

# Dépannage de l'erreur CloudCenter « Impossible de communiquer avec l'orchestrateur » avec l'erreur 408

## Contenu

[Introduction](#)

[Conditions préalables](#)

[Conditions requises](#)

[Components Used](#)

[Problème](#)

[Solution](#)

[AMQP](#)

[CCO](#)

[CCM](#)

[Vérification](#)

[CCM](#)

[AMQP](#)

[CCO](#)

## Introduction

Ce document décrit comment dépanner l'erreur CloudCenter « Impossible de communiquer avec l'orchestrateur » avec l'erreur 408

## Conditions préalables

### Conditions requises

Cisco vous recommande de prendre connaissance des rubriques suivantes :

- Interface Linux
- Environnement de machine virtuelle
- VIM

### Components Used

Cisco recommande les connaissances suivantes :

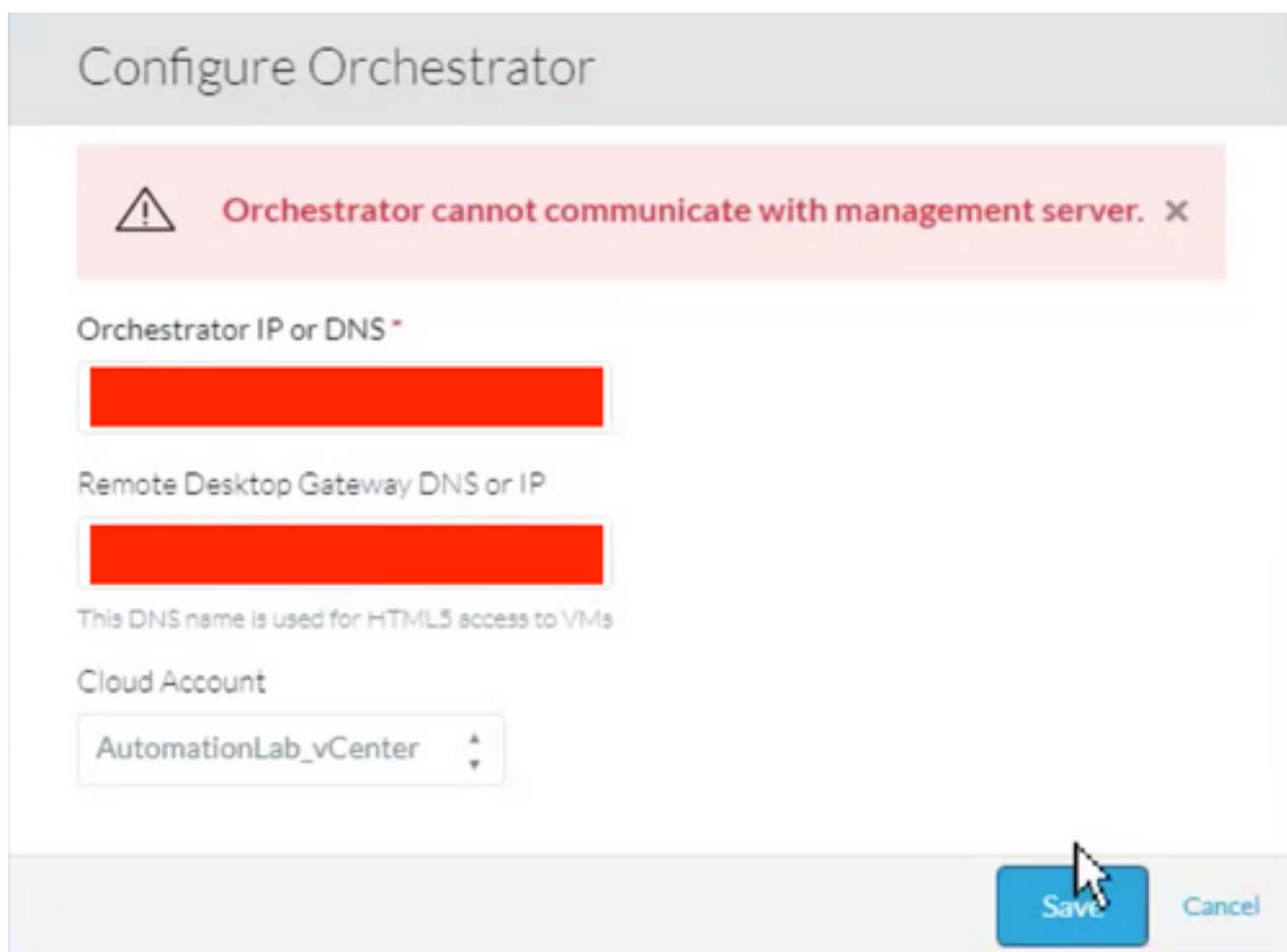
- Appareils CloudCenter
- Architecture CloudCenter
- Linux O.S.
- CCM (Gestion CloudCenter)

- CCO (CloudCenter Orchestrator)
- AMQP (Advanced Message Queuing Protocol)

Les informations de ce document ont été créées à partir des périphériques d'un environnement de travaux pratiques privé spécifique. All of the devices used in this document started with a cleared (default) configuration. Si votre réseau est en ligne, assurez-vous de bien comprendre l'incidence possible des commandes.

## Problème

Les pannes d'alimentation, les redémarrages inattendus ou les pannes de réseau pendant de longues périodes peuvent entraîner la désynchronisation de l'apppliance CloudCenter. Il est nécessaire d'effectuer les vérifications suivantes afin de savoir que les appliances sont correctement connectées. Lors de la configuration de l'orchestrateur sur l'interface utilisateur graphique de CloudCenter Manager (interface utilisateur graphique CCM), les utilisateurs peuvent obtenir l'erreur comme indiqué dans l'image.



Lors de la vérification des journaux CCO, l'erreur suivante peut s'afficher :

```
Caused by: java.net.ConnectException: Connection refused (Connection refused)
at java.net.PlainSocketImpl.socketConnect(Native Method)
at java.net.AbstractPlainSocketImpl.doConnect(AbstractPlainSocketImpl.java:350)
at java.net.AbstractPlainSocketImpl.connectToAddress(AbstractPlainSocketImpl.java:206)
at java.net.AbstractPlainSocketImpl.connect(AbstractPlainSocketImpl.java:188)
at java.net.SocksSocketImpl.connect(SocksSocketImpl.java:392)
at java.net.Socket.connect(Socket.java:589)
at
```

```
org.apache.http.conn.ssl.SSLConnectionSocketFactory.connectSocket(SSLConnectionSocketFactory.jav
a:337)
at
org.apache.http.impl.conn.DefaultHttpClientConnectionOperator.connect(DefaultHttpClientConnectio
nOperator.java:134)
... 87 more
java.lang.RuntimeException: Failed to connect to CCM, please check network connection between
CCM and CCO. JobId: 21912
at
com.osmosix.commons.mgmtserver.impl.MgmtServerServiceImpl.getUserCloudAccountByJobId(MgmtServers
erviceImpl.java:236)
at
com.osmosix.gateway.persistence.impl.hazelcast.AbstractDistributedJobDaoImpl.find(AbstractDistri
butedJobDaoImpl.java:109)
at
com.osmosix.gateway.persistence.impl.hazelcast.AbstractDistributedJobDaoImpl.find(AbstractDistri
butedJobDaoImpl.java:17)
at com.osmosix.gateway.lifecycle.impl.AbstractLifecycle.getJob(AbstractLifecycle.java:207)
at com.osmosix.gateway.lifecycle.helpers.LifecycleReaper.reapApp(LifecycleReaper.java:62)
at com.osmosix.gateway.lifecycle.helpers.LifecycleReaper.reapDeadApps(LifecycleReaper.java:45)
at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
at java.lang.reflect.Method.invoke(Method.java:498)
at
org.springframework.scheduling.support.ScheduledMethodRunnable.run(ScheduledMethodRunnable.java:
65)
at
org.springframework.scheduling.support.DelegatingErrorHandlingRunnable.run(DelegatingErrorHandli
ngRunnable.java:54)
at
org.springframework.scheduling.concurrent.ReschedulingRunnable.run(ReschedulingRunnable.java:81)
at java.util.concurrent.Executors$RunnableAdapter.call(Executors.java:511)
at java.util.concurrent.FutureTask.run(FutureTask.java:266)
at
java.util.concurrent.ScheduledThreadPoolExecutor$ScheduledFutureTask.access$201(ScheduledThreadP
oolExecutor.java:180)
at
java.util.concurrent.ScheduledThreadPoolExecutor$ScheduledFutureTask.run(ScheduledThreadPoolExec
utor.java:293)
at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1142)
at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:617)
at java.lang.Thread.run(Thread.java:748)
```

## Solution

Redémarrez les composants CloudCenter un par un pour actualiser la connexion entre les  
composants.

## AMQP

Étape 1. Se connecter en tant que root

Étape 2. Redémarrer le service AMQP

Sur toutes les versions jusqu'à 4.8.1.2

```
# /etc/init.d/tomcatgua restart
```

Sur les versions commençant à 4.8.2

```
# systemctl restart rabbit
```

## CCO

Étape 1. Se connecter en tant que root

Étape 2. Redémarrer le service CCO

Sur toutes les versions jusqu'à 4.8.1.2

```
# /etc/init.d/tomcat restart
```

Sur les versions commençant à 4.8.2

```
# systemctl restart cco
```

## CCM

Étape 1. Se connecter en tant que root

Étape 2. Redémarrer le service CCM

Sur toutes les versions jusqu'à 4.8.1.2

```
# /etc/init.d/tomcat restart
```

Sur les versions commençant à 4.8.2

```
# systemctl restart ccm
```

## Vérification

Il est important que tous les appareils soient correctement connectés. Pour cela, il est nécessaire d'intégrer chacun des composants CloudCenter.

## CCM

Étape 1. Se connecter en tant que root

Étape 2. Vérifiez que tomcat (antérieur à 4.8.2) ou le service CCM (post 4.8.2) sont en cours d'exécution

Sur toutes les versions jusqu'à 4.8.1.2

```
[root@localhost ~]# ps -ef | grep -i tomcat
```

Sur les versions commençant à 4.8.2

```
[root@localhost ~]# systemctl status ccm
```

Étape 3. Si telnet est installé, une tentative peut être effectuée à partir de CCO vers CCM, ce qui permettra de comprendre que la communication est possible

```
[root@cliqr-centos7-base-image ~]# telnet 10.31.127.41 8443
Trying 10.31.127.41...
Connected to 10.31.127.41.
Escape character is '^]'.
```

Si une erreur se produit, aucune communication n'est possible. Cela doit être corrigé.

Étape 4. Si vous souhaitez configurer l'orchestrateur sur l'interface utilisateur graphique CCM, un nom d'hôte sera utilisé, assurez-vous que le nom d'hôte est présent dans le fichier /etc/hosts

```
[root@cliqr-centos7-base-image ~]# cat /etc/hosts
127.0.0.1 localhost localhost.localdomain localhost4 localhost4.localdomain4
::1 localhost localhost.localdomain localhost6 localhost6.localdomain6
127.0.0.1 devCC
10.31.127.42 cco
```

## AMQP

Étape 1. Se connecter en tant que root

Étape 2. Vérifiez qu'une connexion est établie entre le protocole AMQP et chacun des CCO existants.

```
[root@localhost ~]# rabbitmqctl list_connections -p /cliqr
Listing connections ...
cliqr10.31.127.4233062running
cliqr_worker10.31.127.4233130running
cliqr_worker10.31.127.5938596running
cliqr_worker10.31.127.6749781running
cliqr_worker10.31.127.7949778running
cliqr_worker10.31.127.8549786running
```

Dans la commande précédente, les connexions vers le CCO peuvent être vues dans la ligne avec l'utilisateur cliqr (dans ce cas, il n'y a qu'un CCO)

En cas de haute disponibilité (HA) et d'AMQP sous un équilibrEUR de charge, une connexion par CCO est connectée à l'IP de l'équilibrEUR de charge d'AMQP (dans l'exemple suivant, il y a 2 CCO)

```
[root@amqp-azrel ~]# rabbitmqctl list_connections -p /cliqr
Listing connections ...
cliqr15.1.0.1035788running
cliqr15.1.0.1036212running
cliqr_worker15.1.0.1037714running
cliqr_worker15.1.0.1038362running
cliqr_worker15.1.0.1041102running
```

Si ce n'est pas le cas, redémarrez le processus tomcatgua (précédent au 4.8.2) ou le service lapin (après le 4.8.2).

## CCO

## Étape 1. Se connecter en tant que root

Sur toutes les versions jusqu'à 4.8.1.2

```
[root@localhost ~]# ps -ef | grep -i tomcat
```

Sur les versions commençant à 4.8.2

```
[root@localhost ~]# systemctl status cco
```

Étape 3. Vérifiez que les connexions vers le CCM sont établies. Il doit également apparaître dans l'état CLOSE\_WAIT (dans ce cas, notre CCM est sur 10.31.127.41).

```
[root@cliqr-centos7-base-image ~]# netstat -anp | grep 10.31.127.41
tcp      86      0 10.31.127.42:38542        10.31.127.41:8443      CLOSE_WAIT    1330/java
tcp      86      0 10.31.127.42:38562        10.31.127.41:8443      CLOSE_WAIT    1330/java
tcp      86      0 10.31.127.42:38546        10.31.127.41:8443      CLOSE_WAIT    1330/java
tcp      86      0 10.31.127.42:38566        10.31.127.41:8443      CLOSE_WAIT    1330/java
tcp      86      0 10.31.127.42:38556        10.31.127.41:8443      CLOSE_WAIT    1330/java
tcp      86      0 10.31.127.42:38554        10.31.127.41:8443      CLOSE_WAIT    1330/java
tcp      86      0 10.31.127.42:38550        10.31.127.41:8443      CLOSE_WAIT    1330/java
tcp      86      0 10.31.127.42:38564        10.31.127.41:8443      CLOSE_WAIT    1330/java
tcp      86      0 10.31.127.42:38560        10.31.127.41:8443      CLOSE_WAIT    1330/java
tcp      86      0 10.31.127.42:38568        10.31.127.41:8443      CLOSE_WAIT    1330/java
tcp      86      0 10.31.127.42:38552        10.31.127.41:8443      CLOSE_WAIT    1330/java
tcp      86      0 10.31.127.42:38558        10.31.127.41:8443      CLOSE_WAIT    1330/java
tcp      86      0 10.31.127.42:38570        10.31.127.41:8443      CLOSE_WAIT    1330/java
tcp      86      0 10.31.127.42:38548        10.31.127.41:8443      CLOSE_WAIT    1330/java
tcp      86      0 10.31.127.42:38572        10.31.127.41:8443      CLOSE_WAIT    1330/java
tcp      86      0 10.31.127.42:38544        10.31.127.41:8443      CLOSE_WAIT    1330/java
```