SUSE

SUSE Best Practices

SAP

SAP Data Intelligence 3 on SUSE's Kubernetes Stack

SUSE Linux Enterprise Server for SAP Applications 15 SP4
Rancher Kubernetes Engine 2
Harvester
SAP Data Intelligence 3

Dr. Ulrich Schairer, SAP Solutions Architect (SUSE)
Kevin Klinger, SAP Solutions Architect (SUSE)

SUSE

Date: 2025-03-13

SAP Data Intelligence 3 is the tool set to govern big amounts of data, and it runs fully containerized. This document describes the installation and configuration of SAP Data Intelligence 3 deployed on SUSE's Kubernetes stack, including Harvester, Rancher, RKE2 and Longhorn.

Contents

1 Introduction

This guide describes the on-premises installation of SAP Data Intelligence 3.3 using Harvester and Rancher Kubernetes Engine (RKE) 2. In a nutshell, the installation of SAP Data Intelligence 3.3 consists of the following steps:

- Installing Harvester
- Installing Rancher Prime
- Installing RKE 2 Kubernetes cluster on the dedicated nodes
- Deploying SAP Data Intelligence 3.3 on RKE 2 Kubernetes cluster
- Performing post-installation steps for SAP Data Intelligence 3.3
- Testing the installation of SAP Data Intelligence 3.3

To have a fully supported setup, there are two Kubernetes clusters required. One runs Rancher Prime Management server and the other runs the actual workload, which for the purpose of this guide is SAP Data Intelligence.

2 Requirements

2.1 Hardware requirements

This chapter describes the hardware requirements for installing SAP Data Intelligence 3.3 on RKE 2 on top of SUSE Linux Enterprise Server 15 SP4. Only the AMD64/Intel 64 architecture is applicable for our use case.

2.1.1 Hardware Sizing

Correct hardware sizing is very important for setting up SAP Data Intelligence 3.3 on RKE 2.

2.1.1.1 Development systems

- Minimal hardware requirements for a generic SAP Data Intelligence 3 deployment:
  - At least 7 nodes are needed for the Kubernetes cluster.
  - Minimum sizing of the nodes needs to be as shown below:

Server Role	Count	RAM	CPU	Disk space
Management Workstation	1	16 GiB	4	>100 GiB
Master Node	3	16 GiB	4	>120 GiB
Worker Node	4	32 GiB	8	>120 GiB

2.1.1.2 Production systems

- Minimal hardware requirements for an SAP Data Intelligence 3 deployment for production use:
  - At least seven nodes are needed for the Kubernetes cluster.
  - Minimum sizing of the nodes needs to be as shown below:

Server Role	Count	RAM	CPU	Disk space
Management Workstation	1	16 GiB	4	>100 GiB
Master Node	3	16 GiB	4	>120 GiB
Worker Node	4	64 GiB	16	>120 GiB

2.2 Software requirements

The following list contains the software components needed to install SAP Data Intelligence 3.3 on RKE:

- SUSE Linux Enterprise Server 15 SP4
- Rancher Kubernetes Engine 2
- SAP Software Lifecycle Bridge
- SAP Data Intelligence 3.3
- Secure private registry for container images, for example https://documentation.suse.com/sbp/all/single-html/SBP-Private-Registry/index.html
- Access to a storage solution providing dynamically physical volumes
- If it is planned to use Vora's streaming tables checkpoint store, an S3 bucket like object store is needed
- If it is planned to enable backup of SAP Data Intelligence 3.3 during installation access to an S3-compatible object store is needed

2.3 Installation on top of Harvester

When using Harvester to provision the virtual machines for an SAP Data Intelligence installation, the hardware requirements for Harvester need to be added to the requirements of SAP Data Intelligence described at *Section 2.1.1, "Hardware Sizing"*.

2.3.1 Harvester hardware requirements

A full list of requirements for Harvester can be found at https://docs.harvesterhci.io/v1.0/install/requirements

2.3.2 Development systems

The recommended setup for a Harvester cluster to be used for development has the following requirements:

Server Role	Count	RAM	CPU	Disk space
Harvester Node	3	32 GiB	8	>140 GiB

Adding the hardware requirements of SAP Data Intelligence as described in chapter *Section 2.1.1.1, "Development systems"*, the following hardware is required to run an SAP Data Intelligence cluster on top of Harvester for development:

Server Role	Count	RAM	CPU	Disk space
Harvester Node	3	288 GiB	72	>1360 GiB

2.3.3 Production systems

The recommended setup for a Harvester cluster to be used in production has the following requirements:

Server Role	Count	RAM	CPU	"Disk space"	Disk speed
Harvester Node	3	64 GiB	16	>500 GiB	>5000 IOPs

Adding the hardware requirements of SAP Data Intelligence as described in chapter *Section 2.1.1.2, "Production systems"*, the following hardware is required to run an SAP Data Intelligence cluster on top of Harvester in production:

Server Role	Count	RAM	CPU	Disk space
Harvester Node	3	512 GiB	128	>2440 GiB

3 Preparations

- Get a SUSE Linux Enterprise Server 15 SP4 subscription.
- Download the installer for SUSE Linux Enterprise Server 15 SP4.
- Check the storage requirements.
- Create a or get access to a private container registry.
- Get an SAP S-user to access software and documentation by SAP.
- Read the relevant SAP documentation:
  - Release Note for SAP DI 3 (https://launchpad.support.sap.com/#/notes/2871970)
  - Release Note for SAP SLC Bridge (https://launchpad.support.sap.com/#/notes/2589449)
  - Installation Guide at help.sap.com (https://help.sap.com/viewer/a8d90a56d61a49718ebcb5f65014bbe7/3.2.latest/en-US)

Important

Make sure that the Harvester version fits the Rancher Prime version. A support matrix can be found here: https://docs.harvesterhci.io/v1.0/rancher/rancher-integration#rancher—harvester-support-matrix

4 Installing SUSE Rancher Harvester

4.1 Introduction

Harvester is the open source Hyper Converged Infrastructure (HCI) solution running on Kubernetes, Longhorn and Kubevirt. Harvester provides the ability to provision, manage and run virtual machines.

4.2 Prerequistites

Depending on the purpose of the Harvester installation (development, testing or productive use), one or more (virtual) machines are needed. At the time of writing, the system architecture is x86_64 only. For the Harvester deployment, the following information should be handy:

- IP addresses for the hosts running Harvester
- IP address to be used as management address
- Gateway address
- IP address of name server
- Access to a time server
- Access to the Internet (for airgapped installations see: https://docs.harvesterhci.io/v1.1/airgap)

For more information, see the product documentation for Harvester: https://docs.harvesterhci.io/v1.1/

4.3 Preparing the installation

Before the installation of Harvester can be started, the following steps should be performed:

- Download installation media as needed https://github.com/harvester/harvester/releases
- Prepare hardware to run the Harvester installation on, for example, mount the ISO file.
- Network setup (IP addresses, VLAN)

4.4 Installing Harvester

The installation of Harvester is straight forward:

- Boot the machines dedicated to the Harvester cluster from the installation media.
- After booting the machine, a guided setup leads you through the installation.

FIGURE 1: CHOOSE INSTALLATION MODE

- Provide the following information:
  - Device where the installation is targeted to

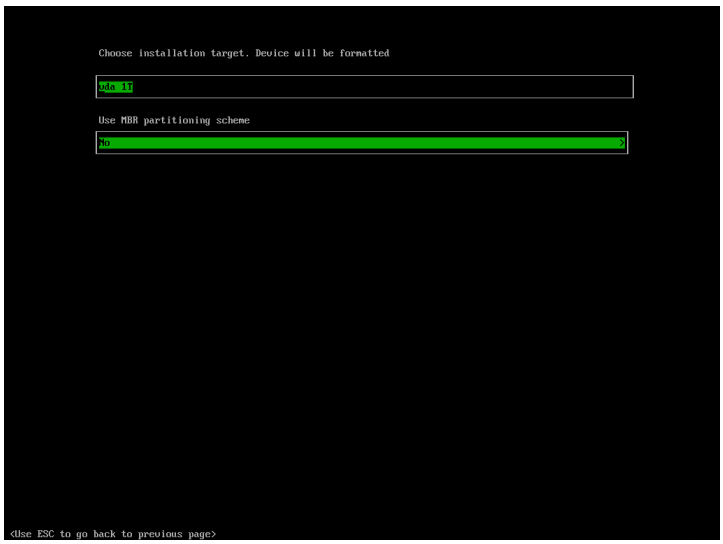

FIGURE 2: INSTALLATION TARGET

- Host name
- IP address
- Network interface to be used
- Gateway

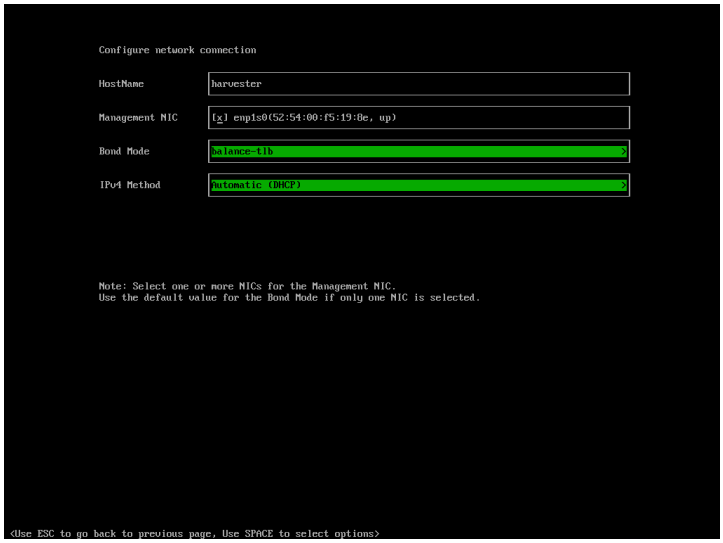

FIGURE 3: NETWORK CONFIGURATION

- DNS servers

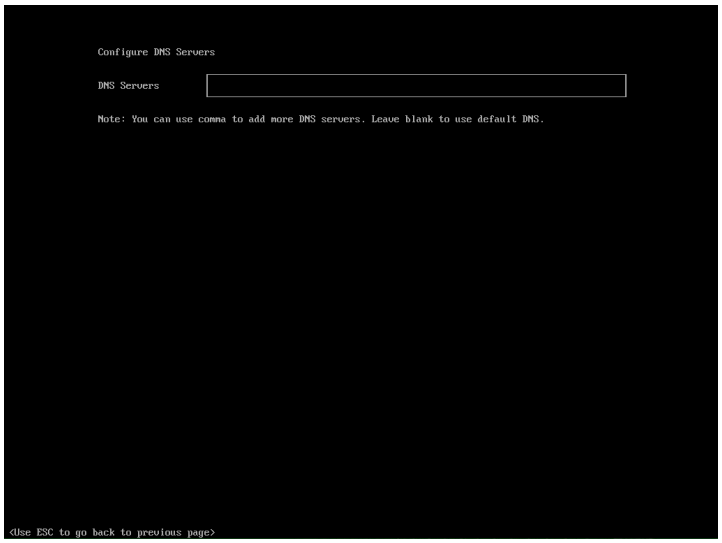

FIGURE 4: DNS CONFIGURATION

- Management IP address

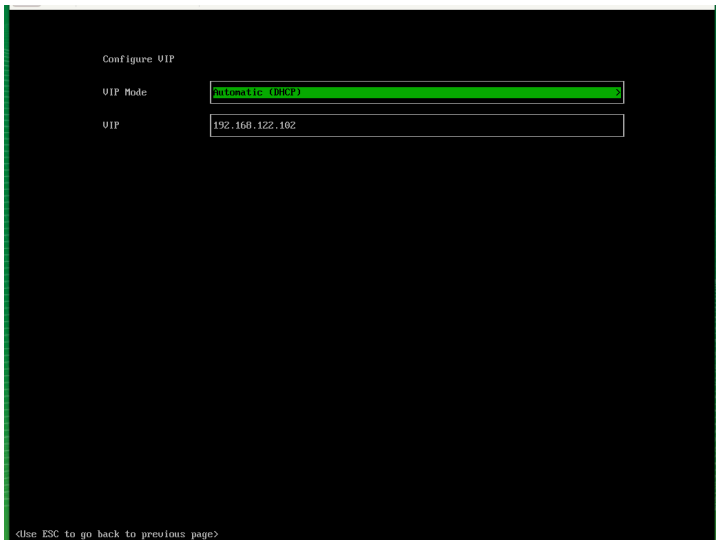

FIGURE 5: CONFIGURE MANAGEMENT IP ADDRESS

- Define cluster token. This is needed when joining other Harvester nodes.

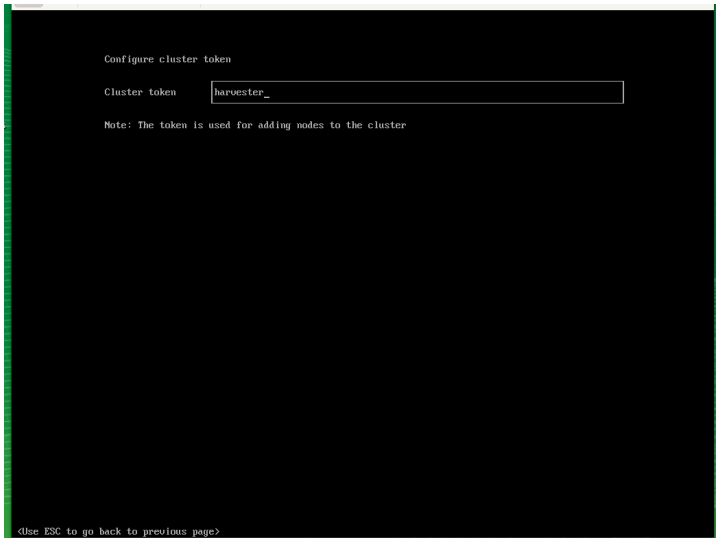

FIGURE 6: DEFINE CLUSTERTOKEN

- Set the node shell access password. Default user is "rancher".

FIGURE 7: SET PASSWORD FOR NODE ACCESS

- Configure the time server.

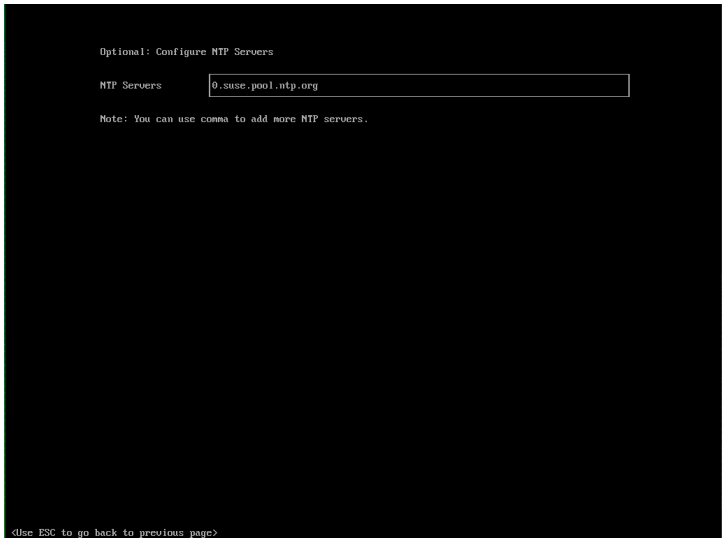

FIGURE 8: TIMEHOST CONFIGURATION

- Proxy servers (optional) are being entered.

Finally, a review panel is displayed.

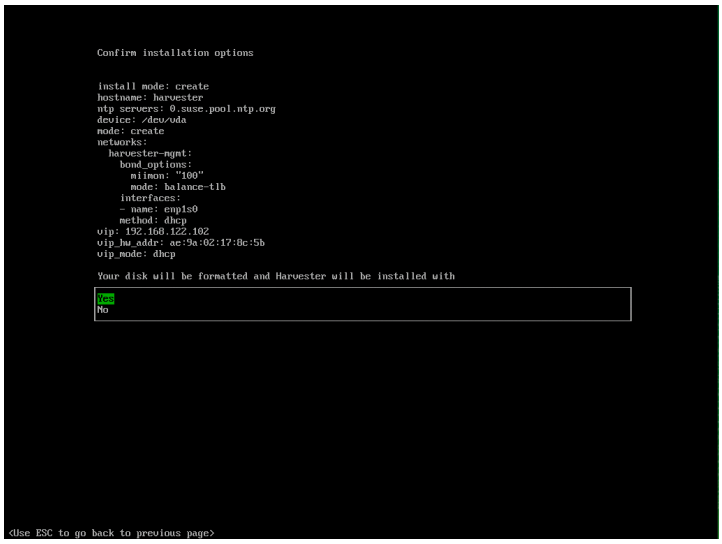

FIGURE 9: REVIEW INSTALLATION SETTINGS

Confirm the configuration. The installation will start.

When the installation is finished, you will see the following screen:

FIGURE 10: INSTALLATION FINISHED

This means that Harvester is up and running. Be patient as it can take some minutes.

For more installation options, see the Harvester documentation at https://docs.harvester-hci.io/v1.1

For productive environments, it is recommended to set up a Harvester cluster consisting of at least three nodes (or a higher odd number). To join nodes to the existing Harvester installation, simply select "Join existing Harvester cluster" after booting the node from the installation media.

FIGURE 11: JOIN HARVESTER CLUSTER

For the installation workflow described, the following information is needed in addition:

- the management VIP
- the cluster token

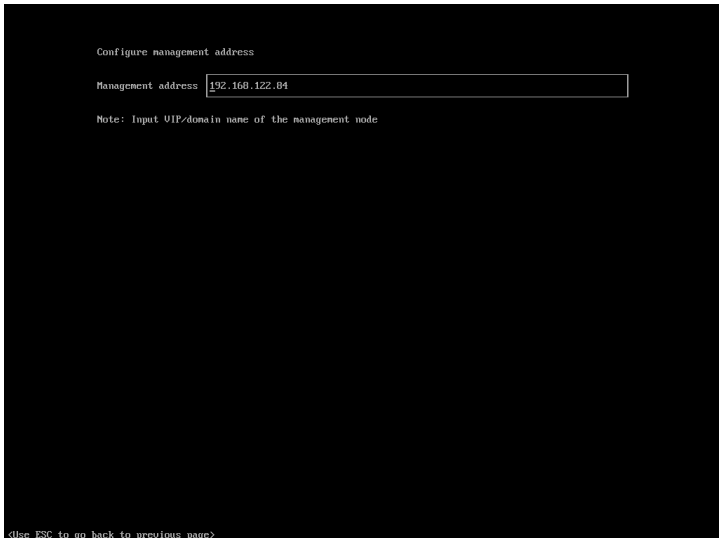

FIGURE 12: HARVESTER VIP

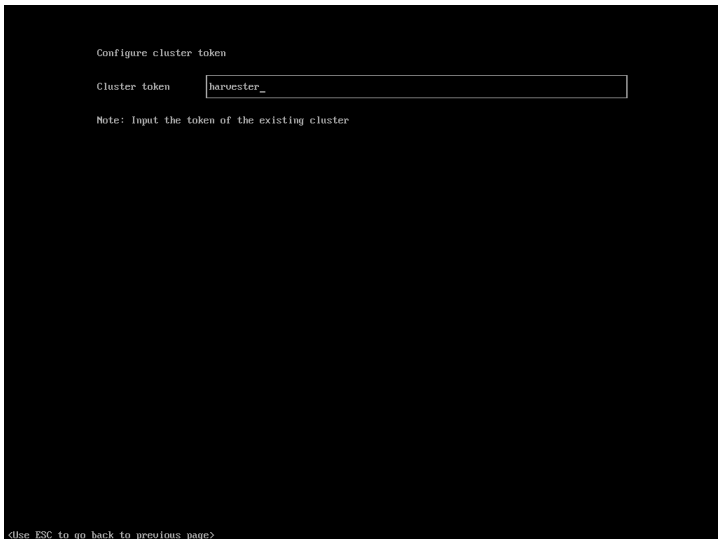

FIGURE 13: CLUSTER TOKEN

4.5 Accessing the management UI

The Harvester HCI is managed via a Web UI:

- Use the management (VIP) address to access the Harvester UI via an Internet browser. Next, set up the administrative account for Harvester.

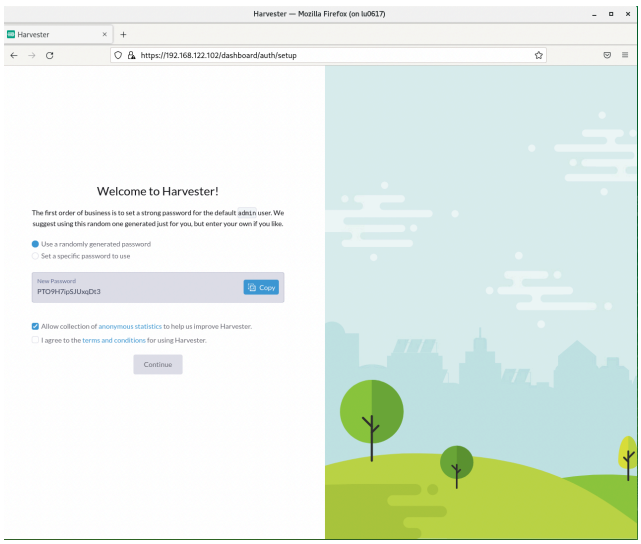

FIGURE 14: FIRST WELCOME

- After logging in, the Harvester Cluster overview dashboard is displayed.

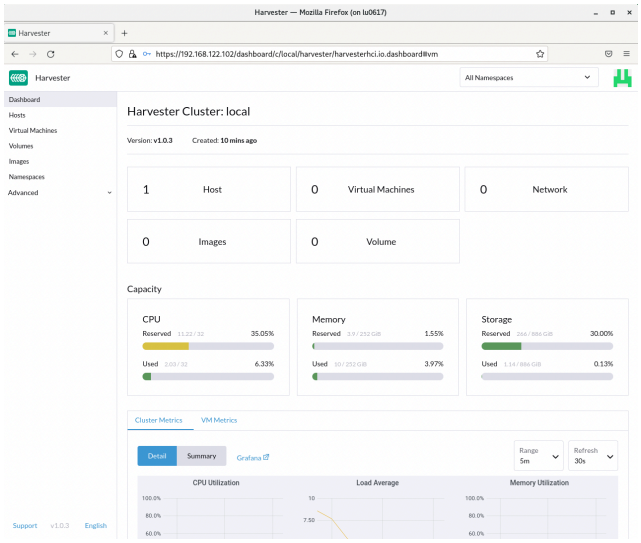

FIGURE 15: HARVESTER DASHBOARD

5 Installing Rancher Prime

5.1 Preparation

To have a highly available Rancher Prime setup, you need a load balancer for your Rancher Prime nodes. This section describes how to set up a custom load balancer using `haproxy`. If you already have a load balancer, you can use that to make Rancher Prime highly available.

If you do not plan to set up a highly available Rancher Prime cluster, you can skip this section.

5.1.1 Installing an haproxy-based load balancer

Set up a virtual machine or a bare metal server with SUSE Linux Enterprise Server and SUSE Linux Enterprise High Availability or use SUSE Linux Enterprise Server for SAP Applications. Install the `haproxy` package.

```
$ zypper in haproxy
```

Create the configuration for `haproxy`. Find an example configuration file for `haproxy` below and adapt for the actual environment.

```
# cat <<EOF > /etc/haproxy/haproxy.cfg
```

```
global
  log /dev/log daemon
  maxconn 32768
  chroot /var/lib/haproxy
  user haproxy
  group haproxy
  daemon
  tune.bufsize 32768
  tune.ssl.default-dh-param 2048
  ssl-default-bind-ciphers ALL:!aNULL:!eNULL:!EXPORT:!DES:!3DES:!MD5:!PSK:!RC4:!ADH:!
LOW@STRENGTH

defaults
  log     global
  mode    tcp
  option  log-health-checks
  option  log-separate-errors
  option  dontlog-normal
  option  dontlognull
  option  tcplog
  retries 3
  option  redispatch
  maxconn 10000
  timeout connect     5s
  timeout client     50s
  timeout server    450s

listen stats
  bind 0.0.0.0:80
  bind :::80 v6only
  stats enable
  stats uri     /
  stats refresh 5s

# access the kubernetes api
frontend kubeapi
  bind *:6443
  mode tcp
  default_backend kubeapibackend

# address to register new nodes
frontend rke2server
  bind *:9345
  mode tcp
  default_backend rke2serverbackend

backend kubeapibackend
```

```
  balance roundrobin
  server mynode1 192.168.122.20:6443 check
  server mynode2 192.168.122.30:6443 check
  server mynode3 192.168.122.40:6443 check


backend rke2serverbackend
  balance roundrobin
  server mynode1 192.168.122.20:9345 check
EOF
```

Check the configuration file:

```
$ haproxy -f /path/to/your/haproxy.conf -c
```

Enable and start the `haproxy` load balancer:

```
$ systemctl enable haproxy
$ systemctl start haproxy
```

Do not forget to restart or reload `haproxy` if any changes are made to the haproxy configuration file.

5.1.2 Installing RKE2

To install RKE2, the script provided at https://get.rke2.io can be used as follows:

```
$ curl -sfL https://get.rke2.io | INSTALL_RKE2_VERSION=v1.28.13-rke2r1 sh
```

For HA setups, it is necessary to create RKE2 cluster configuration files in advance. On the first master node:

```
$ mkdir -p /etc/rancher/rke2
$ cat <<EOF > /etc/rancher/rke2/config.yaml
token: 'your cluster token'
system-default-registry: registry.rancher.com
tls-san:
  - FQDN of fixed registration address on load balancer
  - other hostname
  - IP v4 address
EOF
```

Create configuration files for additional cluster nodes:

```
$ cat <<EOF > /etc/rancher/rke2/config.yaml
server: https://"FQDN of registration address":9345
token: 'your cluster token'
system-default-registry: registry.rancher.com
tls-san:
  - FQDN of fixed registration address on load balancer
  - other hostname
  - IP v4 address
EOF
```

Important

You also need take about ETCD Snapshots and to perfom backups of your Rancher instance. This is not part of this Document and you can find more information in our Documentation.

Important

For security reasons, we generally recommend activating the CIS profile when installing RKE2. This is currently still being validated and will be included in the documentation at a later date.

Now enable and start the RKE2 components and run the following command on each cluster node:

```
$ systemctl enable rke2-server --now
```

To verify the installation, run the following command:

```
$ /var/lib/rancher/rke2/bin/kubectl --kubeconfig /etc/rancher/rke2/rke2.yaml get nodes
```

For convenience, the `kubectl` binary can be added to the **$PATH** and the given `kubeconfig` can be set via an environment variable:

```
$ export PATH=$PATH:/var/lib/rancher/rke2/bin/
$ export KUBECONFIG=/etc/rancher/rke2/rke2.yaml
```

5.1.3 Installing Helm

To install Rancher Prime and some of its required components, you need to use Helm.

One way to install Helm is to run:

```
$ curl https://raw.githubusercontent.com/helm/helm/main/scripts/get-helm-3 | bash
```

5.1.4 Installing cert-manager

To install the `cert-manager` package, do the following:

```
$ kubectl create namespace cert-manager
```

How to create the **imagePullSecret** is described in the *Section 10.1, "Creating an imagePullSecret for the Rancher Application Collection"*.

5.1.4.1 Installing the application

You will need to login to the Rancher Application Collection:

```
$ helm registry login dp.apps.rancher.io/charts -u <yourUser> -p <your-token>
```

```
$ helm install cert-manager oci://dp.apps.rancher.io/charts/cert-manager \
--set crds.enabled=true \
--set-json 'global.imagePullSecrets=[{"name":"application-collection"}]' \
--namespace=cert-manager \
--version 1.15.2
```

5.2 Installing Rancher Prime

To install Rancher Prime, you need to add the related Helm repository. To achieve that, use the following command:

```
$ helm repo add rancher-prime https://charts.rancher.com/server-charts/prime
```

Next, create the `cattle-system` namespace in Kubernetes as follows:

```
$ kubectl create namespace cattle-system
```

The Kubernetes cluster is now ready for the installation of Rancher Prime:

```
$ helm install rancher rancher-prime/rancher \
    --namespace cattle-system \
```

```
    --set hostname=<your.domain.com> \
    --set replicas=3
```

During the rollout of Rancher Prime, you can monitor the progress using the following command:

```
$ kubectl -n cattle-system rollout status deploy/rancher-prime
```

When the deployment is done, you can access the Rancher Prime cluster at https://<your.domain.com>. Here you will also find a description about how to log in for the first time.

6 Using Harvester and Rancher Prime together

Important

If not done already, make sure the desired Harvester installation is compatible with your Rancher Prime setup: https://docs.harvesterhci.io/v1.0/rancher/rancher-integration#rancher—harvester-support-matrix

6.1 Connecting Harvester with Rancher Prime

To connect Harvester with Rancher Prime, the first step is to access Rancher. The menu in the upper left corner allows you to open the Virtualization Management tab.

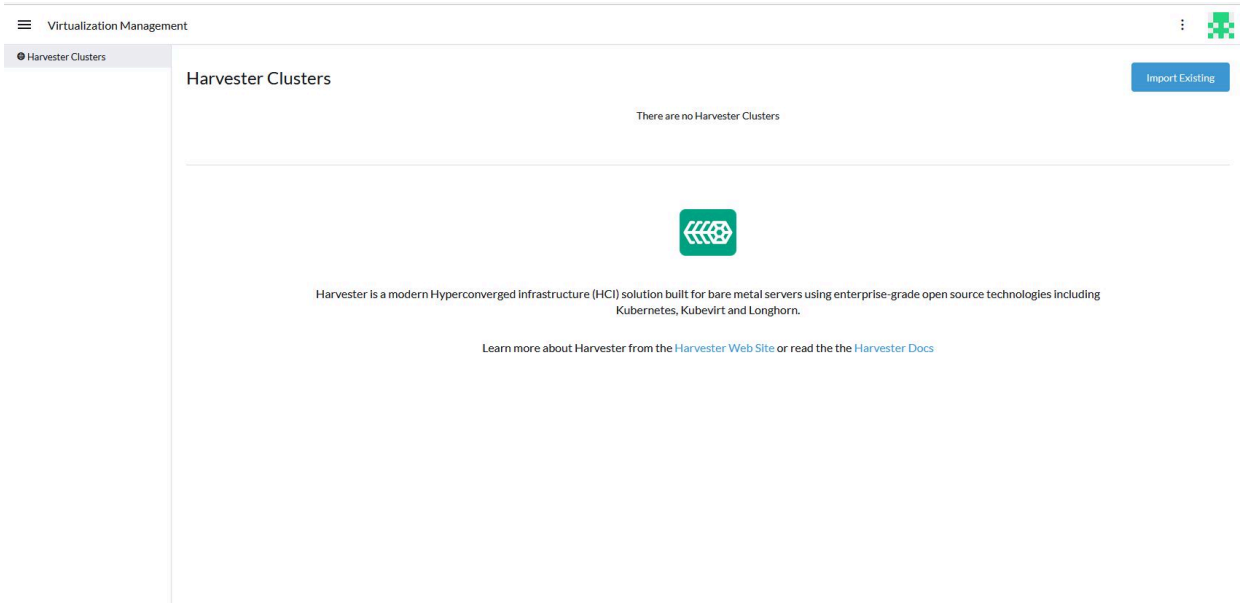

FIGURE 16: RANCHER VIRTUALIZATION MANAGEMENT

The only option available here should be the "Import Existing" button. Click this button.

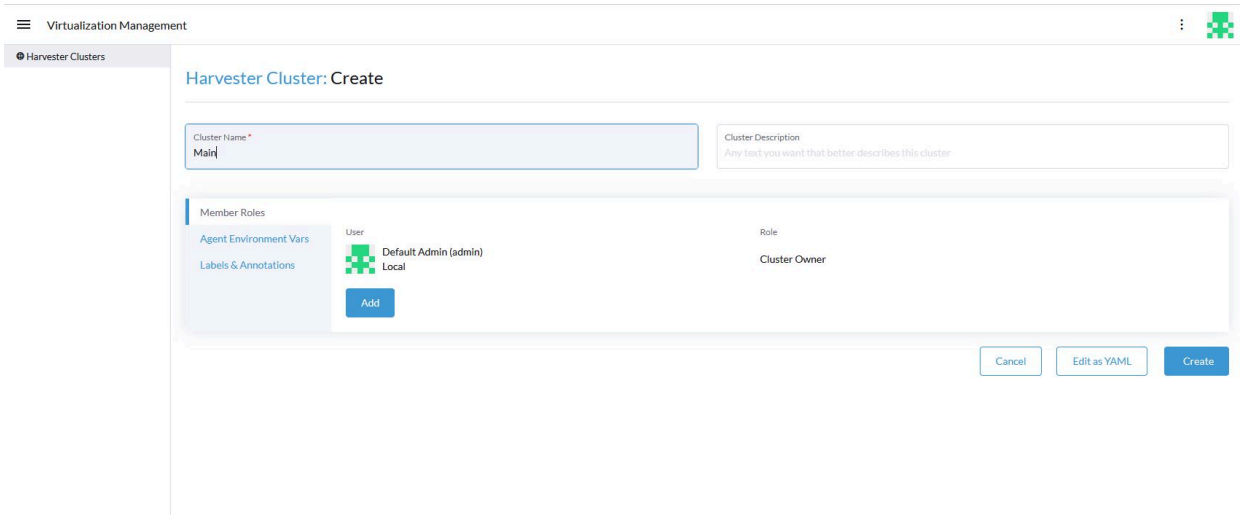

FIGURE 17: CREATE HARVESTER CLUSTER

On the next screen, enter a name for the Harvester cluster.

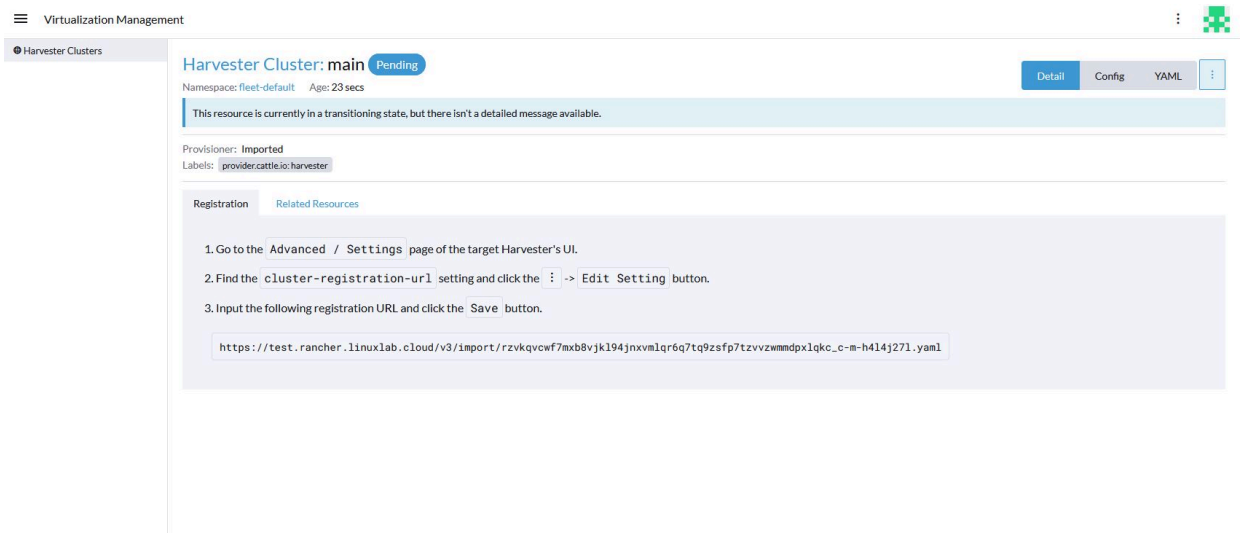

FIGURE 18: CREATE HARVESTER CLUSTER

After clicking the "Create" button, three steps to be executed on the Harvester cluster are shown.

6.2 Provisioning virtual machines with RKE2

After Rancher Prime and Harvester are connected, virtual machines can be provisioned using Rancher Prime. To do so, access Rancher Prime and click "Create" in the home tab. Select the option "Harvester" and make sure that RKE2/K3s is selected.

Next, the "Cluster: Create Harvester" page is loaded.

The first step you perform here is to set the cluster name. The "Machine Pools" section specifies the number of machines to be provisioned, their Kubernetes role, and further VM specific parameters.

The "Cluster Configuration" section allows you to set some Kubernetes-specific parameters. Here, select the Kubernetes version to fulfill the requirements of the desired workload.

Within the scope of this guide it is recommended to deploy three master nodes (roles: etcd & Control Plane) and four worker nodes (roles: Worker) for SAP Data Intelligence. Check *Section 2.1.1, "Hardware Sizing"* to fill out the machine specific configurations to meet the requirements for the given purpose and *Section 6.2, "Provisioning virtual machines with RKE2"* to get an overview how to roll out an RKE2 cluster using Harvester.

When all machines are provisioned and the RKE2 cluster is up and running, a storage must be made available for SAP Data Intelligence. SUSE offers Longhorn which is a validated storage for SAP Data Intelligence workloads. The next chapter describes how to set up Longhorn.

Important

SAP Data Intelligence requires an S3 compatible storage for its backups, which is not delivered by Longhorn. Check the related SAP Note (https://launchpad.support.sap.com/#/notes/2871970) to get an overview of the supported storage solutions together with RKE2.

7 Installing Longhorn

This chapter details the minimum requirements to install Longhorn and describes three different ways for the installation. For more details, visit https://longhorn.io/docs/1.6.2/deploy/install/

7.1 Requirements

Before Longhorn can be installed on a Kubernetes cluster, all nodes must have the `open-iscsi` package installed, and the ISCSI daemon needs to be started. To do so, run:

```
# zypper in -y open-iscsi
# systemctl enable iscsid  --now
```

To esure a node is prepared for Longhorn, you can use the following script to check:

```
$ curl -sSfL https://raw.githubusercontent.com/longhorn/longhorn/v1.6.2/scripts/
environment_check.sh | bash
```

7.2 Installing Longhorn using Rancher Prime

Up-to-date and detailed instructions how to install Longhorn using Rancher Prime can be found at https://longhorn.io/docs/1.6.2/deploy/install/install-with-rancher/

7.3 Installing Longhorn using Helm

To install Longhorn using Helm, run the following commands:

```
$ helm repo add longhorn https://charts.longhorn.io
$ helm repo update
$ helm install longhorn longhorn/longhorn --namespace longhorn-system --create-namespace
```

These commands will add the Longhorn Helm charts to the list of Helm repositories, update the Helm repository, and execute the installation of Longhorn. = = = Installing Longhorn using `kubectl`

You can install Longhorn using `kubectl` with the following command:

```
$ kubectl apply -f https://raw.githubusercontent.com/longhorn/longhorn/v1.6.2/deploy/
longhorn.yaml
```

7.4 Exposing Longhorn UI by creating an Ingress with Basic Authentication

- Create a basic *auth* file:

  ```
  $ USER=<USERNAME_HERE>; \
    PASSWORD=<PASSWORD_HERE>; \
  ```

```
  echo "${USER}:$(openssl passwd -stdin -apr1 <<< ${PASSWORD})" >> auth
```

- Create a Secret from the file *auth*:

```
$ kubectl -n longhorn-system create secret generic basic-auth --from-file=auth
```

- Create the Ingress with basic authentication:

```
$ cat <<EOF > longhorn-ingress.yaml
apiVersion: networking.k8s.io/v1beta1
kind: Ingress
metadata:
  name: longhorn-ingress
  namespace: longhorn-system
  annotations:
    # type of authentication
    nginx.ingress.kubernetes.io/auth-type: basic
    # prevent the controller from redirecting (308) to HTTPS
    nginx.ingress.kubernetes.io/ssl-redirect: 'false'
    # name of the secret that contains the user/password definitions
    nginx.ingress.kubernetes.io/auth-secret: basic-auth
    # message to display with an appropriate context why the authentication is
 required
    nginx.ingress.kubernetes.io/auth-realm: 'Authentication Required '
spec:
  rules:
  - http:
      paths:
      - path: /
        backend:
          serviceName: longhorn-frontend
          servicePort: 80
EOF

$ kubectl -n longhorn-system apply -f longhorn-ingress.yaml
```

For more details, visit https://longhorn.io/docs/1.6.2/deploy/accessing-the-ui/longhorn-ingress/.

8 Installing SAP Data Intelligence 3.3

This section describes the installation of SAP Data Intelligence 3.3 on an RKE 2-powered Kubernetes cluster.

8.1 Preparation

The following steps need to be executed before the deployment of SAP Data Intelligence 3.3 can start:

- Create a namespace for SAP Data Intelligence 3.3.
- Create an access to a secure private registry.
- Create a default storage class.
- Download and install SAP SLC Bridge.
- Download the *stack.xml* file for provisioning the DI 3.3 installation.
- Check if the `nfsd` and `nfsv4` kernel modules are loaded and/or loadable on the Kubernetes nodes.

8.1.1 Creating namespace for SAP Data Intelligence 3.3 in the Kubernetes cluster

Log in to your management workstation and create the namespace in the Kubernetes cluster where DI 3.3 will be deployed.

```
$ kubectl create ns <NAMESPACE for DI 31>
$ kubectl get ns
```

8.1.2 Creating *cert* file to access the secure private registry

Create a file named *cert* that contains the SSL certificate chain for the secure private registry. This imports the certificates into SAP Data Intelligence 3.3.

```
$ cat CA.pem > cert
$ kubectl -n <NAMESPACE for DI 31> create secret generic cmcertificates --from-file=cert
```

8.2 Creating default storage class

To install SAP Data Intelligence 3.3, a default storage class is needed to provision the installation with physical volumes (PV). Below find an example for a `ceph/rbd` based storage class that uses the CSI.

- Create the *yaml* files for the storage class. Contact your storage admin to get the required information.

- Create `config-map`:

```
$ cat << EOF > csi-config-map.yaml
---
apiVersion: v1
kind: ConfigMap
data:
  config.json: |-
    [
      {
        "clusterID": "<ID of your ceph cluster>",
        "monitors": [
          "<IP of Monitor 1>:6789",
          "<IP of Monitor 2>:6789",
          "<IP of Monitor 3>:6789"
        ]
      }
    ]
metadata:
  name: ceph-csi-config
EOF
```

- Create a secret to access the storage:

```
$ cat << EOF > csi-rbd-secret.yaml
---
apiVersion: v1
kind: Secret
metadata:
  name: csi-rbd-secret
  namespace: default
stringData:
  userID: admin
  userKey: AQCR7htglvJzBxAAtPN0YUeSiDzyTeQe0lveDQ==
EOF
```

- Download the file:

```
$ curl -LO https://raw.githubusercontent.com/ceph/ceph-csi/master/deploy/rbd/
kubernetes/csi-rbdplugin-provisioner.yaml
```

- Download the file:

```
$ curl -LO https://raw.githubusercontent.com/ceph/ceph-csi/master/deploy/rbd/
kubernetes/csi-rbdplugin.yaml
```

- Create a pool on the Ceph storage where the PVs will be created, and insert the pool name and the Ceph cluster ID:

```
$ cat << EOF > csi-rbd-sc.yaml
---
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
   name: csi-rbd-sc
provisioner: rbd.csi.ceph.com
parameters:
   clusterID: <your ceph cluster id>
   pool: <your pool>
   csi.storage.k8s.io/provisioner-secret-name: csi-rbd-secret
   csi.storage.k8s.io/provisioner-secret-namespace: default
   csi.storage.k8s.io/node-stage-secret-name: csi-rbd-secret
   csi.storage.k8s.io/node-stage-secret-namespace: default
reclaimPolicy: Delete
mountOptions:
   - discard
EOF
```

- Create *config* for encryption. This is needed, else the deployment of the CSI driver for `ceph/rbd` will fail.

```
$ cat << EOF > kms-config.yaml
---
apiVersion: v1
kind: ConfigMap
data:
  config.json: |-
    {
      },
      "vault-tokens-test": {
          "encryptionKMSType": "vaulttokens",
          "vaultAddress": "http://vault.default.svc.cluster.local:8200",
          "vaultBackendPath": "secret/",
          "vaultTLSServerName": "vault.default.svc.cluster.local",
          "vaultCAVerify": "false",
          "tenantConfigName": "ceph-csi-kms-config",
          "tenantTokenName": "ceph-csi-kms-token",
```

```
          "tenants": {
              "my-app": {
                  "vaultAddress": "https://vault.example.com",
                  "vaultCAVerify": "true"
              },
              "an-other-app": {
                  "tenantTokenName": "storage-encryption-token"
              }
          }
       }
    }
metadata:
  name: ceph-csi-encryption-kms-config
EOF
```

- Deploy the `ceph`/`rbd` CSI and storage class:

```
$ kubectl apply -f csi-config-map.yaml
$ kubectl apply -f csi-rbd-secret.yaml
$ kubectl apply -f \
  https://raw.githubusercontent.com/ceph/ceph-csi/master/deploy/rbd/kubernetes/csi-
provisioner-rbac.yaml
$ kubectl apply -f \
  https://raw.githubusercontent.com/ceph/ceph-csi/master/deploy/rbd/kubernetes/csi-
nodeplugin-rbac.yaml
$ kubectl apply -f csi-rbdplugin-provisioner.yaml
$ kubectl apply -f csi-rbdplugin.yaml
$ kubectl apply -f csi-rbd-sc.yaml
$ kubectl apply -f kms-config.yaml
$ kubectl patch storageclass csi-rbd-sc \
  -p '{"metadata": {"annotations":{"storageclass.kubernetes.io/is-default-
class":"true"}}}'
```

- Check your storage class:

```
$ kubectl get sc
NAME                   PROVISIONER        RECLAIMPOLICY   VOLUMEBINDINGMODE
 ALLOWVOLUMEEXPANSION   AGE
csi-rbd-sc (default)   rbd.csi.ceph.com   Delete          Immediate           false
                 103m
```

8.3 Downloading the SLC Bridge

The SLC Bridge can be obtained:

- from the SAP software center at https://support.sap.com/en/tools/software-logistics-tools.html#section_622087154. Choose "Download SLC Bridge".
- via the information in the release notes of the SLC Bridge at https://launchpad.support.sap.com/#/notes/2589449.
- via https://help.sap.com/viewer/a8d90a56d61a49718ebcb5f65014bbe7/3.3.latest/en-US/8ae38791d71046fab1f25ee0f682dc4c.html.

Download the SLC Bridge software to the management workstation.

8.4 Installing the SLC Bridge

Rename the SLC Bridge binary to `slcb` and make it executable. Deploy the SLC Bridge to the Kubernetes cluster.

```
$ mv SLCB01_XX-70003322.EXE slcb
$ chmod 0700 slcb
$ export KUBECONFIG=<KUBE_CONFIG>
$ ./slcb init
```

During the interactive installation, the following information is needed:

- URL of secure private registry
- Choose **expert mode**
- Choose **NodePort** for the service

Take a note of the service port of the SLC Bridge. It is needed for the installation of SAP Data Intelligence 3.3 or for the reconfiguration of DI 3.3, for example to enable backup. If you forgot to note it down, the following command will list the service port:

```
$ kubectl -n sap-slcbridge get svc
```

8.5 Creating and downloading Stack XML for the SAP Data Intelligence installation

Follow the steps described in the chapter Install SAP Data Intelligence with SLC Bridge in a Cluster with Internet Access (https://help.sap.com/viewer/a8d90a56d61a49718ebcb5f65014bbe7/3.3.latest/en-US/7e4847e241c340b3a3c50a5db11b46e2.html) of the SAP Data Intelligence 3.3 Installation Guide.

8.5.1 Creating Stack XML

You can create the Stack XML via the SAP Maintenance Planner. Access the tool via https://support.sap.com/en/alm/solution-manager/processes-72/maintenance-planner.html. Go to the Maintenance Planner at https://apps.support.sap.com/sap/support/mp published on the SAP Web site and generate a Stack XML file with the container image definitions of the SAP Data Intelligence release that you want to install. Download the Stack XML file to a local directory. Copy *stack.xml* to the management workstation.

8.6 Running the installation of SAP Data Intelligence

The installation of SAP Data Intelligence 3.3 is invoked by:

```
$ export KUBECONFIG=<path to kubeconfig>
$ ./slcb execute --useStackXML MP_Stack_XXXXXXXXXX_XXXXXXXX_.xml --url https://
<node>:<service port>/docs/index.html
```

This starts an interactive process for configuring and deploying SAP Data Intelligence 3.3.

The table below lists some parameters available for an SAP Data Intelligence 3.3 installation:

Parameter	Condition	Recommendation
Kubernetes Namespace	Always	set to namespace created beforehand
Installation Type	installation or update	either
Container Registry	Always	add the uri for the secure private registry

Parameter	Condition	Recommendation
Checkpoint Store Configuration	installation	whether to enable Checkpoint Store
Checkpoint Store Type	if Checkpoint Store is enabled	use S3 object store from SES
Checkpoint Store Validation	if Checkpoint is enabled	Object store access will be verified
Container Registry Settings for Pipeline Modeler	optional	used if a second container registry is used
StorageClass Configuration	optional, needed if a different StorageClass is used for some components	leave the default
Default StorageClass	detected by SAP Data Intelligence installer	The Kubernetes cluster shall have a storage class annotated as default SC
Enable Kaniko Usage	optional if running on Docker	enable
Container Image Repository Settings for SAP Data Intelligence Modeler	mandatory	
Container Registry for Pipeline Modeler	optional	Needed if a different container registry is used for the pipeline modeler images
Loading NFS Modules	optional	Make sure that nfsd and nfsv4 kernel modules are loaded on all worker nodes
Additional Installer Parameters	optional	

For more details about input parameters for an SAP Data Intelligence 3.3 installation, visit the section Required Input Parameters (https://help.sap.com/viewer/a8d90a56d61a49718e-bcb5f65014bbe7/3.3.latest/en-US/abfa9c73f7704de2907ea7ff65e7a20a.html) of the SAP Data Intelligence Installation Guide.

8.7 Post-installation tasks

After the installation workflow is successfully finished, you need to carry out some additional tasks:

- Obtain or create an SSL certificate to securely access the SAP Data Intelligence installation:
  - Create a certificate request using `openssl`, for example:

    ```
    $ openssl req -newkey rsa:2048 -keyout <hostname>.key -out <hostname>.csr
    ```

  - Decrypt the key:

    ```
    $ openssl rsa -in <hostname>.key -out decrypted-<hostname>.key
    ```

  - Let a CA sign the <hostname>.csr You will receive a <hostname>.crt.
  - Create a secret from the certificate and the key in the SAP Data Intelligence 3 namespace:

    ```
    $ export NAMESPACE=<{di} 3 namespace>
    ```

```
$ kubectl -n $NAMESPACE create secret tls vsystem-tls-certs --key  decrypted-
<hostname>.key--cert <hostname>.crt
```

- Deploy an `nginx-ingress` controller:
  - For more information, see https://kubernetes.github.io/ingress-nginx/deploy/#bare-metal.
  - Create the `nginx-ingress` controller as a **nodePort** service according to the Ingress `nginx` documentation:

    ```
    $ kubectl apply -f https://raw.githubusercontent.com/kubernetes/ingress-nginx/
    controller-v0.46.0/deploy/static/provider/baremetal/deploy.yaml
    ```

  - Determine the port the `nginx` controller is redirecting HTTPS to:

    ```
    $ kubectl -n ingress-nginx get svc ingress-nginx-controller
    ```

    The output should be similar to the below:

    ```
    kubectl -n ingress-nginx get svc ingress-nginx-controller
    NAME                       TYPE       CLUSTER-IP    EXTERNAL-IP   PORT(S)
                   AGE
    ingress-nginx-controller   NodePort   10.43.86.90   <none>        80:31963/
    TCP,443:{di_version}06/TCP   53d
    ```

    In our example here, the TLS port is be 3.306. Note the port IP down as you will need it to access the SAP Data Intelligence installation from the outside.

- Create an Ingress to access the SAP Data Intelligence installation:

```
$ cat <<EOF > ingress.yaml
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  annotations:
    kubernetes.io/ingress.class: nginx
    nginx.ingress.kubernetes.io/force-ssl-redirect: "true"
    nginx.ingress.kubernetes.io/secure-backends: "true"
    nginx.ingress.kubernetes.io/backend-protocol: HTTPS
    nginx.ingress.kubernetes.io/proxy-body-size: "0"
    nginx.ingress.kubernetes.io/proxy-buffer-size: 16k
    nginx.ingress.kubernetes.io/proxy-connect-timeout: "30"
    nginx.ingress.kubernetes.io/proxy-read-timeout: "1800"
    nginx.ingress.kubernetes.io/proxy-send-timeout: "1800"
  name: vsystem
```

```
spec:
  rules:
  - host: "<hostname FQDN must match SSL certificate"
    http:
      paths:
      - backend:
          serviceName: vsystem
          servicePort: 8797
        path: /
  tls:
  - hosts:
    - "<hostname FQDN must match SSL certificate>"
    secretName: vsystem-tls-certs
EOF
$ kubectl apply -f ingress.yaml
```

- Connecting to https://hostname:<ingress service port> brings up the SAP Data Intelligence login dialog.

8.8 Testing the SAP Data Intelligence 3 installation

Finally, the SAP Data Intelligence 3 installation should be verified with some very basic tests:

- Log in to SAP Data Intelligence's launchpad
- Create example pipeline
- Create ML Scenario
- Test machine learning
- Download `vctl`

For details, see the SAP Data Intelligence 3 Installation Guide (https://help.sap.com/viewer/a8d90a56d61a49718ebcb5f65014bbe7/3.3.latest/en-US/1551785f3d7e4d37af7fe99185f7acb6.html)

9 Maintenance tasks

This section provides some tips about what should and could be done to maintain the Kubernetes cluster, the operating system and the SAP Data Intelligence 3 deployment.

9.1 Backup

It is good practice to keep backups of all relevant data to be able to restore the environment in case of a failure. To perform regular backups, follow the instructions as outlined in the respective documentation below:

- For RKE 2, consult section Backups and Disaster Recovery (https://rancher.com/docs/rke/latest/en/etcd-snapshots/)
- SAP Data Intelligence 3 can be configured to create regular backups. For more information, visit help.sap.com: https://help.sap.com/viewer/a8d90a56d61a49718ebcb5f65014bbe7/3.3.latest/en-US/e8d4c33e6cd648b0af9fd674dbf6e76c.html.

9.2 Upgrade or update

This section explains how you can keep your installation of SAP Data Intelligence, RKE 2 and SUSE Linux Enterprise Server up-to-date.

9.2.1 Updating the operating system

To obtain updates for SUSE Linux Enterprise Server 15 SP4, the installation must be registered either to SUSE Customer Center, an SMT or RMT server, or SUSE Manager with a valid subscription.

- SUSE Linux Enterprise Server 15 SP4 can be updated on the command line using `zypper`:

```
$ sudo zypper ref -s
$ sudo zypper lu
$ sudo zypper patch
```

- Other methods for updating SUSE Linux Enterprise Server 15 SP4 are described in the product documentation (https://documentation.suse.com/sles).

If an update requires a reboot of the server, make sure that this can be done safely.

- For example, block access to SAP Data Intelligence, and drain and cordon the Kubernetes node before rebooting:

```
$ kubectl edit ingress <put in some dummy port>
```

```
$ kubectl drain <node>
```

- Check the status of the node:

```
$kubectl get node <node>
```

The node should be marked as **not schedulable**.

- On RKE 2 master nodes, run the command:

```
$ sudo systemctl stop rke2-server
```

- On RKE 2 worker nodes, run the command:

```
$ sudo systemctl stop rke2-agent
```

- Update SUSE Linux Enterprise Server 15 SP4:

```
$ ssh node
$ sudo zypper patch
```

- Reboot the nodes if necessary or start the appropriate RKE 2 service.
  - On master nodes, run the command:

```
$ sudo systemctl start rke2-server
```

  - On worker nodes, run the command:

```
$ sudo systemctl start rke2-agent
```

- Check if the respective nodes are back and uncordon them.

```
$ kubectl get nodes
$ kubectl uncordon <node>
```

10 Appendix

10.1 Creating an imagePullSecret for the Rancher Application Collection

To make the resources available for deployment, you need to create an imagePullSecret. In this guide we use the name *application-collection* for it.

10.1.1 Creating an imagePullSecret using kubectl

Using `kubectl` to create the imagePullSecret is quite easy. Get your user name and your access token for the Rancher Application Collection. Then run:

```
$ kubectl -n <namespace> create secret docker-registry application-collection --docker-
server=dp.apps.rancher.io --docker-username=<yourUser> --docker-password=<yourPassword>
```

As secrets are namespace-sensitive, you need to create this for every namespace needed.

10.1.2 Creating an imagePullSecret using Rancher Prime

You can also create an imagePullSecret using Rancher Prime. Therefore, open Rancher Prime and enter your cluster.

Navigate to **Storage** → **Secrets** as shown below:

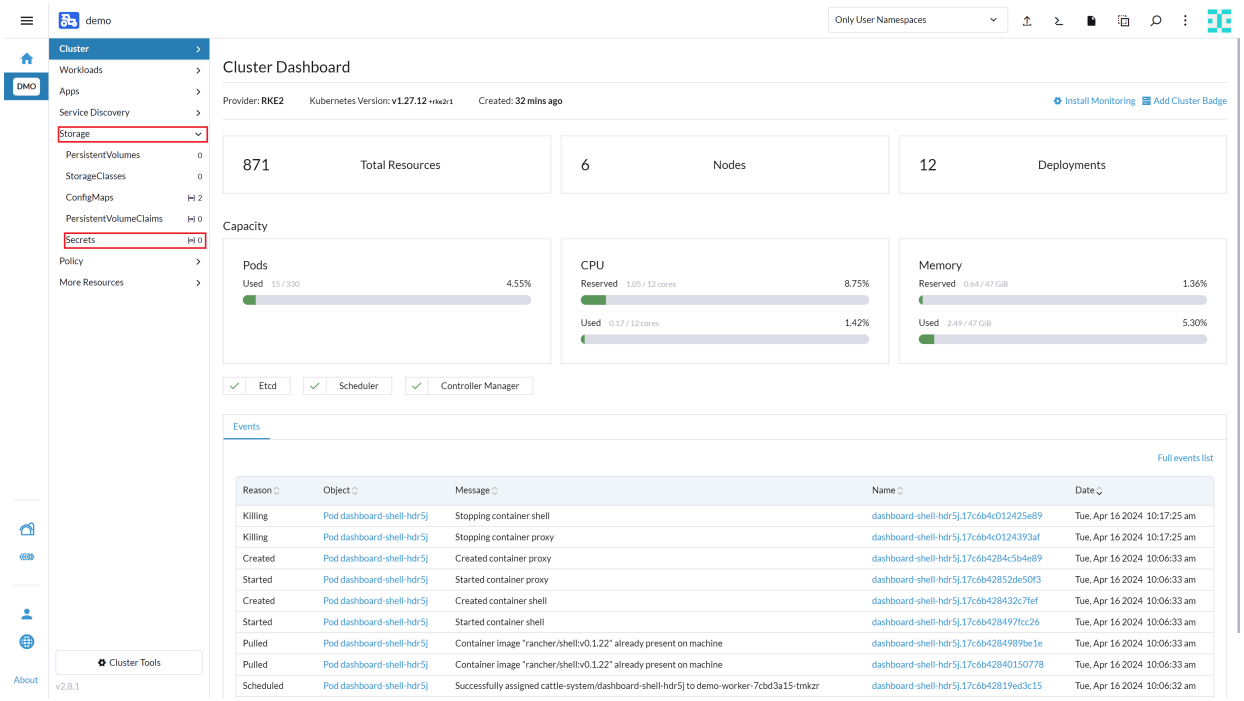

FIGURE 19: SECRETS MENU

Click the **Create** button in the top right corner.

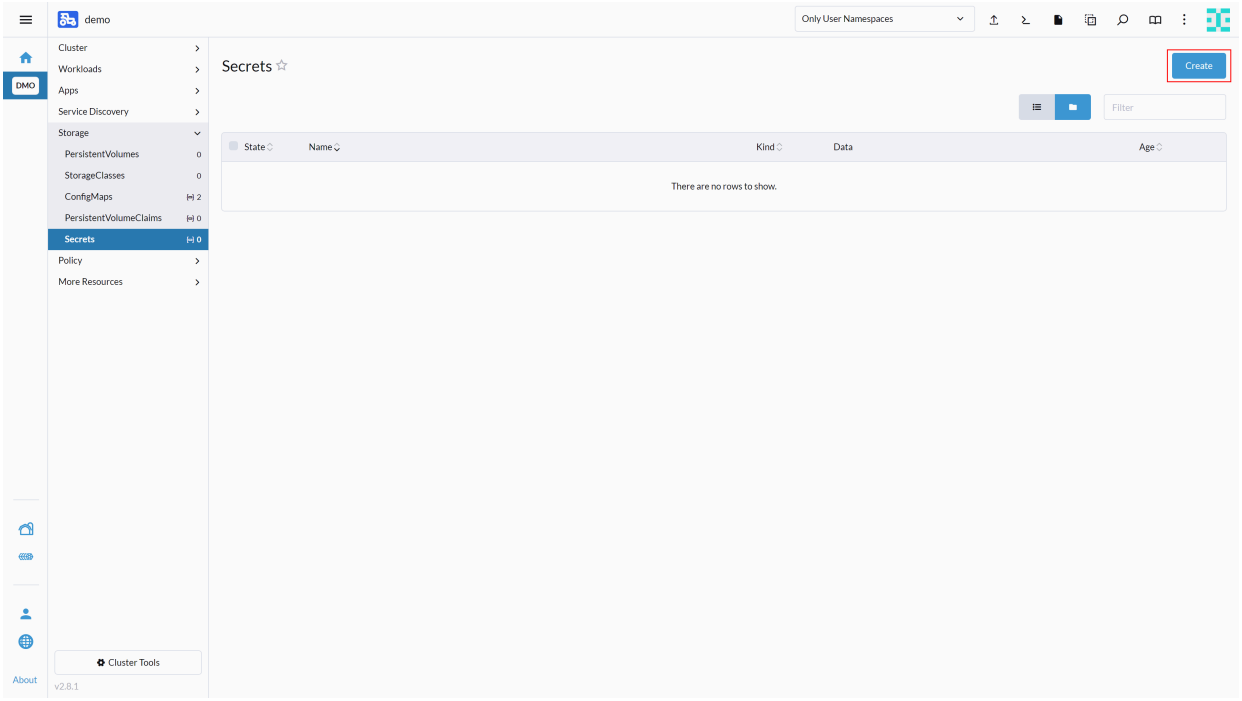

FIGURE 20: SECRETS OVERVIEW

A window will appear asking you to select the Secret type. Select **Registry** as shown here:

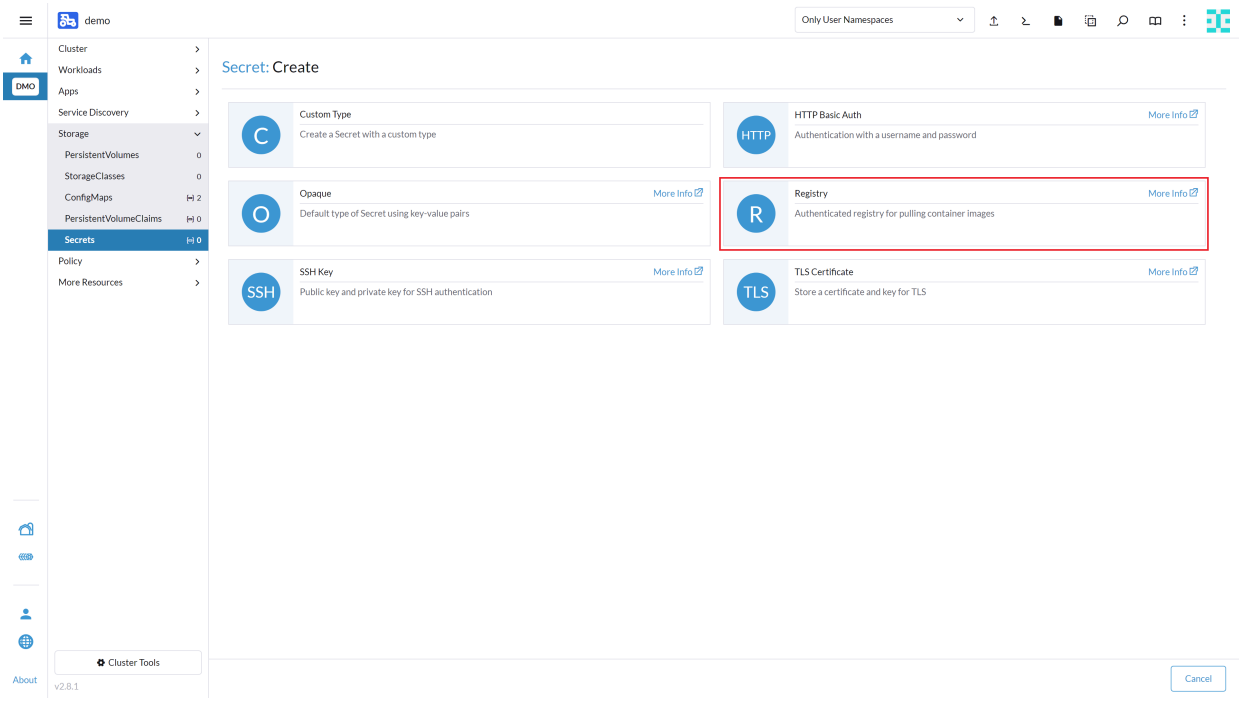

FIGURE 21: SECRETS TYPE SELECTION

Enter a name such as *application-collection* for the Secret. In the text box **Registry Domain Name**, enter *dp.apps.rancher.io*. Enter your user name and password and click the **Create** button at the bottom right.

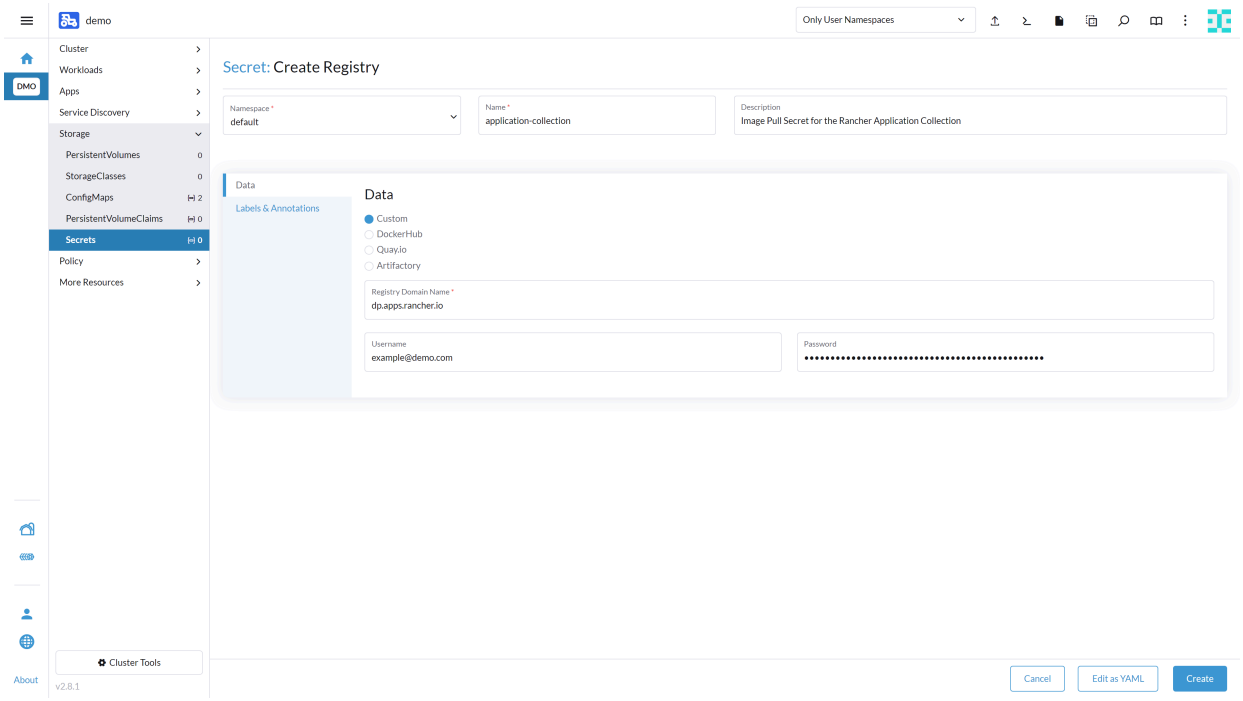

FIGURE 22: SECRETS CREATION STEP

11 Legal notice

12 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.