



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

Genesys Pulse Private Edition Guide

Tenant Provisioning

Contents

- 1 Prerequisites
 - 1.1 Single namespace
- 2 Tenant provisioning
 - 2.1 Install init tenant chart
 - 2.2 Install dcu helm chart
 - 2.3 Install ids helm chart
 - 2.4 Install permissions helm chart
- 3 Troubleshooting

-
- Administrator

Learn how to provision Genesys Pulse.

Related documentation:

-
-
-
-

RSS:

- [For private edition](#)

Prerequisites

Before performing the steps described on this page, complete the Before you begin instructions, and ensure that you have the following information:

- Versions:
 - = 100.0.000.0015
 - = 100.0.000+0015
- K8S namespace pulse
- Project Name pulse
- Postgres credentials:
 -
 -
 -
 -
 -
 -
- Docker credentials:
 -
 -
- Redis credentials:
 -
 -
 -
 -
- Tenant service variables:
 -
 -
 -

-
- GAAuth/GWS service variables:
 -
 -
 -
 -
 -
 - Storage class:
 -
 -
 - Pulse:
 -

Single namespace

Single namespace deployments have a software-defined networking (SDN) with multitenant mode, where namespaces are network isolated. If you plan to deploy Pulse into the single namespace, ensure that your environment meets the following requirements for inputs:

- Back-end services deployed into the single namespace must include the string *pulse*:
 -
 -
- The hostname used for Ingress must be unique, and must include the string *pulse*:
 -
- Internal service-to-service traffic must use the service endpoints, rather than the Ingress Controller:
 -
 -

Tenant provisioning

Install init tenant chart

Get the **init-tenant** helm chart:

```
helm repo update
helm search repo /init-tenant
```

Prepare the **override file**:

- Update the `values-override-init-tenant.yaml` file (AKS):

```
# Default values for init-tenant.
# This is a YAML-formatted file.
# Declare variables to be passed into your templates.

# * Images
# Replace for your values: registry and secret
```

```
image:
  tag: ""
  pullPolicy: IfNotPresent
  registry: ""
  imagePullSecrets: [name: ""]

configurator:
  enabled: true
  # set service domain used to access voice service
  # example for GKE VPC case: voice.svc.gke1-uswest1.gcpe002.gcpe.com
  voiceDomain: "voice.svc."
  # set service domain used to access ixn service
  # example for GKE VPC case: ixn.svc.gke1-uswest1.gcpe002.gcpe.com
  ixnDomain: "ixn.svc."
  # set service domain used to access pulse service
  # example for GKE VPC case: pulse.svc.gke1-uswest1.gcpe002.gcpe.com
  pulseDomain: "pulse.svc."
  # set configuration server password, used when create secrets
  cfgUser: "default"
  # set configuration server user, used when create secrets
  cfgPassword: "password"
  # common log configuration
  cfgHost: "tenant-9350e2fc-ald-4c65-8d40-1f75a2e080dd.voice.svc."

log:
  # target directory where log will be stored, leave empty for default
  logDir: ""
  # path where volume will be mounted
  volumeMountPath: /data/log
  # log volume type: none | hostpath | pvc
  volumeType: none
  # log volume hostpath, used with volumeType "hostpath"
  volumeHostPath: /mnt/log
  # log PVC parameters, used with volumeType "pvc"
  pvc:
    name: pulse-init-tenant-logs
    accessModes:
      - ReadWriteMany
    capacity: 10Gi
    class:

# * Tenant info
# Replace for your values
tenant:
  # Tenant UUID
  id:
  # Tenant SID (like 0001)
  sid:

# common configuration.
config:
  dbName: ""
  # set "true" when need @host added for username
  dbUserWithHost: true
  # set "true" for CSI secrets
  mountSecrets: false
  # Postgres config map name
  postgresConfig: "pulse-postgres-configmap"
  # Postgres secret name
  postgresSecret: "pulse-postgres-secret"
  # Postgres secret key for user
  postgresSecretUser: "META_DB_ADMIN"
  # Postgres secret key for password
```

```
    postgresSecretPassword: "META_DB_ADMINPWD"

## Service account settings
serviceAccount:
  # Specifies whether a service account should be created
  create: false
  # Annotations to add to the service account
  annotations: {}
  # The name of the service account to use.
  # If not set and create is true, a name is generated using the fullname template
  name: ""

## Add annotations to all pods
##
podAnnotations: {}

## Specifies the security context for all Pods in the service
##
podSecurityContext: {}

## Resource requests and limits
## ref: http://kubernetes.io/docs/user-guide/compute-resources/
##
resources:
  limits:
    memory: 256Mi
    cpu: 200m
  requests:
    memory: 128Mi
    cpu: 100m

## Priority Class
## ref: https://kubernetes.io/docs/concepts/configuration/pod-priority-preemption/
##
priorityClassName: ""

## Node labels for assignment.
## ref: https://kubernetes.io/docs/user-guide/node-selection/
##
nodeSelector: {}

## Tolerations for assignment.
## ref: https://kubernetes.io/docs/concepts/configuration/taint-and-toleration/
##
tolerations: []

# * Templates
templates:
  - Agent_Group_Status.gpb
  - Agent_KPIs.gpb
  - Agent_Login.gpb
  - Alert_Widget.gpb
  - Callback_Activity.gpb
  - Campaign_Activity.gpb
  - Campaign_Callback_Status.gpb
  - Campaign_Group_Activity.gpb
  - Campaign_Group_Status.gpb
  - Chat_Agent_Activity.gpb
  - Chat_Queue_Activity.gpb
  - Chat_Service_Level_Performance.gpb
  - Chat_Waiting_Statistics.gpb
  - Email_Agent_Activity.gpb
  - Email_Queue_Activity.gpb
```

- Facebook_Media_Activity.gpb
- IFRAME.gpb
- IWD_Agent_Activity.gpb
- IWD_Queue_Activity.gpb
- Queue_KPIs.gpb
- Queue_Overflow_Reason.gpb
- Static_Text.gpb
- Twitter_Media_Activity.gpb
- eServices_Agent_Activity.gpb
- eServices_Queue_KPIs.gpb

- Update the values-override-init-tenant.yaml file (GKE):

Important

Enable configurator only for configurations in GKE with VPC scoped DNS.

- ```
Default values for init-tenant.
This is a YAML-formatted file.
Declare variables to be passed into your templates.

* Images
Replace for your values: registry and secret
image:
 tag: ""
 pullPolicy: IfNotPresent
 registry: ""
 imagePullSecrets: [name: ""]

configurator:
 enabled: true
 # set service domain used to access voice service
 # example for GKE VPC case: voice.svc.gke1-uswest1.gcpe002.gencpe.com
 voiceDomain: "voice.svc."
 # set service domain used to access ixn service
 # example for GKE VPC case: ixn.svc.gke1-uswest1.gcpe002.gencpe.com
 ixnDomain: "ixn.svc."
 # set service domain used to access pulse service
 # example for GKE VPC case: pulse.svc.gke1-uswest1.gcpe002.gencpe.com
 pulseDomain: "pulse.svc."
 # set configuration server password, used when create secrets
 cfgUser: "default"
 # set configuration server user, used when create secrets
 cfgPassword: "password"
 # common log configuration
 cfgHost: "tenant-.voice.svc."

log:
 # target directory where log will be stored, leave empty for default
 logDir: ""
 # path where volume will be mounted
 volumeMountPath: /data/log
 # log volume type: none | hostpath | pvc
 volumeType: none
 # log volume hostpath, used with volumeType "hostpath"
 volumeHostPath: /mnt/log
 # log PVC parameters, used with volumeType "pvc"
 pvc:
 name: pulse-init-tenant-logs
```

---

```
 accessModes:
 - ReadWriteMany
 capacity: 10Gi
 class: nfs-client

* Tenant info
Replace for your values
tenant:
 # Tenant UUID
 id:
 # Tenant SID (like 0001)
 sid:

common configuration.
config:
 dbName: ""
 # set "true" when need @host added for username
 dbUserWithHost: true
 # set "true" for CSI secrets
 mountSecrets: false
 # Postgres config map name
 postgresConfig: "pulse-postgres-configmap"
 # Postgres secret name
 postgresSecret: "pulse-postgres-secret"
 # Postgres secret key for user
 postgresSecretUser: "META_DB_ADMIN"
 # Postgres secret key for password
 postgresSecretPassword: "META_DB_ADMINPWD"

Service account settings
serviceAccount:
 # Specifies whether a service account should be created
 create: false
 # Annotations to add to the service account
 annotations: {}
 # The name of the service account to use.
 # If not set and create is true, a name is generated using the fullname
template
 name: ""

Add annotations to all pods
##
podAnnotations: {}

Specifies the security context for all Pods in the service
##
podSecurityContext:
 fsGroup: null
 runAsUser: null
 runAsGroup: 0
 runAsNonRoot: true

Resource requests and limits
ref: http://kubernetes.io/docs/user-guide/compute-resources/
##
resources:
 limits:
 memory: 256Mi
 cpu: 200m
 requests:
 memory: 128Mi
 cpu: 100m
```

---

---

```

Priority Class
ref: https://kubernetes.io/docs/concepts/configuration/pod-priority-preemption/
##
priorityClassName: ""

Node labels for assignment.
ref: https://kubernetes.io/docs/user-guide/node-selection/
##
nodeSelector: {}

Tolerations for assignment.
ref: https://kubernetes.io/docs/concepts/configuration/taint-and-toleration/
##
tolerations: []

* Templates
templates:
 - Agent_Group_Status.gpb
 - Agent_KPIS.gpb
 - Agent_Login.gpb
 - Alert_Widget.gpb
 - Callback_Activity.gpb
 - Campaign_Activity.gpb
 - Campaign_Callback_Status.gpb
 - Campaign_Group_Activity.gpb
 - Campaign_Group_Status.gpb
 - Chat_Agent_Activity.gpb
 - Chat_Queue_Activity.gpb
 - Chat_Service_Level_Performance.gpb
 - Chat_Waiting_Statistics.gpb
 - Email_Agent_Activity.gpb
 - Email_Queue_Activity.gpb
 - Facebook_Media_Activity.gpb
 - IFRAME.gpb
 - IWD_Agent_Activity.gpb
 - IWD_Queue_Activity.gpb
 - Queue_KPIS.gpb
 - Queue_Overflow_Reason.gpb
 - Static_Text.gpb
 - Twitter_Media_Activity.gpb
 - eServices_Agent_Activity.gpb
 - eServices_Queue_KPIS.gpb

```

**Install the init-tenant helm chart:**

To install the init-tenant helm chart, run the following command:

```
helm upgrade --install "pulse-init-tenant-" pulsehelmrepo/init-tenant --wait --wait-for-jobs --version="" --namespace=pulse -f values-override-init-tenant.yaml
```

If installation is successful, the exit code 0 appears.

**Validate the init-tenant helm chart:**

To validate the init-tenant helm chart, run the following command:

```
kubectl get pods -n="pulse" -l "app.kubernetes.io/name=init-tenant,app.kubernetes.io/instance=pulse-init-tenant"
```

If the deployment was successful, the pulse-init-tenant job is listed as Completed/. For example:

| NAME                            | READY | STATUS    | RESTARTS | AGE   |
|---------------------------------|-------|-----------|----------|-------|
| pulse-init-tenant-100-job-qszgl | 0/1   | Completed | 0        | 2d20h |

---

## Install dcu helm chart

### Get the dcu helm chart:

```
helm repo update
helm search repo /dcu
```

### Prepare the override file:

- Update the values-override-dcu.yaml file (AKS):

```
Default values for dcu.
This is a YAML-formatted file.
Declare variables to be passed into your templates.

replicaCount: ""

* Tenant info
tenant identification, or empty for shared deployment
tenant:
 # Tenant UUID
 id: ""
 # Tenant SID (like 0001)
 sid: ""

* Common log configuration
log:
 # target directory where log will be stored, leave empty for default
 logDir: ""
 # path where volume will be mounted
 volumeMountPath: /data/log
 # log volume type: none | hostpath | pvc
 volumeType: pvc
 # log volume hostpath, used with volumeType "hostpath"
 volumeHostPath: /mnt/log
 # log PVC parameters, used with volumeType "pvc"
 pvc:
 name: pulse-dcu-logs
 accessModes:
 - ReadWriteMany
 capacity: 10Gi
 class:

* Config info
Set your values.
config:
 dbName: ""
 # set "true" when need @host added for username
 dbUserWithHost: true
 mountSecrets: false
 postgresConfig: "pulse-postgres-configmap"
 # Postgres secret name
 postgresSecret: "pulse-postgres-secret"
 # Postgres secret key for user
 postgresSecretUser: "META_DB_ADMIN"
 # Postgres secret key for password
 postgresSecretPassword: "META_DB_ADMINPWD"
 redisConfig: "pulse-redis-configmap"
 # Redis secret name
 redisSecret: "pulse-redis-secret"
 # Redis secret key for access key
 redisSecretKey: "REDIS01_KEY"
```

---

```
* Image
container image common settings
image:
 tag: ""
 pullPolicy: IfNotPresent
 registry: ""
 imagePullSecrets: [name: ""]

Service account settings
serviceAccount:
 # Specifies whether a service account should be created
 create: false
 # Annotations to add to the service account
 annotations: {}
 # The name of the service account to use.
 # If not set and create is true, a name is generated using the fullname
 template
 name: ""

Add annotations to all pods
##
podAnnotations: {}

Specifies the security context for all Pods in the service
##
podSecurityContext: {}

Add labels to all pods
##
podLabels: {}

HPA Settings
Not supported in this release!
hpa:
 enabled: false

Priority Class
ref: https://kubernetes.io/docs/concepts/configuration/pod-priority-
preemption/
##
priorityClassName: ""

Node labels for assignment.
ref: https://kubernetes.io/docs/user-guide/node-selection/
##
nodeSelector: {}

Tolerations for assignment.
ref: https://kubernetes.io/docs/concepts/configuration/taint-and-
toleration/
##
tolerations: []

Pod Disruption Budget Settings
podDisruptionBudget:
 enabled: false

Affinity for assignment.
Ref: https://kubernetes.io/docs/concepts/configuration/assign-pod-
node/#affinity-and-anti-affinity
##
affinity: {}
```

---

```

* Monitoring settings
monitoring:
 # enable the Prometheus metrics endpoint
 enabled: false
 # enable golden signals metrics (not supported for PE)
 goldenSignals:
 enabled: false
 # port number of the Prometheus metrics endpoint
 port: 9091
 # HTTP path to scrape for metrics
 path: /metrics
 # additional annotations required for monitoring PODs
 # you can reference values of other variables as
 {{.Values.variable.full.name}}
 podAnnotations: {}
 # prometheus.io/scrape: "true"
 # prometheus.io/port: "{{.Values.monitoring.port}}"
 # prometheus.io/path: "/metrics"
 podMonitor:
 # enables PodMonitor creation for the POD
 enabled: true
 # interval at which metrics should be scraped
 scrapeInterval: 30s
 # timeout after which the scrape is ended
 scrapeTimeout:
 # namespace of the PodMonitor, defaults to the namespace of the POD
 namespace:
 additionalLabels: {}
 alerts:
 # enables alert rules
 enabled: true
 # alert condition duration
 duration: 5m
 # namespace of the alert rules, defaults to the namespace of the POD
 namespace:
 additionalLabels: {}

#####

* Configuration for the Collector container
collector:
 # resource limits for container
 resources:
 # minimum resource requirements to start container
 requests:
 # minimal amount of memory required to start a container
 memory: "300Mi"
 # minimal CPU to reserve
 cpu: "200m"
 # resource limits for containers
 limits:
 # maximum amount of memory a container can use before being evicted
 # by the OOM Killer
 memory: "4Gi"
 # maximum amount of CPU resources that can be used and should be tuned
 # to reflect
 # what the application can effectively use before needing to be
 # horizontally scaled out
 cpu: "8000m"
 # securityContext: {}

* Configuration for the StatServer container

```

---

---

```

statsserver:
 # resource limits for container
 resources:
 # minimum resource requirements to start container
 requests:
 # minimal amount of memory required to start a container
 memory: "300Mi"
 # minimal CPU to reserve
 cpu: "100m"
 # resource limits for containers
 limits:
 # maximum amount of memory a container can use before being evicted
 # by the OOM Killer
 memory: "4Gi"
 # maximum amount of CPU resources that can be used and should be tuned
to reflect
 # what the application can effectively use before needing to be
horizontally scaled out
 cpu: "4000m"
 # securityContext: {}

* Configuration for the monitor sidecar container
monitorSidecar:
 # resource limits for container
 resources:
 # disabled: true
 # minimum resource requirements to start container
 requests:
 # minimal amount of memory required to start a container
 memory: "30Mi"
 # minimal CPU to reserve
 cpu: "2m"
 # resource limits for containers
 limits:
 # maximum amount of memory a container can use before being evicted
 # by the OOM Killer
 memory: "70Mi"
 # maximum amount of CPU resources that can be used and should be tuned
to reflect
 # what the application can effectively use before needing to be
horizontally scaled out
 cpu: "10m"
 # securityContext: {}

#####

* Configuration for the Configuration Server Proxy container
csproxy:
 # define domain for the configuration host
 params:
 cfgHost: "tenant-.voice."
 # resource limits for container
 resources:
 # minimum resource requirements to start container
 requests:
 # minimal amount of memory required to start a container
 memory: "200Mi"
 # minimal CPU to reserve
 cpu: "50m"
 # resource limits for containers
 limits:
 # maximum amount of memory a container can use before being evicted
 # by the OOM Killer

```

---

---

```

 memory: "2Gi"
 # maximum amount of CPU resources that can be used and should be tuned
to reflect
 # what the application can effectively use before needing to be
horizontally scaled out
 cpu: "1000m"
 # securityContext: {}

volumeClaims contains persistent volume claims for services
All available storage classes can be found here:
https://github.com/genesysengage/tfm-azure-core-aks/blob/master/k8s-module/
storage.tf
volumeClaims:
 # statserverBackup is storage for statserver backup data
 statserverBackup:
 name: statserver-backup
 accessModes:
 - ReadWriteOnce
 # capacity is storage capacity
 capacity: "1Gi"
 # class is storage class. Must be set explicitly.
 class:

```

- Update the values-override-dcu.yaml file (GKE):

```

Default values for dcu.
This is a YAML-formatted file.
Declare variables to be passed into your templates.

replicaCount: ""

* Tenant info
tenant identification, or empty for shared deployment
tenant:
 # Tenant UUID
 id: ""
 # Tenant SID (like 0001)
 sid: ""

* Common log configuration
log:
 # target directory where log will be stored, leave empty for default
 logDir: ""
 # path where volume will be mounted
 volumeMountPath: /data/log
 # log volume type: none | hostpath | pvc
 volumeType: pvc
 # log volume hostpath, used with volumeType "hostpath"
 volumeHostPath: /mnt/log
 # log PVC parameters, used with volumeType "pvc"
 pvc:
 name: pulse-dcu-logs
 accessModes:
 - ReadWriteMany
 capacity: 10Gi
 class:

* Config info
Set your values.
config:
 dbName: ""
 # set "true" when need @host added for username
 dbUserWithHost: true

```

---

---

```
mountSecrets: false
postgresConfig: "pulse-postgres-configmap"
Postgres secret name
postgresSecret: "pulse-postgres-secret"
Postgres secret key for user
postgresSecretUser: "META_DB_ADMIN"
Postgres secret key for password
postgresSecretPassword: "META_DB_ADMINPWD"
redisConfig: "pulse-redis-configmap"
Redis secret name
redisSecret: "pulse-redis-secret"
Redis secret key for access key
redisSecretKey: "REDIS01_KEY"

* Image
container image common settings
image:
 tag: ""
 pullPolicy: IfNotPresent
 registry: ""
 imagePullSecrets: [name: ""]

Service account settings
serviceAccount:
 # Specifies whether a service account should be created
 create: false
 # Annotations to add to the service account
 annotations: {}
 # The name of the service account to use.
 # If not set and create is true, a name is generated using the fullname
 name: ""

Add annotations to all pods
##
podAnnotations: {}

Specifies the security context for all Pods in the service
##
podSecurityContext:
 runAsNonRoot: true
 runAsUser: 500
 runAsGroup: 500
 fsGroup: 0

Add labels to all pods
##
podLabels: {}

HPA Settings
Not supported in this release!
hpa:
 enabled: false

Priority Class
ref: https://kubernetes.io/docs/concepts/configuration/pod-priority-preemption/
##
priorityClassName: ""

Node labels for assignment.
ref: https://kubernetes.io/docs/user-guide/node-selection/
##
```

---

```

nodeSelector: {}

Tolerations for assignment.
ref: https://kubernetes.io/docs/concepts/configuration/taint-and-
toleration/
##
tolerations: []

Pod Disruption Budget Settings
podDisruptionBudget:
 enabled: false

Affinity for assignment.
Ref: https://kubernetes.io/docs/concepts/configuration/assign-pod-
node/#affinity-and-anti-affinity
##
affinity: {}

* Monitoring settings
monitoring:
 # enable the Prometheus metrics endpoint
 enabled: false
 # enable golden signals metrics (not supported for PE)
 goldenSignals:
 enabled: false
 # port number of the Prometheus metrics endpoint
 port: 9091
 # HTTP path to scrape for metrics
 path: /metrics
 # additional annotations required for monitoring PODs
 # you can reference values of other variables as
 {{.Values.variable.full.name}}
 podAnnotations: {}
 # prometheus.io/scrape: "true"
 # prometheus.io/port: "{{.Values.monitoring.port}}"
 # prometheus.io/path: "/metrics"
 podMonitor:
 # enables PodMonitor creation for the POD
 enabled: true
 # interval at which metrics should be scraped
 scrapeInterval: 30s
 # timeout after which the scrape is ended
 scrapeTimeout:
 # namespace of the PodMonitor, defaults to the namespace of the POD
 namespace:
 additionalLabels: {}
 alerts:
 # enables alert rules
 enabled: true
 # alert condition duration
 duration: 5m
 # namespace of the alert rules, defaults to the namespace of the POD
 namespace:
 additionalLabels: {}

#####

* Configuration for the Collector container
collector:
 # resource limits for container
 resources:
 # minimum resource requirements to start container

```

---

---

```

requests:
 # minimal amount of memory required to start a container
 memory: "300Mi"
 # minimal CPU to reserve
 cpu: "200m"
resource limits for containers
limits:
 # maximum amount of memory a container can use before being evicted
 # by the OOM Killer
 memory: "4Gi"
 # maximum amount of CPU resources that can be used and should be tuned
to reflect
 # what the application can effectively use before needing to be
horizontally scaled out
 cpu: "8000m"
securityContext:
runAsUser: 500
runAsGroup: 500

* Configuration for the StatServer container
statsserver:
resource limits for container
resources:
 # minimum resource requirements to start container
requests:
 # minimal amount of memory required to start a container
 memory: "300Mi"
 # minimal CPU to reserve
 cpu: "100m"
resource limits for containers
limits:
 # maximum amount of memory a container can use before being evicted
 # by the OOM Killer
 memory: "4Gi"
 # maximum amount of CPU resources that can be used and should be tuned
to reflect
 # what the application can effectively use before needing to be
horizontally scaled out
 cpu: "4000m"
securityContext:
runAsUser: 500
runAsGroup: 500

* Configuration for the monitor sidecar container
monitorSidecar:
resource limits for container
resources:
 # disabled: true
 # minimum resource requirements to start container
requests:
 # minimal amount of memory required to start a container
 memory: "30Mi"
 # minimal CPU to reserve
 cpu: "2m"
resource limits for containers
limits:
 # maximum amount of memory a container can use before being evicted
 # by the OOM Killer
 memory: "70Mi"
 # maximum amount of CPU resources that can be used and should be tuned
to reflect
 # what the application can effectively use before needing to be
horizontally scaled out

```

---

```

 cpu: "10m"
 # securityContext:
 # runAsUser: 500
 # runAsGroup: 500

#####

* Configuration for the Configuration Server Proxy container
csproxy:
 # define domain for the configuration host
 params:
 cfgHost: "tenant-.voice."
 # resource limits for container
 resources:
 # minimum resource requirements to start container
 requests:
 # minimal amount of memory required to start a container
 memory: "200Mi"
 # minimal CPU to reserve
 cpu: "50m"
 # resource limits for containers
 limits:
 # maximum amount of memory a container can use before being evicted
 # by the OOM Killer
 memory: "2Gi"
 # maximum amount of CPU resources that can be used and should be tuned
 # to reflect
 # what the application can effectively use before needing to be
 # horizontally scaled out
 cpu: "1000m"
 # securityContext:
 # runAsUser: 500
 # runAsGroup: 500

volumeClaims contains persistent volume claims for services
All available storage classes can be found here:
https://github.com/genesysengage/tfm-azure-core-aks/blob/master/k8s-module/
storage.tf
volumeClaims:
 # statserverBackup is storage for statserver backup data
 statserverBackup:
 name: statserver-backup
 accessModes:
 - ReadWriteOnce
 # capacity is storage capacity
 capacity: "1Gi"
 # class is storage class. Must be set explicitly.
 class:

```

#### Install the dcu helm chart

To install the dcu helm chart, run the following command:

```
helm upgrade --install "pulse-dcu-" pulsehelmrepo/dcu --wait --reuse-values --
version= --namespace=pulse -f values-override-dcu.yaml
```

#### Validate the dcu helm chart

To validate the dcu helm chart, run the following command:

```
kubectl get pods -n=pulse -l "app.kubernetes.io/name=dcu,app.kubernetes.io/
instance=pulse-dcu-"
```

Check the output to ensure that all pulse-dcu pods are running, for example:

| NAME | READY | STATUS | RESTARTS | AGE |
|------|-------|--------|----------|-----|
|------|-------|--------|----------|-----|

---

```
pulse-dcu-100-0 3/3 Running 0 5m23s
pulse-dcu-100-1 3/3 Running 0 4m47s
```

## Install lds helm chart

### Get the lds helm chart:

```
helm repo update
helm search repo /lds
```

### Prepare the override file:

- Update values in the values-override-lds.yaml file (AKS):

```
Default values for lds.
This is a YAML-formatted file.
Declare variables to be passed into your templates.

replicaCount: 2

* Tenant info
tenant identification, or empty for shared deployment
tenant:
 # Tenant UUID
 id: ""
 # Tenant SID (like 0001)
 sid: ""

* Common log configuration
log:
 # target directory where log will be stored, leave empty for default
 logDir: ""
 # path where volume will be mounted
 volumeMountPath: /data/log
 # log volume type: none | hostpath | pvc
 volumeType: pvc
 # log volume hostpath, used with volumeType "hostpath"
 volumeHostPath: /mnt/log
 # log PVC parameters, used with volumeType "pvc"
 pvc:
 name: pulse-lds-logs
 accessModes:
 - ReadWriteMany
 capacity: 10Gi
 class:

* Container image common settings
image:
 tag: ""
 pullPolicy: IfNotPresent
 registry: ""
 imagePullSecrets: [name: ""]

Service account settings
serviceAccount:
 # Specifies whether a service account should be created
 create: false
 # Annotations to add to the service account
 annotations: {}
 # The name of the service account to use.
 # If not set and create is true, a name is generated using the fullname
 template
```

---

```
 name: ""

 ## Add annotations to all pods
 ##
 podAnnotations: {}

 ## Specifies the security context for all Pods in the service
 ##
 podSecurityContext: {}

 ## Add labels to all pods
 ##
 podLabels: {}

 ## HPA Settings
 ## Not supported in this release!
 hpa:
 enabled: false

 ## Priority Class
 ## ref: https://kubernetes.io/docs/concepts/configuration/pod-priority-
 ## preemption/
 ##
 priorityClassName: ""

 ## Node labels for assignment.
 ## ref: https://kubernetes.io/docs/user-guide/node-selection/
 ##
 nodeSelector: {}

 ## Tolerations for assignment.
 ## ref: https://kubernetes.io/docs/concepts/configuration/taint-and-
 ## toleration/
 ##
 tolerations: []

 ## Pod Disruption Budget Settings
 podDisruptionBudget:
 enabled: false

 ## Affinity for assignment.
 ## Ref: https://kubernetes.io/docs/concepts/configuration/assign-pod-
 ## node/#affinity-and-anti-affinity
 ##
 affinity: {}

 # * Monitoring settings
 monitoring:
 # enable the Prometheus metrics endpoint
 enabled: false
 # enable golden signals metrics (not supported for PE)
 goldenSignals:
 enabled: false
 # port number of the Prometheus metrics endpoint
 port: 9091
 # HTTP path to scrape for metrics
 path: /metrics
 # additional annotations required for monitoring PODs
 # you can reference values of other variables as
 # {{.Values.variable.full.name}}
 podAnnotations: {}
 # prometheus.io/scrape: "true"
 # prometheus.io/port: "{{.Values.monitoring.port}}"
```

---

---

```

 # prometheus.io/path: "/metrics"
podMonitor:
 # enables PodMonitor creation for the POD
 enabled: true
 # interval at which metrics should be scraped
 scrapeInterval: 30s
 # timeout after which the scrape is ended
 scrapeTimeout:
 # namespace of the PodMonitor, defaults to the namespace of the POD
 namespace:
 additionalLabels: {}
alerts:
 # enables alert rules
 enabled: true
 # alert condition duration
 duration: 5m
 # namespace of the alert rules, defaults to the namespace of the POD
 namespace:
 additionalLabels: {}

* Configuration for the LDS container
lds:
 # resource limits for container
 resources:
 # minimum resource requirements to start container
 requests:
 # minimal amount of memory required to start a container
 memory: "50Mi"
 # minimal CPU to reserve
 cpu: "50m"
 # resource limits for containers
 limits:
 # maximum amount of memory a container can use before being evicted
 # by the OOM Killer
 memory: "4Gi"
 # maximum amount of CPU resources that can be used and should be tuned
 # to reflect
 # what the application can effectively use before needing to be
 # horizontally scaled out
 cpu: "4000m"
 # securityContext: {}

* Configuration for the monitor sidecar container
monitorSidecar:
 # resource limits for container
 resources:
 # minimum resource requirements to start container
 requests:
 # minimal amount of memory required to start a container
 memory: "30Mi"
 # minimal CPU to reserve
 cpu: "2m"
 # resource limits for containers
 limits:
 # maximum amount of memory a container can use before being evicted
 # by the OOM Killer
 memory: "70Mi"
 # maximum amount of CPU resources that can be used and should be tuned
 # to reflect
 # what the application can effectively use before needing to be
 # horizontally scaled out
 cpu: "10m"
 # securityContext: {}

```

---

---

```

* Configuration for the Configuration Server Proxy container
csproxy:
 # define domain for the configuration host
 params:
 cfgHost: "tenant-.voice."
 resources:
 # minimum resource requirements to start container
 requests:
 # minimal amount of memory required to start a container
 memory: "200Mi"
 # minimal CPU to reserve
 cpu: "50m"
 # resource limits for containers
 limits:
 # maximum amount of memory a container can use before being evicted
 # by the OOM Killer
 memory: "2Gi"
 # maximum amount of CPU resources that can be used and should be tuned
 # to reflect
 # what the application can effectively use before needing to be
 # horizontally scaled out
 cpu: "1000m"
 # securityContext: {}

```

- Update values in the values-override-lds.yaml file (GKE):

```

Default values for lds.
This is a YAML-formatted file.
Declare variables to be passed into your templates.

replicaCount: 2

* Tenant info
tenant identification, or empty for shared deployment
tenant:
 # Tenant UUID
 id: ""
 # Tenant SID (like 0001)
 sid: ""

* Common log configuration
log:
 # target directory where log will be stored, leave empty for default
 logDir: ""
 # path where volume will be mounted
 volumeMountPath: /data/log
 # log volume type: none | hostpath | pvc
 volumeType: pvc
 # log volume hostpath, used with volumeType "hostpath"
 volumeHostPath: /mnt/log
 # log PVC parameters, used with volumeType "pvc"
 pvc:
 name: pulse-lds-logs
 accessModes:
 - ReadWriteMany
 capacity: 10Gi
 class:

* Container image common settings
image:
 tag: ""
 pullPolicy: IfNotPresent

```

---

---

```
registry: ""
imagePullSecrets: [name: ""]

Service account settings
serviceAccount:
 # Specifies whether a service account should be created
 create: false
 # Annotations to add to the service account
 annotations: {}
 # The name of the service account to use.
 # If not set and create is true, a name is generated using the fullname
template
 name: ""

Add annotations to all pods
##
podAnnotations: {}

Specifies the security context for all Pods in the service
##
podSecurityContext:
 runAsNonRoot: true
 runAsUser: 500
 runAsGroup: 500
 fsGroup: 0

Add labels to all pods
##
podLabels: {}

HPA Settings
Not supported in this release!
hpa:
 enabled: false

Priority Class
ref: https://kubernetes.io/docs/concepts/configuration/pod-priority-
preemption/
##
priorityClassName: ""

Node labels for assignment.
ref: https://kubernetes.io/docs/user-guide/node-selection/
##
nodeSelector: {}

Tolerations for assignment.
ref: https://kubernetes.io/docs/concepts/configuration/taint-and-
toleration/
##
tolerations: []

Pod Disruption Budget Settings
podDisruptionBudget:
 enabled: false

Affinity for assignment.
Ref: https://kubernetes.io/docs/concepts/configuration/assign-pod-
node/#affinity-and-anti-affinity
##
affinity: {}

* Monitoring settings
```

---

```

monitoring:
 # enable the Prometheus metrics endpoint
 enabled: false
 # enable golden signals metrics (not supported for PE)
 goldenSignals:
 enabled: false
 # port number of the Prometheus metrics endpoint
 port: 9091
 # HTTP path to scrape for metrics
 path: /metrics
 # additional annotations required for monitoring PODs
 # you can reference values of other variables as
 {{.Values.variable.full.name}}
 podAnnotations: {}
 # prometheus.io/scrape: "true"
 # prometheus.io/port: "{{.Values.monitoring.port}}"
 # prometheus.io/path: "/metrics"
 podMonitor:
 # enables PodMonitor creation for the POD
 enabled: true
 # interval at which metrics should be scraped
 scrapeInterval: 30s
 # timeout after which the scrape is ended
 scrapeTimeout:
 # namespace of the PodMonitor, defaults to the namespace of the POD
 namespace:
 additionalLabels: {}
 alerts:
 # enables alert rules
 enabled: true
 # alert condition duration
 duration: 5m
 # namespace of the alert rules, defaults to the namespace of the POD
 namespace:
 additionalLabels: {}

* Configuration for the LDS container
lds:
 # resource limits for container
 resources:
 # minimum resource requirements to start container
 requests:
 # minimal amount of memory required to start a container
 memory: "50Mi"
 # minimal CPU to reserve
 cpu: "50m"
 # resource limits for containers
 limits:
 # maximum amount of memory a container can use before being evicted
 # by the OOM Killer
 memory: "4Gi"
 # maximum amount of CPU resources that can be used and should be tuned
 # to reflect
 # what the application can effectively use before needing to be
 # horizontally scaled out
 cpu: "4000m"
 # securityContext:
 # runAsUser: 500
 # runAsGroup: 500

* Configuration for the monitor sidecar container
monitorSidecar:
 # resource limits for container

```

---

---

```

resources:
 # minimum resource requirements to start container
 requests:
 # minimal amount of memory required to start a container
 memory: "30Mi"
 # minimal CPU to reserve
 cpu: "2m"
 # resource limits for containers
 limits:
 # maximum amount of memory a container can use before being evicted
 # by the OOM Killer
 memory: "70Mi"
 # maximum amount of CPU resources that can be used and should be tuned
to reflect
 # what the application can effectively use before needing to be
horizontally scaled out
 cpu: "10m"
 # securityContext:
 # runAsUser: 500
 # runAsGroup: 500

* Configuration for the Configuration Server Proxy container
csproxy:
 # define domain for the configuration host
 params:
 cfgHost: "tenant-.voice."
 resources:
 # minimum resource requirements to start container
 requests:
 # minimal amount of memory required to start a container
 memory: "200Mi"
 # minimal CPU to reserve
 cpu: "50m"
 # resource limits for containers
 limits:
 # maximum amount of memory a container can use before being evicted
 # by the OOM Killer
 memory: "2Gi"
 # maximum amount of CPU resources that can be used and should be tuned
to reflect
 # what the application can effectively use before needing to be
horizontally scaled out
 cpu: "1000m"
 # securityContext:
 # runAsUser: 500
 # runAsGroup: 500

```

**Update values in the values-override-lds-vq.yaml file:**

```

Default values for lds.
This is a YAML-formatted file.
Declare variables to be passed into your templates.

```

```

lds:
 params:
 cfgApp: "pulse-lds-vq-$(K8S_POD_INDEX % 2)"

log:
 pvc:
 name: pulse-lds-vq-logs

```

**Install the lds helm chart:**

To install the lds helm chart, run the following command:

---

```
helm upgrade --install "pulse-lds-" pulsehelmrepo/lds --wait --version= --
namespace=pulse -f values-override-lds.yaml
helm upgrade --install "pulse-lds-vq-" pulsehelmrepo/lds --wait --version= --
namespace=pulse -f values-override-lds.yaml -f values-override-lds-vq.yaml
```

If the installation is successful, the exit code 0 appears.

**Validate the lds helm chart:**

To validate the lds helm chart, run the following command:

```
kubectl get pods -n=pulse -l "app.kubernetes.io/name=lds,app.kubernetes.io/
instance=pulse-lds-"
```

Verify that the command reports all pulse-lds-vq pods as Running, for example:

| NAME            | READY | STATUS  | RESTARTS | AGE   |
|-----------------|-------|---------|----------|-------|
| pulse-lds-100-0 | 3/3   | Running | 0        | 2d20h |
| pulse-lds-100-1 | 3/3   | Running | 0        | 2d20h |

## Install permissions helm chart

**Get the permissions helm chart**

```
helm repo update
helm search repo /permissions
```

**Prepare the override file:**

- Update values in the values-override-permissions.yaml file (AKS):

```
Default values for permissions.
This is a YAML-formatted file.
Declare variables to be passed into your templates.

* Image configuration
image:
 tag: ""
 pullPolicy: IfNotPresent
 registry: ""
 imagePullSecrets: [name: ""]

* Tenant info
tenant identification, or empty for shared deployment
tenant:
 # Tenant UUID
 id: ""
 # Tenant SID (like 0001)
 sid: ""

common configuration.
config:
 dbName: ""
 # set "true" when need @host added for username
 dbUserWithHost: true
 # set "true" for CSI secrets
 mountSecrets: false
 # Postgres config map name
 postgresConfig: "pulse-postgres-configmap"
 # Postgres secret name
 postgresSecret: "pulse-postgres-secret"
 # Postgres secret key for user
 postgresSecretUser: "META_DB_ADMIN"
 # Postgres secret key for password
```

---

```

 postgresSecretPassword: "META_DB_ADMINPWD"
 # Redis config map name
 redisConfig: "pulse-redis-configmap"
 # Redis secret name
 redisSecret: "pulse-redis-secret"
 # Redis secret key for access key
 redisSecretKey: "REDIS01_KEY"

* Configuration for the Configuration Server Proxy container
csproxy:
 # define domain for the configuration host
 params:
 cfgHost: "tenant-.voice."
 # resource limits for container
 resources:
 # minimum resource requirements to start container
 requests:
 # minimal amount of memory required to start a container
 memory: "200Mi"
 # minimal CPU to reserve
 cpu: "50m"
 # resource limits for containers
 limits:
 # maximum amount of memory a container can use before being evicted
 # by the OOM Killer
 memory: "2Gi"
 # maximum amount of CPU resources that can be used and should be tuned
 # to reflect
 # what the application can effectively use before needing to be
 # horizontally scaled out
 cpu: "1000m"
 # securityContext: {}

* Common log configuration
log:
 # target directory where log will be stored, leave empty for default
 logDir: ""
 # path where volume will be mounted
 volumeMountPath: /data/log
 # log volume type: none | hostpath | pvc
 volumeType: pvc
 # log volume hostpath, used with volumeType "hostpath"
 volumeHostPath: /mnt/log
 # log PVC parameters, used with volumeType "pvc"
 pvc:
 name: pulse-permissions-logs
 accessModes:
 - ReadWriteMany
 capacity: 10Gi
 class:

Specifies the security context for all Pods in the service
##
podSecurityContext: {}

Resource requests and limits
ref: http://kubernetes.io/docs/user-guide/compute-resources/
##
resources:
 limits:
 memory: "1Gi"
 cpu: "500m"

```

---

---

```

requests:
 memory: "400Mi"
 cpu: "50m"

HPA Settings
Not supported in this release!
hpa:
 enabled: false

Priority Class
ref: https://kubernetes.io/docs/concepts/configuration/pod-priority-
preemption/
##
priorityClassName: ""

Node labels for assignment.
ref: https://kubernetes.io/docs/user-guide/node-selection/
##
nodeSelector: {}

Tolerations for assignment.
ref: https://kubernetes.io/docs/concepts/configuration/taint-and-
toleration/
##
tolerations: []

Pod Disruption Budget Settings
podDisruptionBudget:
 enabled: false

Affinity for assignment.
Ref: https://kubernetes.io/docs/concepts/configuration/assign-pod-
node/#affinity-and-anti-affinity
##
affinity: {}

```

- Update values in the values-override-permissions.yaml file (GKE):

```

Default values for permissions.
This is a YAML-formatted file.
Declare variables to be passed into your templates.

* Image configuration
image:
 tag: ""
 pullPolicy: IfNotPresent
 registry: ""
 imagePullSecrets: [name: ""]

* Tenant info
tenant identification, or empty for shared deployment
tenant:
 # Tenant UUID
 id: ""
 # Tenant SID (like 0001)
 sid: ""

common configuration.
config:
 dbName: ""
 # set "true" when need @host added for username
 dbUserWithHost: true
 # set "true" for CSI secrets

```

---

```

mountSecrets: false
Postgres config map name
postgresConfig: "pulse-postgres-configmap"
Postgres secret name
postgresSecret: "pulse-postgres-secret"
Postgres secret key for user
postgresSecretUser: "META_DB_ADMIN"
Postgres secret key for password
postgresSecretPassword: "META_DB_ADMINPWD"
Redis config map name
redisConfig: "pulse-redis-configmap"
Redis secret name
redisSecret: "pulse-redis-secret"
Redis secret key for access key
redisSecretKey: "REDIS01_KEY"

* Configuration for the Configuration Server Proxy container
csproxy:
define domain for the configuration host
params:
 cfgHost: "tenant-.voice."
resource limits for container
resources:
minimum resource requirements to start container
requests:
minimal amount of memory required to start a container
memory: "200Mi"
minimal CPU to reserve
cpu: "50m"
resource limits for containers
limits:
maximum amount of memory a container can use before being evicted
by the OOM Killer
memory: "2Gi"
maximum amount of CPU resources that can be used and should be tuned
to reflect
what the application can effectively use before needing to be
horizontally scaled out
cpu: "1000m"
securityContext:
runAsUser: 500
runAsGroup: 500

* Common log configuration
log:
target directory where log will be stored, leave empty for default
logDir: ""
path where volume will be mounted
volumeMountPath: /data/log
log volume type: none | hostpath | pvc
volumeType: pvc
log volume hostpath, used with volumeType "hostpath"
volumeHostPath: /mnt/log
log PVC parameters, used with volumeType "pvc"
pvc:
 name: pulse-permissions-logs
 accessModes:
 - ReadWriteMany
 capacity: 10Gi
 class:

Specifies the security context for all Pods in the service

```

---

---

```

##
podSecurityContext:
 fsGroup: null
 runAsUser: null
 runAsGroup: 0
 runAsNonRoot: true

Resource requests and limits
ref: http://kubernetes.io/docs/user-guide/compute-resources/
##
resources:
 limits:
 memory: "1Gi"
 cpu: "500m"
 requests:
 memory: "400Mi"
 cpu: "50m"

HPA Settings
Not supported in this release!
hpa:
 enabled: false

Priority Class
ref: https://kubernetes.io/docs/concepts/configuration/pod-priority-preemption/
##
priorityClassName: ""

Node labels for assignment.
ref: https://kubernetes.io/docs/user-guide/node-selection/
##
nodeSelector: {}

Tolerations for assignment.
ref: https://kubernetes.io/docs/concepts/configuration/taint-and-toleration/
##
tolerations: []

Pod Disruption Budget Settings
podDisruptionBudget:
 enabled: false

Affinity for assignment.
Ref: https://kubernetes.io/docs/concepts/configuration/assign-pod-node/#affinity-and-anti-affinity
##
affinity: {}

```

**Install the permissions helm chart:** To install the permissions helm chart, run the following command:

```
helm upgrade --install "pulse-permissions-" pulsehelmrepo/permissions --wait --version="" --namespace=pulse -f values-override-permissions.yaml
```

If installation is successful, the exit code 0 appears.

**Validate the permissions helm chart:**

To validate the permissions helm chart, run the following command:

```
kubectl get pods -n=pulse -l "app.kubernetes.io/name=permissions,app.kubernetes.io/instance=pulse-permissions-"
```

Verify that the command report all pulse-permissions pods as Running, for example:

---

| NAME                                  | READY | STATUS  | RESTARTS | AGE   |
|---------------------------------------|-------|---------|----------|-------|
| pulse-permissions-100-c5ff8bb7d-jl7d7 | 2/2   | Running | 2        | 2d20h |

## Troubleshooting

### **Check init-tenant helm chart manifests:**

To output the manifest into the **helm-template** directory, run the following command:

```
helm template --version= --namespace=pulse --debug --output-dir helm-template pulse-init-tenant- pulsehelmrepo/init-tenant -f values-override-init-tenant.yaml
```

### **Check dcu helm chart manifests:**

To output the dcu Helm chart manifest into the **helm-template** directory, run the following command:

```
helm template --version= --namespace=pulse --debug --output-dir helm-template pulse-dcu-pulsehelmrepo/dcu -f values-override-dcu.yaml
```

### **Check lds helm chart manifests:**

To output the lds chart manifest into the **helm-template** directory, run the following command:

```
helm template --version= --namespace=pulse --debug --output-dir helm-template pulse-lds-pulsehelmrepo/lds -f values-override-lds.yaml
```

### **Check permissions Helm chart manifests:**

To output the Helm chart manifest into the **helm-template** directory, run the following command:

```
helm template --version= --namespace=pulse --debug --output-dir helm-template pulse-permissions pulsehelmrepo/permissions -f values-override-permissions.yaml
```