0.4.0	https://github.com/intel/platform-aware-scheduling
Platform Aware Scheduling (GAS)	GAS 0.5.1	https://github.com/intel/platform-aware-scheduling
k8s-prometheus-adapter	0.10.0	https://github.com/kubernetes-sigs/prometheus-adapter
K8s node-exporter	1.5.0	https://quay.io/repository/prometheus/node-exporter?tab=tags
K8s prometheus-operator	0.63.0	https://quay.io/repository/prometheus-operator/prometheus-operator?tab=tags
K8s kube-rbac-proxy	0.14.0	https://github.com/brancz/kube-rbac-proxy/releases
Node Feature Discovery	0.12.1-minimal	https://github.com/kubernetes-sigs/node-feature-discovery
Multus CNI	3.9.3	https://github.com/k8snetworkplumbingwg/multus-cni/tags

SOFTWARE FUNCTION	SOFTWARE COMPONENT	LOCATION
calico	v3.24.5	https://github.com/projectcalico/calico/tags
cilium	v1.12.1	https://github.com/cilium/cilium/tags
flannel	v0.20.2	https://github.com/flannel-io/flannel/tags
SR-IOV CNI	2.7.0	https://github.com/k8snetworkplumbingwg/sriov-cni/releases
SR-IOV network device plugin	3.5.1	https://github.com/k8snetworkplumbingwg/sriov-network-device-plugin/releases/
SR-IOV Network Operator	1.2.0	https://github.com/k8snetworkplumbingwg/sriov-network-device-plugin/releases/
Whereabouts Service	05cc22a9c8165c5cba875bebfa58d1b504a2e6c9	https://github.com/k8snetworkplumbingwg/helm-charts.git
Device Plugins Operator	0.26.0	https://github.com/intel/intel-device-plugins-for-kubernetes
QAT device plugin	0.26.0	https://github.com/intel/intel-device-plugins-for-kubernetes
GPU device plugin	0.26.0	https://github.com/intel/intel-device-plugins-for-kubernetes
Intel® SGX device plugin	0.26.0	https://github.com/intel/intel-device-plugins-for-kubernetes
Intel DLB device plugin	0.26.0	https://github.com/intel/intel-device-plugins-for-kubernetes
Intel DSA device plugin	0.26.0	https://github.com/intel/intel-device-plugins-for-kubernetes
Userspace CNI	1.3	https://github.com/intel/userspace-cni-network-plugin
Bond CNI plugin	9800813	https://github.com/k8snetworkplumbingwg/bond-cni
Intel® Ethernet Drivers	i40e v2.22.8 ice v1.10.1.2.2 iavf v4.7.0	https://sourceforge.net/projects/e1000/files/i40e%20stable/2.22.8/ https://sourceforge.net/projects/e1000/files/ice%20stable/1.10.1.2.2/ https://sourceforge.net/projects/e1000/files/iavf%20stable/4.7.0/
Intel® Ethernet NVM Update Package for Intel Ethernet 700 Series	9.20	https://www.intel.com/content/www/us/en/download/18190/non-volatile-memory-nvm-update-utility-for-intel-ethernet-network-adapter-700-series.html
Intel® Ethernet NVM Update Package for Intel Ethernet 800 Series	4.20	https://www.intel.com/content/www/us/en/download/19626/non-volatile-memory-nvm-update-utility-for-intel-ethernet-network-adapters-e810-series-linux.html
DDP Profiles	Dynamic Device Personalization for Intel® Ethernet 700 Series Version 25.4	https://downloadmirror.intel.com/28940/eng/mplsogreudp.zip https://downloadmirror.intel.com/28040/eng/ppp-oe-ol2tpv2.zip https://downloadmirror.intel.com/29446/eng/esp-ah.zip https://downloadmirror.intel.com/29780/eng/ecpri.zip
	Intel® Ethernet 800 Series Telecommunication (Comms) Dynamic Device Personalization (DDP) Package 1.3.37.0	https://www.intel.com/content/www/us/en/download/19660/intel-ethernet-800-series-telecommunication-comms-dynamic-device-personalization-ddp-package.html
Intel® Ethernet Operator	22.11	https://github.com/intel/intel-ethernet-operator.git
Intel® Ethernet Operator SDK	1.26.0	https://github.com/operator-framework/operator-sdk.git
Intel® Ethernet UFT	22.11	https://github.com/intel/UFT.git
Intel® QAT Drivers	QAT20.L.1.0.0-00021	https://www.intel.com/content/www/us/en/download/765501/intel-quickassist-technology-driver-for-linux-hw-version-2-0.html
Intel® QAT Driver Card	QAT.L.4.20.0-00001	https://www.intel.com/content/www/us/en/download/19734/intel-quickassist-technology-driver-for-linux-hw-version-1-7.html?wapkw=qat%20driver
Intel QATLib	23.02.0	https://github.com/intel/qatlib/tags
OpenSSL	openssl-3.0.8	https://github.com/openssl/openssl https://www.openssl.org/source/
OpenSSL QAT Engine	0.6.18	https://github.com/intel/QAT_Engine
Intel ipsec-mb	1.3	https://github.com/intel/intel-ipsec-mb
Intel® SGX DCAP Drivers	1.41	https://download.01.org/intel-sgx/sgx-dcap/1.10.3/linux/

SOFTWARE FUNCTION	SOFTWARE COMPONENT	LOCATION
Intel® SGX SDK	2.18.100.3	https://download.01.org/intel-sgx/sgx-dcap/1.10.3/linux/
Intel® KMRA	2.3	https://01.org/key-management-reference-application-kmra
Intel® KMRA AppHSM	2.3	https://hub.docker.com/r/intel/apphsm
Intel® KMRA CTK	2.3	https://hub.docker.com/r/intel/ctk_loadkey
Intel® KMRA PCCS	2.3	https://hub.docker.com/r/intel/pccs
Istio operator	1.17.1	https://github.com/istio/istio
Intel Istio operator	1.16.1-intel.0	https://hub.docker.com/r/intel/istioctl/
istio-intel/pilot	1.16.1-intel.0	https://hub.docker.com/r/intel/pilot/
istio-intel/proxyv2	1.16.1-intel.0	https://hub.docker.com/r/intel/proxyv2/
istio-intel/trusted-certificate-issuer	0.4.0	https://github.com/intel/trusted-certificate-issuer
istio-intel/trusted-attestation-controller	0.4.0	https://github.com/intel/trusted-attestation-controller
CNDP DP	0.0.2	https://github.com/intel/afxdp-plugins-for-kubernetes.git
CNDP CNI	22.08.0	https://github.com/CloudNativeDataPlane/cndp/tags
MinIO operator	4.5.8	https://github.com/minio/operator
MinIO console	0.22.5	https://github.com/minio/console
Power Manager Operator	1.0.2	https://hub.docker.com/r/intel/power-operator
Power Node Agent Operator	1.0.2	https://hub.docker.com/r/intel/power-node-agent
Intel® RDT	4.4.1	https://github.com/intel/intel-cmt-cat
FEC Operator	23.05	https://github.com/smart-edge-open/sriov-fec-operator
FEC Operator SDK	1.26.0	https://github.com/operator-framework/operator-sdk.git
Operator Package Manager	1.26.3	https://github.com/operator-framework/operator-registry/releases/
FlexRAN™ software	22.11	
OpenTelemetry	0.24.0	https://github.com/open-telemetry/opentelemetry-operator
Jaeger	1.42.0	https://github.com/jaegertracing/jaeger-operator
cadvisor	2.2.4	https://github.com/ckotzbauer/helm-charts
Linkerd	2.12.4	https://helm.linkerd.io/
TADK	22.09	https://hub.docker.com/r/intel/tadk-waf
ADQ-K8s-plugin	22.06-1	https://github.com/intel/adq-k8s-plugins
Intel OneAPI	2022.1.2.146	https://www.intel.com/content/www/us/en/developer/tools/oneapi
Go Lang	1.19.3	https://go.dev/dl/
Intel CPU Control Plugin	1.0	https://github.com/intel/cpu-control-plane-plugin-for-kubernetes
Rook Ceph	1.10.10	https://github.com/rook/rook.git
Multus-service	sha256:f53a6fcf3f728bec8fc6ceb1a6e5ad0ee0cc912ceb3c6610a3c8a468cb2736b9	https://github.com/k8snetworkplumbingwg/multus-service/pkgs/container/multus-service
GStreamer	1.20.3.173	https://github.com/GStreamer/gstreamer
Intel DL Streamer	2022	https://github.com/dlstreamer/dlstreamer
OpenVINO	2022.1.0-643	https://github.com/openvinotoolkit/openvino
FlexRAN Container	22.07	https://hub.docker.com/r/intel/flexran_vdu

4.2 Software Components Compatibility Matrices

Legend for the tables in this section	
🟥	Indicates that the combination is unsupported
🟩	Indicates that the combination is supported and tested
🟨	Indicates that the combination is expected to work but untested
⬜	Indicates that the combination is not applicable

Note: Features that are not listed have been verified to work for all combinations

Feature / Platform Compatibility Limitations			
	3rd Gen Intel® Xeon® Scalable Processor	4th Gen Intel® Xeon® Scalable Processor	Intel® Xeon® D Processor
Intel® DSA	⬜	🟩	⬜
Intel® DLB	⬜	🟩	⬜
Intel® SST-BF	🟩	🟩	🟥
Intel® SST-CP	🟩	🟩	🟥
Intel® SST-TF	🟩	🟩	🟥
Intel® SST-PP	🟥	🟩	🟥
SST Operator	🟩	🟩	🟥
MinIO / NVMe	🟩	🟩	🟨
KMRA	🟩	🟥	🟩
Intel® DC GPU	🟩	🟩	🟨
Intel® QAT off-chip	🟩	🟨	🟩
Intel® QAT on-chip	⬜	🟩	⬜
Intel® Scalable IOV	⬜	🟩	⬜
Intel® FEC off-chip	🟩	🟥	⬜
Intel® FEC on-chip	⬜	🟩	⬜
Intel® FEC Operator	🟩	🟩	⬜

Profile / Platform Validation Matrix			
	3rd Gen Intel® Xeon® Scalable Processor	4th Gen Intel® Xeon® Scalable Processor	Intel® Xeon® D Processor
Access Edge	🟩	🟩	🟩
Basic	🟩	🟩	🟩
On-Premises Edge	🟩	🟩	🟩
Remote Central Office-Forwarding	🟩	🟩	🟩
Regional Data Center	🟩	🟩	🟨
Build-Your-Own	🟩	🟩	🟩

Feature / OS Compatibility Limitations				
	Ubuntu 22.04 5.15.0-25-generic	RHEL 8.6 4.18.0-372.9.1.rt7.166.el8.x86_64-64	RHEL 9.0 5.14.0-70.13.1.el9_0.x86_64	Rocky Linux 9.1 5.14.0-162.6.1.el9_1.x86_64
Intel® DSA	🟩	▨	🟩	🟩
Intel® DLB	🟩	▨	🟩	🟩
FlexRAN™ software	🟩 NDA only	🟩 NDA only	🟥	🟥
Intel® DC GPU	🟩	▨	🟥	🟥

Feature / Feature Compatibility Limitations								
	DPDK 22.11.1	DPDK 22.07	OVS 3.0.3	Telegraf	Intel® SST-BF	Intel® SST-CP	Intel® SST-TF	Intel® SST-PP
OVS 3.0.3	🟥	🟩	▨	▨	▨	▨	▨	▨
VPP	▨	▨	🟥	▨	▨	▨	▨	▨
collectd	▨	▨	▨	🟥	▨	▨	▨	▨
Intel® SST-PP	▨	▨	▨	▨	🟥	🟥	🟥	▨
Power Manager	▨	▨	▨	▨	🟥	🟥	🟥	🟥

5 Post Deployment Verification Guidelines

This section describes a set of processes that you can use to verify the components deployed by the scripts. The processes are not Configuration Profile-specific but relate to individual components that may not be available in all profiles. Details for each of the Configuration Profiles are described in Sections 7 through 12.

Many verification guidelines and output examples can be found on GitHub, as listed in Table 34, and others are described after the table.

Table 34. Links to Verification Guidelines on GitHub

VERIFICATION STEP
Check the Kubernetes Cluster
Check Intel SST-BF and Intel SST-CP on 3rd Gen Intel Xeon Scalable Processor
Check Intel SST-PP with Intel SST-TF on 3rd and 4th Gen Intel Xeon Scalable Processors
Check DDP Profiles on Intel® Ethernet 700 and 800 Series Network Adapters
Check Node Feature Discovery
Check Topology Manager
Check SR-IOV Network Operator
Check SR-IOV Device Plugin
Check QAT Device Plugin
Check SGX Device Plugin
Check DSA Device Plugin
Check GPU Device Plugin
Check Multus CNI Plugin
Check SR-IOV CNI Plugin
Check Userspace CNI Plugin
Check Bond CNI Plugin
Check Telemetry Aware Scheduling
Check Intel® Server GPU Device and Driver
Check Intel QAT Engine with OpenSSL
Check MinIO Operator/Console and Tenant
Check Intel Power Manager (Balance Performance Power-Profile & Sample Power-Pods)

5.1 Check Grafana Telemetry Visualization

BMRA deploys Grafana for telemetry visualization. It is available on every cluster node on port 30000. Due to security reasons, this port is not exposed outside the cluster by default. Default credentials are `admin`/`admin` and you should change the default password after first login.

The Grafana TLS certificate is signed by the cluster certificate authority (CA) and it is available in `/etc/kubernetes/ssl/ca.crt`

Visit Grafana at `https://<node-ip>:30000/`

BMRA comes with a set of dashboards from the kube-prometheus project (kube-prometheus). Dashboards are available in the Dashboards > Manage menu.

5.2 Check Key Management Infrastructure with Intel SGX

To verify the Key Management infrastructure with SGX and use the private keys provisioned to Intel SGX enclaves, see Section 13.1 for step-by-step instructions to set up and run the NGINX workload.

Part 2:
Building a BMRA Step-by-Step

6 BMRA Setup – Applicable for All Configuration Profiles

This section is relevant for generating BMRA Flavors based on their Configuration Profiles. It provides the prerequisites for system setup and includes information that enables you to review BIOS prerequisites and software BOMs at a glance. The information is presented in multi-column tables to provide an easy way to compare and assess the differences between the BMRA Flavors that are available.

After setting up the Kubernetes system, refer to the specific section from the following list to build the BMRA Flavors:

Section 7, BMRA Basic Configuration Profile Setup
Section 8, BMRA Build-Your-Own Configuration Profile Setup
Section 9, BMRA On-Premises Edge Configuration Profile Setup
Section 10, BMRA Access Edge Configuration Profile Setup
Section 11, BMRA Remote Central Office-Forwarding Configuration Profile Setup
Section 12, BMRA Regional Data Center Configuration Profile Setup

6.1 Set Up an Ansible Host

BMRA Kubernetes clusters require an Ansible host that stores information about all managed remote nodes. In general, any machine running a recent Linux distribution can be used as Ansible host for any of the supported BMRA deployments (regardless of target OS on the control and worker nodes) as long as it meets the following basic requirements:

- Network connectivity to the control and worker nodes, including SSH
- Internet connection (using proxy if necessary)
- Git utility installed
- Python 3 installed
- Ansible version 5.7.1 installed (Ansible-base at 2.12.5)

Step-by-step instructions for building the Ansible host are provided below for the same list of operating systems that are supported for the control and worker nodes (see Section 2.3.3).

6.1.1 RHEL Version 8 as Ansible Host

1. Install the Linux OS. If using the iso image, choose the Minimal iso version, or select the "Minimal Install" (Basic functionality) option under Software Selection.
2. Make the proper configuration during installation for the following key elements: Network (Ethernet) port IP Address, Host Name, Proxies (if necessary), and Network Time Protocol (NTP).
3. After the installation completes and the machine reboots, log in as root and confirm that it has a valid IP address and can connect (ping) to the control and worker nodes.
4. Make sure that the HTTP and HTTPS proxies are set, if necessary, for internet access. The configuration can be completed with the *export* command or by including the following lines in the `/etc/environment` file:
   ```
   http_proxy=http://proxy.example.com:1080
   https_proxy=http://proxy.example.com:1080
   ```
   Then, load the proxies configuration in the current environment:
   ```
   # source /etc/environment
   ```
5. Install Git:
   ```
   # yum install -y git
   ```
6. Install Python 3:
   ```
   # yum -y install python3
   ```

The Ansible host box is now ready to deploy the Container BMRA. Follow the instructions in Section 2.5.

6.1.2 Ubuntu 20.04 LTS as Ansible Host

1. Install the OS using any method supported by the vendor (Canonical Ltd.). Either the Desktop or Server distribution can be used. Select the “Minimal installation” option under “Updates and Other software”.
2. Follow steps 2, 3, and 4 as described above for RHEL.
3. Update the installation:
   ```
   # sudo apt update
   ```
4. Install SSH utilities:
   ```
   # sudo apt install openssh-server
   ```
5. Install Git:
   ```
   # sudo apt install -y git
   ```
6. Install Python 3-pip:
   ```
   # sudo apt install -y python3-pip
   ```

The Ansible host box is now ready to deploy the Container BMRA. Follow the instructions in Section 2.5.

6.2 Set Up the Control and Worker Nodes - BIOS Prerequisites

This section is applicable for all **Configuration Profiles.**

Enter the UEFI or BIOS menu and update the configuration as shown in Table 35 and Table 36.

Note: The method for accessing the UEFI or BIOS menu is vendor-specific, for example: How to boot into the BIOS or the Lifecycle Controller on your PowerEdge Server

The BIOS profile referenced in these tables consists of configurations in the power management, thermal management, and configuration for Intel® platform technologies such as Intel® Virtualization Technology, Intel® Hyper-Threading Technology, Intel SpeedStep® technology, and Intel® Turbo Boost Technology.

The table provides four different BIOS profiles.

- Energy Balance
- Max Performance
- Deterministic
- Low Latency (4th Gen Intel® Xeon® Scalable processor)

The configuration and values set per each BIOS profile are defined in the tables in Section 3.6.

Table 35. BIOS Prerequisites for Control and Worker Nodes for Basic, Storage, and Build-Your-Own Configuration Profiles

PROFILES	BASIC CONFIGURATION PROFILE	STORAGE CONFIGURATION PROFILE	BUILD-YOUR-OWN CONFIGURATION PROFILE
Configuration			
BIOS Profile	Energy Balance	Max Performance	Any
Grub Command Line (values are set by Ansible)			
Isolcpus	Optional	No	Optional
Hugepages	Optional	No	Optional
P-state=disable	Optional	No	Optional
Limit C-state	Optional	No	Optional

Table 36. BIOS Prerequisites for Control and Worker Nodes for On-Premises Edge, Remote Central Office-Forwarding, Regional Data Center, and Access Edge Configuration Profiles

PROFILES	ON-PREMISES EDGE CONFIGURATION PROFILE	REMOTE CENTRAL OFFICE-FORWARDING CONFIGURATION PROFILE	REGIONAL DATA CENTER CONFIGURATION PROFILE	ACCESS EDGE CONFIGURATION PROFILE
Configuration				
BIOS Profile	Max Performance	Deterministic / Energy Balance	Max Performance	Low Latency
Grub Command Line (values are set by Ansible)				
Isolcpus	Yes	Yes	Optional	Yes
Hugepages	Yes	Yes	Optional	Yes
P-state=disable	No	Yes, No-SST-BF	Optional	No
Limit C-state	No	Yes	Optional	Yes

Note: The above values are the recommended configuration options on the Intel S2600WFQ and Intel M50CYP server boards. Some server boards may not provide the same options that are documented in this table. Vendors typically provide options for max performance configuration with virtualization.

6.3 Configuration Dictionary - Group Variables

Table 37 lists the parameters available as group variables with their type (for example, Boolean, string, URL, list, integer). Refer to the section that describes your Configuration Profile to see the parameters enabled for that Configuration Profile.

Table 37. Configuration Dictionary – Group Variables

OPTION	TYPE
profile_name	String

OPTION	TYPE
configured_arch	String
unconfirmed_cpu_models	List
project_root_dir	String
vm_enabled	Boolean
post_deployment_hook_enabled	Boolean
hooks_local	String
hooks_remote	String
kubernetes	Boolean
kube_version	String
container_runtime_only_deployment	Boolean
audit_policy_custom_rules	String
container_runtime	String
intel_cpu_controlplane.enabled	Boolean
intel_cpu_controlplane.allocator	String
intel_cpu_controlplane.agent_namespace_prefix	String
local_volume_provisioner_enabled	Boolean
rook_ceph.enabled	Boolean
rook_ceph.log_level	String
rook_ceph.allow_loop_devices	Boolean
rook_ceph.enable_nfs	Boolean
rook_ceph.enable_discovery_daemon	Boolean
rook_ceph.cluster.enabled	Boolean
rook_ceph.cluster.number_of_mons	Integer
rook_ceph.cluster.allow_multiple_mon_per_node	Boolean
rook_ceph.cluster.number_of_mgrs	Integer
rook_ceph.cluster.allow_multiple_mgr_per_node	Boolean
cadvisor_enabled	Boolean
cadvisor_custom_events_config_on	Boolean
preflight_enabled	Boolean
update_all_packages	Boolean
update_kernel	Boolean
additional_grub_parameters_enabled	Boolean
additional_grub_parameters	String
selinux_state	String
nfd_enabled	Boolean
nfd_namespace	String
nfd_sleep_interval	String
kube_dashboard_enabled	Boolean
native_cpu_manager_enabled	Boolean
topology_manager_enabled	Boolean
topology_manager_policy	String
sriov_network_operator_enabled	Boolean
sriov_network_operator_namespace	String
sriov_net_dp_enabled	Boolean
sriov_net_dp_namespace	String
sriov_net_dp_build_image_locally	Boolean

OPTION	TYPE
sriovdp_config_data	String
intel_power_manager.enabled	Boolean
intel_power_manager.power_profiles	List
intel_power_manager.power_nodes	List
intel_power_manager.build_image_locally	Boolean
intel_power_manager.deploy_example_pods	Boolean
intel_power_manager.global_shared_profile_enabled	Boolean
intel_power_manager.max_shared_frequency	Integer
intel_power_manager.min_shared_frequency	Integer
intel_dp_namespace	String
intel_ai_enabled	Boolean
dlb_dp_enabled	Boolean
dlb_dp_build_image_locally	Boolean
dlb_dp_verbosity	Integer
dsa_dp_enabled	Boolean
dsa_dp_build_image_locally	Boolean
dsa_dp_verbosity	Integer
dsa_shared_devices	Integer
intel_ethernet_operator_enabled	Boolean
intel_ethernet_operator_flow_config_enabled	Boolean
intel_sriov_fec_operator_enabled	Boolean
qat_dp_enabled	Boolean
qat_dp_verbosity	Integer
qat_dp_max_num_devices	Integer
qat_dp_build_image_locally	Boolean
allocation_policy (Commented)	String
qat_supported_pf_dev_ids	List
qat_supported_vf_dev_ids	List
openssl_engine_enabled	Boolean
gpu_dp_enabled	Boolean
gpu_dp_verbosity	Integer
gpu_dp_build_image_locally	Boolean
gpu_dp_shared_devices	Integer
gpu_dp_monitor_resources	Boolean
gpu_dp_fractional_manager	Boolean
gpu_dp_prefered_allocation	String
sgx_dp_enabled	Boolean
sgx_dp_verbosity	Integer
sgx_dp_build_image_locally	Boolean
sgx_aesmd_namespace	String
sgx_aesmd_demo_enable	Boolean
sgx_dp_provision_limit	Integer
sgx_dp_enclave_limit	Integer
istio_service_mesh.enabled	Boolean
istio_service_mesh.profile	String
istio_service_mesh.intel_preview.enabled	Boolean

OPTION	TYPE
istio_service_mesh.tcpip_bypass_ebpf.enabled	Boolean
istio_service_mesh.tls_splicing.enabled	Boolean
linkerd_service_mesh.enabled	Boolean
pas_namespace	String
tas_enabled	Boolean
tas_build_image_locally	Boolean
tas_enable_demo_policy	Boolean
gas_enabled	Boolean
gas_build_image_locally	Boolean
prometheus_operator	Boolean
collectd_enabled	Boolean
telegraf_enabled	Boolean
jaeger_operator	Boolean
opentelemetry_enabled	Boolean
elasticsearch_enabled	Boolean
kibana_enabled	Boolean
collectd_scrap_interval	Integer
telegraf_scrap_interval	Integer
example_net_attach_defs.sriov_net_dp	Boolean
example_net_attach_defs.userspace_ovs_dpdk	Boolean
example_net_attach_defs.userspace_vpp	Boolean
firewall_enabled	Boolean
http_proxy (Commented)	String
https_proxy (Commented)	String
additional_no_proxy (Commented)	String
dns_disable_stub_listener	Boolean
remove_kubespray_host_dns_settings	Boolean
cluster_name	String
retry_stagger	Integer
cert_manager_enabled	Boolean
kube_controller_manager_bind_address	String
kube_proxy_metrics_bind_address	String
kube_network_plugin	String
calico_network_backend	String
calico_advanced_options	Boolean
wireguard_enabled	Boolean
kube_network_plugin_multus	Boolean
kube_pods_subnet	String
kube_service_addresses	String
kube_proxy_mode	String
calico_bpf_enabled	Boolean
kube_proxy_nodeport_addresses_cidr	String
docker_registry_mirrors (Commented)	List
docker_insecure_registries (Commented)	List
containerd_registries (Commented)	List
crio_registries (Commented)	List

OPTION	TYPE
crio_insecure_registries (Commented)	List
registry_enable	Boolean
registry_nodeport	String
registry_local_address	String
always_pull_enabled	Boolean
minio_enabled	Boolean
minio_tenant_enabled	Boolean
minio_tenant_servers	Integer
minio_tenant_volumes_per_server	Integer
minio_tenant_volume_size	Integer
minio_deploy_test_mode	Boolean
minio_build_image_locally	Boolean
minio_awsclient_pods_enabled	Boolean
minio_ingress_enabled	Boolean
cndp_dp_enabled	Boolean
cndp_net_attach_def_enabled	Boolean
tadk_install	Boolean
intel_flexran_enabled	Boolean
intel_flexran_type	String
intel_flexran_mode	String
intel_flexran_bbu_front_haul	String
intel_flexran_bbu_ptp_sync	String
intel_flexran_oru_front_haul	String
intel_flexran_oru_ptp_sync	String
adq_dp.enabled	Boolean
adq_dp.interface_address	String
adq_dp.interface_name	String

6.4 Configuration Dictionary - Host Variables

Table 38 lists the parameters available as host variables with their type (for example, Boolean, string, URL, list, integer). Refer to the section that describes your Configuration Profile to see the parameters enabled for that Configuration Profile.

Table 38. Configuration Dictionary – Host Variables

OPTION	TYPE
profile_name	String
configured_arch	String
configured_nic	String
iommu_enabled	Boolean
dataplane_interfaces	List
update_nic_drivers	Boolean
i40e_driver_version (Commented)	String
i40e_driver_checksum (Commented)	String
ice_driver_version (Commented)	String
ice_driver_checksum (Commented)	String
iavf_driver_version (Commented)	String
iavf_driver_checksum (Commented)	String

OPTION	TYPE
update_nic_firmware	Boolean
nvmupdate (Commented)	List
install_ddp_packages	Boolean
enable_ice_systemd_service	Boolean
sriov_cni_enabled	Boolean
custom_sriov_network_policies_dir (Commented)	String
bond_cni_enabled	Boolean
install_dpdk	Boolean
dpdk_version	String
dpdk_local_patches_dir (Commented)	String
dpdk_local_patches_strip (Commented)	Integer
userspace_cni_enabled	Boolean
ovs_dpdk_enabled	Boolean
ovs_version	String
ovs_dpdk_lcore_mask	String
ovs_dpdk_socket_mem	String
vpp_enabled	Boolean
hugepages_enabled	Boolean
default_hugepage_size	String
number_of_hugepages_1G	Integer
number_of_hugepages_2M	Integer
configure_dlb_devices	Boolean
configure_dsa_devices	Boolean
dsa_devices	List
intel_ethernet_operator.ddp_update	Boolean
intel_ethernet_operator.fw_update	Boolean
intel_ethernet_operator.node_flow_config_enabled	Boolean
intel_ethernet_operator.flow_config_dir (Commented)	String
fec_acc	String
update_qat_drivers	Boolean
qat_drivers_dir (Commented)	String
enabled_qat_service	String
disabled_qat_service	String
enable_intel_qatlibs	Boolean
enable_qat_svm	Boolean
qat_sriov_numvfs_required	Integer
qat_vf_driver_required	String
qat_devices	List
openssl_install	Boolean
isolcpus_enabled	Boolean
isolcpus	String
cpusets_enabled	Boolean
cpusets	String
native_cpu_manager_system_reserved_cpus	String
native_cpu_manager_kube_reserved_cpus	String
native_cpu_manager_reserved_cpus (Commented)	String

OPTION	TYPE
intel_pstate_enabled	Boolean
intel_pstate	String
turbo_boost_enabled	Boolean
cstate_enabled	Boolean
cstates.C<1,6>.cpu_range	String
cstates.C<1,6>.enable	Boolean
ufs_enabled	Boolean
ufs.min	Integer
ufs.max	Integer
sst_pp_configuration_enabled	Boolean
sst_pp_config_list.sst_bf	String
sst_pp_config_list.sst_cp	String
sst_pp_config_list.sst_tf	String
sst_pp_config_list.sst_tf.online_cpus_range	String
configure_sgx	Boolean
configure_gpu	Boolean
enable_intel_pmu_plugin	Boolean
intel_pmu_plugin_monitored_cores	String
intel_rdt_plugin_monitored_cores	String
exclude_collectd_plugins	List
cndp_enabled	Boolean
cndp_dp_pools	List
adq_dp.enabled	Boolean
adq_dp.interface_address	String
local_shared_profile.enabled	Boolean
local_shared_profile.node_max_shared_frequency	Integer
local_shared_profile.node_min_shared_frequency	Integer
shared_workload.enabled	Boolean
shared_workload.reserved_cpus	List
shared_workload.shared_workload_type	String
enable_dhclient_systemd_service	Boolean
minio_pv	List

7 BMRA Basic Configuration Profile Setup

This section contains a step-by-step description of how to set up a BMRA Basic Flavor.

To use the Basic Configuration Profile, perform the following steps:

1. Choose your hardware, set it up, and configure the BIOS. Refer to Section 7.1 for details.
   You also need to build your Kubernetes cluster.
2. Download the Ansible playbook for your Configuration Profile. Refer to Section 7.2 for details.
3. Configure the optional Ansible parameters using the information in the Configuration Profile tables. Refer to Section 7.3 for details.
4. Deploy the platform. Refer to Section 7.4 for details.
5. Validate the setup of your Kubernetes cluster. Refer to the tasks in Section 5 and run the validation processes according to the hardware and software components that you have installed.

Be aware of the definitions of terminology used in tables in this section.

TERM	DESCRIPTION
Hardware Taxonomy	
ENABLED	Setting must be enabled in the BIOS (configured as Enabled, Yes, True, or similar value)
DISABLED	Setting must be disabled in the BIOS (configured as Disabled, No, False, or any other value with this meaning.)
OPTIONAL	Setting can be either disabled or enabled, depending on user's workload. Setting does not affect the Configuration Profile or platform deployment.
Software Taxonomy	
TRUE	Feature is included and enabled by default.
FALSE	Feature is included but disabled by default - can be enabled and configured by user.
N/A	Feature is not included and cannot be enabled or configured.

7.1 Step 1 - Set Up Basic Configuration Profile Hardware

The table in this section lists the hardware BOM for the Basic Configuration Profile, including control node, worker node base, and worker node plus. We recommend that you set up at least one control node and one worker node.

Table 39. Hardware Setup for Basic Configuration Profile

NODE OPTIONS	3RD GEN INTEL XEON SCALABLE PROCESSOR	4TH GEN INTEL XEON SCALABLE PROCESSOR	INTEL XEON D PROCESSOR
Control node options	Controller_3rdGen_1	Controller_4thGen_1	Controller_Xeon_D_1
Worker node options	Worker_3rdGen_Base_1	Worker_4thGen_Base_1	Worker_Xeon_D_Base_1

7.2 Step 2 - Download Basic Configuration Profile Ansible Playbook

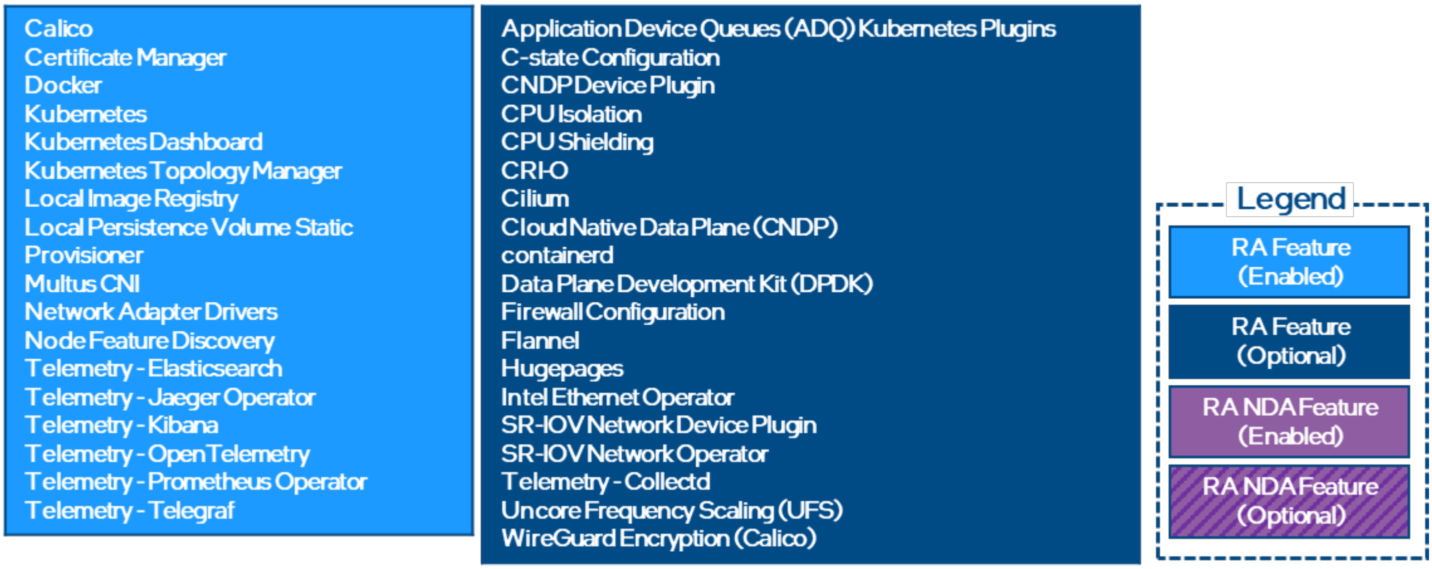
This section contains details for downloading the Basic Configuration Profile Ansible playbook. It also provides an overview of the Ansible playbook and lists the software that is automatically installed when the playbook is deployed.

Download the Basic Configuration Profile Ansible playbook using the steps described in Section 2.5.

7.2.1 Basic Configuration Profile Ansible Playbook Overview

The Ansible playbook for the Basic Configuration Profile allows you to provision a production-ready Kubernetes cluster. Every capability included in the Basic Configuration Profile playbook can be disabled or enabled. Refer to the diagram and group and host variables tables below to see which Ansible roles are included and executed by default.

The diagram shows the architecture of the Ansible playbooks that are included in the Basic Configuration Profile.



Calico
Certificate Manager
Docker
Kubernetes
Kubernetes Dashboard
Kubernetes Topology Manager
Local Image Registry
Local Persistence Volume Static Provisioner
Multus CNI
Network Adapter Drivers
Node Feature Discovery
Telemetry - Elasticsearch
Telemetry - Jaeger Operator
Telemetry - Kibana
Telemetry - OpenTelemetry
Telemetry - Prometheus Operator
Telemetry - Telegraf

Application Device Queues (ADQ) Kubernetes Plugins
C-state Configuration
CNDP Device Plugin
CPU Isolation
CPU Shielding
CRI-O
Cilium
Cloud Native Data Plane (CNDP)
containerd
Data Plane Development Kit (DPDK)
Firewall Configuration
Flannel
Hugepages
Intel Ethernet Operator
SR-IOV Network Device Plugin
SR-IOV Network Operator
Telemetry - Collectd
Uncore Frequency Scaling (UFS)
WireGuard Encryption (Calico)

Legend
RA Feature (Enabled)
RA Feature (Optional)
RA NDA Feature (Enabled)
RA NDA Feature (Optional)

Figure 2. Basic Configuration Profile Ansible Playbook

7.3 Step 3 - Set Up Basic Configuration Profile

Review the optional Ansible group and host variables in this section and select options that match your desired configuration.

1. Update the `inventory.ini` file with your environment details as described in Section 2.5.3.
2. Create `host_vars` files for all worker nodes as specified in Section 2.5.4.
3. Update group and host variables to match your desired configuration as specified in Section 2.3.4. Refer to the tables in Section 7.3.1 and Section 7.3.2.

Variables are grouped into two main categories:

1. Group variables – apply to both control and worker nodes and have cluster-wide impact.
2. Host variables – scope is limited to a single worker node.

The tables below are a summary of group and host variables. For lists showing all configurable properties, see Section 6.3 and Section 6.4. All of the variables are important but pay special attention to variables in **bold** as they almost always need to be updated to match the target environment.

7.3.1 Basic Configuration Profile Group Variables

Table 40. Basic Configuration Profile – Group Variables

COMPONENT	VALUE	
Kubernetes	true	For the list of all configurable properties, see Section 6.3
nfd_enabled	true	
topology_manager_enabled	true	
sriov_network_operator_enabled	false	
sriov_net_dp_enabled	false	
example_net_attach_defs	false	
collectd_enabled	false	
telegraf_enabled	true	

7.3.2 Basic Configuration Profile Host Variables[3]

Table 41. Basic Configuration Profile – Host Variables

COMPONENT	VALUE
iommu_enabled	false

[3] See backup for workloads and configurations or visit Performance Index. Results may vary.

COMPONENT	VALUE	
sriov_cni_enabled	false	For the list of all configurable properties, see Section 6.4
install_dpdk	false	
isolcpus_enabled	false	
dataplane_interfaces	[]	

7.4 Step 4 – Deploy and Validate Basic Configuration Profile Platform

Deploy the Basic Configuration Profile Ansible playbook using the steps described in Section 2.5.5.

Validate the setup of your Kubernetes cluster. Refer to the tasks in Section 5 and run the validation processes according to the hardware and software components that you have installed.

8 BMRA Build-Your-Own Configuration Profile Setup

This section contains a step-by-step description of how to set up a BMRA Build-Your-Own Flavor.

To use the Build-Your-Own Configuration Profile, perform the following steps:

1. Choose your hardware, set it up, and configure the BIOS. Refer to Section 8.1 for details.
   You also need to build your Kubernetes cluster.
2. Download the Ansible playbook for your Configuration Profile. Refer to Section 8.2 for details.
3. Configure the optional Ansible parameters using the information in the Configuration Profile tables. Refer to Section 8.3 for details.
4. Deploy the platform. Refer to Section 8.4 for details.
5. Validate the setup of your Kubernetes cluster. Refer to the tasks in Section 5 and run the validation processes according to the hardware and software components that you have installed.

Be aware of the definitions of terminology used in tables in this section.

TERM	DESCRIPTION
Hardware Taxonomy	
ENABLED	Setting must be enabled in the BIOS (configured as Enabled, Yes, True, or similar value)
DISABLED	Setting must be disabled in the BIOS (configured as Disabled, No, False, or any other value with this meaning.)
OPTIONAL	Setting can be either disabled or enabled, depending on user's workload. Setting does not affect the Configuration Profile or platform deployment.
Software Taxonomy	
TRUE	Feature is included and enabled by default.
FALSE	Feature is included but disabled by default - can be enabled and configured by user.
N/A	Feature is not included and cannot be enabled or configured.

8.1 Step 1 - Set Up Build-Your-Own Configuration Profile Hardware

The table in this section lists the hardware BOM for the Build-Your-Own Configuration Profile, including control node, worker node base, and worker node plus. We recommend that you set up at least one control node and one worker node.

Table 42. Hardware Setup for Build-Your-Own Configuration Profile

NODE OPTIONS	3RD GEN INTEL XEON SCALABLE PROCESSOR	4TH GEN INTEL XEON SCALABLE PROCESSOR	INTEL XEON D PROCESSOR
Control node options	Controller_3rdGen_1	Controller_4thGen_1	Controller_Xeon_D_1
Worker node options	Worker_3rdGen_Base_1	Worker_4thGen_Base_1	Worker_Xeon_D_Base_1

8.2 Step 2 - Download Build-Your-Own Configuration Profile Ansible Playbook

This section contains details for downloading the Build-Your-Own Configuration Profile Ansible playbook. It also provides an overview of the Ansible playbook and lists the software that is automatically installed when the playbook is deployed.

Download the Build-Your-Own Configuration Profile Ansible playbook using the steps described in Section 2.5.

8.2.1 Build-Your-Own Configuration Profile Ansible Playbook Overview

The Ansible playbook for the Build-Your-Own Configuration Profile allows you to provision a production-ready Kubernetes cluster. Every capability included in the Build-Your-Own Configuration Profile playbook can be disabled or enabled. Refer to the diagram and group and host variables tables below to see which Ansible roles are included and executed by default.

The diagram shows the architecture of the Ansible playbooks that are included in the Build-Your-Own Configuration Profile.

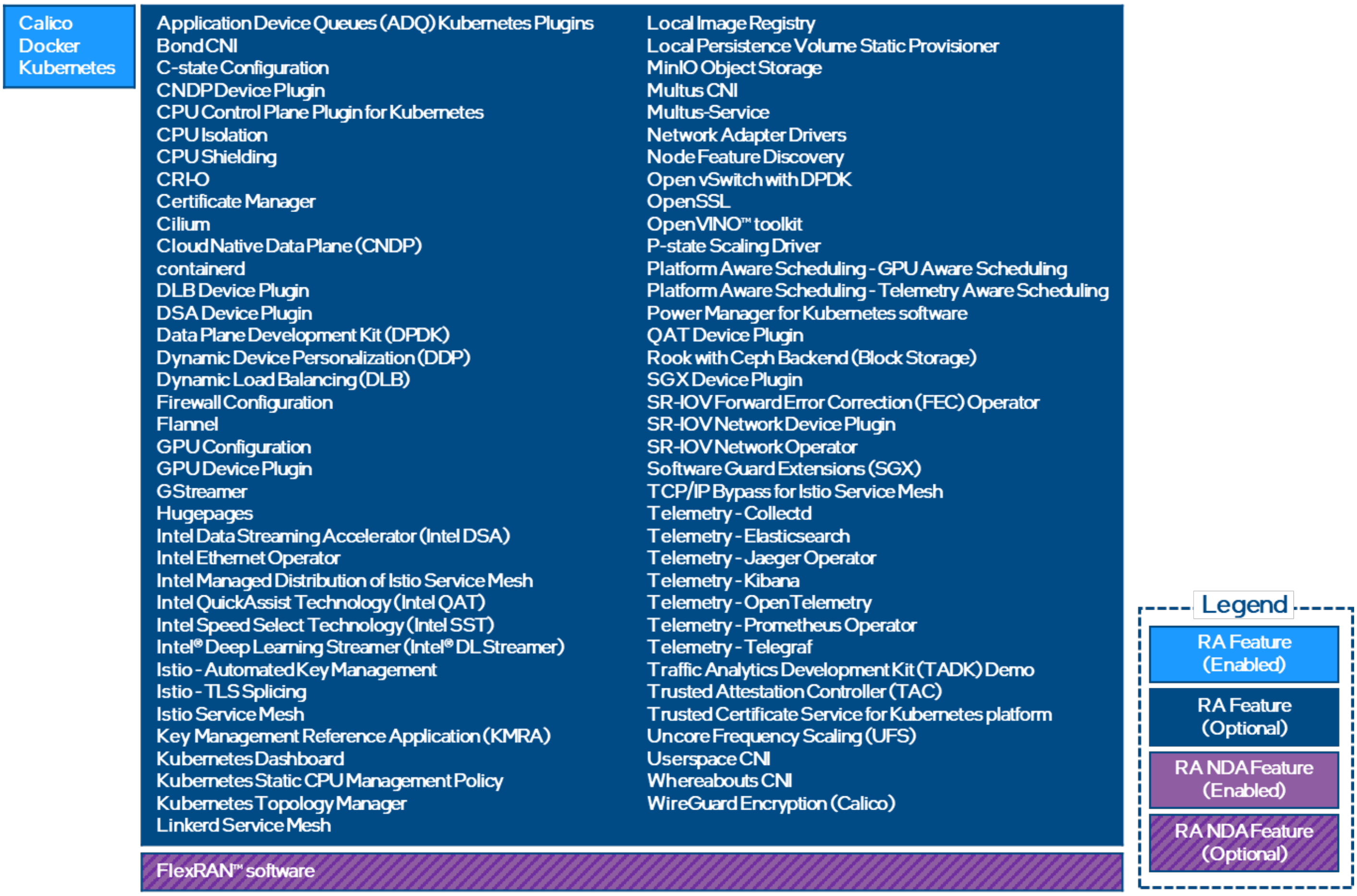

Calico
Docker
Kubernetes
Application Device Queues (ADQ) Kubernetes Plugins
Bond CNI
C-state Configuration
CNDP Device Plugin
CPU Control Plane Plugin for Kubernetes
CPU Isolation
CPU Shielding
CRI-O
Certificate Manager
Cilium
Cloud Native Data Plane (CNDP)
containerd
DLB Device Plugin
DSA Device Plugin
Data Plane Development Kit (DPDK)
Dynamic Device Personalization (DDP)
Dynamic Load Balancing (DLB)
Firewall Configuration
Flannel
GPU Configuration
GPU Device Plugin
GStreamer
Hugepages
Intel Data Streaming Accelerator (Intel DSA)
Intel Ethernet Operator
Intel Managed Distribution of Istio Service Mesh
Intel QuickAssist Technology (Intel QAT)
Intel Speed Select Technology (Intel SST)
Intel® Deep Learning Streamer (Intel® DL Streamer)
Istio - Automated Key Management
Istio - TLS Splicing
Istio Service Mesh
Key Management Reference Application (KMRA)
Kubernetes Dashboard
Kubernetes Static CPU Management Policy
Kubernetes Topology Manager
Linkerd Service Mesh
Local Image Registry
Local Persistence Volume Static Provisioner
MinIO Object Storage
Multus CNI
Multus-Service
Network Adapter Drivers
Node Feature Discovery
Open vSwitch with DPDK
OpenSSL
OpenVINO™ toolkit
P-state Scaling Driver
Platform Aware Scheduling - GPU Aware Scheduling
Platform Aware Scheduling - Telemetry Aware Scheduling
Power Manager for Kubernetes software
QAT Device Plugin
Rook with Ceph Backend (Block Storage)
SGX Device Plugin
SR-IOV Forward Error Correction (FEC) Operator
SR-IOV Network Device Plugin
SR-IOV Network Operator
Software Guard Extensions (SGX)
TCP/IP Bypass for Istio Service Mesh
Telemetry - Collectd
Telemetry - Elasticsearch
Telemetry - Jaeger Operator
Telemetry - Kibana
Telemetry - OpenTelemetry
Telemetry - Prometheus Operator
Telemetry - Telegraf
Traffic Analytics Development Kit (TADK) Demo
Trusted Attestation Controller (TAC)
Trusted Certificate Service for Kubernetes platform
Uncore Frequency Scaling (UFS)
Userspace CNI
Whereabouts CNI
WireGuard Encryption (Calico)
FlexRAN™ software
Legend
RA Feature (Enabled)
RA Feature (Optional)
RA NDA Feature (Enabled)
RA NDA Feature (Optional)

Figure 3. Build-Your-Own Configuration Profile Ansible Playbook

8.3 Step 3 - Set Up Build-Your-Own Configuration Profile

Review the optional Ansible group and host variables in this section and select options that match your desired configuration.

1. Update the `inventory.ini` file with your environment details as described in Section 2.5.3.
2. Create `host_vars` files for all worker nodes as specified in Section 2.5.4.
3. Update group and host variables to match your desired configuration as specified in Section 2.3.4. Refer to the tables in Section 8.3.1 and Section 8.3.2.

Variables are grouped into two main categories:

1. Group variables – apply to both control and worker nodes and have cluster-wide impact.
2. Host variables – scope is limited to a single worker node.

The tables below are a summary of group and host variables. For lists showing all configurable properties, see Section 6.3 and Section 6.4. All of the variables are important but pay special attention to variables in **bold** as they almost always need to be updated to match the target environment.

8.3.1 Build-Your-Own Configuration Profile Group Variables

Table 43. Build-Your-Own Configuration Profile – Group Variables

COMPONENT	VALUE	
Kubernetes	true	For the list of all configurable properties, see Section 6.3
nfd_enabled	false	
topology_manager_enabled	false	
sriov_network_operator_enabled	false	
sriov_net_dp_enabled	false	
example_net_attach_defs	false	
collectd_enabled	false	

COMPONENT	VALUE
telegraf_enabled	false

8.3.2 Build-Your-Own Configuration Profile Host Variables[4]

Table 44. Build-Your-Own Configuration Profile – Host Variables

COMPONENT	VALUE	
iommu_enabled	false	For the list of all configurable properties, see Section 6.4
sriov_cni_enabled	false	
install_dpdk	false	
isolcpus_enabled	false	
dataplane_interfaces	[]	

8.4 Step 4 - Deploy and Validate Build-Your-Own Configuration Profile Platform

Deploy the Build-Your-Own Configuration Profile Ansible playbook using the steps described in Section 2.5.5.

Validate the setup of your Kubernetes cluster. Refer to the tasks in Section 5 and run the validation processes according to the hardware and software components that you have installed.

[4] See backup for workloads and configurations or visit Performance Index. Results may vary.

9 BMRA On-Premises Edge Configuration Profile Setup

This section contains a step-by-step description of how to set up a BMRA On-Premises Edge Flavor.

To use the On-Premises Edge Configuration Profile, perform the following steps:

1. Choose your hardware, set it up, and configure the BIOS. Refer to Section 9.1 for details.
   You also need to build your Kubernetes cluster.
2. Download the Ansible playbook for your Configuration Profile. Refer to Section 9.2 for details.
3. Configure the optional Ansible parameters using the information in the Configuration Profile tables. Refer to Section 9.3 for details.
4. Deploy the platform. Refer to Section 9.4 for details.
5. Validate the setup of your Kubernetes cluster. Refer to the tasks in Section 5 and run the validation processes according to the hardware and software components that you have installed.

Be aware of the definitions of terminology used in tables in this section.

TERM	DESCRIPTION
Hardware Taxonomy	
ENABLED	Setting must be enabled in the BIOS (configured as Enabled, Yes, True, or similar value.)
DISABLED	Setting must be disabled in the BIOS (configured as Disabled, No, False, or any other value with this meaning.)
OPTIONAL	Setting can be either disabled or enabled, depending on user's workload. Setting does not affect the Configuration Profile or platform deployment.
Software Taxonomy	
TRUE	Feature is included and enabled by default.
FALSE	Feature is included but disabled by default - can be enabled and configured by user.
N/A	Feature is not included and cannot be enabled or configured.

9.1 Step 1 - Set Up On-Premises Edge Configuration Profile Hardware

The table in this section lists the hardware BOM for the On-Premises Edge Configuration Profile, including Control Node, Worker Node Base, and Worker Node Plus. We recommend that you set up at least one control node and one worker node.

Table 45. Hardware Setup for On-Premises Edge Configuration Profile

NODE OPTIONS	3RD GEN INTEL XEON SCALABLE PROCESSOR	4TH GEN INTEL XEON SCALABLE PROCESSOR	INTEL XEON D PROCESSOR
Control node options	Controller_3rdGen_1	Controller_4thGen_1	Controller_Xeon_D_1
Worker node options	Worker_3rdGen_Base_2 or Worker_3rdGen_Plus_1	Worker_4thGen_Base_2 or Worker_4thGen_Plus_1	Worker_Xeon_D_Base_2 or Worker_Xeon_D_Plus_1

9.2 Step 2 - Download On-Premises Edge Configuration Profile Ansible Playbook

This section contains details for downloading the On-Premises Edge Configuration Profile Ansible playbook. It also provides an overview of the Ansible playbook and lists the software that is automatically installed when the playbook is deployed.

Download the On-Premises Edge Configuration Profile Ansible playbook using the steps described in Section 2.5.

9.2.1 On-Premises Edge Configuration Profile Ansible Playbook Overview

The Ansible playbook for the On-Premises Edge Configuration Profile allows you to provision a production-ready Kubernetes cluster. It also applies any additional requirements, such as host OS configuration or network adapter drivers and firmware updates. Every capability included in the On-Premises Edge Configuration Profile playbook can be disabled or enabled. Refer to the diagram and group and host variables tables below to see which Ansible roles are included and executed by default.

The diagram shows the architecture of the Ansible playbooks roles that are included in the On-Premises Edge Configuration Profile.

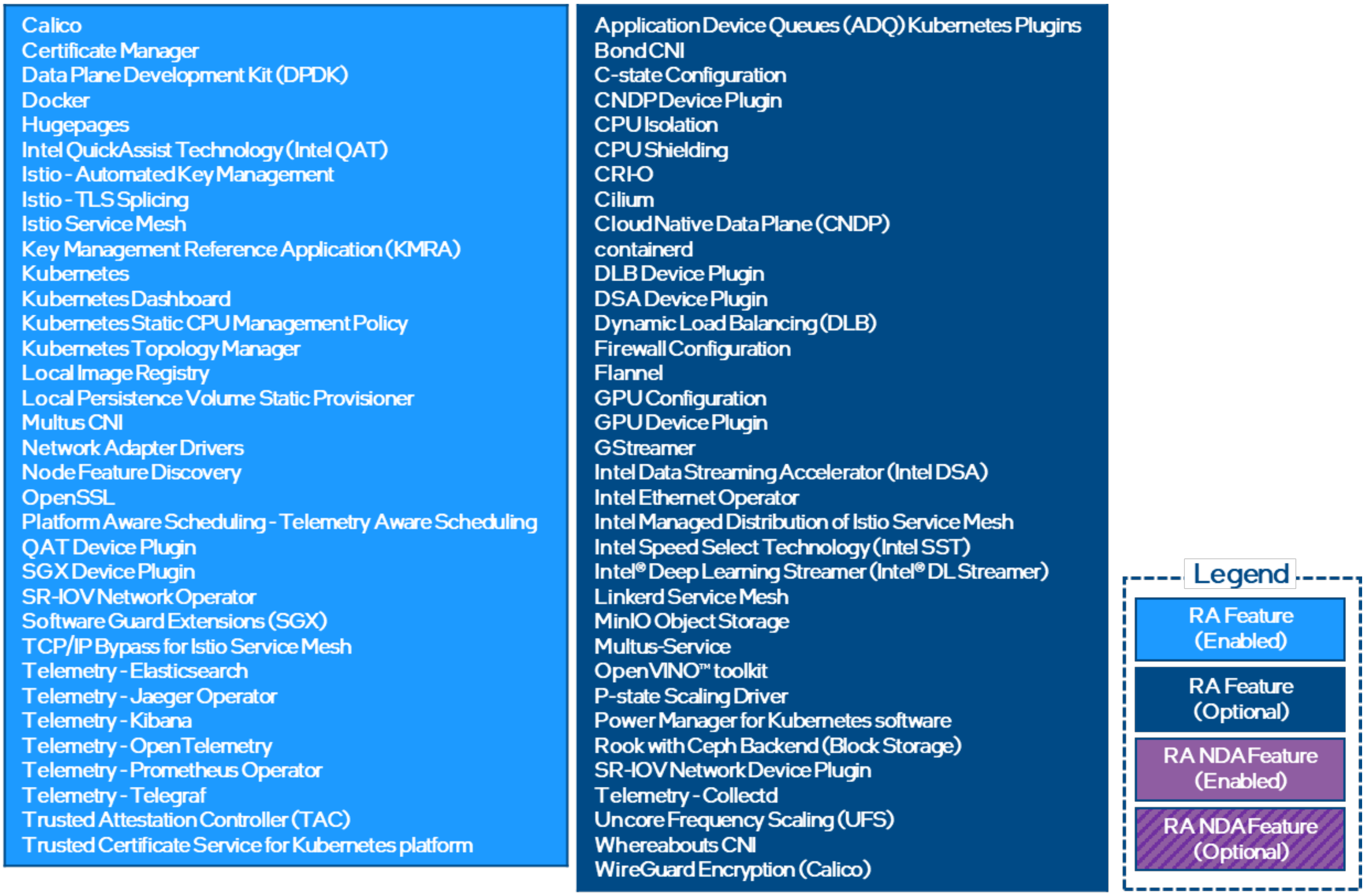

Calico
Certificate Manager
Data Plane Development Kit (DPDK)
Docker
Hugepages
Intel QuickAssist Technology (Intel QAT)
Istio - Automated Key Management
Istio - TLS Splicing
Istio Service Mesh
Key Management Reference Application (KMRA)
Kubernetes
Kubernetes Dashboard
Kubernetes Static CPU Management Policy
Kubernetes Topology Manager
Local Image Registry
Local Persistence Volume Static Provisioner
Multus CNI
Network Adapter Drivers
Node Feature Discovery
OpenSSL
Platform Aware Scheduling - Telemetry Aware Scheduling
QAT Device Plugin
SGX Device Plugin
SR-IOV Network Operator
Software Guard Extensions (SGX)
TCP/IP Bypass for Istio Service Mesh
Telemetry - Elasticsearch
Telemetry - Jaeger Operator
Telemetry - Kibana
Telemetry - OpenTelemetry
Telemetry - Prometheus Operator
Telemetry - Telegraf
Trusted Attestation Controller (TAC)
Trusted Certificate Service for Kubernetes platform
Application Device Queues (ADQ) Kubernetes Plugins
Bond CNI
C-state Configuration
CNDP Device Plugin
CPU Isolation
CPU Shielding
CRI-O
Cilium
Cloud Native Data Plane (CNDP)
containerd
DLB Device Plugin
DSA Device Plugin
Dynamic Load Balancing (DLB)
Firewall Configuration
Flannel
GPU Configuration
GPU Device Plugin
GStreamer
Intel Data Streaming Accelerator (Intel DSA)
Intel Ethernet Operator
Intel Managed Distribution of Istio Service Mesh
Intel Speed Select Technology (Intel SST)
Intel® Deep Learning Streamer (Intel® DL Streamer)
Linkerd Service Mesh
MinIO Object Storage
Multus-Service
OpenVINO™ toolkit
P-state Scaling Driver
Power Manager for Kubernetes software
Rook with Ceph Backend (Block Storage)
SR-IOV Network Device Plugin
Telemetry - Collectd
Uncore Frequency Scaling (UFS)
Whereabouts CNI
WireGuard Encryption (Calico)
Legend
RA Feature (Enabled)
RA Feature (Optional)
RA NDA Feature (Enabled)
RA NDA Feature (Optional)

Figure 4. On-Premises Edge Configuration Profile Ansible Playbook

9.3 Step 3 - Set Up Default On-Premises Edge Configuration Profile

Review the optional Ansible group and host variables in this section and select options that match your desired configuration.

1. Update the `inventory.ini` file with your environment details as described in Section 2.5.3.
2. Create `host_vars` files for all worker nodes as specified in Section 2.5.4.
3. Update group and host variables to match your desired configuration as specified in Section 2.3.4. Refer to the tables in Section 9.3.1 and Section 9.3.2.

Variables are grouped into two main categories:

1. Group variables – apply to both control and worker nodes and have cluster-wide impact.
2. Host variables – scope is limited to a single worker node.

The tables below are a summary of group and host variables. For lists showing all configurable properties, see Section 6.3 and Section 6.4. All of the variables are important but pay special attention to variables in **bold** as they almost always need to be updated to match the target environment.

9.3.1 On-Premises Edge Configuration Profile Group Variables

Table 46. On-Premises Edge Configuration Profile – Group Variables

COMPONENT	VALUE	
Kubernetes	true	For the list of all configurable properties, see Section 6.3
nfd_enabled	true	
native_cpu_manager_enabled	true	
topology_manager_enabled	true	
sriov_network_operator_enabled	true	
sriov_net_dp_enabled	false	
sgx_dp_enabled	true	

COMPONENT	VALUE	
qat_dp_enabled	true	
openssl_engine_enabled	true	
kmra_enabled	true	
tas_enabled	true	
example_net_attach_defs	false	
collectd_enabled	false	
telegraf_enabled	true	
service_mesh	true	
power_manager	false	
intel_ai_enabled	false	

9.3.2 On-Premises Edge Configuration Profile Host Variables[5]

Table 47. On-Premises Edge Configuration Profile – Host Variables

COMPONENT	VALUE	
iommu_enabled	true	For the list of all configurable properties, see Section 6.4
sriov_cni_enabled	false	
bond_cni_enabled	false	
hugepages_enabled	true	
isolcpus_enabled	false	
sst_pp_configuration_enabled	false	
install_dpdk	true	
qat_devices	[]	
dataplane_interfaces	[]	

9.4 Step 3 - Set Up On-Premises Edge Configuration Profile for VSS

Review the optional Ansible group and host variables in this section and select options that match your desired configuration.

1. Update the `inventory.ini` file with your environment details as described in Section 2.5.3.
2. Create `host_vars` files for all worker nodes as specified in Section 2.5.4.
3. Update group and host variables to match your desired configuration as specified in Section 2.3.4. Refer to the tables in Section 9.3.1 and Section 9.3.2.

Variables are grouped into two main categories:

1. Group variables – apply to both control and worker nodes and have cluster-wide impact.
2. Host variables – scope is limited to a single worker node.

The tables below are a summary of group and host variables. For lists showing all configurable properties, see Section 6.3 and Section 6.4. All of the variables are important but pay special attention to variables in **bold** as they almost always need to be updated to match the target environment.

9.4.1 On-Premises Edge Configuration Profile for VSS Group Variables

Table 48. On-Premises Edge Configuration Profile – Group Variables

COMPONENT	VALUE	
intel_ai_enabled	true	For the list of all configurable properties, see Section 6.4
gpu_dp_enabled	true	

[5] See backup for workloads and configurations or visit Performance Index. Results may vary.

9.4.2 On-Premises Edge Configuration Profile for VSS Host Variables[6]

Table 49. On-Premises Edge Configuration Profile – Host Variables

COMPONENT	VALUE	
configure_gpu	true	For the list of all configurable properties, see Section 6.4

9.5 Step 4 - Deploy and Validate On-Premises Edge Configuration Profile Platform

Deploy the On-Premises Edge Configuration Profile Ansible playbook using the steps described in Section 2.5.5.

Validate the setup of your Kubernetes cluster. Refer to the tasks in Section 5 and run the validation processes according to the hardware and software components that you have installed.

To deploy and validate the On-Premises Edge-VSS profile, run the sample application included. Run the commands inside the pod:

```
kubectl exec -it -n intel-ai intel-ai  --  bash
./run_vehicle_detection_attribute.sh
```

To confirm success, run the following command to verify that a new file named *cars-on-highway-annotated.mp4* is inside the `/tmp` folder.

```
ls -al /tmp
```

[6] See backup for workloads and configurations or visit Performance Index. Results may vary.

10 BMRA Access Edge Configuration Profile Setup

This section contains a step-by-step description of how to set up a BMRA Access Edge Flavor.

To use the Access Edge Configuration Profile, perform the following steps:

1. Choose your hardware, set it up, and configure the BIOS. Refer to Section 10.1 for details.
   You also need to build your Kubernetes cluster.
2. Download the Ansible playbook for your Configuration Profile. Refer to Section 10.2 for details.
3. Configure the optional Ansible parameters using the information in the Configuration Profile tables. Refer to Section 10.3 for details.
4. Deploy the platform. Refer to Section 10.4 for details.
5. Validate the setup of your Kubernetes cluster. Refer to the tasks in Section 5 and run the validation processes according to the hardware and software components that you have installed.

Be aware of the definitions of terminology used in tables in this section.

TERM	DESCRIPTION
Hardware Taxonomy	
ENABLED	Setting must be enabled in the BIOS (configured as Enabled, Yes, True, or similar value)
DISABLED	Setting must be disabled in the BIOS (configured as Disabled, No, False, or any other value with this meaning.)
OPTIONAL	Setting can be either disabled or enabled, depending on user's workload. Setting does not affect the Configuration Profile or platform deployment.
Software Taxonomy	
TRUE	Feature is included and enabled by default.
FALSE	Feature is included but disabled by default - can be enabled and configured by user.
N/A	Feature is not included and cannot be enabled or configured.

10.1 Step 1 - Set Up Access Edge Configuration Profile Hardware

The table in this section lists the hardware BOM for the Access Edge Configuration Profile, including control node, worker node base, and worker node plus. We recommend that you set up at least one control node and one worker node.

Table 50. Hardware Setup for Access Edge Configuration Profile

NODE OPTIONS	3RD GEN INTEL XEON SCALABLE PROCESSOR	4TH GEN INTEL XEON SCALABLE PROCESSOR	INTEL XEON D PROCESSOR
Control node options	Controller_3rdGen_1	Controller_4thGen_1	Controller_Xeon_D_1
Worker node options	See Note	See Note	Worker_Xeon_D_Base_1

Note: Refer to BIOS Settings for FlexRAN™ Reference Architecture Platforms Based on Intel® Xeon® Processors

10.2 Step 2 - Download Access Edge Configuration Profile Ansible Playbook

This section contains details for downloading the Access Edge Configuration Profile Ansible playbook. It also provides an overview of the Ansible playbook and lists the software that is automatically installed when the playbook is deployed.

Download the Access Edge Configuration Profile Ansible playbook using the steps described in Section 2.5.

10.2.1 Access Edge Configuration Profile Ansible Playbook Overview

The Ansible playbook for the Access Edge Configuration Profile allows you to provision a production-ready Kubernetes cluster. Every capability included in the Access Edge Configuration Profile playbook can be disabled or enabled. Refer to the diagram and group and host variables tables below to see which Ansible roles are included and executed by default.

The diagram shows the architecture of the Ansible playbooks that are included in the Access Edge Configuration Profile.

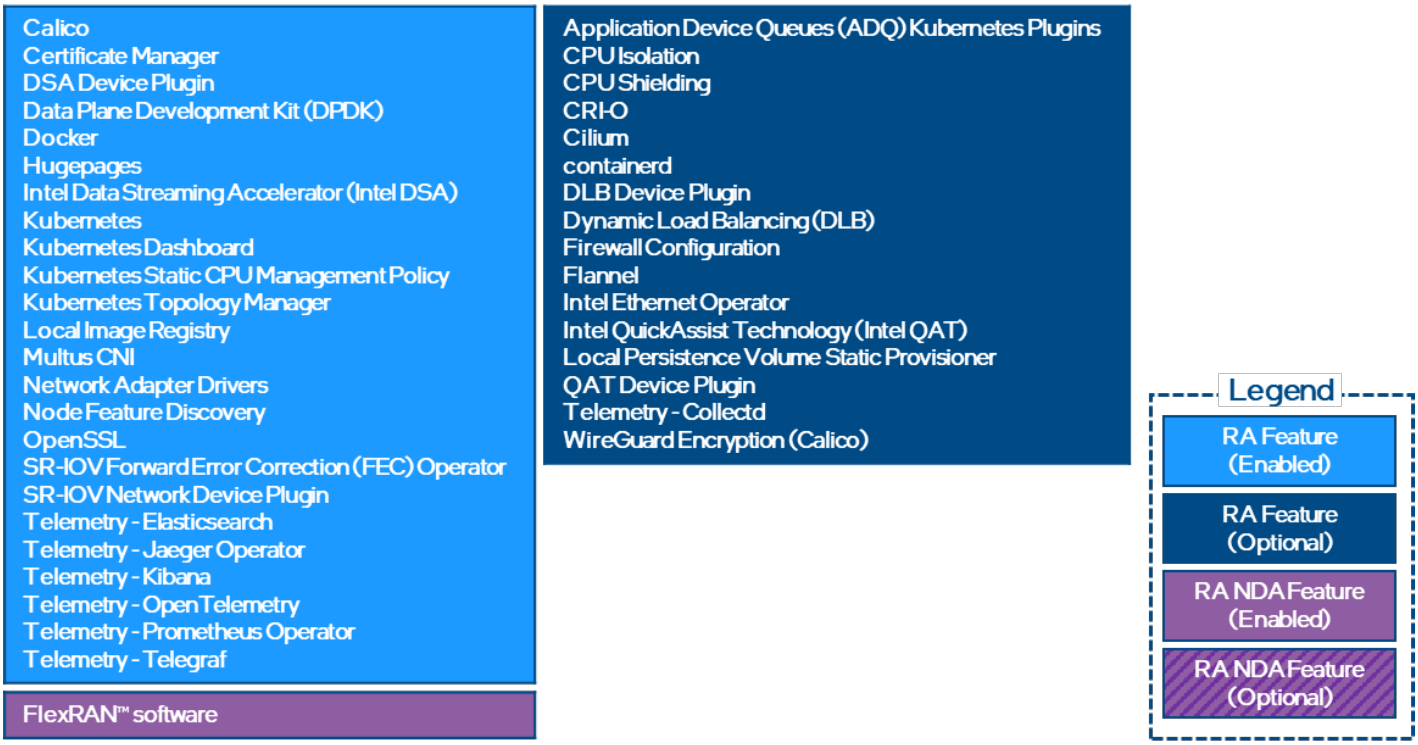

Calico
Certificate Manager
DSA Device Plugin
Data Plane Development Kit (DPDK)
Docker
Hugepages
Intel Data Streaming Accelerator (Intel DSA)
Kubernetes
Kubernetes Dashboard
Kubernetes Static CPU Management Policy
Kubernetes Topology Manager
Local Image Registry
Multus CNI
Network Adapter Drivers
Node Feature Discovery
OpenSSL
SR-IOV Forward Error Correction (FEC) Operator
SR-IOV Network Device Plugin
Telemetry - Elasticsearch
Telemetry - Jaeger Operator
Telemetry - Kibana
Telemetry - OpenTelemetry
Telemetry - Prometheus Operator
Telemetry - Telegraf
FlexRAN™ software
Application Device Queues (ADQ) Kubernetes Plugins
CPU Isolation
CPU Shielding
CRI-O
Cilium
containerd
DLB Device Plugin
Dynamic Load Balancing (DLB)
Firewall Configuration
Flannel
Intel Ethernet Operator
Intel QuickAssist Technology (Intel QAT)
Local Persistence Volume Static Provisioner
QAT Device Plugin
Telemetry - Collectd
WireGuard Encryption (Calico)
Legend
RA Feature (Enabled)
RA Feature (Optional)
RA NDA Feature (Enabled)
RA NDA Feature (Optional)

Figure 5. Access Edge Configuration Profile Ansible Playbook

10.3 Step 3 - Set Up Access Edge Configuration Profile

Review the optional Ansible group and host variables in this section and select options that match your desired configuration.

1. Update the `inventory.ini` file with your environment details as described in Section 2.5.3.
2. Create `host_vars` files for all worker nodes as specified in Section 2.5.4.
3. Update group and host variables to match your desired configuration as specified in Section 2.3.4. Refer to the tables in Section 10.3.1 and Section 10.3.2.

Variables are grouped into two main categories:

1. Group variables – apply to both control and worker nodes and have cluster-wide impact.
2. Host variables – scope is limited to a single worker node.

The tables below are a summary of group and host variables. For lists showing all configurable properties, see Section 6.3 and Section 6.4. All of the variables are important but pay special attention to variables in **bold** as they almost always need to be updated to match the target environment.

10.3.1 Access Edge Configuration Profile Group Variables

Table 51. Access Edge Configuration Profile – Group Variables

COMPONENT	VALUE	
Kubernetes	true	For the list of all configurable properties, see Section 6.3
nfd_enabled	true	
topology_manager_enabled	true	
sriov_network_operator_enabled	true	
sriov_net_dp_enabled	false	
example_net_attach_defs	false	
collectd_enabled	false	
telegraf_enabled	true	

10.3.2 Access Edge Configuration Profile Host Variables[7]

Table 52. Access Edge Configuration Profile – Host Variables

COMPONENT	VALUE	
iommu_enabled	true	For the list of all configurable properties, see Section 6.4
sriov_cni_enabled	false	
install_dpdk	true	
isolcpus_enabled	true	
dataplane_interfaces	[]	

10.4 Step 4 - Deploy and Validate Access Edge Configuration Profile Platform

Deploy the Access Edge Configuration Profile Ansible playbook using the steps described in Section 2.5.5.

Validate the setup of your Kubernetes cluster. Refer to the tasks in Section 5 and run the validation processes according to the hardware and software components that you have installed.

For more information, see the Network and Edge Reference System Architecture with FlexRAN™ Software – Setup on a Single Server Quick Start Guide.

[7] See backup for workloads and configurations or visit Performance Index. Results may vary.

11 BMRA Remote Central Office-Forwarding Configuration Profile Setup

This section contains a step-by-step description of how to set up a BMRA Remote Central Office-Forwarding Flavor.

To use the Remote Central Office-Forwarding Configuration Profile, perform the following steps:

1. Choose your hardware, set it up, and configure the BIOS. Refer to Section 11.1 for details.
   You also need to build your Kubernetes cluster.
2. Download the Ansible playbook for your Configuration Profile. Refer to Section 11.2 for details.
3. Configure the optional Ansible parameters using the information in the Configuration Profile tables. Refer to Section 11.3 for details.
4. Deploy the platform. Refer to Section 11.4 for details.
5. Validate the setup of your Kubernetes cluster. Refer to the tasks in Section 5 and run the validation processes according to the hardware and software components that you have installed.

Be aware of the definitions of terminology used in tables in this section.

TERM	DESCRIPTION
Hardware Taxonomy	
ENABLED	Setting must be enabled in the BIOS (configured as Enabled, Yes, True, or similar value.)
DISABLED	Setting must be disabled in the BIOS (configured as Disabled, No, False, or any other value with this meaning.)
OPTIONAL	Setting can be either disabled or enabled, depending on user's workload. Setting does not affect the Configuration Profile or platform deployment.
Software Taxonomy	
TRUE	Feature is included and enabled by default.
FALSE	Feature is included but disabled by default - can be enabled and configured by user.
N/A	Feature is not included and cannot be enabled or configured.

11.1 Step 1 - Set Up Remote Central Office-Forwarding Configuration Profile Hardware

The table in this section lists the hardware BOM for the Remote Central Office-Forwarding Configuration Profile, including control node, worker node base, and worker node plus. We recommend that you set up at least one control node and one worker node.

Table 53. Hardware Setup for Remote Central Office-Forwarding Configuration Profile

NODE OPTIONS	3RD GEN INTEL XEON SCALABLE PROCESSOR	4TH GEN INTEL XEON SCALABLE PROCESSOR	INTEL XEON D PROCESSOR
Control node options	Controller_3rdGen_2	Controller_4thGen_2	Controller_Xeon_D_2
Worker node options	Worker_3rdGen_Base_3 or Worker_3rdGen_Plus_2	Worker_4thGen_Base_3 or Worker_4thGen_Plus_2	Worker_Xeon_D_Base_3 or Worker_Xeon_D_Plus_2

11.2 Step 2 - Download Remote Central Office-Forwarding Configuration Profile Ansible Playbook

This section contains details for downloading the Remote Central Office-Forwarding Configuration Profile Ansible playbook. It also provides an overview of the Ansible playbook and lists the software that is automatically installed when the playbook is deployed.

Download the Remote Central Office-Forwarding Configuration Profile Ansible playbook using the steps described in Section 2.5.

11.2.1 Remote Central Office-Forwarding Configuration Profile Ansible Playbook Overview

The Ansible playbook for the Remote Central Office-Forwarding Configuration Profile allows you to provision a production-ready Kubernetes cluster. It also applies any additional requirements, such as host OS configuration or Network Adapter drivers and firmware updates. Every capability included in the Remote Central Office-Forwarding Configuration Profile playbook can be disabled or enabled. Refer to the diagram and group and host variables tables below to see which Ansible roles are included and executed by default.

The diagram shows the architecture of the Ansible playbooks that are included in the Remote Central Office-Forwarding Configuration Profile.

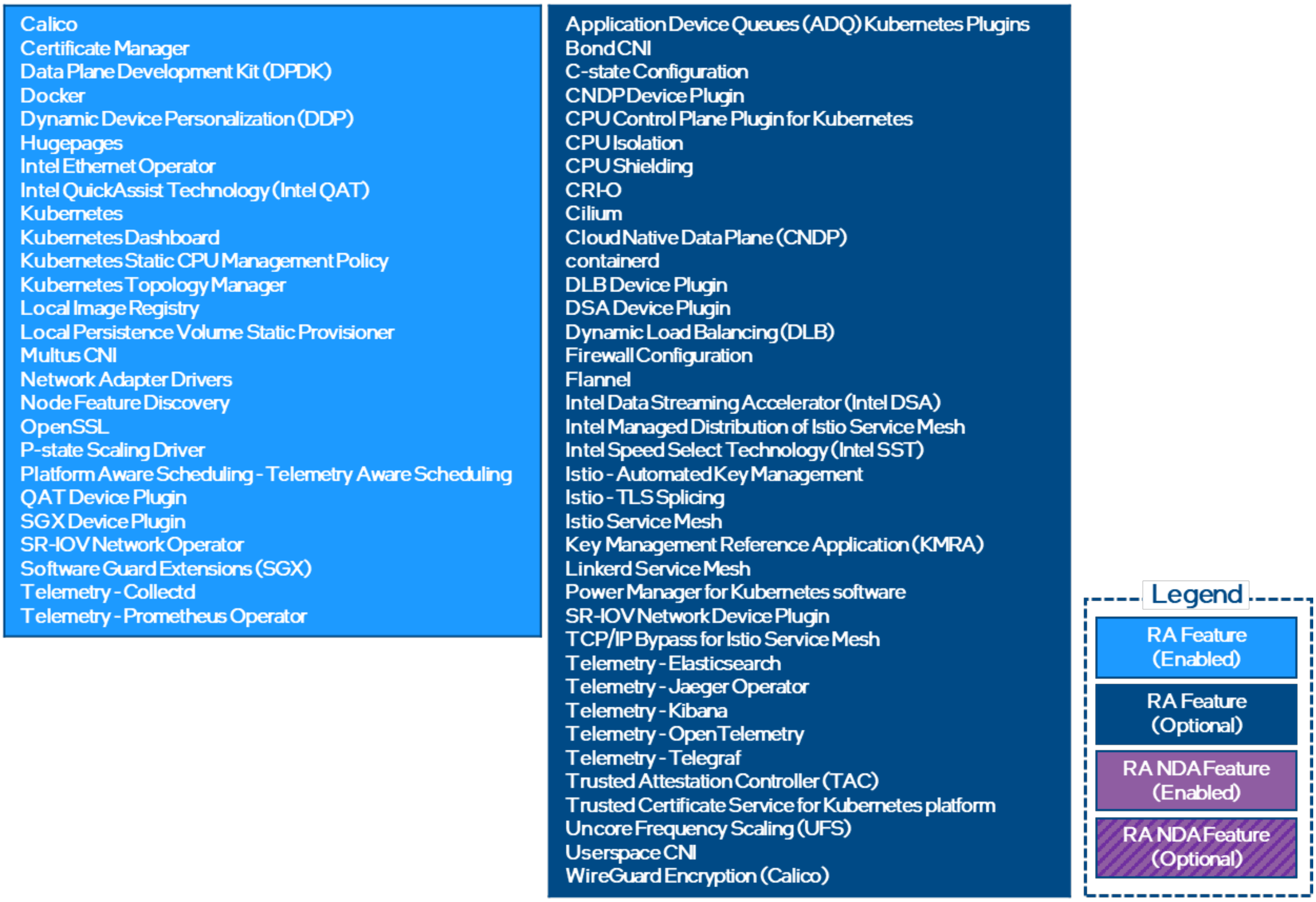

Calico
Certificate Manager
Data Plane Development Kit (DPDK)
Docker
Dynamic Device Personalization (DDP)
Hugepages
Intel Ethernet Operator
Intel QuickAssist Technology (Intel QAT)
Kubernetes
Kubernetes Dashboard
Kubernetes Static CPU Management Policy
Kubernetes Topology Manager
Local Image Registry
Local Persistence Volume Static Provisioner
Multus CNI
Network Adapter Drivers
Node Feature Discovery
OpenSSL
P-state Scaling Driver
Platform Aware Scheduling - Telemetry Aware Scheduling
QAT Device Plugin
SGX Device Plugin
SR-IOV Network Operator
Software Guard Extensions (SGX)
Telemetry - Collectd
Telemetry - Prometheus Operator
Application Device Queues (ADQ) Kubernetes Plugins
Bond CNI
C-state Configuration
CNDP Device Plugin
CPU Control Plane Plugin for Kubernetes
CPU Isolation
CPU Shielding
CRI-O
Cilium
Cloud Native Data Plane (CNDP)
containerd
DLB Device Plugin
DSA Device Plugin
Dynamic Load Balancing (DLB)
Firewall Configuration
Flannel
Intel Data Streaming Accelerator (Intel DSA)
Intel Managed Distribution of Istio Service Mesh
Intel Speed Select Technology (Intel SST)
Istio - Automated Key Management
Istio - TLS Splicing
Istio Service Mesh
Key Management Reference Application (KMRA)
Linkerd Service Mesh
Power Manager for Kubernetes software
SR-IOV Network Device Plugin
TCP/IP Bypass for Istio Service Mesh
Telemetry - Elasticsearch
Telemetry - Jaeger Operator
Telemetry - Kibana
Telemetry - OpenTelemetry
Telemetry - Telegraf
Trusted Attestation Controller (TAC)
Trusted Certificate Service for Kubernetes platform
Uncore Frequency Scaling (UFS)
Userspace CNI
WireGuard Encryption (Calico)
Legend
RA Feature (Enabled)
RA Feature (Optional)
RA NDA Feature (Enabled)
RA NDA Feature (Optional)

Figure 6. Remote Central Office-Forwarding Configuration Profile Ansible Playbook

11.3 Step 3 - Set Up Remote Central Office-Forwarding Configuration Profile

Review the optional Ansible group and host variables in this section and select options that match your desired configuration.

1. Update the `inventory.ini` file with your environment details as described in Section 2.3.3.
2. Create `host_vars` files for all worker nodes as specified in Section 2.5.4.
3. Update group and host variables to match your desired configuration as specified in Section 2.3.4. Refer to the tables in Section 11.3.1 and Section 11.3.2.

Variables are grouped into two main categories:

1. Group variables – apply to both control and worker nodes and have cluster-wide impact.
2. Host variables – scope is limited to a single worker node.

The tables below are a summary of group and host variables. For lists showing all configurable properties, see Section 6.3 and Section 6.4. All of the variables are important but pay special attention to variables in **bold** as they almost always need to be updated to match the target environment.

11.3.1 Remote Central Office-Forwarding Configuration Profile Group Variables

Table 54. Remote Central Office-Forwarding Configuration Profile – Group Variables

COMPONENT	VALUE	
Kubernetes	true	For the list of all configurable properties, see Section 6.3
nfd_enabled	true	
native_cpu_manager_enabled	true	
topology_manager_enabled	true	
sriov_network_operator_enabled	true	
sriov_net_dp_enabled	false	

COMPONENT	VALUE	
sgx_dp_enabled	true	
qat_dp_enabled	false	
openssl_engine_enabled	true	
kmra_enabled	true	
tas_enabled	true	
example_net_attach_defs	false	
collectd_enabled	false	
telegraf_enabled	true	
service_mesh	true	
power_manager	false	

11.3.2 Remote Central Office-Forwarding Configuration Profile Host Variables[8]

Table 55. Remote Central Office-Forwarding Configuration Profile – Host Variables

COMPONENT	VALUE	
iommu_enabled	true	For the list of all configurable properties, see Section 6.4
sriov_cni_enabled	false	
bond_cni_enabled	false	
ddp_enabled	true	
userspace_cni_enabled	false	
hugepages_enabled	true	
isolcpus_enabled	false	
sst_pp_configuration_enabled	false	
install_dpdk	true	
install_ddp_packages	true	
qat_devices	[]	
dataplane_interfaces	[]	

11.4 Step 4 - Deploy and Validate Remote Central Office-Forwarding Configuration Profile Platform

Deploy the Remote Central Office-Forwarding Configuration Profile Ansible playbook using the steps described in Section 2.5.5.

Validate the setup of your Kubernetes cluster. Refer to the tasks in Section 5 and run the validation processes according to the hardware and software components that you have installed.

[8] See backup for workloads and configurations or visit Performance Index. Results may vary.

12 BMRA Regional Data Center Configuration Profile Setup

This section contains a step-by-step description of how to set up your BMRA Regional Data Center Flavor.

To use the Regional Data Center Configuration Profile, perform the following steps:

1. Choose your hardware, set it up, and configure the BIOS. Refer to Section 12.1 for details.
   You also need to build your Kubernetes cluster.
2. Download the Ansible playbook for your Configuration Profile. Refer to Section 12.2 for details.
3. Configure the optional Ansible parameters using the information in the Configuration Profile tables. Refer to Section 12.3 for details.
4. Deploy the platform. Refer to Section 12.4 for details.
5. Validate the setup of your Kubernetes cluster. Refer to the tasks in Section 5 and run the validation processes according to the hardware and software components that you have installed.

Be aware of the definitions of terminology used in tables in this section.

TERM	DESCRIPTION
Hardware Taxonomy	
ENABLED	Setting must be enabled in the BIOS (configured as Enabled, Yes, True, or similar value.)
DISABLED	Setting must be disabled in the BIOS (configured as Disabled, No, False, or any other value with this meaning.)
OPTIONAL	Setting can be either disabled or enabled, depending on user's workload. Setting does not affect the Configuration Profile or platform deployment.
Software Taxonomy	
TRUE	Feature is included and enabled by default.
FALSE	Feature is included but disabled by default - can be enabled and configured by user.
N/A	Feature is not included and cannot be enabled or configured.

12.1 Step 1 - Set Up Regional Data Center Configuration Profile Hardware

The table in this section lists the hardware BOM for the Regional Data Center Configuration Profile, including control node, worker node base, and worker node plus. We recommend that you set up at least one control node and one worker node.

Table 56. Hardware Setup for Regional Data Center Configuration Profile

NODE OPTIONS	3RD GEN INTEL XEON SCALABLE PROCESSOR	4TH GEN INTEL XEON SCALABLE PROCESSOR	INTEL XEON D PROCESSOR
Control node options	Controller_3rdGen_3	N/A*	N/A*
Worker node options	Worker_3rdGen_Plus_3	N/A*	N/A*
*Configuration Profile only tested with 3rd Gen Intel Xeon Scalable processor			

12.2 Step 2 - Download Regional Data Center Configuration Profile Ansible Playbook

This section contains details for downloading the Regional Data Center Configuration Profile Ansible playbook. It also provides an overview of the Ansible playbook and lists the software that is automatically installed when the playbook is deployed.

Download the Regional Data Center Configuration Profile Ansible playbook using the steps described in Section 2.5.

12.2.1 Regional Data Center Configuration Profile Ansible Playbook Overview

The Ansible playbook for the Regional Data Center Configuration Profile allows you to provision a production-ready Kubernetes cluster. It also applies any additional requirements, such as host OS configuration or Network Adapter drivers and firmware updates. Every capability included in the Regional Data Center Configuration Profile playbook can be disabled or enabled. Refer to the diagram and group and host vars tables below to see which Ansible roles are included and executed by default.

The diagram shows the architecture of the Ansible playbooks that are included in the Regional Data Center Configuration Profile.

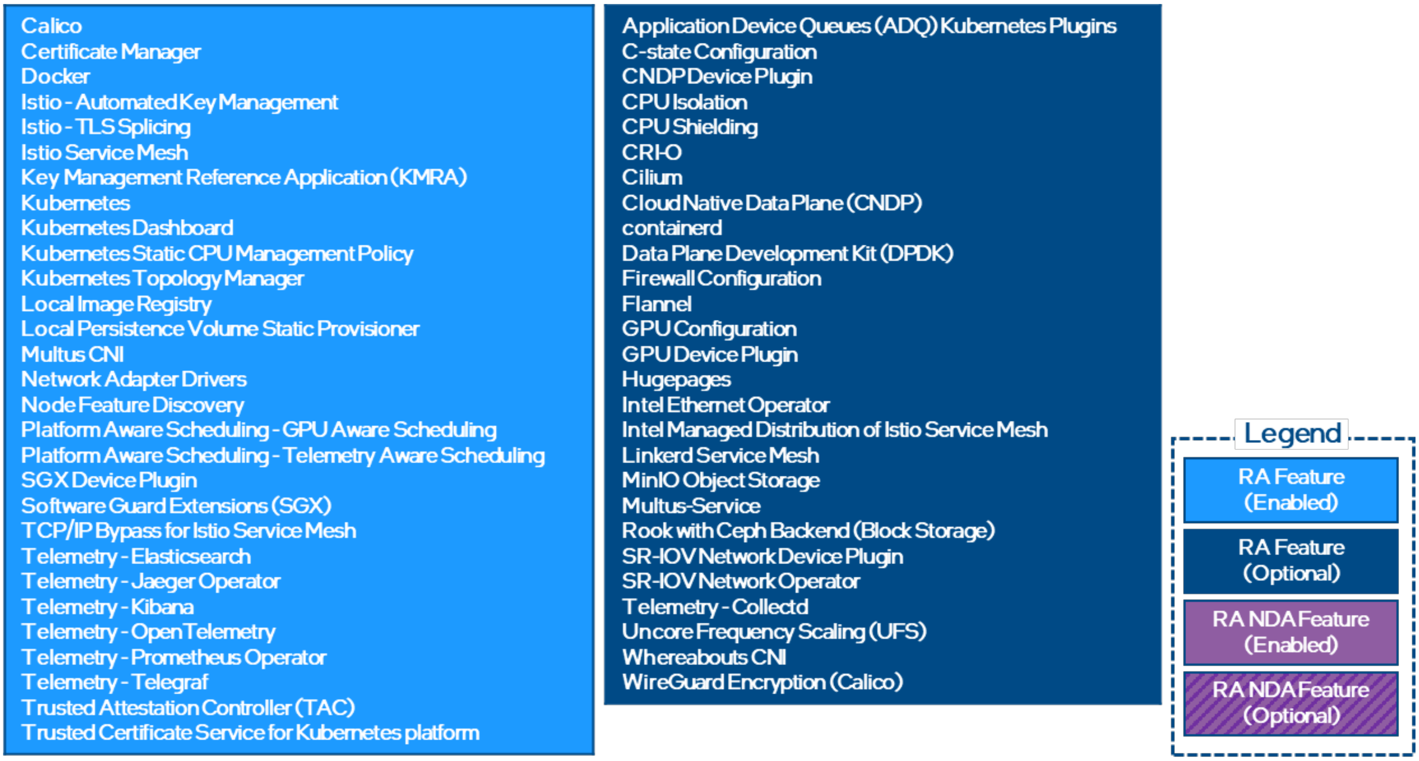

Calico
Certificate Manager
Docker
Istio - Automated Key Management
Istio - TLS Splicing
Istio Service Mesh
Key Management Reference Application (KMRA)
Kubernetes
Kubernetes Dashboard
Kubernetes Static CPU Management Policy
Kubernetes Topology Manager
Local Image Registry
Local Persistence Volume Static Provisioner
Multus CNI
Network Adapter Drivers
Node Feature Discovery
Platform Aware Scheduling - GPU Aware Scheduling
Platform Aware Scheduling - Telemetry Aware Scheduling
SGX Device Plugin
Software Guard Extensions (SGX)
TCP/IP Bypass for Istio Service Mesh
Telemetry - Elasticsearch
Telemetry - Jaeger Operator
Telemetry - Kibana
Telemetry - OpenTelemetry
Telemetry - Prometheus Operator
Telemetry - Telegraf
Trusted Attestation Controller (TAC)
Trusted Certificate Service for Kubernetes platform
Application Device Queues (ADQ) Kubernetes Plugins
C-state Configuration
CNDP Device Plugin
CPU Isolation
CPU Shielding
CRI-O
Cilium
Cloud Native Data Plane (CNDP)
containerd
Data Plane Development Kit (DPDK)
Firewall Configuration
Flannel
GPU Configuration
GPU Device Plugin
Hugepages
Intel Ethernet Operator
Intel Managed Distribution of Istio Service Mesh
Linkerd Service Mesh
MinIO Object Storage
Multus-Service
Rook with Ceph Backend (Block Storage)
SR-IOV Network Device Plugin
SR-IOV Network Operator
Telemetry - Collectd
Uncore Frequency Scaling (UFS)
Whereabouts CNI
WireGuard Encryption (Calico)
Legend
RA Feature (Enabled)
RA Feature (Optional)
RA NDA Feature (Enabled)
RA NDA Feature (Optional)

Figure 7. Regional Data Center Configuration Profile Ansible Playbook

12.3 Step 3 - Set Up Regional Data Center Configuration Profile

Review the optional Ansible group and host variables in this section and select options that match your desired configuration.

1. Update the `inventory.ini` file with your environment details as described in Section 2.5.3.
2. Create `host_vars` files for all worker nodes as specified in Section 2.5.4.
3. Update group and host variables to match your desired configuration as specified in Section 2.3.4. Refer to the tables in Section 12.3.1 and Section 12.3.2.

Variables are grouped into two main categories:

1. Group variables – apply to both control and worker nodes and have cluster-wide impact.
2. Host variables – scope is limited to a single worker node.

The tables below are a summary of group and host variables. For lists showing all configurable properties, see Section 6.3 and Section 6.4. All of the variables are important but pay special attention to variables in **bold** as they almost always need to be updated to match the target environment.

12.3.1 Regional Data Center Configuration Profile Group Variables

Table 57. Regional Data Center Configuration Profile – Group Variables

COMPONENT	VALUE	
Kubernetes	true	For the list of all configurable properties, see Section 6.3
nfd_enabled	true	
native_cpu_manager_enabled	true	
topology_manager_enabled	true	
sriov_network_operator_enabled	false	
sriov_net_dp_enabled	false	
gpu_dp_enabled	true	
tas_enabled	true	
gas_enabled	true	
example_net_attach_defs	false	
collectd_enabled	false	
telegraf_enabled	true	

COMPONENT	VALUE	
service_mesh	true	

12.3.2 Regional Data Center Configuration Profile Host Variables[9]

Table 58. Regional Data Center Configuration Profile – Host Variables

COMPONENT	VALUE	
iommu_enabled	false	For the list of all configurable properties, see Section 6.4
sriov_cni_enabled	false	
hugepages_enabled	false	
isolcpus_enabled	false	
install_dpdk	false	
dataplane_interfaces	[]	

12.4 Step 4 – Deploy and Validate Regional Data Center Configuration Profile Platform

Deploy the Regional Data Center Configuration Profile Ansible playbook using the steps described in Section 2.5.5.

Validate the setup of your Kubernetes cluster. Refer to the tasks in Section 5 and run the validation processes according to the hardware and software components that you have installed.

[9] See backup for workloads and configurations or visit Performance Index. Results may vary.

Part 3:
BMRA Applications

13 Workloads and Application Examples

This section provides examples of how to provision and deploy example applications or workloads.

13.1 Enabling Key Management NGINX Applications

KMRA source code and Dockerfiles: Key Management Reference Application

KMRA docker images on Docker Hub:

- AppHSM: https: //hub.docker.com/r/intel/apphsm
- ctk_loadkey: https://hub.docker.com/r/intel/ctk_loadkey
- PCCS: https://hub.docker.com/r/intel/pccs

KMRA Helm charts are in `/roles/kmra_install/charts`.

Steps to deploy the full KMRA NGINX demo:

1. Generate a new PCCS primary API key and update the `kmra.pccs.api_key` variable in `group_vars/all.yml` (go to Intel® Provisioning Certification Service for ECDSA Attestation and subscribe).
2. Ensure that the `kmra_deploy_demo_workload` variable in the `group_vars/all.yml` is set to `true`.
3. Deploy the `on_prem` or `remote_fp` profile to set up KMRA demo with NGINX. The `kmra` variable must be set to `on` in `profiles/profiles.yml`.

13.2 Enabling Trusted Certificate Service

Trusted Certificate Service (TCS) is a Kubernetes certificate signing solution that uses the security capabilities provided by Intel® SGX. The signing key is stored and used inside the SGX enclaves and is never stored in clear anywhere in the system. TCS is implemented as a cert-manager external issuer by supporting both cert-manager and Kubernetes certificate signing APIs.

To enable TCS on BMRA, follow the guide available at Trusted Certificate Issuer.

13.2.1 Istio Custom CA Integration Using Kubernetes CSR

Istio supports integrating custom certificate authority (CA) using Kubernetes CSR as an experimental feature.

Detailed example steps described in the Istio Custom CA with CSR document show how to provision Istio workload certificates using an Issuer provided by the Trusted Certificate Service (TCS).

Note: Due to misconfiguration of the Istio Demo application, you might need to disable hugepages temporarily to avoid the demo app becoming stuck in the `CrashLoopBackOff` state. To disable hugepages, execute the following command on the worker node:

```
echo 0 > /proc/sys/vm/nr_hugepages
```

13.2.2 Remote Attestation and Manual Key Management

TCS supports SGX remote attestation and the sample key management reference application.

All required steps are described in the Integrate Key Server document.

13.3 Service Mesh Automated Remote Attestation and Key Management with KMRA, TCS, and TCA

Remote attestation is an advanced feature that allows an entity to gain the relying party's trust. Remote attestation gives the relying party increased confidence that the software is running inside an SGX enclave. The attestation results include the identity of the software being attested and an assessment of possible software tampering.

Key management enables external key management systems to deliver the certificates and keys via more secure mechanisms into the SGX enclave. To enable the automated key management feature, KMRA AppHSM, and KMRA PCCS applications must be enabled and configured as well as Trusted Certificate Service (TCS) and Trusted Certificate Attestation (TCA). BMRA tries to install all dependencies and configure the host with reasonable defaults.

KMRA application settings are collected under the `kmra` variable in the `group_vars/all.yml` file and all default values are available for reference in the `roles/kmra_install/defaults/main.yml` file. If you need to overwrite any default value, redefine it in the `group_vars/all.yml` file while keeping the variable structure.

In general, TCS does not require specific configuration. Default values used for TCS deployment are collected in the `roles/tcs_install/vars/main.yml` file and can be redefined in the `group_vars/all.yml` file.

TCA depends on settings of KMRA AppHSM, which should match. Refer to the default values, which can be found in the `roles/tca_install/vars/main.yml` file. Default values can be redefined in the `group_vars/all.yml` file.

Service mesh default settings can be found in the `roles/service_mesh_install/vars/main.yml` file.

For detailed documentation on components involved in this feature, refer to:

- KMRA: Key Management Reference Application
- TCS: Trusted Certificate Issuer
- TCA: Trusted Attestation Controller

13.4 Istio TLS Splicing

To configure Istio with TLS splicing, first enable it in the `group_vars/all.yml` file.

```
service_mesh:
  enabled:true
  tls_splicing:
    enabled: true
```

The config creates an ingress gateway to act as a forward proxy, registers virtual service rule and external service entry to implement TLS passthrough for external service.

A client outside the mesh can use the cluster ingress gateway to access external services with TLS splicing.

```
export INGRESS_PORT=$(kubectl -n istio-system get service istio-ingressgateway -o
jsonpath='{.spec.ports[?(@.name=="http2")].nodePort}')
export SECURE_INGRESS_PORT=$(kubectl -n istio-system get service istio-ingressgateway -o
jsonpath='{.spec.ports[?(@.name=="https")].nodePort}')
export TCP_INGRESS_PORT=$(kubectl -n istio-system get service istio-ingressgateway -o
jsonpath='{.spec.ports[?(@.name=="tcp")].nodePort}')
export INGRESS_HOST=$(kubectl get po -l istio=ingressgateway -n istio-system -o
jsonpath='{.items[0].status.hostIP}')

curl -s -v  --resolve www.example.com:$SECURE_INGRESS_PORT:$INGRESS_HOST
https://www.example.com:$SECURE_INGRESS_PORT
```

13.5 Web Application Firewall Using Traffic Analytics Development Kit

The functionality of the Web Application Firewall (WAF) running in the cluster can be tested from the command line. Start by getting the IP and port of the firewall:

```
# export NODE_PORT=$(kubectl get --namespace modsec-tadk -o
jsonpath="{.spec.ports[0].nodePort}" services tadk-intel-tadkchart)

# export NODE_IP=$(kubectl get nodes --namespace modsec-tadk -o
jsonpath="{.items[0].status.addresses[0].address}")
```

Note: If the `kube_proxy_nodeport_addresses_cidr` option in `group_vars` has not been commented, the nodeport (`NODE_IP`) will not be available externally. In that, case, replace `NODE_IP` with `localhost`.

Start by verifying that the nginx server can be reached:

```
## If nodeports are not available externally (default):
# curl http://localhost:$NODE_PORT

## If nodeports are available externally
# curl http://$NODE_IP:$NODE_PORT
```

The output should be the default “Welcome to nginx” webpage.

Now try sending a message with sample credentials to the firewall:

```
## If nodeports are not available externally (default):
# curl -d "username=admin&password=unknown' or '1'='1" "localhost:$NODE_PORT"

## If nodeports are available externally
# curl -d "username=admin&password=unknown' or '1'='1" "$NODE_IP:$NODE_PORT"
```

The resulting error code should be “403” (Forbidden), showing the firewall has blocked the request.

Part 4:
BMRA Release Notes

Appendix A RA Release Notes

This section lists the notable changes from the previous releases, including new features, bug fixes, and known issues for BMRA, VMRA, and Cloud RA.[10]

A.1 RA 23.02 Release Notes

New Components/Features:

- Media Analytics Libraries
  - Intel® Deep Learning Streamer (Intel® DL Streamer), GStreamer, OpenVINO™ toolkit
  - OpenCL™ software, Level zero GPU, DPC++, and VAAPI from the Intel® GPU toolkit
- FlexRAN™ software running as a Docker container (now available without NDA)
- Rook/Ceph as a storage-related component
- Rocky Linux 9.1 as base operating system (with some limitations mentioned below)
- Non-root user deployment of Virtual Machine Reference System Architecture (VMRA)
- Custom cluster naming in VMRA
- Support for using Amazon Web Services (AWS) and Azure "Cloud" CLIs as an alternative to Terraform
- Azure Kubernetes Service (AKS) support for static CPU Management Policy and Intel® CPU Control Plane Plugin for Kubernetes
- Intel® Software Guard Extensions (Intel® SGX) on AKS

Updates/Changes:

- Software versions upgraded for the majority of RA components (See User Guide for complete BOM and versions)
  Notable updates:
  - Kubernetes to v1.26.1
  - MinIO to v4.5.8
  - DPDK to v22.11.1
  - Service Mesh to v1.17.1
  - VPP to v2302
  - KMRA to v2.3
- Eliminated the BMRA for Object Storage Setup deployment model. The storage-related features (MinIO, LPVSP, and Rook/Ceph) are now provided as optional components in select configuration profiles.
- Support of geo-specific mirrors for Kubespray (for example, in the People's Republic of China)
- Supported Kubernetes versions updated for AKS and Amazon EKS
- Ubuntu images updated for AKS and Amazon EKS
- Ability to deploy more RA software components on Azure and AWS
  - Elasticsearch
  - Kibana

New Hardware (Platforms/CPUs/GPUs/Accelerators):

- N/A

Removed Support:

- `full_nfv` profile
- Ubuntu 20.04 as base operating system
- Rocky Linux 9.0 as base operating system

Known Limitations/Restrictions:

- When using the Cilium CNI, secondary interfaces are not supported
- Intel® Dynamic Load Balancer (Intel® DLB) is not fully supported on Rocky Linux 9.1
- FlexRAN container support is limited to FlexRAN v22.07, Ubuntu 22.04 base operating system, and only on 3rd Gen Intel® Xeon® Scalable processors
- Media Analytics is supported only with Docker runtime
- MinIO is supported only with CRI-O runtime
- VMRA cluster expansion with additional VM nodes might fail
- Trusted Certificate Attestation (TCA) is not fully functional in VMRA

A.2 RA 23.02 Release Updates

The following table lists key features of the 4th Gen Intel Xeon Scalable processor and the support for those features in RA 23.02.

[10] See backup for workloads and configurations or visit www.Intel.com/PerformanceIndex. Results may vary.

Table 59. Status of Support for Key Features of 4th Gen Intel Xeon Scalable Processor in BMRA 23.02

CATEGORY	FEATURE	BMRA 23.02 SUPPORT	BMRA 23.02 STATUS/COMMENTS
CPU / Accelerator	IAX	Yes	
	QAT	Yes	
	DLB	Yes	Not yet available through hypervisor
	DSA	Yes	Not yet available through hypervisor
Power Management	SST-PP, SST-TF SST-BF, SST-CP	Yes	
Security	SGX	Yes	
RAS	RAS	Yes	
ISA	FP-16 (5G ISA)	Yes	
	AMX (TMUL)	No	Not yet supported in RA
	VP2INTERSECT	Yes	
	AIA (MOVDIRI, Power Instrs.)	Yes	
I/O	CXL 1.1	Yes	
	PCI Gen5	Yes	
Virtualization	Intel® Scalable IOV	Yes	
	SVM	Yes	Supported for 4th Gen Intel® Xeon® Scalable processor

Refer to the following tables for other features of 4th Gen Intel Xeon Scalable processor enabled in prior BMRA releases.

A.3 RA 22.11.1 Release Notes

New Components/Features:

- N/A (same as RA 22.11)

Updates/Changes:

- Intel® QAT 2.0 drivers for 4th Gen Intel® Xeon® Scalable processors (formerly code named Sapphire Rapids [SPR]) are sourced from public repo and no longer under NDA. Ignore Guide requirement to provide the *QAT20.L.0.9.9-00019.tar.gz* driver package file.
- Resolved issue regarding downloading CPUID for Rocky Linux 8.5 and RHEL 9.

New Hardware (Platforms/CPUs/GPUs/Accelerators):

- N/A (same as RA 22.11)

Removed Support:

- N/A (same as RA 22.11)

Known Limitations/Restrictions:

- N/A (same as RA 22.11)

A.4 RA 22.11 Release Updates

New Components/Features:

- Intel® Software Guard Extensions (Intel® SGX) support on 4th Gen Intel Xeon Scalable processors (SPR)
- Enabled select features (SGX and QAT) to be deployed through Ansible tags to facilitate interoperability with Intel's Workload Services Framework (WSF)
- Support deployment via hostname or FQDN
- Support for clean-up /re-deploy of the 'Basic' profile
- Support for the Content Delivery Network (CDN) use case
- New observability stack including Opentelemetry and Kibana (and expanding Jaeger support to VMRA)
- Intel's CPU Control Plane Management
- Support for Local Volume Provisioner
- Support for Cilium
- Cloud RA: Support for Azure AKS deployments on top of previous support for AWS EKS

- Cloud RA: Support for generating and deploying configuration profiles and using the generated host/group_vars during deployment
- Cloud RA: Support for Cilium with kube-proxy and eBPF CNI on Azure
- Cloud RA: Proximity Placement Groups for Azure

Updates/Changes:

- Updated Kubernetes to 1.25.3 (min supported 1.23)
- Updated CRI-O to 1.25.1
- Updated Linkerd to 2.12.1
- Upgraded Node Feature Discovery to 0.11.3-minimal
- Updated Key Management Reference Application (KMRA) support to 2.2.2
- Updated FlexRAN support to 22.07.3
- Updated SR-IOV-FEC Operator to image 2.5.0
- Updated TADK to 22.09 Docker image
- Updated Intel device plugins (DPs) to release-0.25.1
- Updated NGINX image to 1.23.2
- Updated Vector Packet Processing (VPP) to version 22.10
- Updated Trusted Attestation Controller (TAC) to version 0.2.0
- Updated Trusted Certificate Issuer (TCS) to version 0.2.0
- Updated Data Plane Development Kit (DPDK) version to 22.11
- Updated Open vSwitch with DPDK to 3.0.1
- Updated Platform Aware Scheduling (PAS) version to 0.9
- Updated collectd
- Updated Telegraf to 1.2
- Updated Grafana to 9.1.8
- Updated Prometheus to 2.39.1
- Updated Prometheus Adapter to 0.10.0
- Updated Prometheus Operator to 0.60.0
- Updated Kube RBAC Proxy to 0.13.1
- Updated OpenTelemetry to 0.1.8.3
- Updated Jaeger to 1.39.0
- Updated cAdvisor to 2.2.2
- Updated Intel® Ethernet firmware and drivers
- Updated Intel® QuickAssist Technology (Intel® QAT) drivers
- Updated OpenSSL QAT Engine to 0.6.17
- Update Intel IPsec MB to 1.3
- Updated Intel Trusted Attestation Controller (TAC) to 0.40.0
- Updated Trusted Certificate Issuer (TCS) to 0.40.0
- Updated Cloud Native Data Plane (CNDP) to 22.08
- Updated MinIO Operator to 4.4.28
- Updated MinIO Console to 0.19.4
- Updated Intel RDT Telemetry Plugin to 4.4.1
- Updated Forward Error Correction (FEC) Operator to 22.38
- Updated Forward Error Correction (FEC) Operator SDK to 1.25.1
- Updated Operator Package Manager to 1.26.2
- Updated ADQ-K8s-plugin to 22.06-1
- Autodetection of the QAT and FEC ACC devices
- Extended Linkerd as a service mesh option for VMRA
- Support for Cluster Flow Config with the updated Intel Ethernet Operator (IEO)
- Enhanced discovery mechanism for Cloud RA

New Hardware (Platforms/CPUs/GPUs/Accelerators):

- Intel® Data Center GPU, code named Arctic Sound-M (ATS-M)
- FEC Accelerator (ACC200) embedded into 4th Gen Intel® Xeon® Scalable processors (Sapphire Rapids) with vRAN Boost
- 3rd Gen Intel® Xeon® Scalable processor CPU (Ice Lake) SKU: 6348
- 4th Gen Intel® Xeon® Scalable processor CPU (Sapphire Rapids) SKUs: 6421N, 6438N, 8480+, 8487C

Removed Support:

- SG1 Graphics card
- Visual Cloud Accelerator Card for Analytics (VCAC-A)

Known Limitations/Restrictions:

- Key Management Reference Application (KMRA) is NOT supported on 4th Gen Intel Xeon Scalable processors (SPR)
- KMRA is NOT supported on CRI-O runtime with 3rd Gen Intel Xeon Scalable processors (ICX)
- CRI-O runtime is NOT supported on Ubuntu 20.04
- NIC Firmware update is NOT supported through Intel Ethernet Operator (IEO)
- VMRA support with containerd runtime environment is limited (unstable) and might exhibit failures of some pods

The following table lists key features of the 4th Gen Intel Xeon Scalable processor and the support for those features in BMRA 22.11.

Table 60. Status of Support for Key Features of 4th Gen Intel Xeon Scalable Processor in BMRA 22.11

CATEGORY	FEATURE	BMRA 22.11 SUPPORT	BMRA 22.11 STATUS/COMMENTS
CPU / Accelerator	IAX	Yes	
	QAT	Yes (NDA)	QAT drivers are NDA and not yet open source
	DLB	Yes	Not yet available through hypervisor
	DSA	Yes	Not yet available through hypervisor
Power Management	SST-PP, SST-TF SST-BF, SST-CP, User Wait Instructions	Yes	
Security	SGX	Yes	
RAS	RAS	Yes	
ISA	FP-16 (5G ISA)	Yes	
	AMX (TMUL)	No	Not yet supported in RA
	VP2INTERSECT	Yes	
	AIA (MOVDIRI, Power Instrs.)	Yes	
I/O	CXL 1.1	Yes	
	PCI Gen5	Yes	
Virtualization	Intel® Scalable IOV	Yes	
	SVM	Yes	Supported for 4th Gen Intel® Xeon® Scalable processor

Refer to the following tables for other features of 4th Gen Intel Xeon Scalable processor enabled in prior BMRA releases.

A.5 RA 22.08 Release Updates

New Components/Features:

- Inclusion of the Cloud RA in the distribution
- Inclusion of OpenTelemetry
- Inclusion of Jaeger
- Inclusion of Linkerd Service Mesh (version 2.12.0)
- Inclusion of standalone cAdvisor
- Inclusion of Intel® Scalable I/O Virtualization (Intel® Scalable IOV) for 4th Gen Intel® Xeon® Scalable processor
- Inclusion of Intel® Data Streaming Accelerator (Intel® DSA) for 4th Gen Intel® Xeon® Scalable processor
- Inclusion of Intel® Dynamic Load Balancer (Intel® DLB) for 4th Gen Intel® Xeon® Scalable processor
- Inclusion of 5G Core support in the Regional Data Center Configuration Profile
- Inclusion of post-deployment hook for additional automation
- Scale up/down cluster nodes after initial deployment
- Support added via DPDK for new User Wait power instructions in 4th Gen Intel Xeon Scalable processors
- Support for Load Balancing on additional interfaces when using Multus CNI
- Support for upgrade/downgrade of network adapter drivers post deployment
- Support binding of QAT to new Virtual Function (VF)
- Support for 3rd Gen Intel® Xeon® Scalable processor platforms for FlexRAN
- Support for xRAN Test Mode for FlexRAN

- Support for RHEL 8.6 Realtime as base operating system for FlexRAN
- Support for Rocky Linux 9.0
- Support for RHEL 9.0
- Tech Preview: Support for Application Device Queues (ADQ)

New Platforms/CPUs:

- Intel Coyote Pass with 8360Y 3rd Gen Intel® Xeon® Scalable processor CPUs
- Intel Fox Creek Pass with XCC E3-QS 4th Gen Intel® Xeon® Scalable processor CPUs
- Intel Ruby Pass platform
- 4th Gen Intel® Xeon® Scalable processor CPU SKUs: 8470N, 8471N, 8490H

Updates/Changes:

- Updated Ansible to 5.7.1 and ansible-core to 2.12.5
- Updated Kubernetes to 1.24.3 (min supported 1.22)
- Updated Key Management Reference Application (KMRA) support to 2.2.1
- Updated FlexRAN support to 22.07
- Updated TADK to 22.3 Docker image
- Updated Intel device plugins (DPs) to release-0.24
- Updated NGINX image to 1.23.1
- Updated Vector Packet Processing (VPP) to version 22.10
- Updated Trusted Attestation Controller (TAC) to version 0.2.0
- Updated Trusted Certificate Issuer (TCS) to version 0.2.0
- Updated Data Plane Development Kit (DPDK) version to 22.07
- Updated Platform Aware Scheduling (PAS) version to 0.8
- Updated Grafana to 8.5.11
- Updated Prometheus to 2.37.1
- Updated Intel® Ethernet firmware and drivers
- Updated Intel® QuickAssist Technology (Intel® QAT) drivers
- Replaced Barometer Collectd with Containerized Collectd
- Enhanced automatic CPU pinning and isolation for Virtual Machine Reference System Architecture (VMRA)
- VM Cluster expansion with new nodes and/or hosts in VMRA

Removed Support:

- RHEL and Rocky Linux 8 series as base operating systems

Known Limitations/Restrictions:

- Intel® Software Guard Extensions (Intel® SGX) and KMRA are NOT supported on 4th Gen Intel Xeon Scalable processors. These features are automatically disabled on all operating systems
- 4th Gen Intel Xeon Scalable processor Intel DSA and Intel DLB features are NOT supported on Ubuntu 20.04
- Enabling support of Intel QAT on 4th Gen Intel Xeon Scalable processor requires an NDA
- CRI-O runtime is not supported on RHEL and Rocky Linux 9.0

The following table lists key features of the 4th Gen Intel Xeon Scalable processor and the support for those features in BMRA 22.08.

Table 61. Status of Support for Key Features of 4th Gen Intel Xeon Scalable Processor in BMRA 22.08

CATEGORY	FEATURE	LINUX KERNEL AVAILABILITY	BMRA 22.08 STATUS
CPU / Accelerator	IAX	5.11	BMRA OS includes the kernel support since BMRA 21.09 release.
	QAT	5.11	Supported and tested. Also validated as part of the NGINX workload since BMRA 21.09 release.
	DLB	5.14	Available as userspace library in DPDK since BMRA 21.09 release. DLB is not up-streamed in a Linux kernel yet, drivers available from 01.org.
	DSA	5.14	DSA supported and tested, including support for the DSA operator since BMRA 21.09 release.
Power Management	SST-PP, SST-TF SST-BF, SST-CP	5.3	SST-BF and SST-PP were available in previous generation. New SST-CP and SST-TF are supported and tested since BMRA 21.09 release.
Security	SGX	5.11	Not yet fully supported on 4th Gen Intel Xeon Scalable processors
	Crytodev and CrytoNI	N/A	Supported and tested through DPDK 21.11 since BMRA 22.01 release. Not supported in BMRA 22.08.

CATEGORY	FEATURE	LINUX KERNEL AVAILABILITY	BMRA 22.08 STATUS
RAS	RAS	5.11	collectd and Telegraf include RAS plugins since BMRA 21.09 release.
ISA	FP-16 (5G ISA)	5.11	BMRA OS includes the kernel support since BMRA 21.09 release.
	AMX (TMUL)	5.16	Not yet supported.
	VP2INTERSECT	5.4	BMRA OS includes the kernel support since BMRA 21.09 release.
	AIA (MOVDIRI, Power Instrs.)	5.10	Supported and tested as part of the DPDK 21.08 release since BMRA 21.09 release.
I/O	CXL 1.1	5.11	Supported but not tested as part of the DPDK 21.08 release and since RA 21.09 release.
	PCI Gen5	5.3	BMRA OS includes the kernel support since RA 21.09 release.
Virtualization	Intel® Scalable IOV	N/A	BMRA OS includes the kernel support since RA 21.09 release.
	SVM	N/A	Not yet supported.

Part 5:
Abbreviations

Appendix B Abbreviations

The following abbreviations are used in this document.

ABBREVIATION	DESCRIPTION
AGF	Access Gateway Function
AI	Artificial Intelligence
AIC	Add In Card
AIA	Accelerator Interfacing Architecture
AMX	Advance Matrix Multiply
API	Application Programming Interface
BIOS	Basic Input/Output System
BKC	Best Known Configuration
BMRA	Bare Metal Reference Architecture
BOM	Bill of Material
CA	Certificate Authority
CDN	Content Delivery Network
CLOS	Class of Service
Cloud RA	Cloud Reference System Architecture
CMK	CPU Manager for Kubernetes
CMTS	Cable Modem Termination System
CNCF	Cloud Native Computing Foundation
CNDP	Cloud Native Data Plane (CNDP)
CNI	Container Network Interface
CO	Central Office
CTK	Crypto-API Toolkit
CU	Central Unit
CXL	Compute Express Link
DDP	Dynamic Device Personalization
DHCP	Dynamic Host Configuration Protocol
DLB	Intel® Dynamic Load Balancer (Intel® DLB)
DNS	Domain Name Service
DPDK	Data Plane Development Kit
DRAM	Dynamic Random Access Memory
DSA	Intel® Data Streaming Accelerator (Intel® DSA)
DU	Distribution Unit
EIST	Enhanced Intel SpeedStep® Technology
FPGA	Field-Programmable Gate Array
FW	Firmware
GAS	GPU Aware Scheduling
GPU	Graphics Processor Unit
HA	High Availability
HCC	High Core Count
HSM	Hardware Security Model
HT	Hyper Threading
IAX	In-Memory Analytics
IMC	Integrated Memory Controller

ABBREVIATION	DESCRIPTION
Intel® AVX	Intel® Advanced Vector Extensions (Intel® AVX)
Intel® AVX-512	Intel® Advanced Vector Extension 512 (Intel® AVX-512)
Intel® DCAP	Intel® Software Guard Extensions Data Center Attestation Primitives (Intel® SGX DCAP)
Intel® DLB	Intel® Dynamic Load Balancer (Intel® DLB)
Intel® DSA	Intel® Data Streaming Accelerator (Intel® DSA)
Intel® HT Technology	Intel® Hyper-Threading Technology (Intel® HT Technology)
Intel® QAT	Intel® QuickAssist Technology (Intel® QAT)
Intel® RDT	Intel® Resource Director Technology (Intel® RDT)
Intel® SecL – DC	Intel® Security Libraries for Data Center (Intel® SecL – DC)
Intel® SGX	Intel® Software Guard Extensions (Intel® SGX)
Intel® SST-BF	Intel® Speed Select Technology – Base Frequency (Intel® SST-BF)
Intel® SST-CP	Intel® Speed Select Technology – Core Power (Intel® SST-CP)
Intel® SST-PP	Intel® Speed Select Technology – Performance Profile (Intel® SST-PP)
Intel® SST-TF	Intel® Speed Select Technology – Turbo Frequency (Intel® SST-TF)
Intel® VT-d	Intel® Virtualization Technology (Intel® VT) for Directed I/O (Intel® VT-d)
Intel® VT-x	Intel® Virtualization Technology (Intel® VT) for IA-32, Intel® 64 and Intel® Architecture (Intel® VT-x)
IOMMU	Input/Output Memory Management Unit
IoT	Internet of Things
ISA	Instruction Set Architecture
I/O	Input/Output
K8s	Kubernetes
KMRA	Key Management Reference Application (KMRA)
KMS	Key Management Service (KMS)
LCC	Low Core Count
LLC	Last Level Cache
LOM	LAN on Motherboard
LPVSP	Local Persistent Volume Static Provisioner
MEC	Multi-Access Edge Compute
mTLS	Mutual Transport Layer Security
NFD	Node Feature Discovery
NFV	Network Function Virtualization
NIC	Network Interface Card (Network Adapter)
NTP	Network Time Protocol
NUMA	Non-Uniform Memory Access
NVM/NVMe	Non-Volatile Memory
OAM	Operation, Administration, and Management
OCI	Open Container Initiative
OS	Operating System
OVS	Open vSwitch
OVS DPDK	Open vSwitch with DPDK
PBF	Priority Based Frequency
PCCS	Provisioning Certification Caching Service
PCI	Physical Network Interface
PCIe	Peripheral Component Interconnect Express
PF	Port Forwarding
PMD	Poll Mode Driver

ABBREVIATION	DESCRIPTION
PMU	Power Management Unit
PXE	Preboot Execution Environment
QAT	Intel® QuickAssist Technology
QoS	Quality of Service
RA	Reference Architecture
RAS	Reliability, Availability, and Serviceability
RDT	Intel® Resource Director Technology
RHEL	Red Hat Enterprise Linux
S3	Amazon Web Services Simple Storage Service
S-IOV	Intel® Scalable I/O Virtualization (Intel® Scalable IOV)
SA	Service Assurance
SGX	Intel® Software Guard Extensions (Intel® SGX)
SR-IOV	Single Root Input/Output Virtualization
SSD	Solid State Drive
SSH	Secure Shell Protocol
SVM	Shared Virtual Memory
TADK	Traffic Analytics Development Kit
TAS	Telemetry Aware Scheduling
TCA	Trusted Certificate Attestation
TCS	Intel® Trusted Certificate Service
TCO	Total Cost of Ownership
TDP	Thermal Design Power
TLS	Transport Layer Security
TME	Total Memory Encryption
TMUL	Tile Multiply
UEFI	Unified Extensible Firmware Interface
UPF	User Plane Function
vBNG	Virtual Broadband Network Gateway
vCDN	Virtualized Content Delivery Network
vCMTS	Virtual Cable Modem Termination System
VF	Virtual Function
VMRA	Virtual Machine Reference Architecture
VPP	Vector Packet Processing
vRAN	Virtual Radio Access Network
VSS	Video Structuring Server
WAF	Web Application Firewall

Performance varies by use, configuration and other factors. Learn more at www.Intel.com/PerformanceIndex.

Performance results are based on testing as of dates shown in configurations and may not reflect all publicly available updates. See backup for configuration details. No product or component can be absolutely secure.

Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

Intel technologies may require enabled hardware, software or service activation.

Intel does not control or audit third-party data. You should consult other sources to evaluate accuracy.

The products described may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Code names are used by Intel to identify products, technologies, or services that are in development and not publicly available. These are not "commercial" names and not intended to function as trademarks.

0323/DN/WIT/PDF 721796-009US