

Tanzu Kubernetes Grid Integrated (formerly Enterprise PKS) integration with vRealize Automation Cloud

This document describes the integration between TKGI & vRAC Cloud Assembly using Cloud Proxy

[Introduction](#)

[High level design](#)

[Overview](#)

[Scope](#)

[Not in Scope](#)

[Design topology](#)

[High level steps](#)

[Prerequisites](#)

[General steps](#)

[Creating/Adding Kubernetes cluster](#)

[Creating cluster](#)

[Using DEPLOY option](#)

[Using Blueprint](#)

[Adding cluster](#)

[Add Existing cluster](#)

[Add External cluster](#)

[Detailed steps](#)

[Prerequisites](#)

[General steps](#)

[Creating/Adding Kubernetes cluster](#)

[Creating cluster](#)

[Using DEPLOY option](#)

[Using Blueprint](#)

[Adding cluster](#)

[Add Existing cluster](#)

[Add External cluster](#)

[Appendix](#)

[Transcript](#)

[Reference Links](#)

Introduction:

vRealize Automation Cloud provides different services as VMware Cloud Assembly, VMware Service Broker, VMware Code Stream.

Cloud Assembly helps construct workload specifications as Blueprints, which we can make available to the business groups and deploy them to the cloud resources.

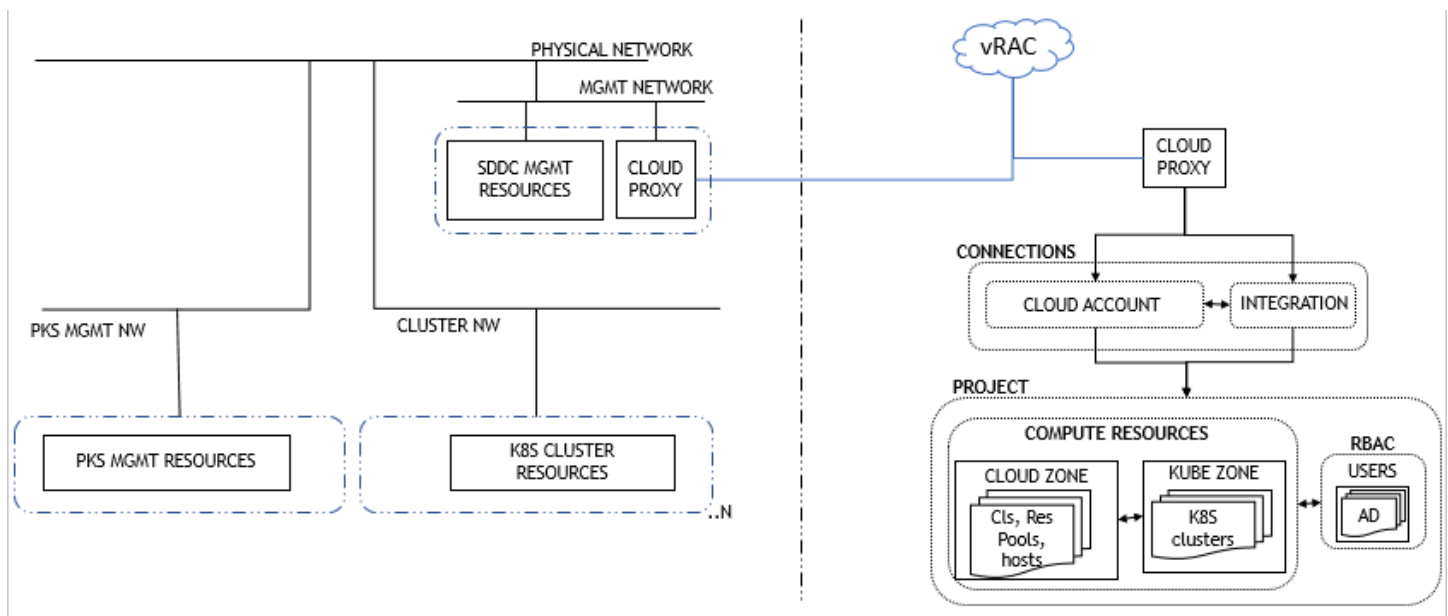
Service Broker provide a catalog of templates and actions to VMware Service Broker consumers.

Code Stream supports DevOps life cycle with pipelines, endpoints, and dashboards using VMware Code Stream.

In this document we'll see the different options Cloud Assembly offers for managing and deploying Kubernetes resources. We'll focus on integration with Pivotal Container Service (PKS) with Cloud Assembly to configure, manage and deploy Kubernetes resources.

We will also understand how to integrate external Kubernetes clusters in Cloud Assembly.

High level design: (Sample topology of integration between vRAC & TKGI)



Brief description of high level design:

The diagram above shows high level integration between TKGI & vRAC. It points out the main components involved in the process such as cloud proxy(for communication between target environment & vRAC), connections as cloud accounts(with SDDC management components) & integrations (with PKS endpoint), configuring compute resources as cloud zone(with clusters, resource pools, hosts) & kubernetes zone(with kubernetes clusters), assigning users & roles.

Overview:

This document describes integration between TKGI and vRAC.

We have assumed that the TKGI configuration part is understood by the partners, so primary focus of this document will be on the integration between vRAC & TKGI.

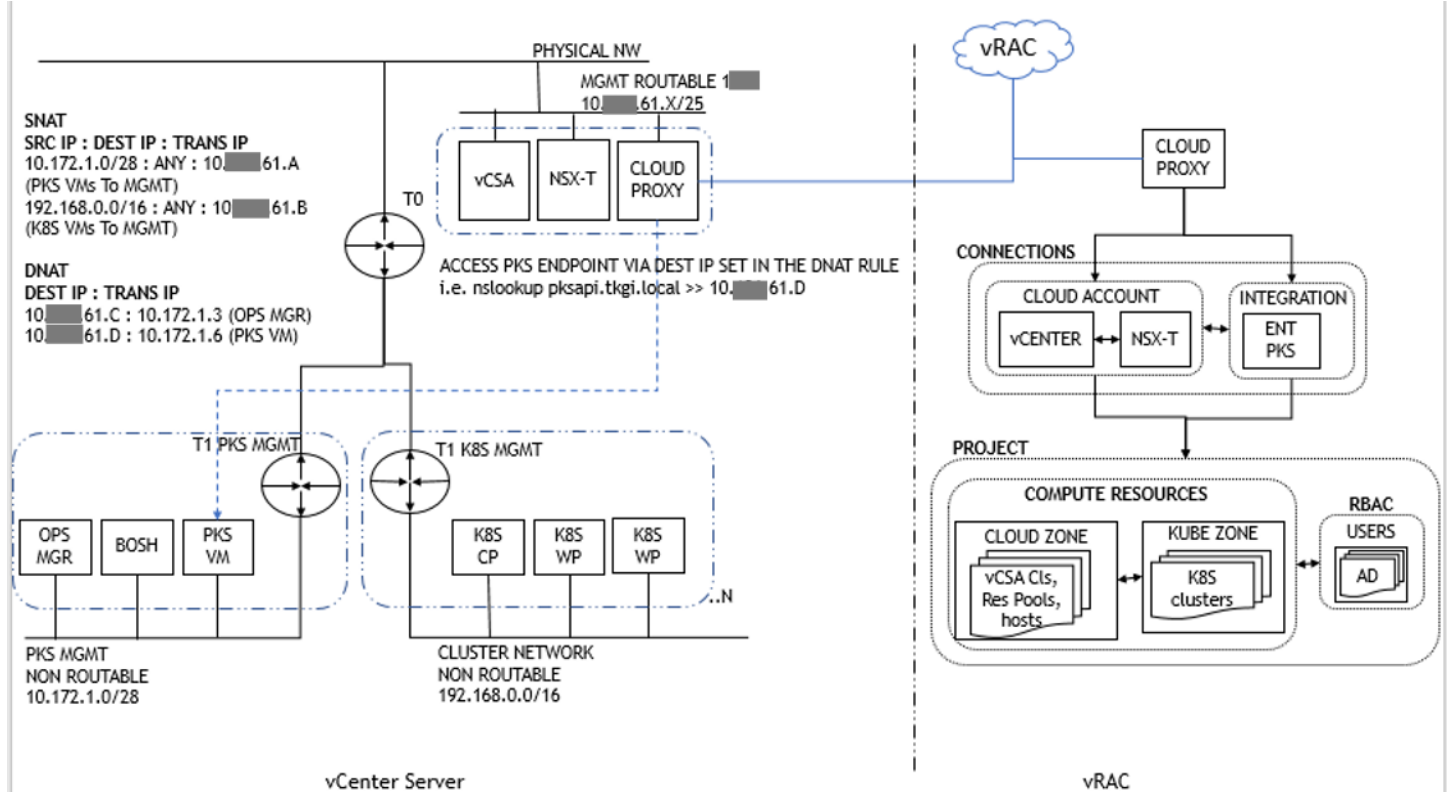
Scope:

- > This is a test setup deployment with minimal required configuration for the demo purpose.
- > Integration with Cloud Accounts (vCenter, NSX-T)
- > Integration with Enterprise PKS endpoint
- > Deployment of TKGI cluster with single control plane.
- > Deployment of cluster using 'DEPLOY' option
- > Deployment of cluster using Blueprint.
- > Adding existing cluster
- > Adding external cluster

Not in scope:

- > TKGI cluster creation with multiple control plane nodes
- > Explore usage of services as Code Stream, Service Broker
- > Any other operation not listed in Scope section above

Design topology (Please maximize to get a clear view of the diagram)



Brief description of topology:

Above topology describes integration between vRAC & TKGI(EntPKS) endpoint. The TKGI lab is configured with NAT topology. Cloud Proxy needs to be installed on the target environment. A cloud proxy is a remote virtual appliance that is created in a target vCenter by deploying the supplied cloud proxy OVA. The cloud proxy allows data collection and other communication between a specific cloud account in Cloud Assembly and a specific on-premises endpoint in vCenter. After cloud proxy is installed, it can be used when creating and associating a cloud account with an on-premises endpoint (i.e. vCenter, NSX-T etc). In a similar way cloud proxy can be used to communicate with some integrations, in our case EntPKS. The cloud proxy deployed on a target vCenter Server manages the information between Cloud Assembly and the integration service or application.

High level steps:

Below will be the high-level steps:

Prerequisites:

TKGI(EntPKS) lab configured with NAT topology

General steps

Login to VMware Cloud Services

Click on VMware Cloud Assembly

Create Cloud Proxy for the target environment

Create Cloud Accounts (vCenter, NSX-T) for the target environment

Create Integration for the (TKGI(EntPKS)) for the target environment

Create Cloud Zone

Create Kubernetes Zone

Create Project

 # Add Cloud Zone, Kubernetes Zone in the project

 # Add Users in the Project

Create cluster deployment blueprint & assign it to the project

Creating/Adding Kubernetes cluster

Creating cluster

Using DEPLOY option

> Go to Infrastructure > Resources > Kubernetes > Clusters > DEPLOY > Enter the necessary values > Click DEPLOY

Using Blueprint

> Go to Design > Select blueprint > Enter the necessary values > Click DEPLOY

Adding cluster

Add Existing cluster

> Go to Infrastructure > Resources > Kubernetes > Clusters > ADD EXISTING > Enter the necessary values > Click ADD

Add External cluster

> Go to Infrastructure > Resources > Kubernetes > Clusters > ADD EXTERNAL > Enter the necessary values > Click ADD

Detailed steps

Below will the detailed steps:

Prerequisites:

TKGI(EntPKS) lab configured with NAT topology

General steps:

Login to VMware Cloud Services

<#> Click on VMware Cloud Assembly

<#> Create Cloud Proxy for the target environment

- > Click Infrastructure > Connections > Cloud Proxies > NEW > Download OVA > Copy the One Time Key
- > Import OVA file in vCenter instance > Select a name & folder > Select a compute resource
- > Review details > Accept license agreements > Select Storage > Select Networks (Management network in SDDC) >
- > Customize template by adding the One Time Key, root user password > cloud proxy display name > Network properties for the cloud proxy appliance
- > On Ready to complete page review the details & click FINISH
- > Power On the cloud proxy & wait for some time
- > Check the status is Active in the Cloud assembly > Infrastructure > Connections > Cloud Proxies for the proxy created

Create Cloud Accounts (vCenter, NSX-T) for the target environment

[>>](#) Create vCenter Cloud Account

- > Click Infrastructure > Connections > Cloud Accounts > ADD CLOUD ACCOUNT > Select vCenter >
- > In New Cloud Account page > Enter Name for the cloud account > Enter vCenter IP address/FQDN > In Cloud Proxy drop down Select the cloud proxy created for the target environment > Enter vCenter username & password >
- Click VALIDATE > After successful validation Click ADD > Check the vCenter Cloud Account is displayed in the Cloud Accounts

[>>](#) Create NSX-T Cloud Account

- > Click Infrastructure > Connections > Cloud Accounts > ADD CLOUD ACCOUNT > Select NSX-T >
- > In New Cloud Account page > Enter Name for the cloud account > Enter NSX-T IP address/FQDN > In Cloud Proxy drop down Select the cloud proxy created for the target environment > Enter NSX-T username & password >
- Click VALIDATE > After successful validation Click ADD > Check the NSX-T Cloud Account is displayed in the Cloud Accounts

Create Integration for the (TKGI(EntPKS)) target environment

> Click Infrastructure > Connections > Integrations > ADD INTEGRATION > Select Integration Type as VMware Enterprise PKS >

> In New Integration page > Enter Name for the integration > Enter PKS Endpoint IP address/FQDN > In Location dropdown Select Private Cloud > In Cloud Proxy drop down Select the cloud proxy created for the target environment > Enter username & password to access the PKS Endpoint > Enter the CA certificate (i.e. Certificate to secure the PKS API from Enterprise PKS tile) >

Click VALIDATE > After successful validation Click ADD > Check the Enterprise PKS integration is displayed in the Integrations

Create Cloud Zone

> Click Infrastructure > Configure > Cloud Zones > NEW CLOUD ZONE >

> In New Cloud Zone page > In Account / region section search for the vCenter Cloud account created & select the same > Enter name for the Cloud Zone > In the Compute tab select the Compute resources(of type clusters, resource pools, host) to be applied to the Cloud Zone > Click on CREATE > Check the Cloud Zone is displayed in the Cloud Zones

Create Kubernetes Zone

> Click Infrastructure > Configure > Kubernetes Zones > NEW KUBERNETES ZONE >

> In New Kubernetes Zone page > In Account section search for the Enterprise PKS integration created & select the same > Enter name for the Kubernetes Zone > In the On-demand tab select the Deployment Plan as required > Enable the Allow Provisioning for the plan > Click on SAVE > Check the Kubernetes Zone is displayed in the Kubernetes Zones

Create Project

> Click Infrastructure > Configure > Projects > NEW PROJECT >

> In New Project page > Enter the Name of the project >

> In Users tab add the users & define their role >

> In Provisioning tab Click ADD CLOUD ZONE > Search & Select the Cloud Zone created & Click ADD >

> In Kubernetes Provisioning tab > Click ADD ZONE > Search & Select the Kubernetes Zone & Click SAVE

> Click CREATE > Check the Project is displayed in the Projects

Create cluster deployment blueprint & assign it to the project

> Click Design > Blueprints > NEW > Enter the name for the blueprint > Search & Select the Project to which blueprint will be tagged > Select Blueprint sharing in Service Broker as required > Drag K8S Cluster resource type in Design canvas > Modify the code as required > Click Version > mention the Version > Select Release this version to catalog > Click CREATE > Check the Blueprint is displayed in the Blueprints

Creating/Adding Kubernetes cluster

Creating cluster

Using DEPLOY option

> Click Infrastructure > Resources > Kubernetes > Clusters > DEPLOY > In Account section Search & Select the EntPKS endpoint > Enter the cluster name > Select sharing as required - in case of sharing at Project level then select the Project's name > Enter the Master hostname(e.g. clsname.tkg.local) > Select the Plan > Enter the Worker nodes count (e.g. 1) to be deployed > Click DEPLOY > Monitor the deployment > Check for the cluster created in Infrastructure > Resources > Kubernetes > Clusters

Using Blueprint

> Click Design > Blueprints > Select blueprint > Click DEPLOY > Enter Deployment name > Select Blueprint version > Click Next > Click DEPLOY > Enter the Hostname(e.g. clsname.tkg.local) > In Size Enter the Worker nodes count (e.g. 1) to be deployed > Select the Plan > Click DEPLOY > Monitor the deployment > Check for the cluster created in Infrastructure > Resources > Kubernetes > Clusters

Adding cluster

Add Existing cluster

> Click Infrastructure > Resources > Kubernetes > Clusters > ADD EXISTING > In Account Search & Select the EntPKS integration endpoint > In Cluster select the cluster to be added > Select Connect by Master IP > Select sharing as required - in case of sharing at Project level then select the Project's name > Click ADD > Check for the cluster added in Infrastructure > Resources > Kubernetes > Clusters

Add External cluster

> Click Infrastructure > Resources > Kubernetes > Clusters > ADD EXTERNAL > Enter the name of cluster > Select sharing as required - in case of sharing at Project level then select the Project's name > In Address enter the Server Address for the cluster > In CA certificate enter the clusters CA certificate i.e. certificate-authority-data > Select location as Private Cloud > Select the Cloud proxy created for the target environment >

> In case Credentials type is Certificate > Enter Public certificate i.e. client-certificate-data & Private certificate i.e. client-key-data for the cluster

> In case Credentials type is Bearer token > Enter the Bearer token

Click VALIDATE > Click ADD > > Check for the cluster added in Infrastructure > Resources > Kubernetes > Clusters

Appendix:

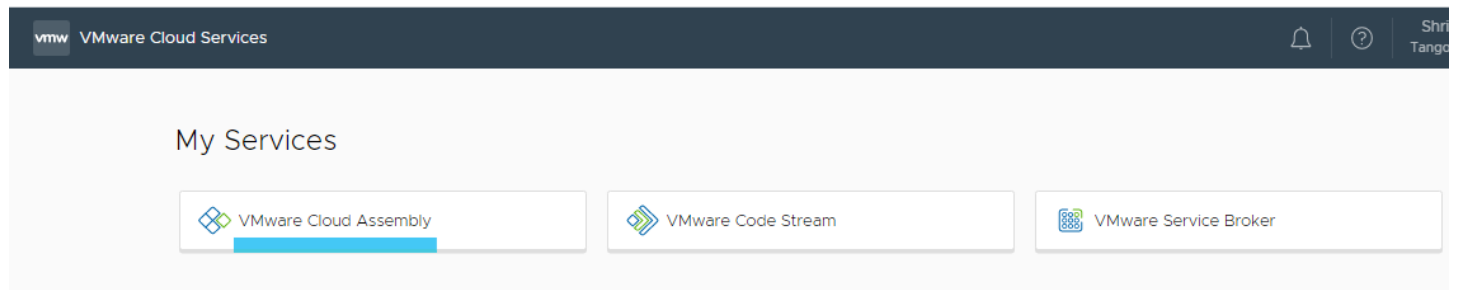
This section contains the detailed steps with screen prints, blueprint etc.

It also has References section where important links are mentioned.

Transcript:

Login to VMware Cloud Services

Click on VMware Cloud Assembly



Create Cloud Proxy for the target environment

> Click Infrastructure > Connections > Cloud Proxies > NEW > Download OVA > Copy the One Time Key

Install Cloud Proxy

1 Download the Cloud Proxy ova file.

The time required for the download depends on your network.

DOWNLOAD OVA

<https://ci-data-collector.s3.amazonaws.com/VMware-Cloud-Services-Data-Collector.ova>

COPY LINK

2 Import the .ova file to the vCenter Server and start the installation. ⓘ

3 When asked for the key, copy and use the following key:

```
eyJyZWdpc3RyYXRpb25VcmwiOiJodHRwczovL2FwaS5tZ210LmNsb3VkLnZtd2FyZS5jb20vN2M2YjM2ZWYtN
zgyOSO0NmE4LTk2MTgtNWlyOTRmOGMxNjFliwib3RrjoiR0s0Ry1CMIZFLURWNzMTTdlGilslnRlbnFudElkjoil
3RlbnFudHMvb3JnYW5pemF0aW9uLzdjNmizNmVmLTc4MjktNDZhOC05NjE4LTViMjk0ZjhMTYxZSIsInByb3h

```

COPY

4 It takes a few minutes to detect your Cloud Proxy after it is deployed and powered up in vCenter.

Moving away from this page will not interrupt the detection.

DONE

- > Import OVA file in vCenter instance > Select a name & folder > Select a compute resource
- > Review details > Accept license agreements > Select Storage > Select Networks (Management network in SDDC) >
- > Customize template by adding the One Time Key, root user password > cloud proxy display name > Network properties for the cloud proxy appliance
- > On Ready to complete page review the details & click FINISH
- > Power On the cloud proxy & wait for some time
- > Check the status is Active in the Cloud assembly > Infrastructure > Connections > Cloud Proxies for the proxy created

tkgi-cloudproxy-ext	Active	10.196.61	Jun 12, 2020	Cloud Assembly - SDDC
---------------------	--------	-----------	--------------	-----------------------

Create Cloud Accounts (vCenter, NSX-T) for the target environment

>> Create vCenter Cloud Account

> Click Infrastructure > Connections > Cloud Accounts > ADD CLOUD ACCOUNT > Select vCenter >

> In New Cloud Account page > Enter Name for the cloud account > Enter vCenter IP address/FQDN > In Cloud Proxy drop down Select the cloud proxy created for the target environment > Enter vCenter username & password >

Click VALIDATE > After successful validation Click ADD > Check the vCenter Cloud Account is displayed in the Cloud Accounts

The screenshot displays the VMware Cloud Assembly interface. The top navigation bar includes 'Deployments', 'Design', 'Infrastructure' (selected), 'Extensibility', and 'Marketplace'. A left-hand sidebar lists various categories: Network Profiles, Storage Profiles, Pricing Cards, Tags, Resources (with a dropdown arrow), Compute, Networks, Security, Storage, Machines, Volumes, Kubernetes, Activity (with a dropdown arrow), Requests, Events Log, Connections (with a dropdown arrow), Cloud Accounts (highlighted in yellow), Integrations, and Cloud Proxies. The main content area shows the configuration for a cloud account named 'tkgi-cloudacc-vcenter'. At the top, it indicates 'Available for deployment' with a green checkmark and an 'UPDATE' button. The configuration fields include: Name (tkgi-cloudacc-vcenter), Description (empty text area), vCenter Server Credentials section with vCenter IP address / FQDN (tkgi-vc.tkgi.local), Cloud proxy (tkgi-cloudproxy-ext) with a '+ NEW CLOUD PROXY' button, Username (administrator@vsphere.local), and Password (empty). Below these fields is a 'VALIDATE' button and a notification box that says 'Validate credentials before making changes.' with a close button. The 'Configuration' section includes 'Allow provisioning to these datacenters' with a checked checkbox for 'tkgi-dc' and 'NSX cloud account' with a search icon and 'tkgi-cloudacc-nsxt'.

>> Create NSX-T Cloud Account

> Click Infrastructure > Connections > Cloud Accounts > ADD CLOUD ACCOUNT > Select NSX-T >

> In New Cloud Account page > Enter Name for the cloud account > Enter NSX-T IP address/FQDN > In Cloud Proxy drop down Select the cloud proxy created for the target environment > Enter NSX-T username & password >

Click VALIDATE > After successful validation Click ADD > Check the NSX-T Cloud Account is displayed in the Cloud Accounts

The screenshot displays the VMware Cloud Assembly interface. The top navigation bar includes 'Deployments', 'Design', 'Infrastructure' (highlighted), 'Extensibility', and 'Marketplace'. The left sidebar shows a menu with categories like 'Network Profiles', 'Storage Profiles', 'Pricing Cards', 'Tags', 'Resources', 'Compute', 'Networks', 'Security', 'Storage', 'Machines', 'Volumes', 'Kubernetes', 'Activity', 'Requests', 'Events Log', 'Connections', 'Cloud Accounts' (highlighted), 'Integrations', 'Cloud Proxies', and 'Onboarding'. The main content area shows the configuration for the integration 'tkgi-cloudacc-nsxt'. The 'Status' section indicates 'Data collection completed 10 minutes ago'. The 'Name' field is 'tkgi-cloudacc-nsxt'. The 'Description' field is empty. The 'NSX-T Credentials' section includes 'NSX-T IP address / FQDN' set to 'tkgi-nsxmgr-vip.tkgi.local' and 'Cloud proxy' set to 'tkgi-cloudproxy-ext'. There is a '+ NEW CLOUD PROXY' button. The 'Username' field is 'admin' and the 'Password' field is empty. A 'VALIDATE' button is present, along with a warning message: 'Validate credentials before making changes.' The 'Configuration' section shows 'vCenter cloud account' set to 'tkgi-cloudacc-vcenter'. The 'Capabilities' section is partially visible at the bottom.

Create Integration for the (TKGI(EntPKS)) target environment

> Click Infrastructure > Connections > Integrations > ADD INTEGRATION > Select Integration Type as VMware Enterprise PKS >

> In New Integration page > Enter Name for the integration > Enter PKS Endpoint IP address/FQDN > In Location dropdown Select Private Cloud > In Cloud Proxy drop down Select the cloud proxy created for the target environment > Enter username & password to access the PKS Endpoint > Enter the CA certificate (i.e. Certificate to secure the PKS API from Enterprise PKS tile) >

Click VALIDATE > After successful validation Click ADD > Check the Enterprise PKS integration is displayed in the Integrations

The screenshot displays the VMware Cloud Assembly interface. The top navigation bar includes 'Deployments', 'Design', 'Infrastructure' (highlighted), 'Extensibility', and 'Marketplace'. A left-hand sidebar lists various categories: Network Profiles, Storage Profiles, Pricing Cards, Tags, Resources (with a dropdown arrow), Compute, Networks, Security, Storage, Machines, Volumes, Kubernetes, Activity (with a dropdown arrow), Requests, Events Log, Connections (with a dropdown arrow), Cloud Accounts, Integrations (highlighted), and Cloud Proxies. The main content area shows the configuration for a Cloud Proxy named 'tkgilab1'. The configuration includes:

- Name:** tkgilab1
- Description:** (Empty text area)
- PKS Endpoint Credentials:**
 - IP address / FQDN:** pksapi.tkgi.local
 - Configure ports manually:** (Toggle switch is off)
 - Location:** Private Cloud
 - Cloud proxy:** tkgi-cloudproxy-ext
 - + NEW CLOUD PROXY** (Button)
- Username:** pksadmin
- Password:** (Empty text area)
- CA certificate:** -----BEGIN CERTIFICATE-----
MIIDeTCCAmGgAwIBAgIUJmk9CPfoahj1aGR8FvN9utfmGcwDQYJKoZIhvcN
AQEL

At the bottom, there is a 'VALIDATE' button and a message: 'Validate credentials before making changes.' with a close button (X).

Create Cloud Zone

> Click Infrastructure > Configure > Cloud Zones > NEW CLOUD ZONE >

> In New Cloud Zone page > In Account / region section search for the vCenter Cloud account created & select the same > Enter name for the Cloud Zone > In the Compute tab select the Compute resources(of type clusters, resource pools, host) to be applied to the Cloud Zone > Click on CREATE > Check the Cloud Zone is displayed in the Cloud Zones

The screenshot displays the VMware Cloud Assembly interface. At the top, the 'vmw Cloud Assembly' logo is visible. Below it, navigation tabs include 'Deployments', 'Design', 'Infrastructure' (which is selected), 'Extensibility', and 'Marketplace'. A left-hand sidebar contains a menu with categories: 'Configure' (with sub-items: Projects, Cloud Zones, Kubernetes Zones, Flavor Mappings, Image Mappings, Network Profiles, Storage Profiles, Pricing Cards, Tags), 'Resources' (with sub-items: Compute, Networks, Security, Storage, Machines, Volumes, Kubernetes), and 'Activity'. The 'Cloud Zones' menu item is highlighted in blue. The main content area shows the configuration page for a cloud zone named 'tkgi-cloudacc-vcenter / tkgi-dc', with a 'DELETE' link. The page has three tabs: 'Summary' (selected), 'Compute', and 'Projects'. A descriptive text states: 'A cloud zone defines a set of compute resources that can be used for provisioning.' The configuration fields are: 'Account / region' set to 'tkgi-cloudacc-vcenter / tkgi-dc'; 'Name' set to 'tkgi-cloudacc-vcenter / tkgi-dc'; 'Description' with an empty text box; 'Placement policy' set to 'DEFAULT'; and 'Folder' with a search input 'Select folder'. Below these fields is a 'Capabilities' section with the text: 'Capability tags are effectively applied to all compute resources in this cloud zone, but only in the context of this cloud zone.' At the bottom of this section is a 'Capability tags' input field with the placeholder 'Enter capability tags'. At the very bottom of the configuration area are two buttons: 'SAVE' and 'CANCEL'.

The screenshot shows the VMware Cloud Assembly interface. The left sidebar is expanded to 'Configure' > 'Cloud Zones'. The main content area is titled 'tkgi-cloudacc-vcenter / tkgi-dc' and has tabs for 'Summary', 'Compute', and 'Projects'. The 'Compute' tab is active, displaying a list of compute resources. A filter tag input field is present above the table. The table has columns for Name, Account / Region, and Type.

Name	Account / Region	Type
<input type="checkbox"/> tkgi-comp	tkgi-cloudacc-vcenter / tkgi-dc	Cluster
<input type="checkbox"/> tkgi-comp / RP-INFRA	tkgi-cloudacc-vcenter / tkgi-dc	Resource Pool
<input type="checkbox"/> tkgi-comp / RP-MGMT-PKS	tkgi-cloudacc-vcenter / tkgi-dc	Resource Pool
<input type="checkbox"/> tkgi-comp / RP-PKS-AZ-1	tkgi-cloudacc-vcenter / tkgi-dc	Resource Pool
<input type="checkbox"/> tkgi-comp / RP-PKS-AZ-2	tkgi-cloudacc-vcenter / tkgi-dc	Resource Pool
<input type="checkbox"/> w1-hs2-o2406.tkgi.local	tkgi-cloudacc-vcenter / tkgi-dc	Host

Buttons for 'SAVE' and 'CANCEL' are visible at the bottom of the table.

The screenshot shows the VMware Cloud Assembly interface. The left sidebar is expanded to 'Configure' > 'Cloud Zones'. The main content area is titled 'tkgi-cloudacc-vcenter / tkgi-dc' and has tabs for 'Summary', 'Compute', and 'Projects'. The 'Projects' tab is active, displaying a list of projects allowed to provision to this cloud zone. A table with columns for Name, Priority, Instances, Memory Limit (MB), CPU Limit, and Storage Limit (GB) is shown.

Name	Priority	Instances	Memory Limit (MB)	CPU Limit	Storage Limit (GB)
tkgi-project-test	0	Unlimited	Unlimited	Unlimited	Unlimited
tkgi-project-dev	0	Unlimited	Unlimited	Unlimited	Unlimited
tkgi-project-1	0	Unlimited	Unlimited	Unlimited	Unlimited

Buttons for 'SAVE' and 'CANCEL' are visible at the bottom of the table.

Create Kubernetes Zone

> Click Infrastructure > Configure > Kubernetes Zones > NEW KUBERNETES ZONE >

> In New Kubernetes Zone page > In Account section search for the Enterprise PKS integration created & select the same > Enter name for the Kubernetes Zone > In the On-demand tab select the Deployment Plan as required > Enable the Allow Provisioning for the plan > Click on SAVE > Check the Kubernetes Zone is displayed in the Kubernetes Zones

The screenshot shows the 'vmw Cloud Assembly' interface. The top navigation bar includes 'Deployments', 'Design', 'Infrastructure' (highlighted), 'Extensibility', and 'Marketplace'. A left sidebar contains a 'Configure' section with options like 'Projects', 'Cloud Zones', 'Kubernetes Zones' (highlighted), 'Flavor Mappings', 'Image Mappings', 'Network Profiles', 'Storage Profiles', 'Pricing Cards', and 'Tags'. Below this is a 'Resources' section with 'Compute' and 'Networks'. The main content area is titled 'tkgi-kube-zone' with a 'DELETE' button. It has tabs for 'Summary' (highlighted), 'On-demand', 'Clusters', and 'Projects'. The 'Summary' tab contains a description: 'A kubernetes zone defines a set of compute resources that can be used for provisioning of clusters and namespaces.' Below this are fields for 'Account' (tkgilab1), 'Name' (tkgi-kube-zone), and 'Description' (empty). A 'Capabilities' section follows, with a note: 'Capability tags are effectively applied to all compute resources in this Kubernetes zone, but only in the context of this Kubernetes zone.' There is a 'Capability tags' input field with the placeholder 'Enter capability tags'. At the bottom are 'SAVE' and 'CANCEL' buttons.

This screenshot shows the 'On-demand' tab for the 'tkgi-kube-zone' configuration. The navigation and sidebar are the same as in the previous screenshot. The main content area is titled 'tkgi-kube-zone' with a 'DELETE' button. The 'On-demand' tab is active, showing 'Cluster Deployment Plans'. A sub-section 'TAGS' is visible. Below it is a table with columns: 'Name', 'Priority', 'Description', 'Tags', and 'Allow Provisioning'. One entry is shown with 'Name' 'small', 'Priority' '0', and a description: 'Example. This plan will configure a lightweight kubernetes cluster. Not recommended for production workloads.' The 'Allow Provisioning' toggle for this entry is turned on. At the bottom are 'SAVE' and 'CANCEL' buttons.

Name	Priority	Description	Tags	Allow Provisioning
small	0	Example. This plan will configure a lightweight kubernetes cluster. Not recommended for production workloads.		<input checked="" type="checkbox"/>

The screenshot shows the VMware Cloud Assembly interface. The top navigation bar includes 'Deployments', 'Design', 'Infrastructure', 'Extensibility', and 'Marketplace'. The left sidebar shows a 'Configure' menu with options like 'Projects', 'Cloud Zones', 'Kubernetes Zones', 'Flavor Mappings', 'Image Mappings', 'Network Profiles', 'Storage Profiles', 'Pricing Cards', and 'Tags'. The main content area is titled 'tkgi-kube-zone' and has a 'DELETE' button. Below the title are tabs for 'Summary', 'On-demand', 'Clusters', and 'Projects'. The 'Projects' tab is active, showing a heading 'Projects that are allowed to use this kubernetes zone for provisioning.' and a table with columns 'Name' and 'Description'. The table contains one entry: 'tkgi-project-1'. At the bottom of the table are 'SAVE' and 'CANCEL' buttons.

Create Project

- > Click Infrastructure > Configure > Projects > NEW PROJECT >
- > In New Project page > Enter the Name of the project >
- > In Users tab add the users & define their role >
- > In Provisioning tab Click ADD CLOUD ZONE > Search & Select the Cloud Zone created & Click ADD >
- > In Kubernetes Provisioning tab > Click ADD ZONE > Search & Select the Kubernetes Zone & Click SAVE
- > Click CREATE > Check the Project is displayed in the Projects

vmw Cloud Assembly

Deployments Design **Infrastructure** Extensibility Marketplace

Configure

- Projects
- Cloud Zones
- Kubernetes Zones
- Flavor Mappings
- Image Mappings
- Network Profiles
- Storage Profiles
- Pricing Cards
- Tags

Resources

- Compute
- Networks
- Security
- Storage
- Machines
- Volumes
- Kubernetes

Activity

tkgi-project-1 DELETE

Summary Users Provisioning Kubernetes Provisioning Integrations

Name **tkgi-project-1**

Description

Overview

Administrators	3
Members	0
Viewers	0
Cloud zones	1
Blueprints	2
Deployments	10
K8s zones	1
Actions	0
Custom resources	0
Resource actions	0

SAVE CANCEL

vmw Cloud Assembly

Deployments Design **Infrastructure** Extensibility Marketplace

Configure

- Projects
- Cloud Zones
- Kubernetes Zones
- Flavor Mappings
- Image Mappings
- Network Profiles
- Storage Profiles
- Pricing Cards
- Tags

Resources

- Compute
- Networks

tkgi-project-1 DELETE

Summary **Users** Provisioning Kubernetes Provisioning Integrations

Deployment sharing Deployments are shared between all users in the project

User roles

Specify the users and groups related to this project.

+ ADD USERS + ADD GROUPS X REMOVE Search users or groups

<input type="checkbox"/>	Name	Account	Role
<input type="checkbox"/>	Alka Gupta	alka@vmware.com	Administrator
<input type="checkbox"/>	Vimal Pal	vpal@vmware.com	Administrator
<input type="checkbox"/>	Aman Basotra	abasotra@vmware.com	Administrator
<input type="checkbox"/>			

SAVE CANCEL

vmw Cloud Assembly

Deployments Design **Infrastructure** Extensibility Marketplace

tkgi-project-1 DELETE

Summary Users **Provisioning** Kubernetes Provisioning Integrations

Cloud Zones

Specify the cloud zones that can be used when users provision deployments in this project. ⓘ

+ ADD CLOUD ZONE X REMOVE

<input type="checkbox"/>	Name	Description	Priority	Instances	Memory Limit (MB)	CPU Limit	Storage Limit (GB)	Capability Tags
<input type="checkbox"/>	tkgi-cloudacc-vcenter ...		0	Unlimited	Unlimited	Unlimited	Unlimited	

1 - 1 of 1 cloud zones

Resource Tags

vmw Cloud Assembly

Deployments Design **Infrastructure** Extensibility Marketplace

tkgi-project-1 DELETE

Summary Users Provisioning **Kubernetes Provisioning** Integrations

Kubernetes Zones

Specify the kubernetes zones that can be used in this project for provisioning of clusters.

+ ADD ZONE X REMOVE

<input type="checkbox"/>	Name	Description
<input type="checkbox"/>	tkgi-kube-zone	

SAVE CANCEL

Create cluster deployment blueprint & assign it to the project

- > Click Design > Blueprints > NEW > Enter the name for the blueprint > Search & Select the Project to which blueprint will be tagged > Select Blueprint sharing in Service Broker as required > Drag K8S Cluster resource type in Design canvas > Modify the code as required > Click Version > mention the Version > Select Release this version to catalog > Click CREATE > Check the Blueprint is displayed in the Blueprints

The screenshot displays the VMware Cloud Assembly interface for a blueprint named 'tkgi-kube-blueprint-1'. The interface is divided into several sections:

- Navigation Bar:** Includes 'Deployments', 'Design' (active), 'Infrastructure', 'Extensibility', and 'Marketplace'.
- Blueprint Header:** Shows 'tkgi-kube-blueprint-1' with tabs for 'SETTINGS', 'VERSION HISTORY', and 'ACTIONS'.
- Resource Catalog:** A sidebar on the left lists resource types under categories like 'Cloud Agnostic', 'Kubernetes', and 'vSphere'. 'K8S Cluster' is highlighted.
- Canvas:** A central workspace with a grid background. A resource 'Cloud_K8S_Clu...' is being added, with a tooltip showing the template: `$(pkcs:" + to_low...`.
- Code Editor:** A panel on the right shows the blueprint's code in a JSON-like format:


```

1 formatVersion: 1
2 inputs:
3   hostname:
4     type: string
5     size:
6       type: integer
7       minimum: 1
8       maximum: 6
9   plan:
10    type: string
11    enum:
12      - small
13      - medium
14      - large
15 resources:
16   Cloud_K8S_cluster_1:
17     type: Cloud.K8S.Cluster
18     properties:
19       hostname: '${input.hostname}'
20       workers: '${input.size}'
21     constraints:
22       - tag: '${pkcs:" + to_lower(input.plan)}'
23

```

Creating/Adding Kubernetes cluster

Creating cluster

Using DEPLOY option

> Click Infrastructure > Resources > Kubernetes > Clusters > DEPLOY > In Account section Search & Select the EntPKS endpoint > Enter the cluster name > Select sharing as required - in case of sharing at Project level then select the Project's name > Enter the Master hostname(e.g. clsname.tkg.local) > Select the Plan > Enter the Worker nodes count (e.g. 1) to be deployed > Click DEPLOY > Monitor the deployment > Check for the cluster created in Infrastructure > Resources > Kubernetes > Clusters

vmw Cloud Assembly

Deployments Design **Infrastructure** Extensibility Marketplace

Configure ▾

- Projects
- Cloud Zones
- Kubernetes Zones
- Flavor Mappings
- Image Mappings
- Network Profiles
- Storage Profiles
- Pricing Cards
- Tags

Resources ▾

- Compute
- Networks
- Security
- Storage
- Machines
- Volumes
- Kubernetes**

Activity ▾

- Requests

Account * tkgilab1

Name * cls1

Description

Sharing * Global (shareable via kubernetes zones or namespaces) Project (access limited to a single project)

Project * tkgi-project-1

Cluster Details

Master hostname * cls1.tkgi.local

Master host port 8443

Connect by * Master IP Master hostname

Plan * small

Plan details Example: This plan will configure a lightweight kubernetes cluster. Not recommended for production workloads.

Master nodes 1

Worker nodes * 1

Using Blueprint

> Click Design > Blueprints > Select blueprint > Click DEPLOY > Enter Deployment name > Select Blueprint version > Click Next > Click DEPLOY > Enter the Hostname(e.g. clsname.tkg.local) > In Size Enter the Worker nodes count (e.g. 1) to be deployed > Select the Plan > Click DEPLOY > Monitor the deployment > Check for the cluster created in Infrastructure > Resources > Kubernetes > Clusters

vmw Cloud Assembly

Deployments Design Infrastructure Extensibility Marketplace

tkgi-kube-blueprint-1 SETTINGS VERSION HISTORY ACTIONS

Search Resource Types

- Cloud Agnostic
 - Machine
 - Load Balancer
 - Network
 - Security Group
 - Volume
- Kubernetes
 - K8S Cluster
 - K8S Namespace**
- vSphere
 - Machine
 - Disk
 - Network
- NSX
 - Load Balancer
 - Network

Deploy tkgi-kube-bluepri...

1 Deployment Type

2 Deployment Inputs

Deployment Type

Create a new deployment

Deployment Name * deployment

Blueprint Version * Current Draft

Description

CANCEL NEXT

vmw Cloud Assembly

Deployments Design Infrastructure Extensibility Marketplace

tkgi-kube-blueprint-1 SETTINGS VERSION HISTORY ACTIONS

Search Resource Types

- Cloud Agnostic
 - Machine
 - Load Balancer
 - Network
 - Security Group
 - Volume
- Kubernetes
 - K8S Cluster
 - K8S Namespace**
- vSphere
 - Machine
 - Disk
 - Network
- NSX
 - Load Balancer
 - Network

Deploy tkgi-kube-bluepri...

1 Deployment Type

2 Deployment Inputs

Deployment Inputs

Hostname * cls2.tkgi.local

Size * 1

Plan * small

CANCEL PREVIOUS DEPLOY

Adding cluster

Add Existing cluster

> Click Infrastructure > Resources > Kubernetes > Clusters > ADD EXISTING > In Account Search & Select the EntPKS integration endpoint > In Cluster select the cluster to be added > Select Connect by Master IP > Select sharing as required - in case of sharing at Project level then select the Project's name > Click ADD > Check for the cluster added in Infrastructure > Resources > Kubernetes > Clusters

vmw Cloud Assembly

Deployments Design **Infrastructure** Extensibility Marketplace

Network Profiles
Storage Profiles
Pricing Cards
Tags
Resources
Compute
Networks
Security
Storage
Machines
Volumes
Kubernetes
Activity

Add Existing Cluster

Account * tkgi-lab-2

Cluster * cls1

Connect by *
 Master IP
 Master hostname

Description

Sharing *
 Global (shareable via kubernetes zones or namespaces)
 Project (access limited to a single project)

Project * tkgi-project-1

ADD CANCEL

Add External cluster

> Click Infrastructure > Resources > Kubernetes > Clusters > ADD EXTERNAL > Enter the name of cluster > Select sharing as required - in case of sharing at Project level then select the Project's name > In Address enter the Server Address for the cluster > In CA certificate enter the clusters CA certificate i.e. certificate-authority-data > Select location as Private Cloud > Select the Cloud proxy created for the target environment >

> In case Credentials type is Certificate > Enter Public certificate i.e. client-certificate-data & Private certificate i.e. client-key-data for the cluster

> In case Credentials type is Bearer token > Enter the Bearer token

Click VALIDATE > Click ADD > > Check for the cluster added in Infrastructure > Resources > Kubernetes > Clusters

*Note: Please refer configuration file for the necessary input data (config file path is generally \$HOME/.kube/config)

vmw Cloud Assembly

Deployments Design **Infrastructure** Extensibility Marketplace

Project (access limited to a single project)

Project

Cluster Credentials

Address

CA certificate

Location

Cloud proxy

[+ NEW CLOUD PROXY](#)

Credentials type

Public certificate

Private certificate

[VALIDATE](#) [ADD](#) [CANCEL](#)

vmw Cloud Assembly

Deployments Design **Infrastructure** Extensibility Marketplace

Network Profiles
Storage Profiles
Pricing Cards
Tags
Resources
Compute
Networks
Security
Storage
Machines
Volumes
Kubernetes
Activity
Requests
Events Log
Connections
Cloud Accounts
Integrations
Cloud Proxies
Onboarding

Name: tkgi-entpks

Description:

Sharing:

- Global (shareable via kubernetes zones or namespaces)
- Project (access limited to a single project)

Project: tkgi-project-1

Cluster Credentials

Address: https://devenvclst.tkgi.local:8443

CA certificate:

```

Ci0tLS0tOktVHSU4gO0V5VEIGSUNBVEUjLS0tLQpNSUjDK3pDQ0FjT2dBd0l
CQWdJVVUZ5eUlxVjc0ZUJOTUxpRXEiUit1QTdrei9MUXdEUVlKS29aSWh2Y0
5BUUVMckJROXdEVEVMTUFR0EjVUVBeE1DWTJFd0h0Y05NakF3TmPFe
  
```

Location: Private Cloud

Cloud proxy: tkgi-cloudproxy-ext

+ NEW CLOUD PROXY

Credentials type: Bearer token

Bearer token:

```

WdR30NvY00g4R1PFRXIIIPIrozBt0E1B0E0BRVvKCKV3R01VdIPST0PnQK
FmOHdEUVlKS29aSWh2Y05BUUVMQlFBRGdnRUJBQnLLWFBDcW9QY0ISQi
t5MFBaNmU2MUgkYThRTytJbEVtYXZoOXikdUtrOGVGOENRMVRXRUpxMF
R5T3qvWkJUd0RZa1hQMv9iWDlON09oMilHM2hMdDZxNapiR2lFdlUrcFpDS
  
```

Reference links:

<https://docs.vmware.com/en/VMware-Cloud-Assembly/services/Using-and-Managing/GUID-081EA313-129F-4098-B4CC-587A42E7BFFF.html>