Hewlett Packard
Enterprise

HPE Ezmeral Runtime Enterprise 5.6 Documentation

Published: May 2024
Edition: 5.6.0

HPE Ezmeral Runtime Enterprise 5.6 Documentation

Abstract

HPE Ezmeral Container Platform is a unified container platform built on open source Kubernetes and designed for both cloud-native applications and non-cloud-native applications running on any infrastructure either on-premises, in multiple public clouds, in a hybrid model, or at the edge.

Published: May 2024
Edition: 5.6.0

Issues and Workarounds

This topic describes issues and workarounds in version 5.6.x of HPE Ezmeral Runtime Enterprise.

This topic describes issues and workarounds in HPE Ezmeral Runtime Enterprise version 5.6.x.

Installation Issues (prior releases)

The following issues were identified in HPE Ezmeral Runtime Enterprise 5.4.1. Unless otherwise noted, these issues also apply to later releases.

EZCP-2639: Add-Ons missing after installing HPE Ezmeral Runtime Enterprise on a reused Controller or Shadow host

Symptom: After you uninstall HPE Ezmeral Runtime Enterprise from a Controller or Shadow host, and then reuse that host as a Controller or Shadow host in a new deployment, expected system add-ons are not displayed on the Application Configurations screen when creating or editing a Kubernetes cluster.

Cause: The uninstall process did not delete the `hpe-cp-manifest` RPM on the host. Consequently, during the installation of HPE Ezmeral Runtime Enterprise on the reused host, the correct manifest RPM is not installed.

Workaround: Manually reinstall the manifest by entering the following command:

```
yum reinstall hpe-cp-manifest
```

To reuse a host:

After you uninstall HPE Ezmeral Runtime Enterprise from a host that will be used in another deployment, if the host was a Primary Controller or Shadow Controller host, erase the `hpe-cp-manifest` RPM:

- If this host is running RHEL/CentOS, enter the following command:

  ```
  yum erase hpe-cp-manifest
  ```

- If this host is running SLES, enter the following command:

  ```
  zypper rm hpe-cp-manifest
  ```

Upgrade Issues (5.6.x)

The following issues were identified in HPE Ezmeral Runtime Enterprise 5.6.x. Unless otherwise noted, these issues also apply to later releases.

EZCP-3854: After upgrading the **HPE Ezmeral Runtime Enterprise** platform version, then upgrading the Kubernetes add-on versions, then upgrading the Kubernetes version, some pods fail.

Symptom: Some pods fail after performing the following upgrades in order:

1. Upgrading the HPE Ezmeral Runtime Enterprise platform with Settings > Updates > Update.
2. Upgrading the required add-ons with the Kubernetes add-ons upgrade script.
3. Upgrading the Kubernetes version with Clusters > Upgrade Kubernetes > Confirm Upgrade.

Instead of entering Running state, some pods such as `kubeflow` and `airflow` fail. For example:

```
kubeflow              katib-mysql-5bf95ddfcc-gdvc4                 0/1
ContainerCreating   0             76m
kubeflow              minio-6bdd6c645f-p7j4x                     0/2     Init:0/2
0           76m
kubeflow              minio-console-747896b76-6ld4m                  0/1     Init:0/1
0           76m
kubeflow              ml-pipeline-5766c8b8bf-db5cr                 1/2
CrashLoopBackOff    19 (54s ago)     76m
```

Cause: This issue is caused by an incorrect list of namepsaces excluded from OPA Gatekeeper policy constraints.

Workaround: To correct this issue, add all namespaces in the global config to the list of excluded namespaces for OPA Gatekeeper.

Proceed as follows:

1. Use SSH to access the Kubernetes master node.
2. Run the following command to fix the `hpecp-global-config`:

   ```
   kubectl -n hpecp patch hpecpconfig hpecp-global-config --type=json -p "
   [{'op':'replace','path':'/spec/reservedNamespaceNames','value':
   [default,ezmysql,hpecp-falco,istio-system,kubeflow-jobs,ezctl,gatekeeper-
   system,hpe-sparkoperator,hpe-storage,hpe-system,knative-eventing,kubeflow,hpe-
   csi,hpe-secure,kube-node-lease,mapr-external-info,prism-ns,hpe-
   externalclusterinfo,hpe-templates-compute,hpecp-cert-manager,kubeflow-
   operator,ezml-model-mgmt,airflowop-system,kd-spark,knative-serving,kubernetes-
   dashboard,velero,auth,kd-mlops,airflow-base,hpe-nodesvc,hpecp-observability,kube-
   system,cert-manager,hpe-ldap,hpecp-bootstrap,kiali-operator,kube-
   public,argocd,hpe-nfscsi,hpecp,kd-apps,kubeflow-user-example-com]}]"
   ```

3. On the Kubernetes master node, create the following Python script:

   ```
   // Python script to add reservedNamespaceNames to excluded list of gatekeeper
   config
   import os,json

   system_namespaces = os.popen("kubectl get hpecpconfig -n hpecp -o jsonpath=\"
   {.items[0].spec.reservedNamespaceNames}\"").read()

   sna = json.loads(system_namespaces)

   system_namespaces_array = map(lambda x: str(x), sna)

   patch_string = "kubectl patch config config -n gatekeeper-system --type=json -p \"
   [{'op':'replace','path': '/spec/match/0/excludedNamespaces', 'value': %s}]\""%
   (list(set(system_namespaces_array)))

   os.popen(patch_string)
   ```

   This script fetches `reservedNamespaceNames` from `hpecp.global.config` and appends it to the list of exclued namespaces.

4. Run the Python script:

   ```
   # python <python-script-name>
   ```

   For example:

   ```
   # python gatekeeper_update_excluded_namespaces.py
   ```

Upgrade Issues (5.6.x)

The following issues were identified in HPE Ezmeral Runtime Enterprise 5.6.x. Unless otherwise noted, these issues also apply to later releases.

EZCP-3808: Kiali dashboard is not accessible, in Kubernetes 1.24.X or later versions, and HPE Ezmeral Runtime Enterprise 5.6.0 and earlier versions

Symptom: In earlier Kubernetes versions, when a service account was created, a token would be automatically created,.This token in the tenant namespace is required to access the kiali dashboard . In Kubernetes versions 1.24.x or later, this token does not get created automatically, and must be created manually with hpecp-agent.

Workaround: Create the service account token, by executing the following command:

```
kubectl apply -f - <<EOF
apiVersion: v1
kind: Secret
metadata:
```

```
  name: <token-name>
  namespace: <namespace>
  annotations:
    kubernetes.io/service-account.name: <serviceaccount-name>
type: kubernetes.io/service-account-token
EOF
```

Enter `<token-name>` and `<serviceaccount-name>` for the name of the kiali service account in the tenant namespace. The kiali service account name will be displayed as `<tenant-name>-kiali-service-account`

> **NOTE:** The kiali pod will come up before you create this service account token. You must delete the kiali pod in the namespace after the service account token is created. Then, a new kaili pod gets created automatically, and that pod uses the service token.

EZESC-1521: **`bds-worker`** on controller fails to start during upgrade from 5.3.6 to 5.5.1, or later version

Symptom: When you are upgrading from HPE Ezmeral Runtime Enterprise5.3.6 to 5.5.1, or later versions, upgrade may fail and rollback, as `dtap.ko.signed` gets deleted by new RPM.

In HPE Ezmeral Runtime Enterprise 5.5.1 or later versions, the RPM does not include `dtap.ko.signed` binary, and only includes `dtap.ko` binary. So if `dtap.ko.signed` is used before the upgrade, perform the following workaround to solve the issue.

Workaround:

> **IMPORTANT:** This workaround must be performed on each of the three controllers, and also on each of the workers.

On the primary controller, check if the dtap driver is loaded, by running `lsmod | grep` command, for example:

```
[root@mip-bd-vm134 ~]# lsmod | grep dtap
dtap 196679163 0
```

- If the output is empty, then this workaround is not needed.
- If the output shows that the dtap driver is loaded, check the log file `/var/log/bluedata/bds-worker.log` for **dtap.ko** binary, as follows:
  1. If `/sbin/insmod /opt/bluedata/common-install/data_server/drivers/dtap.ko` is available, this workaround is not needed.
  2. If `/sbin/insmod /opt/bluedata/common-install/data_server/drivers/dtap.ko.signed` is available, run the following command:

     ```
     bdconfig --set bdshared_install_nodtapdriver=true
     ```

  3. Make sure that `/sbin/insmod /opt/bluedata/common-install/data_server/drivers/dtap.ko` is available in the `bds-worker.log` file.

EZCP-3742: After upgrading HPE Ezmeral Runtime Enterprise to a newer version, the Istio add-on appears in the UI but is not deployed, and edit actions on the cluster fail.

Symptom: After upgrading HPE Ezmeral Runtime Enterprise to a newer version, the Istio add-on appears as enabled in the HPE Ezmeral Runtime Enterprise UI, but is not deployed on the backend. Edit actions on the Kubernetes cluster fail until the cluster is submitted with the Istio add-on deployed.

Workaround: You must execute the Kubernetes add-ons upgrade script after upgrading HPE Ezmeral Runtime Enterprise to a newer version. See Kubernetes Add-On Upgrade Script.

Upgrade Issues (5.5.0)

EZML-2059: Upgrading a Kubernetes cluster with a Kubeflow add-on in HPE Ezmeral Runtime Enterprise might fail.

Symptom: If your Kubernetes cluster has an existing Kubeflow add-on, the Kubernetes cluster upgrade might fail with the following message in the platform controlller logs within namespaces `kubeflow`, `knative-serving`, `knative-eventing`:

```
Cannot evict pod as it would violate the pod's disruption budget
```

Cause: This issue is caused by attempting an upgrade on a Kubernetes cluster with a version of Kubeflow lower than 1.6 enabled.

Workaround:

1. Execute the following commands:

   ```
   kubectl delete pdb -n knative-serving --all
   ```

   ```
   kubectl delete pdb -n knative-eventing --all
   ```

   ```
   kubectl delete pdb -n kubeflow --all
   ```

2. Re-run the Kubernetes cluster upgrade, as described in Upgrading Kubernetes.

Upgrade Issues (5.4.x)

EZCP-2582: Upgrading Kubeflow on HPE Ezmeral Runtime Enterprise requires assistance.

If your environment includes Kubeflow and you are upgrading HPE Ezmeral Runtime Enterprise, contact Hewlett Packard Enterprise support for assistance before you begin the upgrade. Several manual steps must be performed to replace the existing version of Kubeflow with the new version of Kubeflow.

HPE Ezmeral Data Fabric on Kubernetes Issues (5.4.1)

The following issues were identified in HPE Ezmeral Runtime Enterprise 5.4.1. Unless otherwise noted, these issues also apply to later releases.

EZSPA-661: HPE Ezmeral Data Fabric on Kubernetes pods and Livy pods not able to resolve AD users

Symptom: HPE Ezmeral Data Fabric on Kubernetes pods and Livy pods could not submit any queries successfully. These queries will fail, if customer's AD/LDAP servers do not support TLS version 1.3. You might encounter error `key too small`.

Workaround: Contact Hewlett Packard Enterprise Technical support for assistance.

HPE Ezmeral Data Fabric on Kubernetes Issues (Prior Releases)

The following issues were identified in a version of HPE Ezmeral Runtime Enterprise prior to version 5.5.0. Unless otherwise noted, these issues also apply to later releases.

EZKDF-627: With HPE Ezmeral Data Fabric on Kubernetes version 1.5.0, If a cldb or mfs pod is deleted, **`mrconfig`** info instances may show an incorrect number of instances.

Symptom: With HPE Ezmeral Data Fabric on Kubernetes version 1.5.0, if you delete a `cldb` or `mfs` pod, `mrconfig` info may show an incorrect number of instances.

Workaround: After the pod is restarted, and is in healthy state, restart MFS repeatedly up to three times, until it shows the correct number. For example, use the following commands to restart MFS upto three times:

```
sudo touch /opt/mapr/kubernetes/maintenance
/opt/mapr/initscripts/mapr-mfs restart
/opt/mapr/initscripts/mapr-mfs restart
/opt/mapr/initscripts/mapr-mfs restart
sudo rm /opt/mapr/kubernetes/maintenance
To verify mfs instances count,
Run mrconfig info instances
```

EZKDF-710: With HPE Ezmeral Data Fabric on Kubernetes version 1.5.0, if a cldb or mfs pod is upgraded, **`mrconfig`** info instances may show an incorrect number of instances.

Symptom: With HPE Ezmeral Data Fabric on Kubernetes version 1.5.0, if you upgrade a `cldb` or `mfs` pod applying a new CR, and the change the cpu, memory and/or disk parameters, `mrconfig` info may show an incorrect number of instances.

Workaround: After the pod is restarted and is in healthy state, restart MFS repeatedly up to three times, till it shows the correct number. For example, use the following commands to restart MFS upto three times:

```
kubectl exec -it <mfs_pod> -n <cluster namespace> bash
Within the mfs_pod or CLDB pod, execute the following commands.
sudo touch /opt/mapr/kubernetes/maintenance
/opt/mapr/initscripts/mapr-mfs restart
/opt/mapr/initscripts/mapr-mfs restart
/opt/mapr/initscripts/mapr-mfs restart
sudo rm /opt/mapr/kubernetes/maintenance
```

EZESC-563: ZooKeeper issue when running the **`saveAsNewAPIHadoopFile`** method on HPE Ezmeral Data Fabric on Kubernetes cluster.

Symptom: Running the `saveAsNewAPIHadoopFile` method on HPE Ezmeral Data Fabric on Kubernetes cluster generates the following error:

```
ERROR MapRZKRMFinderUtils: Unable to determine ResourceManager service
address from Zookeeper at xxx.xxx.xxx.xxx
```

Workaround: Set the `yarn.resourcemanager.ha.custom-ha-enabled` and `yarn.resourcemanager.recovery.enabled` property on `/opt/mapr/hadoop/hadoop-2.7.4/etc/hadoop/yarn-site.xml` configuration file to `false.`

EZKDF-109: After CLDB upgrade, MFS pods remain in a bad state.

Workaround: Use the following command to restart the MAST gateway:

```
kubectl exec -it -n <namespace> <mfs-pod> -- /opt/mapr/initscripts/mapr-
mastgateway restart
```

EZKDF-404: Clusters that Implement HPE Ezmeral Data Fabric on Kubernetes fail to start after Kubernetes version or HPE Ezmeral Runtime Enterprise version upgrade.

The following advice applies to deployments that have separate Data Fabric clusters, and deployments that combine compute and Data Fabric nodes in the same cluster. This advice does not apply to deployments that implement Embedded Data Fabric only.

Attempts to upgrade or patch Kubernetes or upgrade HPE Ezmeral Runtime Enterprise in deployments that include HPE Ezmeral Data Fabric on Kubernetes can fail in ways that require a significant number of recovery steps.

Contact your Hewlett Packard Enterprise support representative for upgrade assistance for any of the following:

- Upgrading or patching the Kubernetes version on any cluster that implements **HPE Ezmeral Data Fabric on Kubernetes**.
- Upgrading **HPE Ezmeral Data Fabric on Kubernetes** independently of an upgrade to HPE Ezmeral Runtime Enterprise.
- Upgrading HPE Ezmeral Runtime Enterprise on deployments that implement HPE Ezmeral Data Fabric on Kubernetes.

If your environment deploys a version of HPE Ezmeral Runtime Enterprise prior to version 5.3.5, Hewlett Packard Enterprise recommends that you upgrade to HPE Ezmeral Runtime Enterprise 5.3.5 or later before you add **HPE Ezmeral Data Fabric on Kubernetes.**

EZESC-563: ZooKeeper issue when running the **`saveAsNewAPIHadoopFile`** method on HPE Ezmeral Data Fabric on Kubernetes cluster.

Symptom: Running the `saveAsNewAPIHadoopFile` method on HPE Ezmeral Data Fabric on Kubernetes cluster generates the following error:

```
ERROR MapRZKRMFinderUtils: Unable to determine ResourceManager service
address from Zookeeper at xxx.xxx.xxx.xxx
```

Workaround: Set the `yarn.resourcemanager.ha.custom-ha-enabled` and `yarn.resourcemanager.recovery.enabled` property on `/opt/mapr/hadoop/hadoop-2.7.4/etc/hadoop/yarn-site.xml` configuration file to `false.`

EZKDF-109: After CLDB upgrade, MFS pods remain in a bad state.

Workaround: Use the following command to restart the MAST gateway:

```
kubectl exec -it -n <namespace> <mfs-pod> -- /opt/mapr/initscripts/mapr-
mastgateway restart
```

Open Policy Agent Issues (5.4.1)

The following issues were identified in HPE Ezmeral Runtime Enterprise 5.4.1. Unless otherwise noted, these issues also apply to later releases.

EZCP-2688: CSI drivers fail to install due to enforcement of the **`psp-privileged-container`** OPA policy.

Symptom: An attempt to install CSI drivers fails with the error `ReplicaFailure`, and gives the following message:

```
Error creating: admission webhook "validation.gatekeeper.sh" denied the request:
[psp-privileged-container] Privileged container is not allowed: csi-provisioner,
securityContext: {"privileged": true}
```

```
[psp-privileged-container] Privileged container is not allowed: direct-csi,
securityContext: {"privileged": true}
```

Workaround:

1. On the master node of the Kubernetes cluster, save the following Python script as `priv_constraint_update_excluded_namespaces.py`:

   ```
   import json,os

   csi_driver_system_namespace = sys.argv[1]

   system_namespaces = os.popen("kubectl get
   k8spspprivilegedcontainer.constraints.gatekeeper.sh/psp-privileged-container -
   o=jsonpath=\"{.spec.match.excludedNamespaces}\"").read()

   sna = json.loads(system_namespaces)

   system_namespaces_array = map(lambda x: str(x), sna)

   system_namespaces_array.append(csi_driver_system_namespace)

   patch_string = "kubectl patch
   k8spspprivilegedcontainer.constraints.gatekeeper.sh/psp-privileged-container --
   type=json -p \"[{'op':'replace','path': '/spec/match/excludedNamespaces', 'value':
   %s}]\""%(list(set(system_namespaces_array)))

   os.popen(patch_string)
   ```

2. On the master node of the Kubernetes cluster, execute the script with the following command:

   ```
   python priv_constraint_update_excluded_namespaces.py <csi-driver-namespace>
   ```

   `<csi-driver-namespace>` refers to the namespace in which you are creating the privileged container. For example, if you want to create the `direct-csi` pod in the namespace `csi`, then execute:

   ```
   python priv_constraint_update_excluded_namespaces.py csi
   ```

Kubernetes Issues (5.6.1)

The following issues were identified in HPE Ezmeral Runtime Enterprise 5.6.1. Unless otherwise noted, these issues also apply to later releases.

EZCP-3936: A Kubernetes cluster's **`kubelet`** service fails, and **`kubectl`** commands stop working.

> ***Symptom:*** A Kubernetes cluster on HPE Ezmeral Runtime Enterprise stops working correctly because its SSL certificates have expired. The `kubelet` service fails, and `kubectl` commands stop working.
>
> ***Workaround:*** Follow the steps in this guide to renew the SSL certificates: **Procedure for updating Kubernetes cluster certificates** .

Kubernetes Issues (5.6.0)

The following issues were identified in HPE Ezmeral Runtime Enterprise 5.6.0. Unless otherwise noted, these issues also apply to later releases.

EZCP-3741: The log for a deployed Kubernetes cluster shows the errors **`etcdserver: timed out`** and **`slow disk`**.

> ***Symptom:*** The log for a deployed Kubernetes cluster shows the errors `etcdserver: timed out` and `leader failed to send out heartbeat on time; took too long, leader is overloaded likely from slow disk`.
>
> For example:

```
Feb 07 14:29:36 example.hpecorp.net etcd[6249]: {"level":"warn","ts":"2023-02-
07T14:29:36.553-0800","caller":"etcdserver/server.go:1159","msg":"failed to revoke
lease","lease-id":"7602862df1792313","error":"etcdserver: request timed out"}
```

```
Feb 07 14:29:36 example.hpecorp.net etcd[6249]: {"level":"warn","ts":"2023-02-
07T14:29:36.651-0800","caller":"v3rpc/interceptor.go:197","msg":"request
stats","start time":"2023-02-07T14:29:29.650-0800","time
spent":"7.000923805s","remote":"127.0.0.1:50504","response
type":"/etcdserverpb.KV/Txn","request count":0,"request size":0,"response
count":0,"response size":0,"request content":""}

Feb 07 14:29:39 example.hpecorp.net etcd[6249]: {"level":"warn","ts":"2023-02-
07T14:29:39.128-0800","caller":"etcdserver/server.go:1159","msg":"failed to revoke
lease","lease-id":"4c87862de93218b3","error":"etcdserver: request timed out"}

Feb 07 14:29:39 example.hpecorp.net etcd[6249]: {"level":"warn","ts":"2023-02-
07T14:29:39.295-0800","caller":"etcdserver/raft.go:415","msg":"leader failed to send
out heartbeat on time; took too long, leader is overloaded likely from slow
disk","to":"973a665ee093f602","heartbeat-interval":"100ms","expected-
duration":"200ms","exceeded-duration":"161.316838ms"}
```

In some cases, other errors may occur. For example, the Kubernetes cluster might fail to enter a Ready state, with the bootstrap log for `hpecp-bootstrap-prometheus` displaying the error `UPGRADE FAILED`:

```
[jenkins@mip-bd-ap07-n3-vm01 install]$ kubectl logs hpecp-bootstrap-prometheus-
868c8b97d-hgx65 -n hpecp-bootstrap
Wed Jan  4 04:40:02 UTC 2023: Starting prometheus reconfigure process
Error: UPGRADE FAILED: pre-upgrade hooks failed: warning: Hook pre-upgrade kube-
prometheus-stack/templates/prometheus-operator/admission-webhooks/job-
patch/serviceaccount.yaml failed: etcdserver: request timed out
failed to reconfigure helm chart
configmap/hpecp-bootstrap-prometheus patched
```

Cause: This issue is caused by insufficent disk I/O when performing etcd operations. This issue can impact any add-on or pod running on a Kubernetes cluster that is also running high volume api-server operations simultaneously.

Workaround: To check whether your environment meets minimum disk speed requirements for etcd, you can run one of the etcd benchmark tools described in the **official etcd documentation** (link opens an external site in a new browser tab or window).

To ensure your environment has the required disk speed for etcd operations, Hewlett Packard Enterprise recommends using a solid state drive.

Kubernetes Issues (5.5.0)

The following issues were identified in HPE Ezmeral Runtime Enterprise 5.5.0. Unless otherwise noted, these issues also apply to later releases.

EZCP-3543: In a deployment that includes Embedded Data Fabric, deleting a Kubernetes cluster does not automatically delete the CSI state volume.

Symptom: In a deployment that uses Embedded Data Fabric, when you delete a Kubernetes cluster, the CSI state volume is not deleted automatically. The volume and a small (a few megabytes) file remain.

Workaround: After you delete the Kubernetes cluster, delete the CSI state volume manually. On the Controller, do the following:

1. Look for the following error message in `/var/log/bluedata/bds-mgmt.log`:

   ```
   got an error trying to delete snapshot state volume ("<cluster-id>")
   ```

   - If the log does not contain the error message, the volume was deleted successfully. No other actions are required.
   - If the log contains the error message, proceed to the next step.

2. From the log message, note the cluster ID of the deleted Kubernetes cluster.
   For example, in the following error message, the cluster ID is `10`:

   ```
   got an error trying to delete snapshot state volume("10")
   ```

3. Delete the CSI state volume by entering the following commands:

```
/opt/bluedata/ezpylib/bluedata/mapr/bds-mapr-config.py deleteVolume --vol-name
apps-k8s-<deleted-cluster-id>-k8s-csi-state
/opt/bluedata/ezpylib/bluedata/mapr/bds-mapr-config.py deleteHadoopDir --dir-name
/apps/k8s-<deleted-cluster-id>
```

For example, if the cluster ID is `10` the commands you enter are the following:

```
/opt/bluedata/ezpylib/bluedata/mapr/bds-mapr-config.py deleteVolume --vol-name
apps-k8s-10-k8s-csi-state
/opt/bluedata/ezpylib/bluedata/mapr/bds-mapr-config.py deleteHadoopDir --dir-name
/apps/k8s-10
```

Kubernetes Issues (5.4.3)

The following issues were identified in HPE Ezmeral Runtime Enterprise 5.4.3. Unless otherwise noted, these issues also apply to later releases.

EZCP-3070: Falco pods are in a **`CrashLoopBackOff`** state due to an incompatible runtime schema version.

Symptom: Falco pods installed on HPE Ezmeral Runtime Enterprise are in a `CrashLoopBackOff` state due to an incompatible runtime schema version. The pod logs show a `Runtime error`. For example:

```
Runtime error: Driver supports schema version 2.0.0, but running version needs
1.0.0.
```

Workaround:

1. Update the Falco kernel driver to the latest version.
2. Ensure the latest Falco pods are in the `hpecp-falco` namespace.
3. **If you are upgrading Falco pods** , you must use the latest `falco-no-driver` images. Download the latest images **here** (link opens an external site in a new browser tab or window).

Kubernetes Issues (5.4.0)

The following issues were identified in HPE Ezmeral Runtime Enterprise 5.4.0. Unless otherwise noted, these issues also apply to later releases.

EZCP-1925: When you delete a Kubernetes cluster, **`iptable`** settings do not get cleaned up on each associated Kubernetes host

Symptom: When you delete a Kubernetes cluster, `iptable` settings do not get cleaned up, on each associated Kubernetes host. Later, if you add these hosts to any other kubenetes cluster, `kubeproxy` uses the existing `iptable` rules to get routed to the appropriate pods. These `iptable` rules must be removed from the Kubernetes host, when you remove the host from the Kubernetes cluster.

Cause: As these existing `iptable` entries were not removed from the host, various networking routing problems may occur, when the same host is added to any other cluster.

Workaround:

You must manually delete `iptable` settings in the file. Contact Hewlett Packard Enterprise technical support to know how to delete the `iptable` settings.

EZCP-2036: Graphs on the Kubernetes Dashboard hanotherg on very large Kubernetes deployments.

Symptom: Graphs on the Kubernetes Dashboard fail to load when displaying information for large scale Kubernetes deployments (such as 1,000 nodes).

Workaround: None at this time.

EZCP-2097, EZESC-1103: Creating a Kubernetes cluster with optional add-ons enabled causes a delay in port service link readiness.

Symptom: Clicking the link for a service endpoint shows a `503 error`, or the links for service endpoints do not appear in the UI.

Cause: When creating Kubernetes clusters with the optional add-ons Istio, Kubeflow, Airflow, and Spark Operator, the HPE Ezmeral Runtime Enterprise gateway port mappings for Argo CD, Istio, Kubeflow, and Kiali NodePort services may take up to twenty minutes to become available after the cluster is ready.

Workaround:

- If the UI shows the links for the service endpoints, but clicking the link shows a `503 error`, then check that all pods are running and ready. Once the pods are in the ready state, they will become available.

- If the UI shows the cluster as ready, but the UI does not show the links for the service endpoints, then delete the `hpecp-agent` pod:

  ```
  kubectl -n hpecp delete pod $(kubectl -n hpecp get pod -l name=hpecp-agent -o
  jsonpath='{.items[0].metadata.name}')
  ```

  The pod will be re-created once the cluster enters the ready state. The services gateway port mappings will immediately be created once the `hpecp-agent` pod is running.

Kubernetes Issues (Prior Releases)

The following issues were identified in a version of HPE Ezmeral Runtime Enterprise prior to version 5.4.0. Unless otherwise noted, these issues also apply to later releases.

EZESC-542: On the Kubernetes Application screen, clicking an ingress service endpoint link, such as for Istio, returns an HTTP or HTTPS error.

Symptom: On the Service Endpoints tab of the Kubernetes Application screen, endpoint links are displayed for Kubernetes ingress controllers, such as the Istio ingress gateway, but clicking the links return HTTP or HTTPS 503 errors that indicate the service is unavailable or a secure connection could not be made.

Cause: HPE Ezmeral Runtime Enterprise automatically configures ingress gateway service endpoints when an ingress gateway such as `istio-ingress` is configured on a Kubernetes cluster. However, for most Kubernetes applications, there is no corresponding service that is automatically configured, so there is no service available through the endpoint.

Workaround: None. Ignore the service endpoint links.

See also EZKDF-404 in HPE Ezmeral Data Fabric on Kubernetes Issues (Prior Releases)

EZKDF-404, "Clusters that Implement HPE Ezmeral Data Fabric on Kubernetes fail to start after Kubernetes version or HPE Ezmeral Runtime Enterprise version upgrade," also applies to upgrading Kubernetes versions in HPE Ezmeral Runtime Enterprise 5.3.5 deployments that implement HPE Ezmeral Data Fabric on Kubernetes.

EZCP-1608, EZCP-2306, and EZCP-2358: When an application (e.g. Istio or Airflow) is deployed in the Kubernetes cluster, one or more worker nodes fail to upgrade the Kubernetes version

Symptom: When an application (e.g. Istio or Airflow) is deployed in the Kubernetes cluster, one or more worker nodes fail to upgrade the Kubernetes version, with the following errors:

- `Warning: one or more workers failed to upgrade` on the Kubernetes Cluster screen.
- `Upgrade error: Failed to drain node` error at the individual Kubernetes Host Status screen

This issue also occurs when the application user deploys PodDisruptionBudget (PDB) objects to the application workloads. For more information about PDB, see **https://kubernetes.io/docs/concepts/workloads/pods/disruptions/**

Cause: There are PDB objects for Istio (or any other application) resources with minimum replica as 1. This prevents the "kubectl drain" from succeeding during the Kubernetes upgrade.

Workaround: Execute the following commands on the Kubernetes Master before initiating the Kubernetes Upgrade from the Kubernetes Cluster screen. The following example is for Istio:

```
kubectl -n istio-system delete poddisruptionbudget/istiod
kubectl -n istio-system delete poddisruptionbudget/istio-ingressgateway
kubectl -n istio-system delete poddisruptionbudget/istio-egressgateway
```

> **NOTE:** This workaround can also be applied if the Kubernetes upgrade fails with `Failed to drain node` error on the Kubernetes hosts/workers. In such case, execute the preceeding `kubectl` commands on the Kubernetes Master, and continue with the Kubernetes upgrade on the remaining workers using the Retry Kubernetes Upgrade on Failed Workers action on the cluster from the Kubernetes Cluster screen.

Before doing Kubernetes Upgrade, make sure you have drained all the pods on the node. If an application has Pod disruption budget (PDB) violation, that pod will not get drained and Kubernetes upgrade will fail. This typically happens when you have smaller cluster with limited resources.

PDB voilation will show a similar message like:

```
kubectl drain mip-bd-vm694.mip.storage.hpecorp.net --delete-local-data
--ignore-daemonsets --timeout=5m
evicting pod airflow-base/af-base-nfs-0
evicting pod airflow-base/af-base-postgres-0
error when evicting pod "af-base-nfs-0" (will retry after 5s): Cannot
evict pod as it would violate the pod's disruption budget.
error when evicting pod "af-base-postgres-0" (will retry after 5s):
Cannot evict pod as it would violate the pod's disruption budget.
```

EZCP-561: When Istio mTLS is enabled in `STRICT` mode, the Kiali Dashboard and KubeDirector service endpoints are not accessible through NodePort

Symptom: When Istio is configured to use Mutual Transport Layer Security (mTLS) in `STRICT` mode, the following issues occur:

- None of the KubeDirector service endpoints are accessible through the NodePort service.
- If mTLS in `STRICT` mode is enabled in a tenant, the Kiali Dashboard is not accessible through NodePort. Clicking on the endpoint results in an error.

Workaround: If possible, configure Istio to use `PERMISSIVE` mode (the default mode).

EZESC-232: "Failed to pull image" ImagePullBackoff Errors received on Kubernetes clusters

When working with Kubernetes clusters in HPE Ezmeral Runtime Enterprise, you receive errors similar to the following:

```
Failed to pull image "bluedata/hpe-agent:1.1.5": rpc error: code = Unknown desc = Error response from daemon: toomanyrequests: You have reached your pull rate limit. You may increase the limit by authenticating and upgrading: https://www.docker.com/increase-rate-limit
```

Cause: Kubernetes clusters running on any version of HPE Ezmeral Runtime Enterprise can occasionally encounter problems caused by the pull rate limit that Docker Hub applies to all free and anonymous accounts. These limits can cause cluster creation and application deployment to fail. If Kubernetes pods in a non-Air-gap environment are failing to come into Ready state and are showing ImagePullBackoff or related errors, this is the most likely cause.

Workaround: Do one of the following:

- Wait until the current rate limiting timeout has expired, then re-try.
- Create a local image registry, then configure the air-gap settings to use that registry. For more information about air gap, see Kubernetes Air-Gap Requirements.

  NOTE:

  Hewlett Packard Enterprise strongly recommends performing air-gap configuration steps before adding Kubernetes hosts to the HPE Ezmeral Runtime Enterprise environment. Kubernetes hosts do not implement air-gap changes until the hosts are rebooted or the Kubernetes version is upgraded.

- Upgrade your Docker Hub account as described in **https://www.docker.com/increase-rate-limits** (link opens an external website in a new browser tab/window), then on all hosts, do the following:

  1. Execute a `docker login` operation with the credentials of the ugpgraded account.
     Docker will create or update its `config.json` file after a successful login (or you might want to use an existing configuration file).
  2. Ensure that kubelet uses the new `config.json` file by placing it in one of the known search locations kubelet uses for credential files:
     a. Create a `.docker` directory directly under the root of the filesystem and place the config.json file in that directory. For example: `/.docker/config.json`
     b. Restart kubelet:
        ```
        systemctl restart kubelet
        ```
     c. Verify that kublet has restarted:

```
systemctl status kubelet
```

Kubelet will then choose that `config.json` file and use the paid account that generated that config, ensuring that no image pull rate limit will be exceeded.

The following article (link opens an external website in a new browser tab/window) shows all the locations that kubelet searches for Docker credentials files:

https://kubernetes.io/docs/concepts/containers/images/#configuring-nodes-to-authenticate-to-a-private-registry

- Create a Docker proxy cache as described in the following article (link opens an external website in a new browser tab/window):

  https://docs.docker.com/registry/recipes/mirror/

EZCP-811: Webterms do not work for imported clusters. You will encounter an error if you try to start a webterm on an imported cluster.

Workaround: Execute the following command using either the Kubeconfig file used to import the cluster or a Kubeconfig file for the imported cluster downloaded from the HPE Ezmeral Runtime Enterprise as described in **Downloading Admin Kubeconfig**:

```
kubectl patch hpecpconfigs hpecp-global-config -n hpecp --type merge --patch
'{"spec":{"fsMount":{"enabled":false} } }'
```

After the command is issued, starting a webterm should not generate an error.

EZCP-823: Kubernetes Upgrade dialog empty or not showing latest Kubernetes version after upgrade to HPE Ezmeral Runtime Enterprise 5.3.x.

Workaround: Refresh the browser screen.

BDP-574: Unable to add a Kubernetes host when Platform HA (High Availability) is being enabled.

Workaround: Wait until Platform HA finishes before adding the Kubernetes host.

HAATHI-15093 : A GPU is visible in a non-GPU-requesting pod.

Symptom:When an app spawns on a device having a GPU, it is able to access the GPU even when there are no requests for one. This is a known issue with the NVIDIA k8s-device-plugin.

Workaround: You must manually create an environment variable in the `Kubedirectorcluster` YAML that 'hides' the GPU from the App. The variable is named `NVIDIA_VISIBLE_DEVICES` with value `VOID`.

For example:

```
apiVersion: "kubedirector.bluedata.io/apiVersion" kind: "KubeDirectorCluster"
metadata: name: "sample-name" spec: app: sample-app roles: - id: samplerole
resources: requests: memory: "4Gi" cpu: "2" limits: memory: "4Gi" cpu: "2" env: -
name : "NVIDIA_VISIBLE_DEVICES" value: "VOID"
```

Spark on Kubernetes Issues (5.6.0)

The following issues were identified in HPE Ezmeral Runtime Enterprise 5.6.0. Unless otherwise noted, these issues also apply to later releases.

Livy Session: PySpark code in Livy session results in an error

Symptom: Running PySpark code in Livy session returns the following error:

```
'JavaPackage' object is not callable
```

Cause: PythonSQLUtils is not imported in java_gateway.jvm

Workaround: Perform explicit imports by running the following commands in Livy session:

```
from py4j.java_gateway import java_import
jvm = SparkContext._jvm
java_import(jvm, "org.apache.spark.sql.api.python.*")
```

EZSPA-1037: Data Fabric DB OJAI jobs fails with ANTLR incompatibility exception

Symptom: Data Fabric DB OJAI jobs will fail with ANTLR incompatibility exception.

Workaround: Contact Hewlett Packard Enterprise Technical Support.

EZSPA-1010: Some pyspark APIs do not work, due to python version compatibility

Symptom: Some pyspark APIs do not work as expected.

Cause: Some pyspark APIs do not work, due to python version compatibility issues.

Workaround: Contact Hewlett Packard Enterprise Technical Support.

EZSPA-1008: Livy session fails when group names for users in Active Directory are not POSIX compliant.

Symptom: When you start a Livy session on HPE Ezmeral Runtime Enterprise as `user1` and group names for users in Active Directory are not POSIX compliant, the following error occurs:

```
groupadd: 'Domain Users' is not a valid group name
```

Cause: The main group name of the `user1` user in Active Directory database is `Domain Users`. `Domain Users` group name contains a space symbol which makes it an invalid group name in Linux.

Workaround: The group names for users in Active Directory need to be POSIX compliant. The set of **valid user names** in POSIX is defined as **lower and upper case ASCII letters, digits, period, underscore, and hyphen** . Note that hyphen is not permitted as first character of the user name.

Spark on Kubernetes Issues (5.5.0)

The following issues were identified in HPE Ezmeral Runtime Enterprise 5.5.0. Unless otherwise noted, these issues also apply to later releases.

EZCP-3572: Add-ons upgrade for Spark Operator add-on fails after upgrading from 5.4.x to 5.5.x or later version of HPE Ezmeral Runtime Enterprise.

Symptom: When you perform the following steps:

1. Create a Kubernetes cluster in 5.4.x and 5.5.x.
2. Enable the Spark Operator add-on.
3. Upgrade the platform to 5.5.x from 5.4.x.
4. Run Kubernetes add-ons upgrade script.

The Spark Operator add-on upgrade fails and you'll see the following warning message:

```
2022-10-25 04:40:37,032 INFO add-ons upgrade failed: cluster
    state: warning
```

Cause: Spark Operator is running with an old Spark Operator image in a cluster.

Workaround: Contact Hewlett Packard Enterprise support team.

Spark on Kubernetes Issues (5.4.1)

The following issues were identified in HPE Ezmeral Runtime Enterprise 5.4.1.

EZCP-2624: Launching KubeDirector application tiles for Spark results in Config Error.

Symptom: After upgrading to HPE Ezmeral Runtime Enterprise 5.4.1 from HPE Ezmeral Runtime Enterprise 5.3.x; when you launch the Livy, Spark History Server, Spark Thrift Server, and Hive Metastore in standard tenants after running the `sparkapps.sh` command, you will get the following error:

```
Config Error Detail: execution of app config failed: configure failed with exit status
{120}
Last Config Data Generation: 1
Last Configured Container:
docker://d7f5c968a029f494889da2d06d26ff066b52f342538e4ad822e5d88638e5718
1
Last Connection Version: 0
Last Known Container State: unresponsive
Last Setup Generation: 1
Start Script Stderr Message: Error from server (Forbidden): configmaps "cluster-cm"
is forbidden: User "system:serviceaccount:nonml:ecp-tenant-member-sa" cannot get
resource "configmaps" in API group "" in the namespace "<namespace>"
```

```
Start Script Stdout Message: Error: expected at most two arguments, unexpected
arguments: image.tag=<spark-tenant-services-image-tag>
Failed to exec: helm install <spark-tenant-services-name> /<path-to-spark-tenant-
services-chart> --namespace --set image.tag= <spark-tenant-services-image-tag>
```

Cause: The `ecp-tenant-member-sa` service account was added in HPE Ezmeral Runtime Enterprise 5.4.0. The `member` rolebinding do not have the `ecp-tenant-member-sa` service account binding on the tenants that were created prior to 5.4.0 releases.

Workaround: Delete the existing `member` rolebinding. To delete the rolebinding, run:

```
kubectl delete rolebinding <name_of_rolebinding> -n <tenant-namespace>
```

Deleting an existing `member` rolebinding will automatically create a new `member` rolebinding with `ecp-tenant-member-sa` service account binding providing an access to the current tenant services.

Spark on Kubernetes Issues (5.4.0)

The following issues were identified in HPE Ezmeral Runtime Enterprise 5.4.0. Unless otherwise noted, these issues also apply to later releases.

EZESC-1211: Unable to update Helm charts for Hive Metastore, Livy, Spark History Server, and Spark Thrift Server using **`kubectl apply`** command.

Symptom: When you run the `kubectl apply -f <spark-services-yaml-file>` command to update the Helm charts for Hive Metastore, Livy, Spark History Server, and Spark Thrift Server in the same cluster, update fails with the following error message:

```
Error: release: already exists
Failed to exec: helm install <spark-services-release-name> /<path-to-helm-chart> --
namespace <tenant-namepace> --set image.tag=202202161825P150 --set
eventlogstorage.kind=pvc --set eventlogstorage.storageSize=10Gi --set
eventlogstorage.pvcStoragePath=/<path-to-storage>
```

Cause: Only one instance of Hive Metastore, Livy, Spark History Server, and Spark Thrift Server can be installed in the single tenant within a cluster. When you try to install the multiple instances of the Spark services, Helm with throw an error since the same cluster name is used as the release name for all the Helm installation of the Spark services.

Workaround: To update the Helm charts for Hive Metastore, Livy, Spark History Server, and Spark Thrift Server, see Updating Helm Charts for Spark Services.

EZSPA-576: Authentication fails on Spark tenant services when the permissions for External User Groups on tenants are set at a higher level than the External Groups on Data Fabric clusters.

Symptom: The authentication on tenant services, for example, Livy, Spark History Server, Spark Thrift Server, Hive Metastore fails with the following error:

```
INFO login.PasswordAuthentication: Failed authentication for user qa1:
javax.security.auth.login.FailedLoginException: Permission denied.
ERROR server.BasicAuthHandler: User Principal is null while trying to authenticate
with Basic Auth
```

Cause: You have set the permissions for External User Groups on tenants at a higher level than the External Groups on Data Fabric clusters.

Workaround: Ensure the permissions for External Groups on Creating Kubernetes Cluster step is set at a broader level than the permissions for External User Groups on Creating New K8s Tenant step. See Kubernetes Tenant/Project External Authentication and Creating a New Kubernetes Cluster .

EZSPA-566: Spark Thrift Server restarts continuously when Hive Metastore ConfigMap is not set.

Symptom: When you do not enter the ConfigMap with `hive-site.xml` configuration of the Hive Metastore during the Spark Thrift Server installation, Spark Thrift Server restarts continuously and gives the following error:

```
Error: Unable to instantiate
org.apache.hadoop.hive.ql.metadata.SessionHiveMetaStoreClient
```

Cause: The ConfigMap with `hive-site.xml` configuration of the Hive Metastore was not identified and is therefore missing during the Spark Thrift Server installation.

Workaround: You can enter ConfigMap values using YAML or HPE Ezmeral Runtime Enterprise GUI and there are three separate workarounds for three situations. See Integrating Spark Thrift Server with Hive Metastore .

EZSPA-508: Spark submit fails when using the third-party dependency jars on MinIO.

Symptom: When you submit the Spark applications configured using the third-party dependency jars on MinIO, the `spark-submit` fails with the following exception:

```
Exception in thread "main" com.amazonaws.SdkClientException: Unable to execute
HTTP request
```

Cause: Unable to add CLI options to the spark-submit command.

Workaround: None at this time.

EZSPA-504: Livy and Hive Metastore integration fails in the non Data Fabric type tenants.

Symptom: Livy and Hive Metastore integration fails in non Data Fabric (none) type tenants with the following message:

```
java.lang.RuntimeException: java.io.IOException: Could not create FileClient err: 104
```

Workaround: None at this time.

EZSPA-446: Spark application fails when `jars` option is set with non-file URI scheme for SparkR.

Symptom:

When you set the jars option for DataTap with non-file schema, for example, `- local:///opt/bdfs/bluedata-dtap.jar`, Spark applications fail with the following exception:

```
Exception in thread "main" java.lang.IllegalArgumentException: URI scheme is not
"file"
```

Cause: The `jars` option is set with non-file URI scheme for SparkR.

```
deps:
    jars:
      - local:///opt/bdfs/bluedata-dtap.jar
```

Workaround: To integrate SparkR with DataTap, configure SparkR with the file URI scheme.

For example: Set the `files` option to add DataTap jar to classpath for SparkR.

```
deps:
   files:
     - local:///opt/bdfs/bluedata-dtap.jar
```

EZSPA-442: Authentication fails on SAML environment.

Symptom: When you authenticate Livy on SAML, authentication fails with the following message:

```
INFO login.PasswordAuthentication: Failed authentication for user <user1>:
javax.security.auth.login.FailedLoginException: Permission denied.
```

Workaround: None at this time.

EZSPA-232: Livy and Hive Metastore integration fails when using DataTap to access the data from same Hive Metastore.

Symptom: When you create Livy sessions in the DataTap integration enabled environment, you are unable to use Hive Metastore. For example: You are unable to view the tables created in one Livy session from the another Livy session.

Workaround: To use the Hive Metastore in Livy, remove `"spark.driver.extraClassPath"` option from Livy session configurations. However, in this case, you are unable to pass the application dependencies using `dtap` in Livy.

EZCP-1808: After upgrading to HPE Ezmeral Runtime Enterprise 5.4.0, launching KubeDirector Spark applications as Kubernetes Tenant Admin or Kubernetes Tenant member, fails with an error.

Symptom: After you upgrade to HPE Ezmeral Runtime Enterprise 5.4.0, if you launch KubeDirector Spark applications as Kubernetes Tenant Admin or Kubernetes Tenant member, applications fail with an error.

Workaround:

1. Access Kubernetes cluster as Cluster Administrator and download the `kubeconfig` file. To download the `kubeconfig` file, you can either follow the steps in Downloading Admin Kubeconfig or SSH to Kubernetes Master using following command:

   ```
   kubectl get hpecptenant -n hpecp
   ```

Example of output:

```
NAME       AGE
hpecp-tenant-4   9h
hpecp-tenant-5   8h
hpecp-tenant-6   8h
```

2. Replace the **<tenant-name>** with the desired value using following command:

   ```
   kubectl edit hpecptenant <tenant-name> -n hpecp
   ```

3. Add **Patch** verb at `kubedirectorclusters` resources for following Role Ids:
   Patch verb is added in the following examples:

   Add in **Default Admin RBACS:**

   ```
   - apiGroups:
         - kubedirector.hpe.com
         resources:
         - kubedirectorclusters
         - kubedirectorapps
         verbs:
         - get
         - list
         - watch
         - create
         - update
         - delete
         - patch
   ```

   In **Default Member RBACS:**

   ```
   - apiGroups:
         - kubedirector.hpe.com
         resources:
         - kubedirectorclusters
         verbs:
         - create
         - update
         - delete
         - get
         - list
         - watch
         - patch
   ```

   Also, Add at **secrets** resources of **Default Member RBACS**

   ```
   - apiGroups:
         - ""
         resources:
         - secrets
         verbs:
         - get
         - create
         - update
         - patch
   ```

4. Save and exit the file.

Spark on Kubernetes Issues (Prior Releases)

The following issues were identified in HPE Ezmeral Runtime Enterprise 5.3.x. Unless otherwise noted, these issues also apply to later releases.

- You may encounter a `certificate generation failed` error when executing `spark-submit` or `spark- shell` commands in the `tenantcli` or `spark-client` pods. You can avoid this issue by executing the command using the `--conf spark.ssl.enabled=false` option. Doing so disables encryption for the Spark driver UI. The UI is not exposed outside of the Kubernetes cluster, so it is safe to use this option.
- The pod restarts continuously instead of transitioning to an **Error** state if `hivesitesource` points to an existing ConfigMap that does not have a `hive-site.xml` key.
- The Autoticket-generation feature does not work for scheduled Spark applications. Manually create your user secrets using the `ticketcreator.sh` script in the `tenantcli` pod for this purpose.

KubeDirector Issues (5.4.0)

The following issues were identified in HPE Ezmeral Runtime Enterprise 5.4.0. Unless otherwise noted, these issues also apply to later releases.

EZML-810: The Jupyter notebook does not appear in the UI.

> ***Symptom:*** When creating or launching a Jupyter notebook with `kubectl apply`, the Jupyter notebook does not appear in the UI. However, the Jupyter endpoint is visible.
>
> ***Workaround:*** Add the user ID in the label in `nb.yaml` as:
>
> ```
> ---
> metadata:
>   labels:
>     kubedirector.hpe.com/createdBy:
> ```
>
> This will prevent user ID mismatch, allowing the logged-in user to view the Jupyter notebook.

EZML-994: When opening an R-kernel in Jupyter notebook, a **`TypeError`** occurs.

> ***Symptom:*** When opening an R-kernel in Jupyter notebook, a pop-up appears with the message `TypeError`.
>
> ***Cause:*** This is a known issue with JupyterLab 2.3.
>
> ***Workaround:*** Click Dismiss and proceed with your R session.

EZML-1037: When submitting a KFP job in a KD notebook using Kale, an **`RPC Error`** occurs.

> ***Symptom:*** When uploading a pipeline in KD notebook, the message `An RPC Error has occurred` is displayed.
>
> ***Workaround:*** Before creating the KFP client, execute:
>
> ```
> %kubeRefresh
> ```
>
> After successful execution, recreate the KFP client. For detailed instructions about the prerequisites of Kale, see: `examples/kubeflow/kale/README.ipynb`.
> If the error persists after you have executed `%kubeRefresh` and the kubeconfig file is still fresh, then dismiss the `RPC Error` message and restart the Kale extension.

KubeDirector Issues (Prior Releases)

The following issues were identified in a version of HPE Ezmeral Runtime Enterprise prior to version 5.4.0. Unless otherwise noted, these issues also apply to later releases.

EZESC-1066: "503 Service Unavailable" error for MinIO or MySQL after MLflow cluster pod automatically restarts.

> ***Symptom:*** When a pod managed by an MLflow cluster is deleted and then automatically recreated, attempts to access the MinIO service endpoint or MySQL result in the error:
>
> ```
> 503 Service Unavailable
> ```
>
> This issue occurs on pods that are managed by an MLflow cluster that is configured with persistent storage (PVC) only.
>
> ***Cause:*** When an MLflow cluster is configured with persistent volumes, KubeDirector does not automatically execute startup scripts when the controller restarts. However, state information for MySQL is not retained by the persistent volume, and MinIO is not restarted because it is not a `systemd` process. The startup script (`startscript`) is responsible for configuring and starting services such as MySQL and MinIO.

Workaround: From the Kubernetes master node, manually execute the startup script for the pod by executing the following commands, where `<kdcluster_pod>` is the name of the pod and `<tenant_ns>` is the tenant namespace:

```
kubectl exec -it <kdcluster_pod>  -n <tenant_ns> bash
opt/guestconfig/appconfig/startscript --configure
exit
```

EZESC-217: 503 Service Unavailable error when connecting to training engine instance

Attempts to connect to a training engine instance from a JupyterLab Notebook fail. When you attempt to connect to the service endpoint of the training engine instance in a browser, the error "503 Service Unavailable" is returned.

Cause: One of the possible cause is when the High Availability Proxy (HAProxy) service is not running on the gateway host. If you are not sure whether HAProxty is running or not, contact HPE support for assistance.

Workaround: If this is HAProxy issue, then start (or restart) the HAproxy service. From the master node, enter the following command::

```
kubectl exec -c app -n <tenant-namespace> <trainingengineinstance-loadbalancer-
pod> - systemctl restart haproxy
```

Airflow Issues (5.5.0)

The following issues were identified in HPE Ezmeral Runtime Enterprise 5.5.0. Unless otherwise noted, these issues also apply to later releases:

EZML-2026: Airflow does not work on Kubernetes clusters set with a custom pod domain.

Symptom: Airflow does not work on Kubernetes clusters set with a pod domain other than `cluster.local`.

Workaround: For each of the four listed resources, perform the steps described below:

- `sts af-cluster-airflowui`
- `sts af-cluster-scheduler`
- `cm af-cluster-airflowui`
- `cm af-cluster-scheduler`

1. Edit the resource:

   ```
   kubectl edit <resource-name> -n <airflow-tenant-ns>
   ```

   For example:

   ```
   kubectl edit sts af-cluster-airflowui -n <airflow-tenant-ns>
   ```

2. Delete all labels in the `metadata` section for the resource.
   For example:

   ```
   <...>
   metadata:
     <...>
     labels:
       custom-resource: v1alpha1.AirflowCluster
       custom-resource-name: af-cluster
       custom-resource-namespace: <...>
       using: <...>
   <...>
   ```

3. Replace all occurrences of `af-base-sql.airflow-base.svc.cluster.local` with `af-base-sql.airflow-base`.

Kubeflow Issues (5.5.0)

The following issues were identified in HPE Ezmeral Runtime Enterprise 5.5.0. Unless otherwise noted, these issues also apply to later releases:

- Error "Could not find CSRF cookie XSRF-TOKEN in the request" is returned when creating a Kserve model in Kubeflow UI exposed via

HTTP. For more information about this issue, see: **https://github.com/kubeflow/manifests/pull/2262**

Kubeflow Issues (5.4.0)

The following issues were identified in HPE Ezmeral Runtime Enterprise 5.4.0. Unless otherwise noted, these issues also apply to later releases.

- Kubeflow does not support groups. See:
  https://github.com/kubeflow/kubeflow/issues/4188
- The Dex authentication component does not support the use of LADP/AD external groups. See:
  https://github.com/dexidp/dex/issues/1562

EZML-616: Only a single AD server configuration is supported, even when multiple LDAP server addresses are provided.

> ***Symptom:*** Only a single AD server configuration is supported, even when multiple LDAP server addresses are provided on the Cluster Configuration tile. A single LDAP server adderss is set when DEX is created, and does not change when the chosen LDAP server address becomes unavailable or inaccessable.
>
> ***Cause:*** Kubeflow DEX does not support the use of multiple AD/LDAP servers for authentication.
>
> ***Workaround:*** If the selected LDAP server that was set during installation becomes inaccessable, specify a different server as follows:

1. Get the current configuration for DEX from the secret:

   ```
   kubectl get secrets -n auth dex-config-secret -o "jsonpath={.data['config\.yaml']}" |
   base64 -d
   ```

   Copy the returned value and save it.

2. In the copied value, locate the string which starts with the substring `host:`. After this substring, replace the existing domain string with the domain for the correct server. There should be only one domain.
   For example:

   ```
   host: example.com:636
   ```

3. Open any base64 encoder and encode the whole modified configuration.
   The following are links to base64 encoders:

   - **https://www.base64encode.org/**
   - **base64**

4. Update the secret:

   ```
   kubectl edit secrets -n auth dex-config-secret
   ```

   Replace the value after `config.yaml` with your modified and encoded value.

5. Save the changes.

6. Restart the DEX deployment:

   ```
   kubectl rollout restart deploy -n auth dex
   ```

EZML-1475: When you deploy a model using InferenceService, the KNative Serving controller fails to fetch the image used by the model from the airgap docker image registry.

> ***Symptom:*** When you deploy a model using InferenceService, the InferenceService fails to become `READY`. The KNative Serving controller fails to fetch the image used by the model from the airgap docker image registry and gives the message: `x509: certificate signed by unknown authority.`
>
> ***Workaround:***

1. After deploying Kubeflow, run the following as Cluster Administrator:

   ```
   kubectl edit cm -n knative-serving config-deployment
   ```

2. Add the following under the `data` field:

   ```
   registriesSkippingTagResolving: "<host-name-of-your-airgap-image-registry>"
   ```

Kubeflow Issues (Prior Releases)

The following issues were identified in a version of HPE Ezmeral Runtime Enterprise prior to version 5.4.0. Unless otherwise noted, these issues also apply to later releases.

- If you specify an external user group, the group is not taken into account when a user logs in to Kubeflow. The user will be allowed to log in to Kubeflow regardless of to which groups that the user belongs. See the following for more information: https://stackoverflow.com/questions/58276195/mandate-group-search-condition-in-dex-ldap-coonector
- Occasionally, the `v1beta1.webhook.cert-manager.ioapiservice` is unavailable for a period of time after deploying Kubeflow services (applying a manifest). To make the service available, restart the service as follows:

  ```
  kubectl delete apiservices v1beta1.webhook.cert-manager.io
  ```
- There is an issue with Istio authorization for HTTP traffic in which the KFServing predict request returns `503 Service Unavailable`. See the following for more information: https://github.com/kserve/kserve/issues/820

Katib Issues (Prior Releases)

The following issues were identified in a version of HPE Ezmeral Runtime Enterprise prior to version 5.4.0. Unless otherwise noted, these issues also apply to later 5.4.x releases.

The following issues occur in Katib, which is a Kubernetes-native project for automated machine learning.

- Suggestion pods running after experiment completes: https://github.com/kubeflow/katib/issues/1043
- Katib with Kubernetes 1.19 and higher: https://github.com/kserve/kserve/issues/1197

  https://github.com/kubeflow/katib/issues/1395

General Platform Issues (5.6.1)

The following issues were identified in HPE Ezmeral Runtime Enterprise 5.6.1. Unless otherwise noted, these issues also apply to later 5.6.x releases.

EZCP-3949: On Python 3 hosts (RHEL 8 and SLES 15.4), clicking Support/Troubleshooting in the HPE Ezmeral Runtime Enterprise UI might result in an error. Attempting to generate SOS logs on the Support/Troubleshooting page results in an error.

> ***Symptom:*** On RHEL 8 OS, clicking Support/Troubleshooting in the HPE Ezmeral Runtime Enterprise UI might return a `404 Page Not Found` error. Attempting to generate SOS logs on the Support/Troubleshooting page results in the following error message:
>
> ```
> Got an error while performing the operation
> Additional Details: Error
> ```
>
> ***Workaround:*** None at this time.

General Platform Issues (5.6.0)

The following issues were identified in HPE Ezmeral Runtime Enterprise 5.6.0. Unless otherwise noted, these issues also apply to later 5.6.x releases.

EZCP-3844: When you upgrade to HPE Ezmeral Runtime Enterprise 5.6.0, and perform the Cluster Upgrade from 1.21.x to 1.22.x, some pods are in CrashLoopBackOff state.

> ***Symptom:***When you upgrade to HPE Ezmeral Runtime Enterprise 5.6.0, and perform the Cluster Upgrade from 1.21.x to 1.22.x, some pods are in `CrashLoopBackOff` state.
>
> ***Cause:*** Pods that are running on the worker are unable to access the pods that are running on the master node, due to missing routes on the master node.
>
> ***Workaround:*** On the master node that is missing the routes to some or all the workers, do the following:
>
> 1. Find the canal pod running in that master node, using the command:
>
>    ```
>    CANAL_POD_NAME=$(kubectl get pods -n kube-system -o wide | grep <hostipaddr>
>    | grep canal | awk '{print $1}')
>    ```

2. Delete the canal pod using the following command:

   ```
   kubectl delete -n kube-system pod ${CANAL_POD_NAME}
   ```

3. Restart the pod. The missing routes will be restored.

General Platform Issues (5.4.1)

The following issues were identified in HPE Ezmeral Runtime Enterprise 5.4.1. Unless otherwise noted, these issues also apply to later 5.4.x releases.

EZCP-2669: When an attempt to enable High Availability (HA) on HPE Ezmeral Runtime Enterprise fails, the log repeats error messages multiple times.

Symptom: When an attempt to enable HA on HPE Ezmeral Runtime Enterprise fails, the `bds_mgmt.log` repeats error messages multiple times, making it difficult to read and debug the issue.

Workaround: To view log files that capture the whole configuration, look at `/var/log/bluedata/install/enableha*`.

General Platform Issues (Prior Releases)

The following issues were identified in a version of HPE Ezmeral Runtime Enterprise prior to version 5.4.0. Unless otherwise noted, these issues also apply to later 5.4.x releases.

EZESC-253: After upgrade, UI becomes inaccessible and browser displays internal error 500.

Symptom: Following an upgrade to HPE Ezmeral Runtime Enterprise 5.3, the UI for the controller is inaccessible, failing with internal error 500. The system fails with the error:

```
No space left on device
```

The `/var/lib/monitoring/logs` director contains large `hpecp-monitoring_access` and `hcep-monitoring_audit` logs.

Cause:

Dangling Search Guard indexes exist after the upgrade. You might see log entries similar to the following:

```
[WARN ][o.e.g.DanglingIndicesState] [xxxxx] [[searchguard/xxx-xxxxxxxxx-xxxxxxx]] can not be imported as a dangling index, as index with same name already exists in cluster metadata
```

Workaround: Search Guard indexes are not used by HPE Ezmeral Runtime Enterprise 5.3. You can remove the Search Guard indexes, delete the large log files, and resume monitoring on the HA nodes.

1. Remove the Search Guard indexes using one of the following methods:

   - If Elasticsearch is running, you can delete the Search Guard index through the Elasticsearch REST API.

     For example:

     ```
     curl --insecure  -u $(bdconfig --getvalue bdshared_elasticsearch_admin):$(bdconfig --getvalue bdshared_elasticsearch_adminpass) --silent -X DELETE https://localhost:9210/searchguard
     ```

   - If Elasticsearch is not able to run, you must identify and delete SearchGuard indexes manually:

     a. Identify the indexes.
        Change the directory to `/var/lib/monitoring/elasticsearch/nodes/0`, then enter the following command:

        ```
        find . -name "state-*.st" -print | xargs grep searchguard
        ```

        All the indices that are from Search Guard are displayed. You can use matching entries to determine which indexes to remove.

        For example, this line identifies a state file related to that contains the word Search Guard. The index name is part of the full file path of that file. In this example, the index name: `xtSTTUb7RgOeUlCXWH8dAg`

        ```
        ./indices/xtSTTUb7RgOeUlCXWH8dAg/_state/state-45.st matches
        ```

     b. Use the `rm` command to remove the index.

        For example:

```
rm -rf ./indices/xtSTTUb7RgOeUlCXWH8dAg
```

2. Delete the large log files.

3. On the HA cluster nodes only, restart monitoring. For example, from the controller, enter the following command:

   ```
   HPECP_ONLY_RESTART_ES=1 /opt/bluedata/bundles/hpe-cp-*/startscript.sh --action
   enable_monitoring
   ```

BDP-2879: The Python ML and DL Toolkit lists a deleted Training cluster in the %attachments list.

Workaround: Ignore the deleted cluster. No jobs will be submitted to deleted clusters.

BDP-841: When enabling multi-domain authentication, the password field must be filled out for all domains before submitting changes to any domain, otherwise the web interface will fail to react.

Workaround: None at this time.

HAATHI-15068: Unable to create a tenant or FS mount if any host is down.

Workaround: Consider removing the Kubernetes host from the Kubernetes cluster or wait until the host is back up and running.

HAATHI-12781: When HPE Ezmeral Runtime Enterprise is installed on RedHat 7.x systems, system reboots are observed under heavy load.

Workaround: Update the RedHat kernel to the newest kernel version.

HAATHI-14220: Adding a license when one or more Worker hosts is in an error state may cause an error.

Workaround: Remove the affected hosts before uploading the license.

HAATHI-12810: After restarting the container that handles monitoring, the service may fail to restart and will show red in the Services tab of the Platform Administrator Dashboard screen.

Workaround: Restart the service manually from the Controller host by executing the command `systemctl restart bds-monitoring`.

HAATHI-12829: For RHEL/CentOS 7.x OS installs, if a server is physically rebooted, some services that depend on network services may be down as shown in the Services tab of the Platform Administrator Dashboard screen.

Workaround: Execute the following commands on the Controller host:

```
$ systemctl stop NetworkManager
$ systemctl disable NetworkManager
$ systemctl restart network
$ systemctl restart bds-controller
$ systemctl restart bds-worker
```

HAATHI 13253: HPE Ezmeral Runtime Enterprise does not compress or archive Nagios log files.

Workaround: Manually archive files as needed in the `/srv/bluedata/nagios` directory on the Controller.

EZCP-463: Platform HA must be enabled before creating Kubernetes clusters.

Workaround: If you enable Platform HA after Kubernetes cluster creation, then reconfigure host monitoring as follows:

1. On a Kubernetes master node bring up the monitoring bootstrap deployment:

   ```
   kubectl -n hpecp-bootstrap scale deployment/hpecp-bootstrap-hpecp-monitoring --
   replicas=1
   ```

2. Exec into the bootstrap pod

   ```
   kubectl -n hpecp-bootstrap exec -it $(kubectl -n hpecp-bootstrap get -o
   jsonpath='{.items[0].metadata.name}' pods -l name=hpecp-bootstrap-hpecp-
   monitoring) -c hpecp-monitoring - bash
   ```

3. Delete running deployment (if exist):

   ```
   kubectl -n kube-system -delete -f /workspace/monitoring.yaml
   ```

4. Export / change any needed bds_xxx env variables (e.g. redeploy after HA enable)

   ```
   export bds_ha_enabled='Yes'
   ```

```
      export bds_ha_nodes='<controller IP list>'
```

(e.g. `export bds_ha_nodes='16.143.21.35,16.143.21.237,16.143.21.38'`)

5. Run startscript install:

   ```
   /usr/local/bin/startscript --install
   ```

   This places `metricbeat.yaml` in the workspace folder.

6. Deploy metricbeat deployment:

   ```
   kubectl -n kube-system create -f /workspace/monitoring.yaml
   ```

7. Exit the bootstrap pod and scale down bootstrap deployment:

   ```
   kubectl -n hpecp-bootstrap scale deployment/hpecp-bootstrap-hpecp-monitoring --
   replicas=0
   ```

BDP-685: Kubernetes cluster creation fails with an "internal error."

> ***Workaround:*** Remove the Kubernetes hosts, verify that all system clocks are synchronized, and then re-add the hosts and recreate the Kubernetes cluster.

BDP-852: All uploaded files and new folders created by AD/LDAP users via the HPE Ezmeral Runtime Enterprise FS mounts interface will have root ownership and full permission for all tenant members.

> ***Workaround:*** None at this time.

BDP-1868: An admin kubeconfig file downloaded from an imported external Kubernetes cluster will not contain expected edits from the HPE Ezmeral Runtime Enterprise web interface.

> ***Workaround:*** Manually edit the kubeconfig file after download.

Application Issues (Prior Releases)

The following issues were identified in a version of HPE Ezmeral Runtime Enterprise prior to version 5.4.0. Unless otherwise noted, these issues also apply to later 5.4.x releases.

HAATHI-14109: When using CEPH for persistent storage, a discrepancy between the client and server versions will cause HPE Ezmeral Runtime Enterprise to fail to load App Store images with the error "Failed to map the volume."

> ***Workaround:*** Remove the persistent storage until the client and server versions are the same.

HAATHI-14192: Running the Impala shell on a container where the Impala daemon is not running.

> ***Workaround:*** Use the `-i` option to refer to the worker node. For example, `impala-shell -i <worker hostname>`.

HAATHI-14461: Notebooks with a name that includes one or more spaces cannot be committed to GitHub.

> ***Symptom:*** When working in an AI/ML project that includes a GitHub repository, creating a Jupyterhub notebook with a name that includes one or more spaces will cause an error when trying to commit that notebook to GitHub.
> ***Workaround:*** Do not include any spaces when naming a Jupyterhub notebook.

HAATHI-10733: Hive jobs that use DataTap paths may fail with a SemanticException error.

> ***Cause:*** When Hive creates a table, the location where the table metadata is stored comes from the Hive configuration parameter `fs.defaultFS` by default (which will point to the cluster file system). If a Hive job references DataTap paths outside of the file system where the table metadata is stored, then the job will fail with a `SemanticException` error because Hive enforces that all data sources must come from the same file system.
> ***Workaround:*** Explicitly set the table metadata location to a path on the same DataTap that you will use for the job inputs and/or outputs, using the `LOCATION` clause when creating the table. For example, if you intend to use the **TenantStorage** DataTap, you would set the table metadata location to some path on that DataTap such as:

```
CREATE TABLE docs (c1 INT, c2 STRING) LOCATION
'dtap://TenantStorage/hive-table-docs'
```

HAATHI-12546: Some http links in applications running on HPE Ezmeral Runtime Enterprise show the hostname of the instance. These links will not work when HPE Ezmeral Runtime Enterprise is installed with the non-routable network option.

> ***Workaround:*** See "Configure Client to use Hostname instead of IP Address, below."

HAATHI-13254: If a user updates an app inside a container instead of via the App Store screen, then cluster expansion will fail.

Workaround: Expand the cluster before performing the upgrade. Once the update is complete, edit `classpath` to point to the correct .jar files, such as `hadoop-common-*.jar`.

DOC-9: Cloudera Manager reports incorrect values for a node's resources.

Cause: Cloudera Manager accesses the Linux `/proc` file system to determine the characteristics of the nodes it is managing. Because container technology is used to implement virtual nodes, this file system reports information about the host rather than about the individual node, causing Cloudera Manager to report inflated values for a node's CPU count, memory, and disk.
Workaround: Use the web interface to see a node's virtual hardware configuration (flavor).

DOC-19: Spark applications may wait indefinitely if no free vCPUs are available.

Cause: This is a general Spark behavior, but it is worth some emphasis in an environment where various virtual hardware resources (possibly in small amounts) can be quickly provisioned for use with Spark.
Workaround:
A Spark application will be stuck in the **Waiting** state if all vCPUs in the cluster are already considered to be in-use (by the Spark framework and other running Spark applications). In Spark version 1.5, the thrift server is configured to use 2 vCPUs on the Spark master node by default. You can reduce this to 1 vCPU by editing the total-executor- cores argument value in the `/etc/init.d/hive-thriftserver` script, and then restarting the thrift server (`$ sudo service hive-thriftserver restart`).

K8S-1887: A MapR software version alarm is generated, indicating that “One or more services on the node are running an unexpected version.” The alarm includes a “recommended action” to stop and restart the node.

Workaround: You can ignore the alarm and recommended action for container-based HPE Ezmeral Data Fabric.

CUDA and GPU Issues (Prior Releases)

The following issue applies to HPE Ezmeral Runtime Enterprise relese 5.3.5 and later.

EZESC-964: CUDA applications fail to run on A100 GPU HGX hosts with NVIDIA NVLink switches

Symptom: CUDA applications fail to run on A100 GPU HGX hosts that have NVIDIA NVLink switches.

Workaround: On A100 GPU HGX systems with NVIDIA NVLink switches, you must install the and configure the NVIDIA Fabric Manager on the system before adding it as a host to HPE Ezmeral Runtime Enterprise.

1. Install the NVIDIA Fabric Manager on the host.
   For instructions, see the **Fabric Manager for NVIDIA NVSwitch Systems User Guide** (link opens an external website in a new browser tab or window)

2. Change the Fabric Manager service start-up options to ensure that the Manager service is started before the kubelet service:
   In the `[Unit]` section of the `nvidia-fabricmanager.service` file, add the following line:

   ```
   Before=kubelet.service
   ```

   For example:

   ```
   [Unit]
   Description=FabricManager service
   Before=kubelet.service
   After=network-online.target
   Requires=network-online.target
   ```

3. Verify the NVLink switches topology to ensure that “NV12” appears between peer GPUs. This result indicates that all 12 NVLinks are trained and available for full bi-directional bandwidth.
   For example, execute the command: `nvidia-smi topo -m`

   The following is an example of a portion of the output:

   ```
           GPU0    GPU1    GPU2
   GPU0     X      NV12    NV12
   GPU1    NV12     X      NV12
   GPU2    NV12    NV12     X
   ```

4. After you add the host to HPE Ezmeral Runtime Enterprise and to the Kubernetes cluster, verify the CUDA Kubernetes application by doing the following:

   a. Create a test pod: `kubectl create -f cuda-test.yaml`

For example, the following pod executes the `nvidia/samples:vectoradd-cuda10.2` test:

```
apiVersion: v1
kind: Pod
metadata:
  name: nvidia-cuda-test
spec:
  restartPolicy: OnFailure
  containers:
  - name: cuda-vector-add
    image: "nvidia/samples:vectoradd-cuda10.2"
    resources:
      limits:
         nvidia.com/gpu: 1
```

b. Verify that the test passed by executing the following command:

```
kubectl logs nvidia-cuda-test
```

Example result:

```
[Vector addition of 50000 elements]
Copy input data from the host memory to the CUDA device
CUDA kernel launch with 196 blocks of 256 threads
Copy output data from the CUDA device to the host memory
Test PASSED
Done
```

For more information about the installing and configuring the Fabric Manager, the following NVIDIA documentation (link opens an external website in a new browser tab or window):

- NVIDIA HGX A100 Software User Guide
- Fabric Manager for NVIDIA NVSwitch Systems User Guide