

Migration Toolkit for Virtualization

About Forklift	2
Planning the migration	4
Installing Forklift	5
Installing Forklift with the web console	5
Installing Forklift from the CLI	5
Adding a VDDK image to Forklift	7
Migrating virtual machines	9
Migrating virtual machines with the web console	9
Adding providers	9
Creating mappings	10
Creating a migration plan	11
Running a migration plan	12
Migrating virtual machines from the CLI	12
Using Migration Analytics	16
About the Migration Analytics report	16
Creating a Migration Analytics report	16
Troubleshooting	18
Collecting logs - to do	18
Known issues - in progress	18
Unsupported VMware features	18

Draft documentation

This draft document is a work in progress.



1. It might not be complete.
2. It might be not accurate.
3. It might break your environment.

Forklift enables you to migrate a large number of virtual machines from VMware vSphere to KubeVirt.

You can download the VMware provider data collected by Forklift and upload it to the Migration Analytics service on cloud.redhat.com for a detailed workload analysis report and migration recommendations.

About Forklift

IN PROGRESS

Placeholder for diagram

Custom resources

Forklift creates the following custom resources (CRs):

- **Provider:**
 - Created when you add a source or target provider
 - Stores provider attributes in order to connect to and interact with the providers
- **Plan:**
 - Created when you create a migration plan
 - Contains a list of VMs that are migrated together with the same migration parameters
 - Selects VMs to fit the available migration resources until all VMs are migrated
- **Mapping:** Created when you map source and target networks or source and target storage
- **Migration:**
 - Created when you run a migration plan
 - Aggregates the migration progress

If a virtual machine fails to migrate, you can run a new migration with the same migration plan. Only the virtual machines that have not migrated will be migrated.

Services

Forklift creates the following services:

- **Provider Inventory:**
 - Connects to the source and target providers
 - Maintains a local inventory for mappings and plans
 - Separate provider inventory pods for each provider enable scalability
- **Validation:**
 - Validates the suitability of a VM for migration to KubeVirt by applying rules
 - Triggered by the Provider Inventory service whenever a VM configuration change is detected
 - Stores the VM validation status
- **User Interface:**
 - Enables you to manipulate the Forklift CRs
 - Indicates the state of the CRs
 - Displays the progress of a migration

- Controller: Implements the logic to reconcile the CRs with the state defined by the user

For example, when you create a Plan, the controllers validate it and add a status label. If a Plan fails validation, its status is **Not ready** and the Plan cannot be used to perform a migration.

- Aggregates the status of related CRs to provide a holistic and consistent view of the state of a migration

Planning the migration

TBD

Installing Forklift

You can install the Forklift with the KubeVirt web console or from the CLI.

Then you create a VMware Virtual Disk Development Kit (VDDK) image and add it to a config map. Forklift uses VDDK to copy the source disks.

Installing Forklift with the web console

You can install the Forklift with the KubeVirt web console.

Prerequisites

- `cluster-admin` privileges

Procedure

1. In the KubeVirt web console, click **Operators** → **OperatorHub**.
2. Click the Operator and then click **Install**.
3. On the **Install Operator** page, click **Install**.

On the **Installed Operators** page, the **Forklift Operator** appears in the **openshift-migration** project with the status **Succeeded**.

4. Click **Forklift Operator**.
5. Under **Provided APIs**, locate the **VirtController** tile, and click **Create Instance**.
6. Click **Create**.
7. Click **Workloads** → **Pods** to verify that the `ProviderInventory`, `Validation`, `UserInterface`, and `Controller` pods are running.

Installing Forklift from the CLI

You can install the Forklift from the CLI.

Prerequisites

- [OpenShift CLI](#) installed
- `cluster-admin` privileges

Procedure

1. Log in to the OpenShift Kubernetes Engine cluster and create an `openshift-migration` project:

```
$ cat << EOF | oc apply -f -
---
apiVersion: project.openshift.io/v1
kind: Project
metadata:
  name: openshift-migration
EOF
```

2. Create an **OperatorGroup** object called **migration**:

```
$ cat << EOF | oc apply -f -
---
apiVersion: operators.coreos.com/v1
kind: OperatorGroup
metadata:
  name: migration
  namespace: openshift-migration
spec:
  targetNamespaces:
    - openshift-migration
EOF
```

3. Create a **Subscription** object for the **virt-operator**:

```
$ cat << EOF | oc apply -f -
---
apiVersion: operators.coreos.com/v1alpha1
kind: Subscription
metadata:
  name: virt-operator
  namespace: openshift-migration
spec:
  channel: development
  installPlanApproval: Automatic
  name: virt-operator
  source: konveyor-for-vms
  sourceNamespace: openshift-marketplace
  startingCSV: "konveyor-virt-operator.v99.0.0"
EOF
```

4. Create a **VirtController** object:


```
$ cat << EOF | oc apply -f -
---
apiVersion: virt.konveyor.io/v1alpha1
kind: VirtController
metadata:
  name: virt-controller
  namespace: openshift-migration
spec:
  olm_managed: true
EOF
```

5. View the resources created by the **VirtController** object:

```
$ oc get pods -n openshift-migration
```

Example output

TBD

Adding a VDDK image to Forklift

You can create a VDDK image and store it in a private image registry. Then, you can add the image to the **v2v-vmware** config map.



Storing the VDDK image in a public registry might violate the terms of the VMware license.

Prerequisites

- An internal [OpenShift Kubernetes Engine image registry](#) or a secure [external registry](#)
- Access to the image registry from your KubeVirt environment
- Podman installed

Procedure

1. Create and navigate to a temporary directory:

```
$ mkdir /tmp/<dir_name> && cd /tmp/<dir_name>
```

2. In a browser, navigate to [VMware code](#) and click **SDKs**.
3. In the **Compute Virtualization** section, click **Virtual Disk Development Kit (VDDK)**.
4. Select the VDDK version that corresponds to your VMware vSphere version, for example, VDDK 7.0 for vSphere 7.0.

5. Click **Download** and save the VDDK archive in the temporary directory.

6. Extract the VDDK archive:

```
$ tar -xzf VMware-vix-disklib-<version>.x86_64.tar.gz
```

7. Create a **Dockerfile**:

```
$ cat > Dockerfile <<EOF
FROM busybox:latest
COPY vmware-vix-disklib-distrib /vmware-vix-disklib-distrib
RUN mkdir -p /opt
ENTRYPOINT ["cp", "-r", "/vmware-vix-disklib-distrib", "/opt"]
EOF
```

8. Build the VDDK image:

```
$ podman build . -t <registry_route_or_server_path>/vddk:<tag> ①
```

① Specify your image registry:

- For an internal OpenShift Kubernetes Engine registry, specify the internal registry route, for example, `image-registry.openshift-image-registry.svc:5000/openshift/vddk:<tag>`.
- For an external registry, specify the server name, path, and tag, for example, `server.example.com:5000/vddk:<tag>`.

9. Push the VDDK image to the registry:

```
$ podman push <registry_route_or_server_path>/vddk:<tag>
```

10. Edit the **v2v-vmware** config map in the **openshift-cnv** project:

```
$ oc edit configmap v2v-vmware -n openshift-cnv
```

11. Add the **vddk-init-image** parameter to the **data** stanza:

```
...
data:
  vddk-init-image: <registry_route_or_server_path>/vddk:<tag>
```

Migrating virtual machines

Migrating virtual machines with the web console

You can add providers, create network and storage mappings, and create and run a migration plan in the web console.

Adding providers

You must add the KubeVirt and VMware providers in the web console.

Adding the KubeVirt provider

You can add the KubeVirt provider in the {mtv-short} web console.

Prerequisites

- **VirtController** service account token

Procedure

1. In the web console, navigate to **Providers** and click **Add provider**.
2. Select **KubeVirt** from the **Type** list.
3. Fill in the following fields:
 - **Cluster name**: OpenShift cluster name to display in the list of providers
 - **URL**: OpenShift cluster API endpoint
 - **Service account token**: VirtController service account token
4. Click **Check connection** to verify the credentials.
5. Click **Add** to add and save the provider.

The provider appears in the list of providers.

Adding the VMware provider

You can add the VMware provider in the {mtv-short} web console.

Prerequisites

- Admin privileges

Procedure

1. In the web console, navigate to **Providers** and click **Add provider**.
2. Select **VMware** from the **Type** list.
3. Fill in the following fields:
 - **Name**: vCenter name to display in the list of providers
 - **Hostname**: vCenter host name or IP address

- **Username:** vCenter admin user name, for example, `administrator@vsphere.local`
 - **Password:** vCenter password
4. Click **Check connection** to verify the credentials.
 5. Click **Add** to add and save the provider.

The provider appears in the list of providers.

Creating mappings

You must create mappings to map the source and target networks and storage.

Creating a network mapping

You can create a network mapping to map VMware networks to KubeVirt networks.



You cannot map an opaque network to a KubeVirt network.

Procedure

1. In the web console, navigate to **Mappings** → **Network**.
2. Click **Create mapping**.
3. Select a **Source provider** and a **Target provider**.
4. Select a network from the **Source networks** list and from the **Target networks** list.
5. Click **Add** to create additional network mappings or to map multiple source networks to a single target network.
6. Click **Create**.

The mapping is displayed in the **Network mappings** list.

Creating a storage mapping

You can create a storage mapping to map VMware data stores to KubeVirt storage classes.

Prerequisites

- The local and shared persistent storage must support VM migration.

Procedure

1. In the web console, navigate to **Mappings** → **Storage**.
2. Click **Create mapping**.
3. Select a **Source provider** and a **Target provider**.
4. Select a data store from the **Source datastores** list and a storage class from the **Target storage classes** list.
5. Click **Add** to create additional storage mappings or to map multiple data stores to a single storage class.

6. Click **Create**.

The mapping is displayed in the **Storage mappings** list.

Creating a migration plan

You can create a migration plan in the web console.

A migration plan allows you to group virtual machines that should be migrated together or with the same migration parameters, for example, a percentage of the members of a cluster or a complete application.

Prerequisites

- The migration network supports migration.
- The web console has source and target providers.

Procedure

1. In the web console, navigate to **Migration plans** and click **Create migration plan**.
2. Fill in the following fields in the **General settings** screen:
 - **Plan name**: Plan name that is displayed in the web console.
 - **Plan description**: Optional. Brief description of the migration plan.
 - **Source provider**: Select a VMware provider.
 - **Target provider**: Select the KubeVirt provider.
3. Click **Next**.
4. Filter the VMs that you want to migrate by selecting data centers, clusters, or folders.
5. Select the VMs to migrate. You can click each VM to view its migration risk assessment.
6. Select an existing storage mapping or create a new storage mapping.
7. If you want to create a new storage mapping:
 - a. Select a data store from the **Source datastores** list and a storage class from the **Target storage classes** list.
 - b. Click **Add** to create additional storage mappings or to map multiple data stores to a single storage class.
8. Click **Next**.
9. Select an existing network mapping or create a new network mapping.
10. If you want to create a new network mapping:
 - a. Select a network from the **Source networks** list and a network from the **Target networks** list.
 - b. Click **Add** to create additional network mappings or to map multiple source networks to a single target network.
11. Click **Next**.

12. Review your migration plan and click **Finish**.

The migration plan is saved in the migration plan list.

Running a migration plan

You can run a migration plan and view its progress in the web console.

Prerequisites

- The VMware VMs must be powered off.

Procedure

1. In the web console, navigate to **Migration plans**.

The **Migration plans** list displays the source and target providers, the number of VMs being migrated, the status of the plan, and a high-level progress bar that shows how many VMs have been migrated.

2. Click **Start** beside a migration plan with a **Ready** status to run the plan.
3. To view the **Migration Details by VM** screen, click the name of a migration plan.

This screen displays the migration start and end time, the amount of data copied, and a progress pipeline for each VM being migrated.

4. Expand a VM to view the migration steps, elapsed time of each step, and the state.

Migrating virtual machines from the CLI

You can migrate virtual machines from the command line (CLI) by creating the following custom resources:

- **Secret:** Contains the VMware vSphere credentials
- **Provider:** Specifies the VMware provider
- **Plan:** Specifies the source and target clusters, network mappings, data store mappings, and VMs to migrate
- **Migration:** Runs the Plan

If a Migration does not complete, you can create a new Migration CR, referencing the same Plan, to migrate the remaining VMs.

Prerequisites

- [OpenShift CLI](#) installed
- Admin privileges

Procedure

1. Get the SSL thumbprint of the vCenter:

```
$ openssl s_client \
  -connect <vcenter.example.com>:443 \ ①
  < /dev/null 2>/dev/null \
  | openssl x509 -fingerprint -noout -in /dev/stdin \
  | cut -d '=' -f 2
```

① Specify the vCenter host name.

2. Create a Secret object with the VMware credentials:

```
$ cat < EOF | oc apply -f -
---
apiVersion: v1
kind: Secret
metadata:
  name: vmware-secret
  namespace: openshift-migration
type: Opaque
stringData:
  user: administrator@vsphere.local
  password: <password> ①
  thumbprint: <01:23:45:67:89:AB:CD:EF:01:23:45:67:89:AB:CD:EF:01:23:45:67> ②
EOF
```

① Specify the password.

② Specify the thumbprint.

3. Create a Provider object for VMware vSphere:

```
$ cat < EOF | oc apply -f -
---
apiVersion: virt.konveyor.io/v1alpha1
kind: Provider
metadata:
  name: vmware-provider
  namespace: openshift-migration
spec:
  type: vsphere
  url: https://<vcenter.example.com>/sdk ①
  secret:
    name: vmware-secret ②
    namespace: openshift-migration
EOF
```

① Specify the vSphere API end point.

② Specify the Secret.

4. Create a Plan object for the migration:

```

$ cat < EOF | oc apply -f -
---
apiVersion: virt.konveyor.io/v1alpha1
kind: Plan
metadata:
  name: test-migration-plan
  namespace: openshift-migration
spec:
  provider:
    source:
      name: vmware-provider
      namespace: openshift-migration
    destination:
      name: host ①
      namespace: openshift-migration
  map:
    networks:
      - source:
          id: <network-31> ②
        destination:
          type: pod
          name: pod
          namespace: openshift-migration
      - source:
          id: <network-723>
        destination:
          type: pod
          name: pod
          namespace: openshift-migration
    datastores:
      - source:
          id: <datastore-30> ③
        destination:
          storageClass: standard
      - source:
          id: <datastore-35>
        destination:
          storageClass: standard
    vms:
      - id: <vm-1630> ④
      - id: <vm-2841>
EOF

```

① The OpenShift provider is added automatically as **host** when the Operator is deployed.

② Specify a source network.

③ Specify a source data store.

④ Specify the IDs of the VMs to migrate.

5. Create a Migration CR to run the Plan CR:


```
$ cat < EOF | oc apply -f -
---
apiVersion: virt.konveyor.io/v1alpha1
kind: Migration
metadata:
  name: migration-test-001
  namespace: openshift-migration
spec:
  plan:
    name: test-migration-plan
    namespace: openshift-migration
EOF
```

The Migration CR creates a `VirtualMachineImport` object for each VM being migrated. You can monitor the progress of the migration by viewing the `VirtualMachineImport` pods in the `openshift-migration` namespace.

Using Migration Analytics

You can use the Migration Analytics service at cloud.redhat.com to analyze your VMware workloads and to provide migration recommendations.

Migration Analytics generates a report that you can view in your browser or download as a CSV file.

About the Migration Analytics report

The Migration Analytics report contains a VM migration summary and a VM migration inventory.

VM migration summary

The VM migration summary displays the following information:

- **Summary:** Providers, hosts, clusters, and VMs. Percentage of VMs that have no issues for migrating to KubeVirt.
- **Migration risk assessment:** Suitability of a VM for migration. The assessment is determined by rules that detect certain configurations or operating systems.
- **Rules summary (issues that impact migration risk):** Information about the rules.
- **Operating systems identified:** Operating systems that can be migrated to Red Hat Enterprise Linux.

VM migration inventory

The VM migration inventory displays a list of virtual machines with detailed information.

You can click a VM to view its migration risk assessment.

You can filter the VM migration inventory and export it as a CSV file.

Creating a Migration Analytics report

You can download VMware provider data in the Forklift web console and upload it to the Migration Analytics service on cloud.redhat.com to create a Migration Analytics report.

Procedure

1. In the web console, navigate to **Providers**.
2. In the **VMware** tab, select a VMware provider and click **Download data**.
3. Launch a browser and navigate to cloud.redhat.com.
4. In the **Migration Services** tile, click **Migration Analytics**.
5. Click **Get Started** if you are creating a report for the first time or navigate to **Payload Files** and click **Upload file**.
6. Browse to the VMware provider data file and click **Open**.
7. Select the data file and click **Create report**.

8. Select **OpenShift Virtualization** as the **Preferred target**.
9. Enter the **Report name** and optional **Report description**.
10. Click **OK**.

The generated report appears in the **Reports** list.

Filtering a Migration Analytics report

You can filter the results of a Migration Analytics report to display virtual machines according to provider, cluster, migration effort, and other parameters.

Procedure

1. Click **VM migration inventory** in the Migration Analytics report.
2. Click **Filter** and select a filter type, for example, **Provider**, **OS type**, or **Rules**.
3. Click **Filter by** <filter_type> and select one or more options.
4. If you are filtering by **VM name**, enter the text in the search field.

The filtered virtual machines are displayed.

Exporting a Migration Analytics report

You can export a complete or filtered Migration Analytics report in CSV format.

Procedure

1. Click **VM migration inventory** in the Migration Analytics report.
2. Optional: Filter the report.
3. Click **Export as CSV**.

The Migration Analytics report is downloaded to your local machine as a CSV file.

Troubleshooting

Collecting logs - to do

TBD

Known issues - in progress

This section describes known issues and steps to resolve them.

Unsupported VMware features

The following VMware features are not supported by KubeVirt:

- VMware Site Recovery Manager (SRM). You can migrate the VM but SRM will not be enabled.
- vSphere Distributed Power Management (DPM). You can migrate the VM but DPM will not be enabled.
- VMDK with SCSI-3 persistent reservation (PR). You can migrate the VM but SCSI-3 PR will not be enabled. NEED INFO