This white paper is intended to help an organization create and deploy an on-premise container-as-a-service platform instance to deploy, orchestrate and manage containerized workloads.

Written by:
Bryan Gartner
SUSE
1.1 Description
As with any software-defined infrastructure, all the classic IT disciplines around networking, computing, and storage are involved and need to be effectively planned and integrated. This document provides a reference implementation from proof-of-concept to production ready deployments with design considerations, implementation suggestions, and best practices. The combination of Dell EMC® Network S-Series Network Switches and PowerEdge R640 Rack Servers computing hardware plus the SUSE® CaaS Platform software solution yields a fully functional, container-as-a-service infrastructure.

- To ease the adoption curve and provide multiple proof points, the infrastructure can be quickly deployed as a proof-of-concept utilizing only virtual machines for evaluation of the solution.
- To address increases in workload capacity, physical systems with more resources can be iteratively added to the existing cluster and even replace some of the initial roles on virtual machines.
- To accommodate the increased adoption of the solution, cluster-wide functions can be migrated to a full, production instance to provide all the required functionality and services for administrators, developers and users to operate at scale.

Each of these modes is detailed, citing considerations and requirements necessary to move seamlessly through the modes. For the final production mode, one has an operationally complete solution that can be monitored and maintained over time. In addition, core functions are deploying in a highly available fashion to help prevent downtime and multiple user interfaces are made available. To complement the solution, an integration with a Ceph-based storage backend is provided the persistent data storage needed to allow stateful components for the containerized workloads.

The deployment process for each of these modes was implemented, tested and validated in the Dell EMC Partner lab.

1.2 Target Audience
The target audience for this document is IT professionals responsible for setting up, configuring, administering and operating a container-ready infrastructure. It is suggested that this document is reviewed in its entirety, along with the referenced supplemental documentation before attempting the deployment.

2.1 Business Problem

2.2 Business Value

In the rapidly evolving era of DevOps, containerized workloads are becoming increasingly popular. Both developers and users need an API-centric platform to continuously integrate with, rapidly deploy to, and utilize applications that can scale as required. Operational teams need the ability to set up and manage such a software-defined infrastructure quickly and easily, yet with the ability to maintain it over time. By providing an easy to deploy and scale container-as-a-service platform, organizations can begin migrating their business applications from legacy monolithic applications to micro-service cloud-native solutions in a low-risk fashion.

2.1 Business Problem

Providing an environment to support containerized workload applications requires a robust, software-defined infrastructure including networking, computing platforms, and persistent storage. Since it is a combination of all the classic disciplines, IT administrators appreciate a solution that is synergistically more than a sum of the individual components and is easy to configure, maintain, secure and adjust over time to react to changing needs.
2.2 Business Value

Each of these three components, networking, compute and software provide individual value-add to address the overall requirements of the solution:

**Dell EMC Network Switches.** Dell EMC data center switching solutions are cost-effective and easy to deploy at any scale, from 1G to multi-rate 100G for optimum connectivity both within the rack or blade chassis and across the data center. The switching solutions also feature a choice of innovative Dell EMC and third-party software options to address virtually any enterprise or service provider use-case or environment. For the physical switching layer, Dell EMC S-Series top-of-rack (ToR) open networking switches like the 10GbE S4048-ON are cost-effective and easy to deploy at any scale for optimum connectivity both within the rack or chassis and across the data center fabric. The switches also feature a choice of innovative Dell EMC or third-party software options. To complete the networking solution, a Dell EMC Networking 1GbE S3048-ON open networking switch is used to handle management functions.

**Dell EMC PowerEdge Servers.** The Dell EMC PowerEdge R640 and R740xd deliver a perfect balance between storage scalability and performance. Both two-socket platforms are ideal for software defined storage (like ScaleIO, VSAN or the PowerEdge XC), service providers or Big Data servers. The platform versatility is highlighted with the ability to mix any drive type to create the optimum configuration of NVMe, SSD and HDD for either performance, capacity or both. The R740xd drives fast response times for business-critical workloads with up to 24 NVMe drives for storage intensive roles with optimum performance. Both platforms support the Dell EMC OpenManage™ portfolio, delivering intelligent, automated management of routine tasks. Combined with unique agent-free management capabilities, the R640/R740xd are simply managed, freeing up time for high profile projects. The R640/R740xd leverage new security features built-into every new PowerEdge server strengthening protection so you can reliably and securely deliver accurate data to your customers no matter where they are. By considering each aspect of system security, from design to retirement, Dell EMC ensures trust and delivers a worry-free, secure infrastructure without compromise.

**Software Defined Infrastructure.** SUSE CaaS Platform is an enterprise-class container management solution that enables IT and DevOps professionals to more easily deploy, manage, and scale container-based applications and services. It includes Kubernetes to automate lifecycle management of modern applications and surrounding technologies that enrich Kubernetes and make the platform itself easy to operate. As a result, enterprises that use SUSE CaaS Platform can reduce application delivery cycle times and improve business agility.
SUSE is focused on delivering an exceptional operator experience with SUSE CaaS Platform. With deep competencies in infrastructure, systems, process integration, platform security, lifecycle management, and enterprise-grade support, SUSE aims to ensure IT operations teams can deliver the power of Kubernetes to their users quickly, securely and efficiently.

Figure 2.1: SUSE CaaS Platform
Container-as-a-Service Platforms require reliability, manageability, and serviceability. These requirements span the multiple layers of such a solution, from the container host operating system, the container runtime engine, and the container orchestration system. Such demands are inherited from previous generations of IT infrastructure expectations and carry forward, even though the containerized workloads themselves have vastly different approaches through agility and resiliency.

With SUSE CaaS Platform you can:

- Achieve faster time to value with an enterprise-ready container management platform, built from industry-leading technologies, and delivered as a complete package, with everything you need to quickly offer container services.
- Simplify management and control of your container platform with efficient installation, easy scaling, and update automation.
Maximize return on your investment, with a flexible container services solution for today and tomorrow.

Figure 3.1: SUSE CaaS Platform Features

4.1 Solution architecture

4.2 Networking architecture

As noted in Section 1.1, “Description” section, this document provides the deployment steps to create a container-as-a-service instance, starting off in a proof-of-concept mode and transitioning through to a full, production mode setup.

Underlying each of these deployment modes, however, is a core set of functionality and architectural components:

- **Container-as-a-Service Platform**

- **Host Operating System.** Typically a small footprint operating system installation, having just enough functionality to support the container runtime engine, leaving as many CPU, memory and I/O resources available for the containerized workloads.
  - SUSE currently delivers this as MicroOS, a read-mostly, minimal operating system based upon SUSE Linux Enterprise Server. This is complemented by a distributed key-value store provided by etcd to retain persistent configuration data. In addition, MicroOS provides a snapshot-driven, transactional-update methodology to perform atomic upgrades.

- **Container Runtime Engine(s).** Comprised of both a format for and service to run containerized applications on top of the host operating system.
  - SUSE provides support for Docker® Community Edition Engine, the current, defacto standard open source format for application containers.
  - SUSE also offers a technical preview of CRI-O, an implementation of Container Runtime Interface (CRI), designed specifically for Kubernetes as a lightweight alternative, using Open Container Initiative (OCI) images.
**Container Networking.** An intra-cluster service and overlay network used for container and orchestration communication.

- SUSE currently utilizes the Container Network Interface (CNI) with the Flannel plugin and a configuration management web-interface to setup and deploy these networks. More details follow in Section 4.2, “Networking architecture” section.

**Container Orchestration.** A service to manage deployments of containerized workload, known as Kubernetes, the current, defacto standard open source implementation for container orchestration.

- SUSE currently delivers and supports a Cloud-Native Computing Foundation (CNCF) certified Kubernetes distribution. Included with this is a role-based access control technology to, as desired, limit access to resources, functions, and services.
- Miscellaneous Infrastructure Components and Services

**Core Infrastructure Components / Services**

- Domain Name Service (DNS) – an external network-accessible service to map IP Addresses to hostnames
- Network Time Protocol (NTP) – an external network-accessible service to obtain and synchronize system times to aid in timestamp 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 SUSE Customer Center (http://scc.suse.com) or from local servers running a SUSE Subscription Management Tool (https://www.suse.com/documentation/sles-12/singlehtml/book_smt/book_smt.htm) (SMT) instance. As each node is deployed, it can be pointed to the respective update service and update notification and applicate will be managed by the configuration management web interface.
- Client System - one or more existing system, with your choice of operating system, used to access the cluster and various services provided from a command line, via kubectl and helm, and web browser.

**4.1 Solution Architecture**

In addition to these high-level architectural components, SUSE CaaS Platform provides and relies upon the following types of nodes / roles:

**Note:** Refer to the "Architectural Overview" section of SUSE CaaS Platform Deployment Guide (https://www.suse.com/documentation/suse-caasp-3/) for more details.

- **Administration Node.** Provides a cluster infrastructure management system, with each service run as containers on this host and providing configuration management plus a web-based dashboard to manage other node types within the cluster.
- **Master Node(s).** Oversees Kubernetes container workload orchestration services across the cluster, and manages the Kubernetes Worker Nodes
- **Worker Node(s).** Where the user-defined containerized workloads and services run in Kubernetes pods

**4.2 Networking Architecture**

The following networking requirements must be in place for a successful deployment:

**Note:** Refer to the "Networking Requirements" section of SUSE CaaS Platform Deployment Guide (https://www.suse.com/documentation/suse-caasp-3/) for more details and port specifics.

**Cluster network**

- Choose a subnet range that will span the total number of cluster nodes. This range can also be segmented or secured for access to specific node roles as desired.
- All of the cluster node types must be able to communicate on the same network, with this primary network interface card. A client system with similar network access is also required for command-line and web browser interaction with the cluster, especially during setup.
Higher speed network interface cards (minimum of 10GigE and above) and switching are preferred since the number of containerized workloads can be high and they share this infrastructure capacity, both from an external and intra-cluster perspective.

**Internal networks**

Known as the Overlay and Service networks, these are used by Kubernetes and the underlying Flannel network plug-in to manage the internal cluster and container connections. These are implemented with bridges to the main cluster network.

**Important:** These internal network ranges should be planned prior to deployment, are usually non-routable network ranges and cannot be changed without redeploying the entire cluster.

**Network services**

Ensure that an external DNS server is accessible, and configured for each cluster node to resolve all node names, uniquely. At least the Administration Node and Kubernetes API Master must resolve in a Fully Qualified Domain Name (FQDN) fashion for external clients to connect to these respective cluster nodes.

Ensure the Administration Node is pointed to a reliable external NTP service and the remaining nodes will, by default, point to the Administration Node.

Ensure all cluster nodes have access to a software update repository to facilitate upgrades over time.

### 5.1 Network Deployment configuration

### 5.2 HW Deployment configuration

### 5.3 SW Deployment configuration

### 5.4 Solution Admin Host

### 5.5 HAProxy

### 5.6 Proof-of-Concept Mode (PoC)

### 5.7 Virtual to Physical System Migration Mode (V2P)

### 5.8 Production Instance Mode

### 5.9 Additional Deployment Considerations

This section is meant as a companion guide to the official network, system, and software product deployment documentation, citing specific settings as needed for this reference implementation. Default settings are assumed to be in use unless otherwise cited to accomplish the respective best practices and design decisions herein.

### 5.1 Network Deployment Configuration

The following considerations for the network switching configuration should be attended to:

- Configure 802.3ad for system port bonding, if used, and for VLT between the top-of-rack switches, if possible, to get the maximum performance of bonded network interfaces
- Connect the Dell EMC PowerEdge server’s iDRAC to a management network, which can be co-reside with the cluster network or entirely distinct, depending upon your local administration policies

**Important:** Ensure that all similar switching devices are consistent and up-to-date with regard to firmware versions to reduce potential troubleshooting issues later.
Tip: Meticulous care of the network wiring from the various resource nodes and switches makes troubleshooting much easier. Where possible, also label connections and stick to consistent patterns of port/placement of connections.

Figure 5.1: Logical View of Deployment Network
The following considerations for various network service configurations should be attended to:

- Setup external DNS A records for all nodes. Decide on subnet ranges and configure the switch ports accordingly to match those nodes in use.
- Ensure that you have access to a valid, reliable external NTP service, as this is a critical requirement for all nodes.

For this reference implementation, the following IP / Hostname settings were utilized and configured in the accessible external DNS service:

- Network IP addressing and IP ranges need proper planning to address current as well as future growth.

Table 5.1: Network Address Configuration #

<table>
<thead>
<tr>
<th>Function</th>
<th>Role</th>
<th>Mode</th>
<th>Hostname</th>
<th>IP Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>core</td>
<td>Solution Admin Host (SAH)</td>
<td>PoC, V2P, Production</td>
<td>sah.suse-dell.net</td>
<td>10.204.92.86</td>
</tr>
<tr>
<td></td>
<td>K8s Master LB</td>
<td>PoC, V2P, Production</td>
<td>mstr-lb.suse-dell.net</td>
<td>10.204.92.245</td>
</tr>
</tbody>
</table>
5.2 HW Deployment configuration #

The following considerations for the system platforms should be attended to:

**Note:** Any SUSE YES (https://www.suse.com/yessearch/) certified Dell EMC platform, like the PowerEdge R640, can be used for the physical nodes of this deployment, as long as the certification refers to the version of the underlying SUSE operating system used by SUSE CaaS Platform.

- Reset the BIOS setup configuration to the default setting to have a known baseline configuration to provide consistency.
- If possible, setup RAID1 mirroring on the storage controller across a pair of drives for the operating system installation.

**Important:** Ensure that all similar system devices are consistent and up-to-date with regard to BIOS/uEFI/device firmware versions to reduce potential troubleshooting issues later.

5.3 SW Deployment Configuration

- From the SUSE Downloads (https://download.suse.com) site, obtain the SUSE CaaS Platform install media (DVD1) and utilize either trial or purchased subscriptions for the cluster nodes to ensure access to support and software updates.
- From the same download site, for the Solution Admin Host, obtain the SUSE Linux Enterprise Server 12-SP3 (DVD1) operating system install media.

5.4 Solution Admin Host #

- Solution Admin Host (SAH) Because of the need for various administrative-like services, a convenient approach is to create a Solution Admin Host (SAH) that consolidates these services. Given a finite number of physical systems, this
consolidation helps to preserve other system nodes for more resource-intensive use by deploying virtual machine guests for various administrative functions.

**Tip:** A simple hypervisor host, using KVM, provides the platform for the SAH and enables further grouping of administrative functions here as virtual machines.

Using an available system, perform a bare-metal installation of the SUSE Linux Enterprise Server 12-SP3 operating system with either physical media or virtual media through iDRAC

**Note:** The default partitioning scheme can be used, but remember to store any virtual machine images into the larger home directory partition or create a distinct partition for /var/lib/libvirt. For more details, refer to SUSE Virtualization Guide (https://www.suse.com/documentation/sles-12)

- A minimal system can be installed, with at least the following patterns include: base, minimal, kvm_server, kvm_tools
- Register the system to the SUSE Customer Center (SCC) or a local SMT server during or after the installation to ensure all the latest software updates are present.
- After the installation completes, use YaST to:
  - Configure the desired networking including:
    - An external network interface for access beyond the cluster environment (using one of the 1GigE NICs, e.g., em3)
    - A bond, mode 802.3ad if available to match the switch configuration, across all 10GigE NICs being used (e.g., em1, em2)
    - A bridge for virtualization on top of the previously bonded network interfaces, configured with an IP address in the cluster network
  - For convenience, install an Administrative VNC server to remotely access this system from other systems, which provides a graphical user interface

### 5.5 HAPerxy

Utilizing HAPerxy for load balancing is an approach to make the cluster, and more specifically some of the core Kubernetes Master Node API functions, accessible to client systems. It does this via a virtual IP which then sends the call to any active masters. While not required for a single master cluster, setting this up in advance allows later expansion and substitutions to happen.

This process can be run on any host or virtual machine with access to the Admin network. The steps to deploy this service are:

- In this implementation, HAPerxy was run as a service on the Solution Admin Host, by adding the respective K8s Master LB Virtual IP as another address on the virtualization bridge, via:

  root@sah # yast2 network

- Install the HAPerxy package, which can be found in the SUSE Linux Enterprise High Availability Extension (https://www.suse.com/products/highavailability/) via:

  root@sah # zypper in haproxy

- Modify the HAPerxy configuration file /etc/haproxy/haproxy.conf to include the following stanzas, to account for both the Kubernetes API and DEX functionality and then save the file.

  ```
  # Kubernetes API server
  listen mstrlb
  bind 10.204.92.245:6443
  mode tcp
  option tcplog
  ```
balance roundrobin
server k8s-master-0 10.204.92.246:6443 check
server mstr-a 10.204.92.50:6443 check
server mstr-b 10.204.92.60:6443 check
server mstr-c 10.204.92.70:6443 check

# DEX (OIDC Connect)
listen kubeconfiglb
  bind 10.204.92.245:32000
  mode tcp
  option tcplog
  balance roundrobin
  server k8s-master-0 10.204.92.246:32000 check
  server mstr-a 10.204.92.50:32000 check
  server mstr-b 10.204.92.60:32000 check
  server mstr-c 10.204.92.70:32000 check

Note: You will notice that all Kubernetes Master Nodes are included in the example, which allows it to be used throughout
the mode transitions. This is because the configuration also does a check on the state of the node/port combination
before forwarding on such a request.

Tip: You should also adjust the "stats" stanza to utilize another, available port, e.g. 12345 as in the following example
configuration, to allow any services setup later that need to access port 80.

# haproxy stats
listen stats
  bind 0.0.0.0:12345
  mode http
  stats enable
  stats hide-version
  stats realm HAProxy\ Statistics
  stats uri /haproxy_stats
  stats refresh 8s

  Then enable and start the HAProxy service, via:

root@sah # systemctl enable haproxy
root@sah # systemctl start haproxy
5.6 Proof-of-Concept Mode (PoC)

The goal of this mode, as shown in the following figure, is to create a preliminary container-as-a-service infrastructure utilizing virtual machines for use in a proof-of-concept mode. Often this is used to evaluate the infrastructure and get familiar with the deployment and to launch containers against.

![Proof-of-Concept Deployment](image)

Figure 5.2: Proof-of-Concept Deployment

**Note:** The installation process used, across all modes and all nodes, whether virtual or physical, were done from ISO images just for consistency in this document. Other options are available as noted in the SUSE CaaS Platform Deployment Guide ([https://www.suse.com/documentation/suse-caasp-3/](https://www.suse.com/documentation/suse-caasp-3/)). Also included in this Deployment Guide are minimum node requirements for (v)CPU, Memory and Storage RAM to be used to setup the virtual machines or even later for the physical nodes being deployed.

**Administration Node.** Install the SUSE CaaS Platform Administration Node as a virtual machine on the SAH

- Using virt-manager (GUI) or virsh (CLI) on the SAH, create a virtual machine that meets or exceeds the minimum requirements for this node’s role as noted in the deployment document
- Allocate a virtual NIC for the cluster network, tied to the virtualization bridge residing on the cluster network
- Configure the following virtual CD drive
  - SUSE CaaS Platform ISO image (bootable)
- Follow the "Installing the Administration Node" process steps described in the SUSE CaaS Platform Deployment Guide ([https://www.suse.com/documentation/suse-caasp-3/](https://www.suse.com/documentation/suse-caasp-3/))
• When the installation is complete and the system reboots, use the client system to access the Velum Dashboard web-interface at the FQDN of the Administration Node. Continue the setup described in the "Configuring the Administration Node" section of the SUSE CaaS Platform Deployment Guide (https://www.suse.com/documentation/suse-caasp-3/). Ensure the following items are addressed:
  • On the home page, "Create Admin" account with a valid email address and password
  • Once logged in:
    • Check the "Install Tiler (Helm's server component)" box in "Cluster Services" as this will be used extensively later.
    • Ensure the Overlay and Service network settings match the desired values, if the default values are not satisfactory.
    • Select the desired container runtime. For this deployment, the Docker open-source engine was used.
  • On the "Bootstrap your CaaS Platform" page:
    • Note the location of the AutoYast file, in case you'd like to automate other node installations
At this point, you are ready to install the remaining cluster nodes.

Kubernetes Master (1) and Kubernetes Worker Nodes (2)

Install the three remaining nodes of a minimal cluster. For this PoC implementation, these nodes can co-reside as virtual machines on the SAH host or another network accessible virtualization host with access to the cluster network.
• Using virt-manager (GUI) or virsh (CLI), create a virtual machine that meets or exceeds the minimum requirements for this node's role from the deployment document
• Allocate a virtual NIC for the cluster network, tied to the virtualization bridge
• Configure the following virtual CD drives
  • SUSE CaaS Platform ISO image (bootable)
  • Complete the installation steps as described in the "Installing Master and Worker Nodes" section of the SUSE CaaS Platform Deployment Guide (https://www.suse.com/documentation/suse-caasp-3/)

Bootstrap the Cluster. When the nodes have completed their installation and rebooted, use the client system again to login and access the Velum Dashboard web-interface at the FQDN of the Administration Node to continue the cluster formation.
• There should be three items listed in the "Pending Nodes" section, so "Accept All Nodes"
• Designate the "Master" and "Worker" to the respective nodes, then "Next"
• Enter the K8s Master LB FQDN setting "mstr-lb.suse-dell.net" for the "External Kubernetes API FQDN"
• Enter the FQDN of the Administration Node, "caasp-admin.suse-dell.net" in "External Dashboard FQDN", then "Bootstrap Cluster"

Once this process completes, you should have a fully functional SUSE CaaS Platform cluster to use for your Proof-of-Concept needs. You can validate this:
• By logging into the Administration Node and running:

root@caasp-admin# kubectl cluster-info
root@caasp-admin# kubectl get nodes
root@caasp-admin# kubectl get pods -n kube-system

• By logging into the client system:
  • Using a web browser, login to the Velum Dashboard web-interface with the admin credentials at the FQDN of the Administration Node
  • Download the kubeconfig file, and put a copy in the default location of ~/.kube/config
  • Ensure the client system has kubectl installed, then run the same set of kubectl commands from the previous section

Tip: If using a SUSE Linux Enterprise 12 or newer release host as the client, both the kubectl and helm commands can be found in SUSE Package Hub (https://packagehub.suse.com/)
Review the following information to:

- Understand the administration aspects of the cluster by reviewing SUSE CaaS Platform Administration Guide (https://www.suse.com/documentation/suse-caasp-3/)
- Become familiar with the usage of kubectl by reviewing Overview of kubectl (https://kubernetes.io/docs/reference/kubectl/overview/)

5.7 Virtual to Physical System Migration Mode (V2P)

The goal of this mode, as shown in the following figure, is to increase the number of Kubernetes Worker Nodes virtual machines with physical systems for increased resource access.

Figure 5.3: Virtual to Physical Deployment

**Preparation.** As container usage increases, which may be the rationale behind the virtual to physical migration of Kubernetes Worker Nodes, it can be instructive to sample the utilization of your cluster and it’s resources.

- Log into the client system’s command line, follow the “Deploying Helm and Tiller” section of the SUSE CaaS Platform Administration Guide (https://www.suse.com/documentation/suse-caasp-3/).
- When completed, survey the resources being used across your cluster’s nodes and for each deployed pod, via:
  - to see CPU and memory usage for each of the systems
    
    ```
    tux@client > kubectl top nodes
    ```
  - to see CPU, memory and I/O usage for each of the pods running containers. You can also append either "-n <namespace>" or "--all-namespaces" to set a more specific set or every pods' resource usage, respectively.
    
    ```
    tux@client > kubectl top pods
    ```
• In addition, you can also view a graphical representation of resource utilization, via the cAdvisor utility, for any of the Kubernetes Worker nodes by pointing a client's web browser at "<FQDNorIPAddressOfWorker>" and port "4194". A sample screenshot is shown below:

![Figure 5.4: Worker Node Resource Utilization via cAdvisor](image)

Installing Additional Kubernetes Worker Nodes

• In an available system, use the SUSE CaaS Platform ISO image as a physical boot media or via the iDRAC virtual media function
• Ensure the suggested storage configuration of a pair of RAID1 mirrored drives for the operating system is used to protect against device failures.
• Repeat the installation steps as described in the "Installing Master and Worker Nodes" section of the SUSE CaaS Platform Deployment Guide (https://www.suse.com/documentation/suse-caasp-3/)

Bootstrap the New Nodes

• As each gets installed and rebooted, there should a corresponding new item listed in the "Pending Nodes" section, then "Accept Node"
• Designate the respective node as a "Worker", then "Next"
Once incorporated into the cluster, you can validate the node's presence by running:

```
root@caasp-admin # kubectl get nodes
```

If desired, you can later "Remove" the existing, virtual-machine-based worker nodes from this same web-based interface. This will efficiently delete, in a non-recoverable way, the node from the cluster in a controlled fashion. Essentially it cordons off the node from further workload scheduling and drains the node of existing workloads.

### 5.8 Production Instance Mode

The goal of this mode, as shown in the following figure, is to upgrade the cluster to a multi-master state, to eliminate that particular single point of failure.

![Production Instance Deployment](image)

**Figure 5.5: Production Instance Deployment**

In addition, some further capabilities are added to increase user-level functionality:

- Add a web-based Kubernetes dashboard, which is itself containerized, to enable ease of use for those deploying containers beyond the kubectl command line interface
- While many containerized workloads are truly stateless, there are some microservices that do need persistent storage options, so a Ceph-based backend, like SUSE Enterprise Storage (https://www.suse.com/products/suse-enterprise-storage), can be integrated to satisfy that need

For administrators of the infrastructure, to help address the increasing needs for higher availability, to validate, manage and monitor the cluster can also be enhanced, specifically:
- Run sample Kubernetes conformance tests to ensure the expected upstream functionality is present even with cluster changes over time
- Install performance metrics gathering and visualization toolsets to assess resource utilization and aid in troubleshooting

**Tip:** Many curated Helm charts (https://github.com/helm/charts) are available for deploying various containerized applications.

**Kubernetes Dashboard**

- Log into the client system’s command line or to the Administration Node, referencing the cluster’s admin kubeconfig, then
- Follow the Helm chart instructions for "kubernetes-dashboard (https://github.com/helm/charts/tree/master/stable/kubernetes-dashboard)" to deploy this functionality. The resulting output of the respective helm install command also provides guidance on how to make this port/service publicly visible to users. Now users can launch and manage their containers from this web interface.

![Kubernetes Dashboard](image)

**Figure 5.6: Kubernetes Dashboard #**

**Ceph-based Persistent Storage**

A companion document, refer to Chapter 7, Appendices, outlines using similar Dell EMC network switches, Dell EMC PowerEdge Servers, and SUSE Enterprise Storage, powered by Ceph. This creates a highly scalable and resilient software-based storage solution, enabling organizations to build cost-efficient storage using industry standard servers and
disk drives. It is self-managing and delivers storage functionality comparable to mid- and high-end storage products at a fraction of the cost.

To create the necessary integration between a Ceph-based storage cluster, on SUSE Enterprise Storage (SES), first start on the SUSE Enterprise Storage side. Login to the respective Admin Node and

- Collect the list of Monitor Node IP Addresses from /etc/ceph/ceph.conf
- Create and validate a dedicated storage pool (e.g. "caasp-pool") for rbd-based applications. Adjust the placement group size (e.g. "512") as desired.

root@ses-admin # ceph osd pool create caasp-pool 512
root@ses-admin # ceph osd pool ls
root@ses-admin # ceph osd pool application enable caasp-pool rbd
root@ses-admin # ceph osd pool application get caasp-pool
- Capture and encode the admin key, which will be used later

root@ses-admin # ceph auth get-key client.admin | base64
- Create a specific user (e.g. "caasp") with necessary capabilities to utilize the "caasp-pool", then capture and encode user the key, which will be used later

root@ses-admin # ceph auth get-or-create client.caasp mon 'allow r' osd 'allow class-read object_prefix rbd_children, allow rwx pool=caasp-pool' -o ceph.client.caasp.keyring
root@ses-admin # ceph auth get-key client.caasp | base64

Then from the SUSE CaaS Platform side, login to the respective Administration Node

- Create the file ceph-secret-admin.yaml

# file - ceph-secret-admin.yaml

apiVersion: v1
kind: Secret
metadata:
  name: ceph-secret-admin
  namespace: default
type: "kubernetes.io/rbd"
data:
  key: #insert output string from client.admin encode step
- Apply and validate deployment of the file ceph-secret-admin.yaml

root@caasp-admin # kubectl apply -f ceph-secret-admin.yaml
root@caasp-admin # kubectl get secrets
- Create the file ceph-secret-caasp.yaml

# file - ceph-secret-caasp.yaml

apiVersion: v1
kind: Secret
metadata:
  name: ceph-secret-caasp
  namespace: default
type: "kubernetes.io/rbd"
data:
  key: #insert output string from client.caasp encode step

- Apply and validate file ceph-secret-caasp.yaml
  
  root@caasp-admin # kubectl apply -f ceph-secret-caasp.yaml
  
  root@caasp-admin # kubectl get secrets

- Create the file ses-rbd-storage-class.yaml, using the Monitor Node IP addresses collected earlier
  
  # file - ses-rbd-storage-class.yaml
  
  kind: StorageClass
  
  apiVersion: storage.k8s.io/v1
  
  metadata:
    name: ses-rbd-sc
  
  provisioner: kubernetes.io/rbd
  
  parameters:
    monitors: <IPAddressMon1>:6789, <IPAddressMon2>:6789, <IPAddressMon3>:6789
    adminId: admin
    adminSecretName: ceph-secret-admin
    adminSecretNamespace: default
    pool: caasp-pool
    userId: caasp
    userSecretName: ceph-secret-caasp

- Apply and validate file ses-rbd-storage-class.yaml
  
  root@caasp-admin # kubectl apply -f ses-rbd-storage-class.yaml
  
  root@caasp-admin # kubectl get sc <-n default>

- Create the storage claim file ses-rbd-persistent-volume-claim.yaml
  
  # file - ses-rbd-storage-class.yaml
  
  kind: PersistentVolumeClaim
  
  apiVersion: v1
  
  metadata:
    name: ses-rbd-pvc
  
  spec:
    storageClassName: ses-rbd-sc
    accessModes:
      - ReadOnlyMany
    resources:
      requests:
        storage: 1Gi # adjust size as needed

- Apply and validate persistent volume claim file ses-rbd-persistent-volume-claim.yaml
  
  root@caasp-admin # kubectl apply -f ses-rbd-persistent-volume-claim.yaml
  
  root@caasp-admin # kubectl get pvc <-n default>
Then to validate a working integration, create a simple container using the persistent volume claim. Refer to the "Creating Pods with Persistent Volumes" section of the SUSE CaaS Platform Administration Guide (https://www.suse.com/documentation/suse-caasp-3/).

**Kubernetes Conformance Tests**

A representative set of the upstream Kubernetes conformance test suite can be seen at Heptio / Sonobuoy (https://github.com/heptio/sonobuoy). This can be easily run via a browser from the client system that has access to the admin kubeconfig file as noted in the "Getting Started" section of this site. An example run can be seen in the following screenshot:

![Sonobuoy Scan Results](image)

**Figure 5.7: Kubernetes validation with Sonobuoy**

**Resource Metrics Gathering / Visualization**

- Log into the client system's command line, using the cluster's admin kubeconfig, then refer to SUSE CaaSP Platform FAQ (https://wiki.microfocus.com/index.php?title=SUSE_CaaS_Platform/FAQ) and follow the steps in the "Monitoring Stack based on Prometheus and Grafana" section.

**Tip:** The Ceph-based RBD storage and persistent volume claims should be utilized as the backing store for the monitoring and visualization containers.
5.9 Additional Deployment Considerations

Beyond the three distinct operational modes described in this document, some very convenient and technologically advanced features are included or can be extended:

- Fine-grained, role-based access control, relying upon a local source can be easily augmented for users with specific roles. In addition, federating to external, like LDAP/AD, authentication/authorization sources can also be accomplished. Refer to the "Managing Users and Groups" and "Role Management" sections in the SUSE CaaS Platform Administration Guide (https://www.suse.com/documentation/suse-caasp-3/) to add other users. Then these configured users can login to the web interface, download their respective kubeconfig files and launch containers into their created or designated namespaces with their roles and access to the specified resources.

- By combining the resiliency of containerized workloads and the orchestration provided by Kubernetes, SUSE CaaS Platform can be continually updated and upgraded. Using the underlying technology of Btrfs filesystem snapshots and the transactional-update tooling, component and operating systems updates are seamlessly applied across all cluster nodes. More details can be found in the "Software Management" section of the SUSE CaaS Platform Administration Guide (https://www.suse.com/documentation/suse-caasp-3/).

- An additional service to consider is a repository site for container images to pull workloads from. This can be a publicly-accessible site or can be a private collection of workload images. Other registry sites, either public or private, can be used to provide files, like Helm Charts, to deploy complete services. An option, to provide more fine-grained user authorization and access to the container images is SUSE Portus (http://port.us.org/).

- Various types of logs are available:
  - The setup and configuration of the cluster
  - The operation of the cluster, including the containerized services and updates
  - The orchestration aspect of Kubernetes
  - More details, including locations to access and how to collect logs for external log servers, are included in the "Logging" section of the SUSE CaaS Platform Administration Guide (https://www.suse.com/documentation/suse-caasp-3/).

- Increasing the node count of the cluster is another consideration. Details and recommended configuration changes can be found in the "Scaling the Cluster" section of the SUSE CaaS Platform Administration Guide (https://www.suse.com/documentation/suse-caasp-3/)

- The density of containerized workloads is another topic to consider. As shown in some previous sections around performance monitoring, one can collect the associated metrics of a given workload, kubectl top pods and the nodes, kubectl top nodes to determine how many such workloads can be accommodated. Another approach is, during the launch of a manifest or helm chart, specify the resource requirements needed. Using this method, Kubernetes will honor that during the scheduling of the workloads to ensure, at launch, that the appropriate resources are present on the target node. It is also important to ensure that both networking and I/O of storage resources are taken into consideration.

- Other factors like certificate management, security, graceful shutdown, and startup of the cluster, and troubleshooting are covered in the SUSE CaaS Platform Administration Guide (https://www.suse.com/documentation/suse-caasp-3/)

Combining the features of Dell EMC Network Switches, PowerEdge R640 server with the software from SUSE CaaS Platform yields a robust, powerful and flexible container-as-a-service infrastructure. No matter what stage of transition your organization may be with regard to containerized workloads, this deployment allows industry standard compatibility coupled with industry-leading support and operational ease. Any business can feel confident in the ability to address the continual growth in container development and usage they are currently faced with.

7.1 Appendix: Bill of Materials

7.1 Appendix: Bill of Materials

Table 7.1: Bill of Materials - Network #
<table>
<thead>
<tr>
<th>Role</th>
<th>Quantity</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Top of Rack Network Switch</td>
<td>1</td>
<td>Dell EMC S3048-ON</td>
<td>connects up to 48 systems, add as needed</td>
</tr>
<tr>
<td>Top of Rack Network Switch</td>
<td>1</td>
<td>Dell EMC S4048T-ON</td>
<td>1 per rack of systems, unless System NIC bonding, then multiply by the number of linked interfaces</td>
</tr>
</tbody>
</table>

**Table 7.2: Bill of Materials - System Counts #**

<table>
<thead>
<tr>
<th>Role</th>
<th>Quantity PoC</th>
<th>Quantity V2P</th>
<th>Quantity Production</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Solution Admin Host</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>Dell PowerEdge R640 Server</td>
<td>n/a</td>
</tr>
<tr>
<td>Kubernetes Master Node(s)</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>Dell EMC PowerEdge R640 Server</td>
<td>requires an odd number to provide high availability</td>
</tr>
<tr>
<td>Kubernetes Worker Node(s)</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>Dell EMC PowerEdge R640 Server</td>
<td>can be scaled up to 100 for a single cluster instance</td>
</tr>
</tbody>
</table>

**Table 7.3: Bill of Materials - Dell EMC PowerEdge R640 System #**

<table>
<thead>
<tr>
<th>Role</th>
<th>Quantity</th>
<th>SKU</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Every System Role</td>
<td>1</td>
<td>210-AKWU</td>
<td>PowerEdge R640 Server</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>461-AAEM</td>
<td>Trusted Platform Module 2.0</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>321-BCQJ</td>
<td>2.5&quot; Chassis with up to 8 Hard Drives and 3PCIe slots</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>340-BKNE</td>
<td>PowerEdge R640 Shipping</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>343-BBEV</td>
<td>PowerEdge R640 x8 Drive Shipping material</td>
</tr>
<tr>
<td>P/N</td>
<td>Description</td>
<td>Details</td>
<td></td>
</tr>
<tr>
<td>-----------</td>
<td>------------------------------------------------------------------------------</td>
<td>--------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>338-BLUU</td>
<td>1 Intel Xeon Gold 5115 2.4G, 10CT/20CT, 10.4GT/s, 14M Cache, Turbo, HT(85W) DDR4-2400</td>
<td></td>
<td></td>
</tr>
<tr>
<td>374-BBPR</td>
<td>1 Intel Xeon Gold 5115 2.4G, 10CT/20CT, 10.4GT/s, 14M Cache, Turbo, HT(85W) DDR4-2400</td>
<td></td>
<td></td>
</tr>
<tr>
<td>370-ABWE</td>
<td>Heatsinks for Midbay Configuration 1 370-ABWE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>370-ADNU</td>
<td>Memory DIMM Type and Speed 2666MT/s RDIMMS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>370-AAIP</td>
<td>Memory Configuration Type Performance Optimized</td>
<td></td>
<td></td>
</tr>
<tr>
<td>370-ADNF</td>
<td>Memory Capacity 32GB RDIMM, 26666MT/s, Dual Rank</td>
<td></td>
<td></td>
</tr>
<tr>
<td>780-BCDS</td>
<td>RAID Configuration C7, Unconfigured RAID for HDDs or SSDs (Mixed Drive Types)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>400-ASZB</td>
<td>RAID/Internal Storage Controllers PERC H740P RAID Controller, 8GB NV Cache, Minicard</td>
<td></td>
<td></td>
</tr>
<tr>
<td>400-ASZB</td>
<td>Hard Drives 1.92TB SSD SATA Mix Use 6GBPS 512e 2.5in Hot-plug Drive, S4600,3 DWPD, 10512 TBW</td>
<td></td>
<td></td>
</tr>
<tr>
<td>421-5736</td>
<td>No Media Required</td>
<td></td>
<td></td>
</tr>
<tr>
<td>385-BBKT, 528-BBWT</td>
<td>iDRAC9 Enterprise with OME Server Configuration Management</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Quantity</td>
<td>Code</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>----------</td>
<td>-------</td>
<td>--------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>379-BCQV</td>
<td>iDRAC Group Manager, Enabled</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>379-BCSG</td>
<td>iDRAC, Legacy Password</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>330-BBGN</td>
<td>PCIe Riser Config 2, 3x16 LP</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>555-BCKP</td>
<td>Network Daughter Card Intel X710 Quad Port 10Gb DA/SFP+ Ethernet</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>384-BBQJ</td>
<td>8 Standard Fans for R640</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>450-ADWS</td>
<td>Dual, Hot-plug, Redundant Power Supply (1+1), 750W</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>450-AALV</td>
<td>NEWA 5-15P to C13 Wall Plug, 125 Volt, 15 AMP, 10 Feet (3m), Power Cord, North America</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>325-BCHH, 350-BBJS</td>
<td>Standard Bezel for x4 and x8 Chassis</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>350-BBKC</td>
<td>Quick Sync 2 (At-the-box mgmt.)</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>750-AABF</td>
<td>BIOS and Advanced System Configuration Setting Power Saving Dell Active Power Controller</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>800-BBDM</td>
<td>Advanced System Configurations UEFI BIOS Boot Mode with GPT Partition</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>770-BBBL</td>
<td>ReadyRails Sliding Rails with Cable Management Arm</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>631-AACK</td>
<td>No System Documentation, No OpenManage DVD Kit</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>813-9255,813-9262,813-9274,989-3439</td>
<td>3 Years ProSupport with Next Business Day Onsite Service</td>
<td></td>
</tr>
</tbody>
</table>
Table 7.4: Bill of Materials - Software

<table>
<thead>
<tr>
<th>Role</th>
<th>Quantity PoC</th>
<th>Quantity V2P</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Software</td>
<td>4</td>
<td>+2</td>
<td>SUSE CaaS Platform, x86-64, 1-2 Sockets or 1 Virtual Machine, L3-Priority Subscription, 3 year</td>
<td>full count includes Administration//Master/Worker Nodes</td>
</tr>
</tbody>
</table>

Dell EMC Network Switches

Dell EMC PowerEdge Servers

SUSE Software
  - Documentation - https://www.suse.com/documentation/suse-enterprise-storage-5/