
Troubleshoot

Operations Bridge Suite 2017.11

Troubleshoot

This section provides information that can help you troubleshoot problems you may encounter when installing and using the ITOM Container Deployment Foundation and the Operations Bridge Suite.

For more information, see these topics:

- [Common problems and limitations](#)
- [Logs](#)
- [Manual verification commands](#)
- [Support toolset](#)

Common problems and limitations

You may encounter the following problems and limitations when installing or administering the Container Deployment Foundation and the Operations Bridge Suite:

- [Management Portal is not accessible](#)
- [Management Portal is not accessible: nginx controller is Pending](#)
- [Management Portal is not accessible: Gateway time out](#)
- [Login to Management Portal is not possible: IDM service is not ready yet](#)
- [Reboot does not work. Pods are in status CrashLoopBackOff.](#)
- ["502 Bad Gateway" error when attempting to launch OMi](#)
- [No server connection: invalid character "{" in host name](#)
- [Pod is in ImagePullBackOff or ErrImagePull status: Image not found](#)
- [Pod is in ImagePullBackOff or ErrImagePull status: Error while pulling image](#)
- [Worker node installation fails with a Flannel related error](#)
- ["503 nginx error" when attempting to run the Suite Installer](#)
- [CDF IDM service fails to start](#)
- ["Failed to start flannel" error](#)
- ["Fail to pull image" error](#)
- [Error while updating suite images \(storage configuration step\)](#)
- [Suite uninstallation fails](#)
- [Login to Docker Hub fails](#)
- [CDF installation failure errors](#)
- [Error when upgrading CDF on a node](#)

- [Worker node does not start](#)
- [Suite images cannot be downloaded](#)
- [idm-postgresql cannot access /var/pgdata](#)
- [System error: read parent: connection reset by peer](#)
- [Renew token failed in http_code=403](#)
- [Docker pull doesn't work: Error while pulling image](#)
- [Pod is in Pending status: Insufficient CPU, insufficient memory](#)

Management Portal is not accessible

Description

After the installation of the Container Deployment Foundation, the Management Portal cannot be accessed at `https://<external_access_host>:5443`.

Possible solutions

- Make sure you entered the correct URL and port.
- Make sure you can access the host: `ping <external_access_host>`
- Check your browser's proxy settings.
- Check the installation logs in `/opt/kubernetes/install-<timestamp>.log`.
- Empty the NFS folder and then reinstall the Container Deployment Foundation.
- See also [Management Portal is not accessible: nginx controller is Pending](#), [Management Portal is not accessible: Gateway time out](#) and [Login to Management Portal is not possible: IDM service is not ready yet](#).

Management Portal is not accessible: nginx controller is Pending

Description

After the installation of the Container Deployment Foundation, the Management Portal cannot be accessed at `https://<external_access_host>:5443`.

When running `kubectl get pods --all-namespaces`, the nginx ingress controller status is Pending.

Cause and solution

The map hash bucket size might be too small. Check if that is the case by running the following commands:

```
kubectl describe nginx-ingress-controller-u69gg
```

```
kubectl logs nginx-ingress-controller-u69gg
```

If an error is displayed similar to nginx: [emerg] could not build map_hash, increase the map_hash_bucket_size as follows:

1. Access the file /opt/kubernetes/objectdefs/nginx-ingress.yaml
2. Locate the specified map_hash_bucket_size (32 by default) and increase it, for example to 128
3. Run the following commands to recreate the nginx-ingress.yaml file:

```
kubectl delete -f /opt/kubernetes/objectdefs/nginx-ingress.yaml
```

```
kubectl create -f /opt/kubernetes/objectdefs/nginx-ingress.yaml
```

4. *Optional.* If you get a warning about failed scheduling, the scheduling constraints could not be fulfilled. Execute the following command to fix this:

```
kubectl label nodes role=loadbalancer --all
```

The nginx pod container should then be started automatically.

5. After the OMi configuration, you must repeat steps 2 and 3 for the OMi nginx controller located at
/var/vols/itom/core/suite-install/opsbridge/output/suite-ingress-controller-configmap.yaml

Management Portal is not accessible: Gateway time out

Description

After the installation of the Container Deployment Foundation, the Management Portal cannot be accessed at https://<external_access_host>:5443. The Docker daemon cannot be started, and displays the error message `Gateway time out` when logging into IDM.

Cause and solution

Kubernetes might not be running. Run the following commands to start Kubernetes:

```
cd $K8S_HOME/bin  
./kube-start.sh
```

Login to Management Portal is not possible: IDM service is not ready yet

Description

After the installation of the Container Deployment Foundation, the Management Portal cannot be accessed at https://<external_access_host>:5443. The login failure error `The IDM service is not ready yet` is displayed, and the pods `autopass-lm`, `idm`, and `suite-installer` all have the status `CrashLoopBackOff`.

Solution

1. Run the following command:

```
kubectl delete -f autopass-lm.yaml; kubectl delete -f autopass-pg.yaml; kubectl delete -f idm.yaml;
kubectl delete -f idm-pg.yaml; kubectl delete -f suite.yaml
```

2. Delete the subfolders located in the NFS subdirectories <NFS_HOME>/baseinfra-1.0/autopass_db, <NFS_HOME>/baseinfra-1.0/idm_db, and <NFS_HOME>/baseinfra-1.0/suite_db.
3. Run the following command:

```
kubectl create -f idm-pg.yaml; kubectl create -f idm.yaml; kubectl create -f autopass-pg.yaml; kubectl
create -f autopass-lm.yaml; kubectl create -f suite.yaml
```

Reboot does not work. Pods are in status CrashLoopBackOff.

Description

After attempting to reboot, the pods have the status CrashLoopBackOff.

Cause and solution

This is related to the vault-renewal container, which does not get a valid token. You have to delete the failed pods. Once the pods are deleted, they are recreated automatically and should run without error.

You can get the status of all pods with the following command:

```
kubectl get pods --all-namespaces
```

First delete all failed database related pods (`suite-db`, `idm-postgresql`, `postgresql-aplm`). Next, delete all failed pods within the namespace `core`. After that delete all failed pods within the namespace `opsbridge`, starting with `postgres`, `ucmdb`, `omi`, `redis`, `bvd`, `obr-server`, `obr-rc`).

Use the following command to delete the failed pods within the namespaces specified above:

```
kubectl delete pod <pod_name> --namespace <pod_namespace>
```

"502 Bad Gateway" error when attempting to launch OMi

Description

After the installation of the Operations Bridge Suite, a 502 Bad Gateway error is displayed when trying to access OMi.

Cause and solution

The 502 error is displayed because OMi is not yet up and running. Depending on the host machine, it might take up to one hour for OMi to start after the initial configuration.

No server connection: invalid character "{" in host name

Description

A connection to the server could not be established. The log displays that the invalid character "{" is used in the host name.

Cause and solution

The firewall might still be enabled on the NFS server. Make sure that the firewall is disabled.

Pod is in ImagePullBackOff or ErrImagePull status: Image not found

Description

After the installation of the Container Deployment Foundation, one of the pods has the status `ImagePullBackOff` or `ErrImagePull`. When running the command `kubectl describe pod <pod_name> -n <namespace>`, the following error message is displayed:

```
Image <image_name> not found
```

Cause and solution

Make sure the images are pushed into the private docker registry. To confirm, run the following command:

```
docker pull <image_name>
```

Pod is in ImagePullBackOff or ErrImagePull status: Error while pulling image

Description

After the installation of the Container Deployment Foundation, one of the pods has the status `ImagePullBackOff` or `ErrImagePull`. When running the command `kubectl describe pod <pod_name> -n <namespace>`, the following error message is displayed:

```
Error while pulling image: Get http://localhost:5000/v1/repositories/xxx:  
dial tcp [::1]:5000: getsockopt: connection refused
```

Cause and solution

To resolve this issue, delete the Docker registry and the registry proxy pods, and then restart them.

Worker node installation fails with a Flannel related error

Description

Setting up one or multiple worker nodes fails during the Container Deployment Foundation installation due to an error related to Flannel.

Cause and solution

To troubleshoot and resolve this issue, do the following:

- Double check if the FQDN is resolved to the correct IP address on the master node.
- On the master node, run `kube-restart.sh`
- Reinstall the worker node from the Management Portal.

"503 nginx error" when attempting to run the Suite Installer

Description

After the installation of the Container Deployment Foundation, a 503 Nginx error is displayed when trying to access the Suite Installer.

Cause and solution

This error might be displayed because the time on the master and worker nodes is different. To resolve this issue, synchronize the time on your nodes by using, for example, NTP or VMWare tools.

CDF IDM service fails to start

Description

After the installation of the Container Deployment Foundation, the CDF IDM service doesn't start.

Solution

Change the proxy settings in the autopass-lm.yaml file to a null value:

```
env:  
- name: http_proxy  
  value: ""  
- name: https_proxy  
  value: ""
```

"Failed to start flannel" error

Description

The CDF installation fails with the error message "failed to start flannel".

Solution

1. Comment out the following line in the etc/hosts file of each node. Skip this step if the host file does not contain this line.
127.0.1.1 <masterfqdn> <masterhostname>
2. Restart all the master nodes.
3. Install the worker nodes from the CDF Management Portal.

"Fail to pull image" error

Description

When installing CDF on a node with multiple network adapters, the installation fails with a "fail to pull image" error.

Solution

Follow these steps to reinstall the CDF on the node:

- a. Uninstall CDF on this node with the **uninstall.sh** command.
- b. Reboot the server.

- c. Configure the FLANNEL_IFACE parameter in the install.properties file.
- d. Follow the installation instructions to install CDF on that node.

Error while updating suite images (storage configuration step)

Description

When you access **SUITE > Management > Actions > Update** to upgrade the suite images, and then click **NEXT** in the storage configuration page, the system displays an error message and the content fails to load.

Solution

To solve this problem, log on to the `https://{{ingresshost}}/update` page in a new tab in the same browser. Then, reload the frame in the suite upgrade page where the error message appears.

Suite uninstallation fails

Description

After CDF fails to install a suite, you cannot uninstall the suite by clicking **SUITE > Management > Actions > Uninstall**.

Solution

To solve this problem, run the following commands to recreate the suite-db pods:

```
kubectl get pods -n core
```

```
kubectl delete pod <pod name of suite-db> -n core
```

Alternatively, restart the virtual machine on which the suite-db pod is installed.

Login to Docker Hub fails

Description

You cannot log in to Docker Hub.

Solution

To solve this problem, try these solutions:

- Make sure the user name and password are correct.
- Make sure the Docker HTTP proxy is configured as follows:
`/usr/lib/systemd/system/docker.service.d/http_proxy.conf`
- Make sure the host HTTP proxy is configured as follows:
`export http_proxy https_proxy`

CDF installation failure errors

Description

When trying to install CDF, you receive errors related to an installation failure.

Solution

To solve this problem, try these solutions:

Error when upgrading CDF on a node

Description

You receive error messages when you try to upgrade CDF on a node.

Solution

1. Follow these steps to troubleshoot the upgrade failure:

Run the following command:

```
source /etc/profile
```

2. (Optional) If the upgrade failed on a master node that was not the first master node that was stopped, run the following command to remove the etcd member from the etcd cluster:
upgrade.sh -d /<Parameter file path>/CDF_upgrade_parameters.txt

You will receive a message that resembles one of the following:

Remove member successfully

Not found this node in etcd cluster

3. Check whether the backup-complete file was created in the */<backup directory>/CDF_201703_backup* directory.
 - If the backup-complete file does not exist, the CDF backup process failed. Follow these steps to back up the CDF, and then follow the remaining steps to troubleshoot the upgrade:
 - a. Delete the */<backup directory>/CDF_201703_backup* folder.
 - b. Run the following command:

```
upgrade.sh -u /<Parameter file path>/CDF_upgrade_parameters.txt
```

- If the backup-complete file exists, the CDF backup completed successfully. Follow the remaining steps to troubleshoot the upgrade.
4. Run the following command to check the status of the kubelet service:
systemctl status kubelet
 - If the kubelet service is not active, delete the kubelet.service file in the */usr/lib/systemd/system* directory.
 - If the kubelet service is active, run the following command to stop the kubelet service. Then, delete the kubelet.service file in the */usr/lib/systemd/system* directory.
systemctl stop kubelet

5. Run the following command to check the docker service status: `systemctl status docker`
 - If the docker service is not active, delete the `docker.service` file in the `/usr/lib/systemd/system` directory.
 - If the docker service is active, run the following command to stop the docker service. Then, delete the `docker.service` file in the `/usr/lib/systemd/system` directory.
`systemctl stop docker`
6. Check the `docker-bootstrap` service status with the command: `systemctl status dockerbootstrap`.
 - If the `docker-bootstrap` is not active, delete the `docker-bootstrap.service` file in the `/usr/lib/systemd/system` directory.
 - If the `docker-bootstrap` is active, run the following command, and then delete the `docker-bootstrap.service` file in the `/usr/lib/systemd/system` directory.

systemctl stop docker-bootstrap

7. Run the following commands to unmount the mounted data:
`for data in $(mount | grep "${K8S_HOME}/data/" | cut -d" " -f3 | sort -r);do umount -f -l $data; done`
`for data in $(mount | grep "/usr/lib/kubelet" | cut -d" " -f3 | sort -r);do umount -f -l $data; done`
8. Reboot the machine that you are retrying the upgrade.
9. Run the following command to delete the `$<K8S_HOME>` directory:

```
rm -rf $<K8S_HOME>
```

10. Run the following command to roll back the `$<K8S_HOME>` directory:

```
mv /<backup directory>/CDF_201703_backup $<K8S_HOME>
```

11. Delete the `backup-complete` file in the `$<K8S_HOME>` directory.
12. Run the following commands to recover the `docker.service` and `docker-bootstrap.service` files:

```
mv ${K8S_HOME}/docker.service /usr/lib/systemd/system/
```

```
mv ${K8S_HOME}/docker-bootstrap.service /usr/lib/systemd/system/
```

13. (Optional) If the upgrade failed on the first master node that was stopped, manually restore the data on the NFS server.
14. Run the following command to retry the upgrade:

```
upgrade.sh -u /<Parameter file path>/CDF_upgrade_parameters.txt
```

Worker node does not start

Description

Due to missing disk space, the worker nodes does not start.

Solution

To solve this problem, make sure that the `/` and `/var` directories have at least 5 GB free disk space.

Suite images cannot be downloaded

Description

The suite image download fails.

Solution

Make sure you have set the correct **http_proxy** and **https_proxy** in `/etc/environment` and `/etc/profile.d/proxy.sh`.

idm-postgresql cannot access /var/pgdata

Description

You receive the error message that idm-postgresql cannot access `/var/pgdata/`.

Solution

Make sure the user is the owner of the right group, for example group ID:1999: User ID: 1999.

An example command: **chown -R 1999:1999 /var/pgdata/**

System error: read parent: connection reset by peer

Description

A container cannot be started and fails with the error message "System error: read parent: connection reset by peer".

Solution

Edit the `kube-registry-proxy.yaml` file by adding the following parameters:

```
name: DOCKER_FIX
value: "dockerfix"
```

Renew token failed in http_code=403

Description

The vault token is expired and the error message "ERROR: Renew token failed in http_code=403" is displayed. A container named `kubernetes-vault-renew` is in this pod.

Solution

- Workaround #1:

Initialize a new token with the following commands on the master node:

```
cd $K8S_HOME/bin
kube-restart.sh
```

- Workaround #2:

- Delete the pod manually if the pod is managed by ReplicationController or Deployment. A new pod will be created automatically.
- If the pod is not managed by a Replication Controller, run the following command on the node where the pod is running:
pdocker restart `docker ps -a |grep <podName> |grep kubernetes-vault-init|awk '{print \$1}'`

Docker pull doesn't work: Error while pulling image

Description

Docker pull doesn't work on worker nodes. The following error message is displayed: "Error while pulling image: Get <http://localhost:5000/v1/repositories/itom-hcm-poca-jenkins/images>: read tcp 127.0.0.1:43074->127.0.0.1:5000: read: connection reset by peer".

Solution

1. Change the subnet mask to 255.255.255.0.
2. Configure the parameter FLANNEL_BACKEND_TYPE as follows (see [Configure the install.properties file](#)).

FLANNEL_BACKEND_TYPE = vxlan

Pod is in Pending status: Insufficient CPU, insufficient memory

Description

A pod is in pending status and the error "Insufficient cpu, Insufficient memory" is displayed.

Solution

- Make sure the host machine has enough CPU available and enough memory.
- Reduce the CPU memory requirement of the pod.
- Add a new worker node.

Logs

To troubleshoot your issue, you can review log files.

- [Suite logs](#)
- [CDF logs](#)
- [Docker logs](#)

Suite logs

Installation

/opt/kubernetes/install-`.log`

NFS share

<NFS_omi_directory>/omi/opt/HP/BSM/log/topaz_all.log
<NFS_omi_directory>/omi/opt/HP/BSM/log/jboss7_boot.log
<NFS_omi_directory>/omi/opt/HP/BSM/log/supervisor/nanny_all.log
<NFS_omi_directory>/opsbridge-opsbridge/pe/logs

Login

<NFS_omi_directory>/omi/opt/HP/BSM/log/jboss/login.log

OBR

Configuration

<NFS_conf_directory>/OBR/reporting/... (OBR server)
<NFS_conf_directory>/OBR/reporting-collector/... (OBR reporting collector)
<NFS_conf_directory>/OBR/reporting-content/... (OBR content pack artifacts)

Logs

<NFS_log_directory>/OBR/reporting/... (OBR server)
<NFS_log_directory>/OBR/reporting-collector/... (OBR reporting collector)

Data

<NFS_data_directory>/OBR/reporting/... (OBR server)
<NFS_data_directory>/OBR/reporting-collector/... (OBR reporting collector)
<NFS_data_directory>/OBR/MgmtDB/... (OBR PostgreSQL instance)

CDF logs

Access pod logs

1. In the ITOM Container Deployment Foundation (CDF) navigation pane, click **RESOURCES > Workloads > Pods**.
2. Click the relevant pod.
3. Click **View logs** in the Pod area.

Note All suite logs are currently stored within a persistent volume so that they are not lost even if the pods go down.

Docker logs

Log and trace Model

Recommendations:

- Pay attention to the log level and do not unnecessarily enable tracing or debug parameters.
- Pay attention to log rotation and switching.

Log rotation

CDF supports log rotation. By default, the maximum log file size is 10 MB, and the maximum number of log files is five. To change the maximum log file size and maximum log file number, follow these steps:

1. Run the following commands to open the docker file:

```
cd /opt/kubernetes/cfg
```

```
vim docker
```

2. Change the values of max-size and max-file variables in the *DOCKER_LOG_OPTS* parameter. For example:

```
DOCKER_LOG_OPTS="--log-driver=json-file --log-opt labels=io.kubernetes.container.name,io.kubernetes.pod.uid --log-opt max-size=12m --log-opt max-file=6"
```

3. Run the following command to restart Docker and enable the changes:

```
systemctl restart docker
```

Note

We recommend that you use the default maximum log size number and maximum log file number. Do not set a large number for the max-size and max-file variables. Overly large maximum size and maximum file numbers may affect the free disk space.

Manual verification commands

The following commands can be used to troubleshoot the ITOM Container Deployment Foundation and the Operations Bridge Suite container deployment, for example to list namespaces and services.

`/opt/kubernetes/bin/kube-status.sh`
Displays the status of the K8S cluster.

`/opt/kubernetes/bin/kube-stop.sh`
Stops the K8S cluster.

`/opt/kubernetes/bin/kube-restart.sh`
Restarts the K8S cluster.

`/opt/kubernetes/bin/kube-start.sh`
Starts the K8S cluster.

`kubectl`
The command to interact with Kubernetes (K8S).

Tip

To shorten the `kubectl` command, run the following command:

```
ln -s /usr/bin/kubectl /usr/bin/kl
```

This enables you to type `kl` instead of `kubectl`.

`kubectl cluster-info`
Summarizes information about some of the services that are running on the cluster, including Kubernetes master, KubeDNS for service discovery, and the endpoints of the KubeRegistry (if you are running a registry).

`kubectl get nodes`
Lists all nodes in the cluster.

`kubectl describe nodes <node_IP>`
Provides more specific information about the node, such as labels, events, capacity, CPU, memory, the maximum number of pods that the node can support, system information on the node, external IP address, the pods that are running, the list of namespaces, and resources.

`kubectl get pods`
Lists all pods in the default namespace (used to separate the Container Deployment Foundation services from the deployed suites).

`kubectl get pods -n=<namespace>`
Lists all the pods that are running on the specified namespace. For example, run `kubectl get pods -n=opsbridge1` to get a list of the pods running in the namespace `opsbridge1`.

`kubectl get pods --all-namespaces`
Lists all the pods that are currently running in the cluster.

`kubectl describe pod <pod_name> -n=<namespace>`
Displays details about a specified pod in a specified namespace, such as the image it is running, the port it is exposing, and the command (/hyperkube) that is running inside the container itself with their options, volumes, and more.

`kubectl exec <pod_name> -c <container> -n <namespace>`
Executes a command in the specified container. If no container is specified, the first container in the pod is selected.

Example

```
kubectl exec omi-1949254658-p3ipj -c omi -n opsbridge1 bash -ti
```

Example

Executes a bash shell in the OMi container with the pod name omi-1949254658-p3ipj and the namespace opsbridge1. By executing a bash shell in the OMi container, you can call CLIs from inside the container. For more information, see [Access Command Line Interfaces](#).

`kubectl get services --all-namespaces`
Displays all the services running in the cluster.

`kubectl logs -n=<namespace>`
Displays the log output for the specified pod.

Support toolset

The support toolset helps to collect information about Docker, Kubernetes, suites, commands, directories, and files as listed below:

- Docker: containers, inspect, docker service systemd logs
- Kubernetes: nodes, pods, namespaces, images, containers, cluster-info, describe, logs
- Suite: suite-db dump, suite data, modules, product deployments, features
- Commands defined by users
- Directories and files defined by users

You can view the summary information on a console. For the detailed output information, you can view them in an encrypted tar file.

Use the toolset

Run the following commands to use the toolset:

1. `cd <K8S_HOME>/tools/support-tool`

2. # ./support-dump [-c <dump_filename_with_path>] [-u <username> [-p <password>]] [-P <package_password>]
3. Unpack the dumpfile:

```
# dd if=xxx.des3 |openssl des3 -d -k <package_password>|tar zxf -
```

Example

- Create a dump file with the default file name in the default directory.
./support-dump
- Create an example dump file dump.des3 in the directory /var/test.
./support-dump -c /var/test/dump.des3
- Create a dump file with the user name admin and the password 123456. Additionally, specify the package password abcdef.
./support-dump -u admin -p 123456 -P abcdef

Configuration file

The support toolset provides a configuration file with some predefined [commands], [files], and [dirs] to specify your deployment's information. You can also define your own commands, files, and directories in the configuration file. Alternatively, create other configuration files in the same directory. The default configuration file is `conf/supportdump.config`.

Example

```
<K8S_HOME>/cfg *_User.json
```

The support toolset collects all files and directories located in `<K8S_HOME>/cfg` except the `*_User.json` file(s).

Dump file

The default support dump file is called `dmp/support_data_YYYYMMDD-hhmmss.des3`. The dump file contains the `support_data_YYYYMMDD-hhmmss.log` of the running support toolset and the `ITOM_Core_Platform` directory for the dump files. The table below shows the dump files in the `ITOM_Core_Platform` directory.

<local_ip>- <NodeType>	The directory of docker information and user defined information on the current node. docker docker information dump files bootstrap: bootstrap_containers.out: docker-bootstrap containers bootstrap_docker_inspect.out: docker-bootstrap inspect journalctl_docker-bootstrap.out: docker-bootstrap service log workload workload: containers.out: docker containers docker_inspect.out: docker inspect information journalctl_docker.out: docker service log os: user defined commands, directories and files commands: directory of output files of commands defined in the [commands] section in .config files. The file name format: <command>.out.	Directory
---	--	-----------

	<p>other directories: directories and files defined in the [files], and [dirs] sections in .config files. The structure of directories will be reserved.</p>	
<p>global</p>	<p>deployment:</p> <p>suite_data: JSON files of all suites data suite-db_pgdump.tar: PostgreSQL db dump file of suite-db pod suite_features.out: suite deployment and features</p> <p>kubernetes:</p> <p>cluster_info::</p> <p>cluster_info.out: output of running command 'kubectl cluster_info dump' other dirs & files: generated by command 'kubectl cluster_info dump'</p> <p>kube_describe.out: describe of all pods</p> <p>kube_summary.out: namespaces, pv, pvc, nodes, deployments, services, pods, ingress</p> <p>platform:</p> <p>containers_by_pod.out: all containers, images on all pods</p>	<p>Directory</p>