SUSE

Technical Reference Documentation

Layered Stack Deployment of Rancher Kubernetes Engine

Integrated with Hewlett Packard Enterprise (R)

SUSE | Hewlett Packard Enterprise

Layered Stack Deployment of Rancher Kubernetes Engine: Integrated with Hewlett Packard Enterprise (R)

SUSE Linux Enterprise Server 15 SP3, Rancher Kubernetes Engine 1.2.16

The purpose of this document is to provide an overview and procedure of implementing SUSE (R) and partner offerings for Rancher Kubernetes Engine (RKE), a Kubernetes distribution that runs entirely within containers on bare-metal and virtualized nodes. RKE solves the problem of installation complexity and the operation is both simplified and easily automated, while entirely accommodating the operating system and platform it is running on.

Publication Date: 2022-04-06

https://documentation.suse.com

Contents

1 Introduction

On the digital transformation journey to a full cloud-native landscape, the use of microservices becomes the main approach with the dominant technology for such container orchestration being Kubernetes.[1] With its large community of developers and abundant features and capabilities, Kubernetes has become the de-facto standard and is included across most container-as-a-service platforms. With all of these technologies in place, both developer and operation teams can effectively deploy, manage and deliver functionality to their end users in a resilient and agile manner.

1.1 Motivation

Once on such a digital transformation journey, also relevant to focus on areas like:

Workload(s)

Determine how to manage and launch internally developed containerized, microservice workloads

Kubernetes

As developers and organizations continue their journey from simple, containerized microservices toward having these workloads orchestrated and deployed where ever they need, being able to install, monitor and use such Kubernetes infrastructures is a core need. Such deployments, being Cloud Native Computing Foundation (CNCF[2]) conformant and certified[3] are essential for both development and production workloads.

- Solving common frustrations around installation complexity, Rancher Kubernetes Engine reduces many host dependencies and provides a stable path for deployment, upgrades, and rollbacks for core use cases.

Compute Platform(s)

To optimize availability, performance, scalability and integrity, assess current system or hosting platforms

1 https://kubernetes.io/

2 https://www.cncf.io/

3 https://www.cncf.io/certification/software-conformance

from Independent Hardware Vendors (IHV), such as Hewlett Packard Enterprise (https://www.hpe.com/us/en/home.html) ® as the platform for physical, bare metal, hypervisors and virtual machines

1.2 Scope

The scope of this document is to provide a layered *reference configuration* for Rancher Kubernetes Engine. This can be done in a variety of scenarios to create an enterprise Kubernetes cluster deployment anywhere.

1.3 Audience

This document is intended for IT decision makers, architects, system administrators and technicians who are implementing a flexible, software-defined Kubernetes platform. One should still be familiar with the traditional IT infrastructure pillars — networking, computing and storage — along with the local use cases for sizing, scaling and limitations within each pillars' environments.

2 Business aspect

Agility is driving developers toward more cloud-native methodologies that focus on microservices architectures and streamlined workflows. Container technologies, like Kubernetes, embody this agile approach and help enable cloud-native transformation.

By unifying IT operations with Kubernetes, organizations realize key benefits like increased reliability, improved security and greater efficiencies with standardized automation. Therefore, Kubernetes infrastructure platforms are adopted by enterprises to deliver:

Cluster Operations

Improved Production and DevOps efficiencies with simplified cluster usage and robust operations

Security Policy & User Management

Consistent security policy enforcement plus advanced user management on any Kubernetes infrastructure

Access to Shared Tools & Services

A high level of reliability with easy, consistent access to a broad set of tools and services

2.1 Business problem

Many organizations are deploying Kubernetes clusters everywhere — in the cloud, on-premises, and at the edge — to unify IT operations. Such organizations can realize dramatic benefits, including:

- Consistently deliver a high level of reliability on any infrastructure
- Improve DevOps efficiency with standardized automation
- Ensure enforcement of security policies on any infrastructure

However, simply relying on upstream Kubernetes alone can introduce extra overhead and risk because Kubernetes clusters are typically deployed:

- Without central visibility
- Without consistent security policies
- And must be managed independently

Deploying a scalable kubernetes infrastructure requires consideration of a larger ecosystem, encompassing many software and infrastructure components and providers. Further, the ability to continually address the needs and concerns of:

Developers

For those who focus on writing code to build their apps securely using a preferred workflow, providing a simple, push-button deployment mechanism of their containerized workloads where needed.

IT Operators

General infrastructure requirements still rely upon traditional IT pillars are for the stacked, underlying infrastructure. Ease of deployment, availability, scalability, resiliency, performance, security and integrity are still core concerns to be addressed for administrative control and observability.

Beyond the core infrastructure software layers of managed Kubernetes clusters, organizations may be also be impacted by:

Compute Platform

Potential inconsistencies and impacts of multiple target system platforms for the distributed deployments of the cluster elements, across:

- physical, baremetal, hypervisors and virtual machines

2.2 Business value

With Rancher Kubernetes Engine, the operation of Kubernetes is easily automated and entirely independent of the operating system and platform running. Using a supported version of the container runtime engine, one can deploy and run Kubernetes with Rancher Kubernetes Engine. It builds a cluster from a single command in a few minutes, and its declarative configuration makes Kubernetes upgrades atomic and safe.

By allowing operation teams to focus on infrastructure and developers to deploy code the way they want too, SUSE and the Rancher offerings helps bring products to market faster and accelerate an organization's digital transformation.

SUSE Rancher is a fundamental part of the complete software stack for teams adopting containers. It provides DevOps teams with integrated tools for running containerized workloads while also addressing the operational and security challenges of managing multiple Kubernetes clusters across any targeted infrastructure.

Developers

SUSE Rancher makes it easy to securely deploy containerized applications no matter where the Kubernetes infrastructure runs -– in the cloud, on-premises, or at the edge. Using Helm or the App Catalog to deploy and manage applications across any or all these environments, ensuring multi-cluster consistency with a single deployment process.

IT Operators

SUSE Rancher not only deploys and manages production-grade Kubernetes clusters from datacenter to cloud to the edge, it also unites them with centralized authentication, access control and observability. Further, it streamlines cluster deployment on bare metal or virtual machines and maintains them using defined security policies.

With this increased consistency of the managed Kubernetes infrastructure clusters, organizations benefit from an even higher level of the Cloud Native Computing model where each layer only relies upon the API and version of the adjacent layer, such as:

Compute Platform

Using the above software application and technology solutions with the server platforms offered by Hewlett Packard Enterprise (HPE (https://www.hpe.com/us/en/home.html)) provides many alternative for scale, cost-effectiveness and performance options that could align with local IT staff platform preferences:

- density-optimized - high performance and efficiency for big data and the most demanding workloads
- mission-critical - systems of intelligence to fuel your digital transformation in a world where time and data are the new currency and business continuity is expected
- composable - fully adaptable and ready for Hybrid-IT to future-proof your data center for today’s workloads and tomorrow’s disruptors
- IoT - realize the potential of the Internet of Things to provide compute at the network edge

- cloud - high-capacity, mass-compute open infrastructure with security and software to match
- and virtualized use cases.

3 Architectural overview

This section outlines the core elements of the Rancher Kubernetes Engine solution, along with the suggested target platforms and components.

3.1 Solution architecture

The figure below illustrates the high-level architecture overview of Kubernetes components on instances like Rancher Kubernetes Engine:

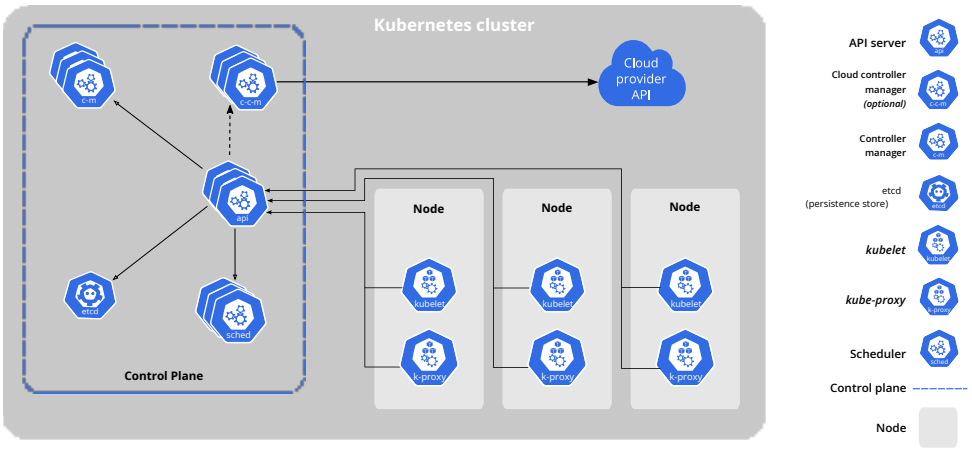

FIGURE 3.1: ARCHITECTURE OVERVIEW - RANCHER KUBERNETES ENGINE

A Kubernetes cluster consists of a set of nodes machines, called workers or agents, that host and run containerized applications in Pods. Every cluster has at least one worker node. The control plane manages the worker nodes and the Pods in the cluster. The provider API is a generic element that allows external interaction with the Kubernetes cluster.

Control Plane Components

The control plane's components make global decisions about the cluster (for example, scheduling), and detecting and responding to cluster events.

- kube-apiserver
  - The API server is a component of the Kubernetes control plane that exposes the Kubernetes API
- etcd

- Consistent and highly-available key value store used as Kubernetes' backing store for all cluster data.

- kube-scheduler
  - Control plane component that watches for newly created Pods with no assigned node, and selects a node for them to run on.

- kube-controller-manager
  - Control plane component that runs controller processes.

Node Components

Node components run on every node, maintaining running pods and providing the Kubernetes runtime environment.

- kubelet
  - An agent that runs on each node in the cluster. It makes sure that containers are running in a Pod.

- kube-proxy
  - A network proxy that runs on each node in your cluster, implementing part of the Kubernetes Service concept.

Note

Regardless of the deployment instance, Rancher Kubernetes Engine could always be deployed directly by SUSE Rancher or imported as a managed, downstream cluster.

4 Component model

This section describes the various components being used to create a Rancher Kubernetes Engine solution deployment, in the perspective of top to bottom ordering. When completed, the Rancher Kubernetes Engine instance can be used as the application infrastructure for cloud-native workloads and can be imported into SUSE Rancher for management.

4.1 Component overview

By using:

- Kubernetes Platform - Rancher Kubernetes Engine
- Operating System - SUSE Linux Enterprise Server
- Compute Platform
    - Hewlett Packard Enterprise ProLiant
    - Hewlett Packard Enterprise Synergy

you can create the necessary infrastructure and services. Further details for these components are described in the following sections.

4.2 Software - Rancher Kubernetes Engine

Rancher Kubernetes Engine is a CNCF-certified Kubernetes distribution that runs entirely within Docker containers. It solves the common frustration of installation complexity with Kubernetes by removing most host dependencies and presenting a stable path for deployment, upgrades, and rollbacks.

With Rancher Kubernetes Engine [RKE], the operation of Kubernetes is easily automated and entirely independent of the operating system and platform you are running. As long as you can run a supported version of Docker, you can deploy and run Kubernetes with RKE. It builds a cluster from a single command in a few minutes, and its declarative configuration makes Kubernetes upgrades atomic and safe.

What is provided with Rancher Kubernetes Engine

- CNCF Certification

    - Rancher Kubernetes Engine CNCF certification means that every release supports the same APIs as upstream Kubernetes. This gives enterprises the confidence that their Kubernetes resources are portable between RKE and other CNCF-certified Kubernetes distributions.

- Simplified installation

    - Installation is via a single binary and it uses a single YAML file, meaning that even non-experts can deploy Kubernetes with a single command. The command connects to remote hosts via SSH, so Rancher or any staff member with SSH access can deploy and manage RKE instances anywhere in the world.

- Automated Operation

    - When used with SUSE Rancher, operators can perform automated installation and upgrades of RKE clusters with a few clicks.

- Vendor Independence

    - RKE is not locked into a specific vendor operating system, Kubernetes Management Platform or proprietary tooling.

- Safe, Atomic Upgrades

    - Since RKE is built using containers, it does not have any touch points with the underlying operating system beyond the container engine. Containers make it easy to upgrade to a new version and to roll back to the previous version if necessary.

- 24x7 Enterprise-level Support

- Ensures around-the-clock support from technical experts when you need it.

The fundamental roles for the nodes and core functionality of Rancher Kubernetes Engine are represented in the following figure:

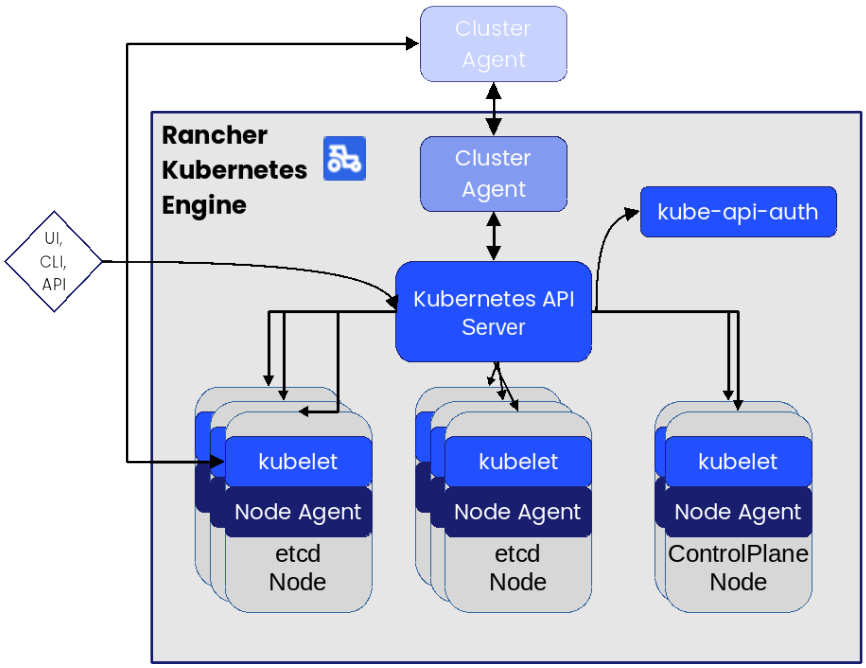

FIGURE 4.1: COMPONENT OVERVIEW - RANCHER KUBERNETES ENGINE

- Kubernetes API Server,
  - interacts with kubelet on all the nodes, plus addresses authentication, user interface (UI), command line interface (CLI) and API for external access and cluster management via SUSE Rancher cluster controller to agent

While all Rancher Kubernetes Engine roles can be installed on a single system, for the best availability, performance and security, the recommended deployment of a Rancher Kubernetes Engine cluster is a pair of nodes for the control plane role, at least three etcd role-based nodes and three or more worker nodes.

Rancher Kubernetes Engine can run as a complete cluster on a single node or can be expanded into a multi-node cluster. Besides the core Kubernetes components, these are also configurable and included:

- Multiple Kubernetes versions
- CoreDNS, Metrics, Ingress controller
- CNI : Canal, Calico, Flannel, Weave
- Support for a Windows worker agent node (only with Flannel)
- Fleet Agent : for GitOps deployment of cloud-native applications

All of these components are configurable and can be swapped out for your implementation of choice. With these included components, you get a fully functional and CNCF-conformant cluster so you can start running apps right away.

Tip

Learn more information about Rancher Kubernetes Engine at https://rancher.com/docs/rke/latest/en/.

While all Rancher Kubernetes Engine roles can be installed on a single system, a multi-node cluster, is a more production-like approach and will be described in the deployment section.

Tip

To improve availability, performance and security, the recommended deployment of a Rancher Kubernetes Engine cluster is a pair of nodes for the control plane role, at least three etcd role-based nodes and three or more worker nodes.

4.3 Software - SUSE Linux Enterprise Server

SUSE Linux Enterprise Server (SLES (https://www.suse.com/products/server/)) is an adaptable and easy-to-manage platform that allows developers and administrators to deploy business-critical workloads on-premises, in the cloud and at the edge. It is a Linux operating system that is

adaptable to any environment – optimized for performance, security and reliability. As a multimodal operating system that paves the way for IT transformation in the software-defined era, this simplifies multimodal IT, makes traditional IT infrastructure efficient and provides an engaging platform for developers. As a result, one can easily deploy and transition business-critical workloads across on-premises and public cloud environments.

Designed for interoperability, SUSE Linux Enterprise Server integrates into classical Unix and Windows environments, supports open standard interfaces for systems management, and has been certified for IPv6 compatibility. This modular, general purpose operating system runs on four processor architectures and is available with optional extensions that provide advanced capabilities for tasks such as real time computing and high availability clustering. SUSE Linux Enterprise Server is optimized to run as a high performing guest on leading hypervisors and supports an unlimited number of virtual machines per physical system with a single subscription. This makes it the perfect guest operating system for virtual computing.

4.4 Compute Platform

Leveraging the enterprise grade functionality of the operating system mentioned in the previous section, many compute platforms can be the foundation of the deployment:

- Virtual machines on supported hypervisors or hosted on cloud service providers
- Physical, baremetal or single-board computers, either on-premises or hosted by cloud service providers

Note

To complete self-testing of hardware with SUSE YES Certified Process (https://www.suse.com/partners/ihv/yes/yes-certified-process), you can download and install the respective SUSE operating system support-pack version of SUSE Linux Enterprise Server and the YES test suite. Then run the tests per the instructions in the test kit, fixing any problems encountered and when corrected, re-run all tests to obtain clean test results. Submit the test results into the SUSE Bulletin System (SBS) for audit, review and validation.

Tip

Certified systems and hypervisors can be verified via SUSE YES Certified Bulletins (https://www.suse.com/yessearch/) and then can be leveraged as supported nodes for this deployment, as long as the certification refers to the respective version of the underlying SUSE operating system required.

Even with the broad certification and support model across the range of available HPE platform models, the following table summarizes which might be a best-practice selection for the various deployment types and focus areas:

TABLE 4.1: HEWLETT PACKARD ENTERPRISE PLATFORM MATRIX FOR DEPLOYMENT TYPES

System Platform	Baremetal	Hypervisor	Virtual Machine
ProLiant	DL360,DL380	DL360,DL380	(hosting)
Synergy	SY480	SY480	(hosting)

As listed in the previous table, multiple server product-line and model options abound in the HPE server portfolio (https://www.hpe.com/uk/en/servers.html), as detailed in the following sections.

4.4.1 Hewlett Packard Enterprise iLO

The Hewlett Packard Enterprise iLO (https://www.hpe.com/us/en/servers/integrated-lights-out-ilo.html) [iLO] arms you with the tools to manage your servers efficiently, resolve issues quickly, and keep your business running – from anywhere in the world, allowing you to manage your entire server environment with ease. Upgrade the basic iLO license for additional functionality, such as graphical remote console, multi-user collaboration, video record/playback, remote management, and much more. The latest iLO innovations include:

- Security and performance
- Support for Simple Certificate Enrollment Protocol [SCEP]
- Enablement for 802.1x protocol to securely onboard servers into a network
- Redfish API Conformance

4.4.2 HPE ProLiant DL Rack Servers

The HPE ProLiant DL (https://www.hpe.com/us/en/servers/proliant-dl-servers.html) family of servers are the most flexible, reliable, and performance-optimized HPE ProLiant rack servers—ever. HPE continues to provide industry-leading compute innovations. The new HPE ProLiant rack portfolio, with flexible choices and versatile design, along with improved energy efficiencies, ultimately lowers your TCO. Integrated with a simplified, but comprehensive management suite and industry-leading support, the HPE ProLiant rack portfolio delivers a more reliable, fast, and secure infrastructure solution, helps increase IT staff productivity, and accelerates service delivery. In addition, the rack portfolio is performance-optimized for multiapplication workloads to significantly increase the speed of IT operations and enable IT to respond to business needs of any size, faster.

Specific models that offer relevant choices for Enterprise Kubernetes are:

HPE ProLiant DL380 (https://www.hpe.com/us/en/product-catalog/servers/proliant-servers/pip.hpe-proliant-dl380-gen10-server.1010026818.html)

The industry-leading HPE DL380 2P/2U server with world-class performance and supreme versatility for multi-workload compute server delivers the latest in security, performance and expandability, backed by a comprehensive warranty. Standardize on the industry's most trusted compute platform. The HPE DL380 server is securely designed to reduce costs and complexity, featuring:

- the First, Second, Third Generation Intel Xeon Processor Scalable Family with up to a 60% performance gain1 and 27% increase in cores2
- the HPE 2933 MT/s DDR4 SmartMemory supporting 3.0 TB
- support of 12 Gb/s SAS, and up to 20 NVMe drive plus a broad range of compute options
- HPE Persistent Memory offers unprecedented levels of performance for databases and analytic workloads to run everything from the most basic to mission-critical applications and deploy with confidence.

HPE ProLiant DL360 (https://www.hpe.com/us/en/product-catalog/servers/proliant-servers/pip.hpe-proliant-dl360-gen10-server.1010007891.html)

Adaptable for diverse workloads and environments, the compact 1U HPE DL360 server delivers security, agility and flexibility without compromise. It supports:

- the Intel Xeon Scalable processor with up to a 60% performance gain1 and 27% increase in cores2
- along with 2933 MT/s HPE DDR4 SmartMemory supporting up to 3.0 TB2 with an increase in performance of up to 82%3
- the added performance that HPE Persistent Memory6, HPE NVDIMMs7 and 10 NVMe bring, the HPE DL360 means business. Deploy, update, monitor and maintain with ease
- automating essential server life cycle management tasks with HPE OneView and HPE Integrated Lights Out to deploy this 2P secure platform for diverse workloads in space constrained environments.

Note

HPE Servers Support & OS Certification Matrices (https://techlibrary.hpe.com/us/en/enterprise/servers/supportmatrix/suse_linux.aspx) outlines the minimum version of SLE required for installation, yet later service pack releases may also be used and supported.

4.4.3 HPE Synergy Servers

HPE Synergy (https://www.hpe.com/us/en/integrated-systems/synergy.html), the first Composable Infrastructure, empowers IT to create and deliver new value easily and continuously. This single infrastructure reduces operational complexity for traditional workloads and increases operational velocity for the new breed of applications and services. Through a single interface, HPE Synergy composes compute, storage and fabric pools into any configuration for any application.

It also enables a broad range of workloads — from bare metal, to virtual machines, to containers, to operational models like hybrid cloud and DevOps. HPE Synergy enables IT to rapidly react to new business demands with the following components:

- HPE Synergy 12000 Frames are uniquely architected as Composable Infrastructure (CI) to match the powerful 'infrastructure-as-code' capabilities of the HPE intelligent software architecture. Flexible access to compute, storage, and fabric resources allows for use and re-purposing. Linking multiple HPE Synergy Frames efficiently scales the infrastructure with a dedicated single view of the entire management network.
    - Creating multiple composable domains in the infrastructure can efficiently deliver available resources to the business. HPE Synergy Frames reduce complexity by using intelligent auto-discovery to find all available resources to accelerate workload deployments. This drives IT efficiency as the business grows and delivers balanced performance across resources to increase solution effectiveness.
- With HPE Synergy SY480 (https://www.hpe.com/us/en/integrated-systems/synergy.html) Compute Module, one gains operational efficiency and control, and can deploy IT resources quickly for any workload through a single interface. HPE Synergy is a powerful software-defined solution. HPE Synergy Composable Compute resources create pools of flexible compute capacity that can be configured almost instantly to rapidly provision infrastructure for a broad range of applications. The HPE Synergy SY480 Compute Module delivers an efficient and flexible two-socket workhorse to support most demanding workloads. Powered by:
    - Intel Xeon Scalable Family of processors
    - up to 4.5 TB DDR4, more storage capacity and controllers
    - a variety of GPU options within a composable architecture HPE Synergy SY480 Compute Module is the ideal platform for general-purpose enterprise workload performance now and in the future.

Note

HPE Servers Support & OS Certification Matrices (https://techlibrary.hpe.com/us/en/enterprise/servers/supportmatrix/suse_linux.aspx) outlines the minimum version of SLE required for installation, yet later service pack releases may also be used and supported.

Note

A sample bill of materials, in the *Chapter 9, Appendix*, cites the necessary quantities of all components, along with a reference to the minimum resource requirements needed by the software components.

5 Deployment

This section describes the process steps for the deployment of the Rancher Kubernetes Engine solution. It describes the process steps to deploy each of the component layers starting as a base functional *proof-of-concept*, having considerations on migration toward *production*, providing *scaling* guidance that is needed to create the solution.

5.1 Deployment overview

The deployment stack is represented in the following figure:

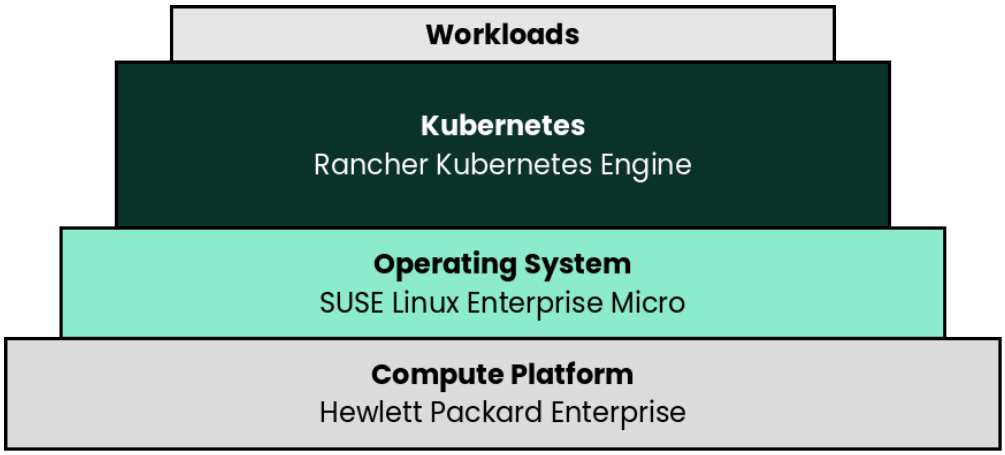

FIGURE 5.1: DEPLOYMENT STACK - RANCHER KUBERNETES ENGINE

and details are covered for each layer in the following sections.

Note

The following section's content is ordered and described from the bottom layer up to the top.

5.2 Compute Platform

The base, starting configuration can reside all within a single Hewlett Packard Enterprise Synergy Frame. Based upon the relatively small resource requirements for a Rancher Kubernetes Engine deployment, a viable approach is to deploy as a virtual machine (VM) on the target nodes, on top of an existing hypervisor, like KVM. For a physical host, there are tools that can be used during the setup of the server, see below.

Preparation(s)

The HPE Integrated Lights Out (https://www.hpe.com/us/en/servers/integrated-lights-out-ilo.html) [iLO] is designed for secure local and remote server management and helps IT administrators deploy, update and monitor HPE servers anywhere, anytime.

1. Upgrade your basic iLO license for additional functionality, such as graphical remote console and virtual media access to allow the remote usage of software image files (ISO files), which can be used for installing operating systems or updating servers.

    - (Optional) - iLO Federation (https://www.hpe.com/us/en/servers/integrated-lights-out-ilo.html) enables you to manage multiple servers from one system using the iLO web interface.

2. For nodes situated in an HPE Synergy enclosure, like HPE Synergy SY480 used in the deployment:

    - Setup the necessary items in the Hewlett Packard Enterprise OneView (https://www.hpe.com/us/en/integrated-systems/software.html) interface, including:

        - Settings → Addresses and Identifiers (Subnets and Address Ranges)
        - Networks → Create (associate subnets and designate bandwidths)
        - Network Sets → Create (aggregate all the necessary Networks)
        - Logical Interconnects → Edit (include the respective Network Sets)
        - Logical Interconnect Groups → Edit (include the respective Network Sets)
        - Server Profile Templates → Create (or use existing hypervisor templates)
        - OS Deployment mode → could be configured to boot from PXE, local storage, shared storage

- Firmware (upgrade to the latest and strive for consistency across node types)
- Manage Connections (assign the Network Set to be bonded across NICs)
- Local Storage (create the internal RAID1 set and request additional drives for the respective roles)
- Manage Boot/BIOS/iLO Settings
- Server Profile → Create (assign the role template to the target model)

- Add Servers and Assign Server Roles
  - Use the Discover function from Hewlett Packard Enterprise OneView to see all of the available nodes that can be assigned to to their respective roles:
  - Then drag and drop the nodes into the roles and ensure there is no missing configuration information, by reviewing and editing each node's server details
  - Manage Settings - setup DNS/NTP, designate Disk Models/NIC Mappings/Interface Model/Networks
  - Manage Subnet and Netmask - edit Management Network information, ensuring a match exists to those setup in Hewlett Packard Enterprise OneView

Deployment Process

On the respective compute module node, determine if a hypervisor is already available for the solution's virtual machines.

1. If this will be the first use of this node, an option is to deploy a KVM hypervisor, based upon SUSE Linux Enterprise Server by following the Virtualization Guide (https://documentation.suse.com/sles/15-SP3/single-html/SLES-virtualization/#book-virt).

- Given the simplicity of the deployment, the operating system and hypervisor can be installed with the SUSE Linux Enterprise Server ISO media and the Hewlett Packard Enterprise Integrated Lights Out virtual media and virtual console methodology.

2. Then for the solution VM, use the hypervisor user interface to allocate the necessary CPU, memory, disk and networking as noted in the link:SUSE Rancher hardware requirements (https://documentation.suse.com/cloudnative/rancher-manager/latest/en/installation-and-upgrade/requirements/requirements.html).

Deployment Consideration(s)

To further optimize deployment factors, leverage the following practices:

- *Automation*
  - For HPE Synergy servers, you can simplify multiple compute module setups and configurations, leveraging the Hewlett Packard Enterprise OneView SDK for Terraform Provider (https://github.com/HewlettPackard/terraform-provider-oneview).
  - For nodes running KVM, you can leverage either virt-install (https://documentation.suse.com/sles/15-SP3/single-html/SLES-virtualization/#book-virt) or Terraform Libvirt Provider (https://registry.terraform.io/providers/dmacvicar/libvirt/latest/docs) to quickly and efficiently automate the deployment of multiple virtual machines.
- *Availability*
  - While the initial deployment only requires a single VM, as noted in later deployment sections, having multiple VMs provides resiliency to accomplish high availability. To reduce single points of failure, it would be beneficial to have the multi-VM deployments spread across multiple hypervisor nodes. So consid-

eration of consistent hypervisor and compute module configurations, with the needed resources for the VMs will yield a robust, reliable production implementation.

5.3 SUSE Linux Enterprise Server

As the base software layer, use an enterprise-grade Linux operating system. For example, SUSE Linux Enterprise Server.

Preparation(s)

To meet the solution stack prerequisites and requirements, SUSE operating system offerings, like SUSE Linux Enterprise Server (https://www.suse.com/products/server/) can be used.

1. Ensure these services are in place and configured for this node to use:

    - Domain Name Service (DNS) - an external network-accessible service to map IP Addresses to host names
    - Network Time Protocol (NTP) - an external network-accessible service to obtain and synchronize system times to aid in time stamp consistency
    - Software Update Service - access to a network-based repository for software update packages. This can be accessed directly from each node via registration to
        - the general, internet-based SUSE Customer Center (https://scc.suse.com) (SCC) or
        - an organization's SUSE Manager (https://www.suse.com/products/suse-manager/) infrastructure or
        - a local server running an instance of Repository Mirroring Tool (https://documentation.suse.com/sles/15-SP3/single-html/SLES-rmt/#book-rmt) (RMT)

Note

During the node's installation, it can be pointed to the respective update service. This can also be accomplished post-installation with the command line tool named SUSEConnect (https://www.suse.com/support/kb/doc/?id=000018564).

Deployment Process

On the compute platform node, install the noted SUSE operating system, by following these steps:

1. Download the SUSE Linux Enterprise Server (https://www.suse.com/download/sles/) product (either for the ISO or Virtual Machine image)

    - Identify the appropriate, supported version of SUSE Linux Enterprise Server by reviewing the support matrix for SUSE Rancher (https://www.suse.com/suse-rancher/support-matrix/all-supported-versions/) versions Web page.

2. The installation process is described and can be performed with default values by following steps from the product documentation, see Installation Quick Start (https://documentation.suse.com/sles/15-SP3/single-html/SLES-installation/#article-installation)

Tip

Adjust both the password and the local network addressing setup to comply with local environment guidelines and requirements.

Deployment Consideration(s)

To further optimize deployment factors, leverage the following practices:

- *Automation*

    - To reduce user intervention, unattended deployments of SUSE Linux Enterprise Server can be automated

- for ISO-based installations, by referring to the AutoYaST Guide (https://documentation.suse.com/sles/15-SP3/single-html/SLES-autoyast/#book-autoyast)

5.4 Rancher Kubernetes Engine

Preparation(s)

1. Identify the appropriate, desired version of the Rancher Kubernetes Engine binary (for example vX.Y.Z) that includes the needed Kubernetes version by reviewing

   - the "Supported Rancher Kubernetes Engine Versions" associated with the respective SUSE Rancher (https://www.suse.com/suse-rancher/support-matrix/all-supported-versions/) version from "Rancher Kubernetes Engine Downstream Clusters" section, or

   - the "Releases" on the Download (https://github.com/rancher/rke/) Web page.

2. On the target node with a default installation of SUSE Linux Enterprise Server operating system, log in to the node either as root or as a user with sudo privileges and enable the required container runtime engine

   ```
   sudo SUSEConnect -p sle-module-containers/15.3/x86_64
   sudo zypper refresh ; zypper install docker
   sudo systemctl enable --now docker.service
   ```

   - Then validate the container runtime engine is working

     ```
     sudo systemctl status docker.service
     sudo docker ps --all
     ```

3. For the underlying operating system firewall service, either

   - enable and configure the necessary inbound ports (https://rancher.com/docs/rke/latest/en/os/#ports) or

   - stop and completely disable the firewall service.

Deployment Process

The primary steps for deploying this Rancher Kubernetes Engine Kubernetes are:

Note

Installing Rancher Kubernetes Engine requires a client system (i.e. admin workstation) that has been configured with kubectl.

1. Download the Rancher Kubernetes Engine binary according to the instructions on product documentation (https://rancher.com/docs/rke/latest/en/) page, then follow the directions on that page, but with the following exceptions:
2. Create the cluster.yml file with the command `rke config`

Note

See product documentation for example-yamls (https://rancher.com/docs/rke/latest/en/example-yamls/) and config-options (https://rancher.com/docs/rke/latest/en/config-options/) for detailed examples and descriptions of the cluster.yml parameters.

- It is recommended to create a unique SSH key for this Rancher Kubernetes Engine cluster with the command `ssh-keygen`
  - Provide the path to that key for the option "Cluster Level SSH Private Key Path"
- The option "Number of Hosts" refers to the number of hosts to configure at this time
  - Additional hosts can be added very easily after Rancher Kubernetes Engine cluster creation
  - For this implementation it is recommended to configure one or three hosts
- Give all hosts the roles of "Control Plane", "Worker", and "etcd"
- Answer "n" for the option "Enable PodSecurityPolicy"

3. Update the cluster.yml file before continuing with the step "Deploying Kubernetes with RKE"

4. If a load balancer has been deployed for the Rancher Kubernetes Engine control-plane nodes, update the cluster.yml file before deploying Rancher Kubernetes Engine to include the IP address or FQDN of the load balancer. The appropriate location is under authentication.sans. For example:

```
LB_IP_Host=""
```

```
authentication:
  strategy: x509
  sans: ["${LB_IP_Host}"]
```

5. Verify password-less SSH is available from the admin workstation to each of the cluster hosts as the user specified in the cluster.yml file

6. When ready, run `rke up` to create the RKE cluster

7. After the `rke up` command completes, the RKE cluster will continue the Kubernetes installation process

    - Monitor the progress of the installation:

        - Export the variable KUBECONFIG to the absolute path name of the kube_config_cluster.yml file. I.e. `export KUBECONFIG=~/rke-cluster/kube_config_cluster.yml`

        - Run the command: `watch -c "kubectl get deployments -A"`

            - The cluster deployment is complete when elements of all the deployments show at least "1" as "AVAILABLE"

            - Use Ctrl+c to exit the watch loop after all deployment pods are running

Tip

To address *Availability* and possible *scaling* to a multiple node cluster, etcd is enabled instead of using the default SQLite datastore.

Deployment Consideration(s)

To further optimize deployment factors, leverage the following practices:

- *Availability*
  - A full high-availability Rancher Kubernetes Engine cluster is recommended for production workloads. For this use case, two additional hosts should be added; for a total of three. All three hosts will perform the roles of control-plane, etcd, and worker.
    1. Deploy the same operating system on the new compute platform nodes, and prepare them in the same way as the first node
    2. Update the cluster.yml file to include the additional node
       - Using a text editor, copy the information for the first node (found under the "nodes:" section)
         - The node information usually starts with "- address:" and ends with the start of another node entry, or the beginning of the "services: " section, i.e.

```
- address: 172.16.240.71
  port: "22"
  internal_address: ""
  role:
  - controlplane
  - worker
  - etcd

. . .

  labels: {}
```

```
  taints: []
```

- Paste the information into the same section, once for each additional host
- Update the pasted information, as appropriate, for each additional host

3. When the cluster.yml file is updated with the information specific to each node, run the command `rke up`
   - Run the command: `watch -c "kubectl get deployments -A"`
     - The cluster deployment is complete when elements of all the deployments show at least "1" as "AVAILABLE"
     - Use Ctrl + c to exit the watch loop after all deployment pods are running

After this successful deployment of the Rancher Kubernetes Engine solution, review the product documentation (https://rancher.com/docs/rke/latest/en/) for details on how to directly use this Kubernetes cluster. Furthermore, by reviewing the SUSE Rancher product documentation (https://documentation.suse.com/cloudnative/rancher-manager/) this solution can also be:

- imported (refer to subsection "Importing Existing Clusters"), then
- managed (refer to subsection "Cluster Administration") and
- accessed (refer to subsection "Cluster Access") to address orchestration of workloads, maintaining security and many more functions are readily available.

6 Summary

Using components and offerings from SUSE (https://www.suse.com) and the Rancher portfolio plus Hewlett Packard Enterprise ProLiant Rack Servers (https://www.hpe.com/us/en/servers/proliant-dl-servers.html) plus Hewlett Packard Enterprise Synergy Servers (https://www.hpe.com/us/en/integrated-systems/synergy.html) streamline the ability to quickly and effectively engage in a digital transformation, taking advantage of cloud-native resources and disciplines. Using such technology approaches lets you deploy and leverage transformations of infrastructure into a durable, reliable enterprise-grade environment.

Simplify

Simplify and optimize your existing IT environments

- Using Rancher Kubernetes Engine enables you to simplify, maintain and scale Kubernetes cluster deployments in a supportable fashion.

Modernize

Bring applications and data into modern computing

- With Rancher Kubernetes Engine, the digital transformation to containerized applications can benefit from the provided, production-quality application infrastructures for each of the respective user bases and to facilitate the actual workload deployments and resilient usage.

Accelerate

Accelerate business transformation through the power of open source software

- Given the open source nature of Rancher Kubernetes Engine and the underlying software components, you can simplify deployment with automation, maintain secure production instance and make significant IT savings as you scale orchestrated microservice deployments anywhere you need to and for whatever use cases are needed, in an agile and innovative way.

7 References

WHITE PAPERS

- A Buyer's Guide to Enterprise Kubernetes Management Platforms - https://more.suse.com/FY22_Buyers_Guide_to_Enterprise_Container_Management_Buyers-Guide-to-Kubernetes-Management-Platforms.html
- How to Build an Enterprise Kubernetes Strategy - https://more.suse.com/FY22-global-web-How-to-Build-Enterprise-K8s-Strategy.html

BOOKS

- Kubernetes Management - https://more.suse.com/rs/937-DCH-261/images/002022021-DummiesGuide.pdf

TRAINING

- SUSE - https://training.suse.com/
  - Rancher - https://rancher.com/training/

WEB SITES

- SUSE - https://www.suse.com
  - SUSE Customer Center (SCC) - https://scc.suse.com
  - Products
    - SUSE Rancher - https://www.suse.com/products/rancher/ (documentation (https://documentation.suse.com/cloudnative/rancher-manager/))
    - Rancher Kubernetes Engine (RKE) - https://rancher.com/products/rke/ (documentation (https://rancher.com/docs/rke/latest/en/))
    - K3s - https://www.suse.com/products/k3s/ (documentation (https://documentation.suse.com/cloudnative/k3s/))
    - SUSE Linux Enterprise Micro (SLEMicro) - https://www.suse.com/products/micro/ (documentation (https://documentation.suse.com/sle-micro/5.5/))
    - SUSE Linux Enterprise Server (SLES) - https://www.suse.com/products/server/ (documentation (https://documentation.suse.com/sles/15-SP3/))

        - SUSE Manager - https://www.suse.com/products/suse-manager/ (documentation (https://documentation.suse.com/suma/4.3/))
        - SUSE Repository Mirroring Tool (RMT) - https://www.suse.com/products/server/ (documentation (https://documentation.suse.com/sles/15-SP3/single-html/SLES-rmt/#book-rmt))
    - Projects
        - Rancher Kubernetes Engine Government (RKE2) - https://github.com/rancher/rke2 (documentation (https://docs.rke2.io/))
- Hewlett Packard Enterprise - https://www.hpe.com/us/en/home.html
    - ProLiant - https://www.hpe.com/us/en/servers/proliant-dl-servers.html
    - Synergy - https://www.hpe.com/us/en/integrated-systems/synergy.html

8 Glossary

- Document Scope

  Reference Configuration

  A guide with the basic steps to deploy the layered stack of components from both the SUSE and partner portfolios. This is considered a fundamental basis to demonstrate a specific, tested configuration of components.

  Reference Architectures[1]

  A guide with the general steps to deploy and validate the structured solution components from both the SUSE and partner portfolios. This provides a shareable template of consistency for consumers to leverage for similar production ready solutions, including design considerations, implementation suggestions and best practices.

  Best Practice

  Information that can overlap both the SUSE and partner space. It can either be provided as a stand-alone guide that provides reliable technical information not covered in other product documentation, based on real-life installation and implementation experiences from subject matter experts or complementary, embedded sections within any of the above documentation types describing considerations and possible steps forward.

- Factor(s)

  Automation[2]

  Infrastructure automation enables speed through faster execution when configuring the infrastructure and aims at providing visibility to help other teams across the enterprise work quickly and more efficiently. Automation removes the risk associated with human error, like manual misconfiguration; removing this can decrease downtime and increase reliability. These outcomes and attributes help the enterprise move toward implementing a culture of DevOps, the combined working of development and operations.

---

1 link: Reference Architecture (https://en.wikipedia.org/wiki/Reference_architecture)

2 link: Infrastructure-as-Code (https://en.wikipedia.org/wiki/Infrastructure_as_code)

Availability[3]

The probability that an item operates satisfactorily, without failure or downtime, under stated conditions as a function of its reliability, redundancy and maintainability attributes. Some major objectives to achieve a desired service level objectives are:

- Preventing or reducing the likelihood and frequency of failures via design decisions within the allowed cost of ownership
- Correcting or coping with possible component failures via resiliency, automated failover and disaster-recovery processes
- Estimating and analyzing current conditions to prevent unexpected failures via predictive maintenance

Integrity[4]

Integrity is the maintenance of, and the insurance of the accuracy and consistency of a specific element over its entire lifecycle. Both physical and logical aspects must be managed to ensure stability, performance, re-usability and maintainability.

Security[5]

Security is about ensuring freedom from or resilience against potential harm, including protection from destructive or hostile forces. To minimize risks, one mus manage governance to avoid tampering, maintain access controls to prevent unauthorized usage and integrate layers of defense, reporting and recovery tactics.

- Deployment Flavor(s)

Proof-of-Concept[6]

A partial or nearly complete prototype constructed to demonstrate functionality and feasibility for verifying specific aspects or concepts under consideration. This is often a starting point when evaluating a new, transitional technology. Sometimes it starts as a Minimum Viable Product (MVP[7]) that has just enough features to satisfy an

3 link: Availability (https://en.wikipedia.org/wiki/Availability)

4 link: Data Integrity (https://en.wikipedia.org/wiki/Data_integrity)

5 link: Security (https://en.wikipedia.org/wiki/Security)

6 link: Proof of Concept (https://en.wikipedia.org/wiki/Proof_of_concept)

7 link: Minimum Viable Product (https://en.wikipedia.org/wiki/Minimum_viable_product)

initial set of requests. After such insights and feedback are obtained and potentially addressed, redeployments may be used to iteratively branch into other realms or to incorporate other known working functionality.

Production

A deployed environment that target customers or users can interact with and rely upon to meet their needs, plus be operationally sustainable in terms of resource usage and economic constraints.

Scaling

The flexibility of a system environment to either vertically scale-up, horizontally scale-out or conversely scale-down by adding or subtracting resources as needed. Attributes like capacity and performance are often the primary requirements to address, while still maintaining functional consistency and reliability.

9 Appendix

The following sections provide a bill of materials listing for the respective component layer(s) of the described deployment.

9.1 Compute platform bill of materials

Sample set of computing platform models, components and resources.

Role	Qty	SKU	Component	Notes
Example 1	1-3	867959-B21 ABA	Hewlett Packard Enterprise ProLiant DL360 Gen10 8SFF CTO server	• items below listed per node
	2	P02592-L21	• Intel Xeon-Gold 5218 (2.3GHz/16-core/125W) Processor Kit	
	12	P00918-B21	• Single Rank x8 DDR4-2933 CAS-21-21-21 Registered Smart Memory Ki	
	2	P18434-B21	• 960GB SATA 6G Mixed Use SFF (2.5in) SC 3yr Wty Multi Vendor SSD	
	1	P01366-B21	• 96W Smart Storage Lithium-ion Battery with 145mm Cable Kit	
	1	804326-B21	• Smart Array E208i-a SR Gen10 (8 Internal Lanes/	

Role	Qty	SKU	Component	Notes
			No Cache) 12G SAS Modular Controller	
	1	879482-B21	• InfiniBand FDR/ Ethernet 40/50Gb 2-port 547FLR-QSFP Adapter	
	1	BD505A	• iLO Advanced 1-server License with 3yr Support on iLO Licensed Features	
Example 2	1-3	868703-B21 ABA	Hewlett Packard Enterprise ProLiant DL380 Gen10 8SFF CTO server	• items below listed per node
	2	P02510-L21	• Intel Xeon-Gold 6242 (2.8GHz/16-core/150W) FIO Processor Ki	
	12	P00922-B21	• 16GB (1x16GB) Dual Rank x8 DDR4-2933	

Role	Qty	SKU	Component	Notes
			CAS-21-21-21 Registered Smart Memory Kit	
	2	P18434-B21	• 960GB SATA 6G Mixed Use SFF (2.5in) SC 3yr Wty Multi Vendor SSD	
	1	P01366-B21	• 96W Smart Storage Lithium-ion Battery with 145mm Cable Kit	
	1	804326-B21	• Smart Array E208i-a SR Gen10 (8 Internal Lanes/	

Role	Qty	SKU	Component	Notes
			No Cache) 12G SAS Modular Controller	
	1	879482-B21	• InfiniBand FDR/ Ethernet 40/50Gb 2-port 547FLR-QSFP Adapter	
	1	BD505A	• iLO Advanced 1-server License with 3yr Support on iLO Licensed Features	
Example 3 Chassis	1	797740-B21	Enclosure : Hewlett Packard Enterprise Synergy 12000 Configurer-to-order Frame with 1x Frame Linke Module, 10x Fans	• items below listed per enclosure
	1	804938-B21	• Frame Rack Rail Kit	
	1	804942-B21	• Frame Link Module	
	2	804353-B21	• Composer	
	2	779218-B21	• Network : 20Gb Interconnect Link Module	
	2	794502-B23	• Hewlett Packard Enterprise Virtual Connect SE 40Gb F8 Module for Synergy	
	2	755985-B21	• Storage : 12G SAS Connectivity Module for Synergy	
Example 3 Node	1-3	871940-B21	• Compute Module : Hewlett Packard Enterprise Synergy SY480 Gen10	• items below listed per node

Role	Qty	SKU	Component	Notes
			CAS-19-19-19 Registered Smart Memory Kit	
	1	804424-B21	• Smart Array P204i-c SR Gen10 (4 Internal Lanes/1GB Cache) 12G SAS Modular Controller	
	2	875478-B21	• 1.92TB SATA 6G Mixed Use SFF (2.5in) SC 3yr WTY Digitally Signed Firmware SSD	

9.2 Software bill of materials

Sample set of software, support and services.

Role	Qty	SKU	Component	Notes
Operating System	1-3	874-006875	SUSE Linux Enterprise Server,	Configuration: • per node (up to 2 sock-

Role	Qty	SKU	Component	Notes
			• x86_64, • Priority Subscription, • 1 Year	ets, stackable) or 2 VMs
Kubernetes Management	1	R-0001-PS1	SUSE Rancher, • x86-64, • Priority Subscription, • 1 Year	Configuration: • per deployed instance
Rancher Management	2	R-0004-PS1	Rancher 10 Nodes • x86-64 or aarch64, • Priority Subscription, • 1 Year,	Configuration: • requires priority server subscription
Consulting and Training	1	R-0001-QSO	Rancher Quick Start, • Go Live Services	

Note

For the software components, other support term durations are also available.

9.3 Documentation configuration / attributes

This document was built using the following AsciiDoc (https://github.com/asciidoc/asciidoc) and DocBook Authoring and Publishing Suite (DAPS (https://github.com/openSUSE/daps)) attributes:

```
Appendix=1 ArchOv=1 Automation=1 Availability=1 BP=1 BPBV=1 CompMod=1 DepConsiderations=1
 Deployment=1 FCTR=1 FLVR=1 GFDL=1 Glossary=1 HWComp=1 HWDepCfg=1 IHV-HPE-ProLiant=1
 IHV-HPE-Synergy=1 IHV-HPE=1 Integrity=1 LN=1 PoC=1 Production=1 RA=1 RC=1 References=1
 Requirements=1 SWComp=1 SWDepCfg=1 Scaling=1 Security=1 docdate=2022-04-06 env-daps=1
 focusRKE1=1 iIHV=1 iK3s=1 iRKE1=1 iRKE2=1 iRMT=1 iRancher=1 iSLEMicro=1 iSLES=1 iSUMa=1
 layerSLES=1
```

10 Legal Notice

11 GNU Free Documentation License

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.

C. State on the Title page the name of the publisher of the Modified Version, as the publisher.

D. Preserve all the copyright notices of the Document.

E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

H. Include an unaltered copy of this License.

I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.

N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.

O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all

Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

ADDENDUM: How to use this License for your documents

```
Copyright (c) YEAR YOUR NAME.
   Permission is granted to copy, distribute and/or modify this document
   under the terms of the GNU Free Documentation License, Version 1.2
```

```
   or any later version published by the Free Software Foundation;
   with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.
   A copy of the license is included in the section entitled “GNU
   Free Documentation License”.
```

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the “ with…Texts.” line with this:

```
with the Invariant Sections being LIST THEIR TITLES, with the
   Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.
```

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.