

# Comment configurer un cluster de base de données de CMS version 3.0

## Contenu

[Introduction](#)

[Conditions préalables](#)

[Conditions requises](#)

[Components Used](#)

[Configuration](#)

[Diagramme du réseau](#)

[Configurations](#)

[Vérification](#)

## Introduction

Ce document décrit étape par étape la configuration d'un cluster de base de données Cisco Meeting Server (CMS).

Contribué par Amadeus Ubaldo et Octavio Miralrio, ingénieurs du TAC Cisco.

## Conditions préalables

### Conditions requises

Cisco vous recommande de prendre connaissance des rubriques suivantes :

- Configuration générale CMS
- Client Secure Shell (SSH)
- Client du protocole de transfert de fichier sécurisé (SFTP)

### Components Used

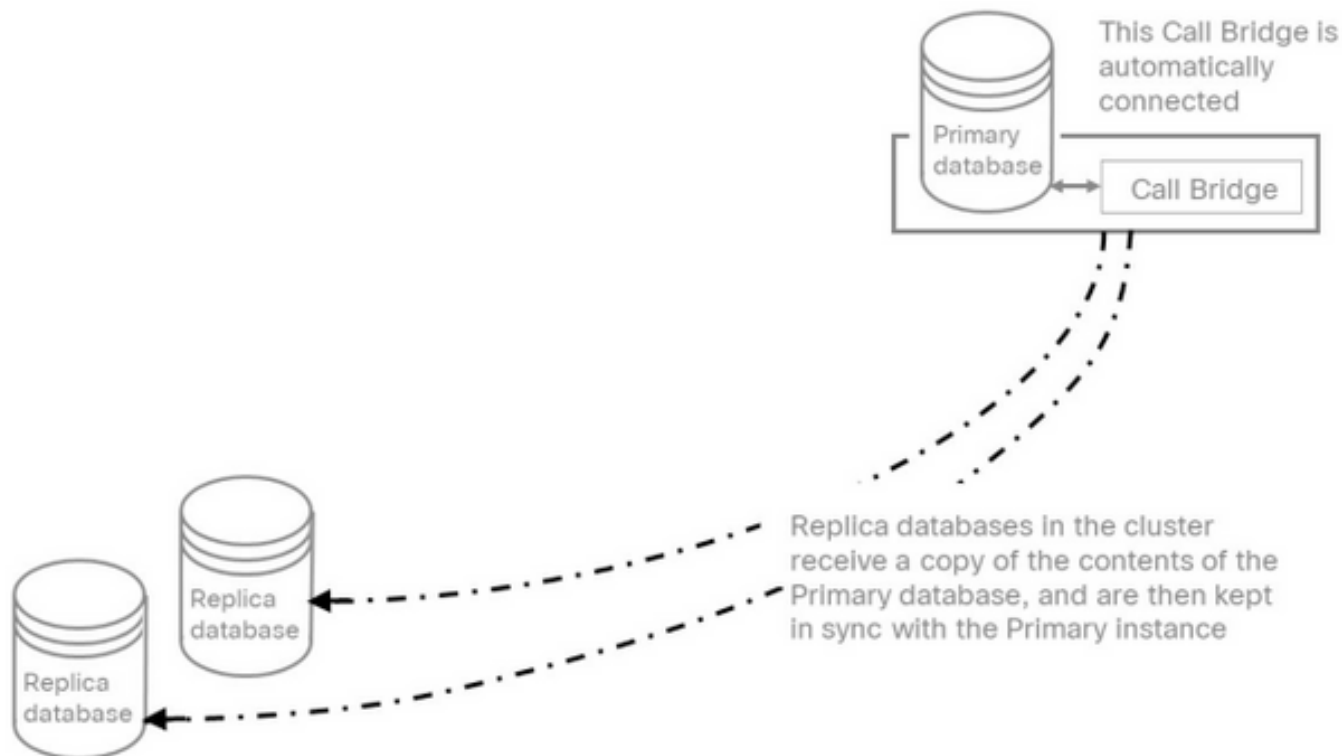
Les informations contenues dans ce document sont basées sur les versions de matériel et de logiciel suivantes :

- CMS version 3.0
- Client SSH putty
- Client Windows Secure Copy (winSCP)

Les informations contenues dans ce document ont été créées à partir des périphériques d'un environnement de laboratoire 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.

# Configuration

## Diagramme du réseau



## Configurations

Étape 1. Sélectionnez un CMS comme noeud principal du cluster de base de données.

Remarque: Le cluster de base de données CMS nécessite un nombre impair de serveurs et un maximum de 5. Consultez la documentation pour plus de détails sur le [Guide de configuration de CMS](#).

Étape 2. Sur le CMS que vous avez sélectionné pour le rôle principal, utilisez le client SSH pour ouvrir son interface MMP (Mainboard Management Processor).

Étape 3. Générer une demande de signature de certificat (CSR) pour le rôle client. Utilisez le format suivant :

Remarque: Le nom commun (CN) de la RSE du client doit être **postgres**.

```
pki csr
```

```
172.16.85.108 - PuTTY
cms01>
cms01> pki csr DBClient CN:postgres
.....
.....
Created key file DBClient.key and CSR DBClient.csr
CSR file DBClient.csr ready for download via SFTP
```

Étape 4. Générer un CSR pour le rôle de serveur. Utilisez le format suivant :

Remarque: Ajoutez les noms de domaine complets (FQDN) des autres serveurs faisant partie du même cluster de base de données que le SAN (Subject Alternative Name).

pki csr

```
cms01> c-cmspod3.lab subjectAltName:cms02.mxc-cmspod3.lab,cms03.mxc-cmspod3.lab
.....
.....
Created key file DBServer.key and CSR DBServer.csr
CSR file DBServer.csr ready for download via SFTP
cms01> █
```

Étape 5. Vérifiez que quatre fichiers ont été créés avec le nom que vous avez utilisé à partir des commandes précédentes.

**DBServer.csr**

**DBServer.key**

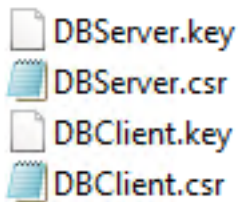
**DBClient.csr**

**DBClient.key**

Étape 6. Suivez les étapes 3 et 4 avec le reste de vos serveurs CMS faisant partie du cluster DB.

Remarque: Les mêmes CSR client et serveur et la même paire de clés générée à partir du premier serveur peuvent être utilisés pour les autres homologues du cluster. Afin de copier les fichiers **.key** et les certificats déjà signés aux autres serveurs CMS avec un logiciel SFTP.

Étape 7. Utilisez un logiciel SFTP de votre préférence pour accéder à chaque serveur afin de collecter les fichiers CSR.



Étape 8. Fournissez les fichiers CSR de chaque serveur à l'autorité de certification que vous prévoyez d'utiliser pour signer le certificat.

Étape 9. Une fois le certificat déjà signé, utilisez un client SFTP pour charger les certificats signés sur chaque serveur CMS, respectivement.

Étape 10. Accédez à votre serveur CMS via l'interface de ligne de commande et affectez le certificat et les fichiers clés au service de cluster de base de données. Utilisez la syntaxe suivante :

```
database cluster certs
```

Étape 11. Répétez l'étape 10 pour le reste de vos serveurs CMS afin d'attribuer les certificats respectivement.

Étape 12. Accédez via l'interface de ligne de commande au CMS désigné comme noeud maître pour le cluster de base de données, sélectionnez l'interface affectée à utiliser pour le service de base de données, dans cet exemple, il s'agit de l'**interface a** et exécutez la commande :

```
database cluster localnode a
```

Étape 13. Répétez l'étape 12 sur les autres serveurs CMS prévus pour faire partie du cluster de base de données CMS.

Étape 14. Accédez à CMS via l'interface de ligne de commande (CLI) avec le rôle de base de données principale pour initialiser le service de base de données. Exécutez la commande suivante :

```
database cluster initialize
```

Remarque: La confirmation est sensible à la casse, assurez-vous de taper la lettre **Y** avec la majuscule. Cette commande doit être exécutée uniquement dans la base de données principale et non dans les noeuds suivants.

Étape 15. L'avertissement suivant s'affiche. Sélectionnez **Y**.

```
database cluster initialize WARNING!!! Are you sure you wish to initialize this node as a new database cluster?(Y/n) The contents of this node's database will become the primary version of the database in the new cluster. Initialization started...
```

Étape 16. Vérifiez l'état de l'initialisation, exécutez la commande **database cluster status**.

```
Status: initializing Nodes: 172.16.85.108 (me): Connected primary Interface: a
```

Remarque: L'état indique l'**initialisation** si le processus d'initialisation du service de base de données n'est pas encore terminé.

Étape 17. Exécutez la commande **database cluster status** jusqu'à ce que l'état soit **Activé**, ce qui signifie que le processus d'initialisation a réussi comme indiqué dans la sortie suivante :

```
Status: Enabled Nodes: 172.16.85.108 (me) : Connected primary Interface: a
```

Étape 18. Accès via l'interface de ligne de commande au prochain homologue CMS faisant partie du cluster de base de données.

Étape 19. Exécutez la commande **database cluster join <primary\_database\_ip\_address>**.

Étape 20. Sélectionnez **Y**.

```
cms> database cluster join 172.16.85.108 WARNING? ! ! Are you sure you wish to attach this node
to an existing database cluster? (Y/n) The contents of this node's database will be destroyed!
Attachment started...
```

Étape 21. Répétez la procédure décrite aux étapes 18, 19 et 20 pour le reste des serveurs.

## Vérification

Vous pouvez vérifier l'état du cluster de base de données directement sur l'interface de ligne de commande. Exécutez la commande **database cluster status**. la sortie suivante indique que le processus de cluster n'est pas encore terminé :

```
cms> database cluster status Status: Attaching Nodes: 172.16.85.108: Connected Primary
172.16.85.117 (me): Connected Replica 172.16.85.118: Connected Replica Interface: a
```

Lorsque le processus de jointure de cluster de base de données est terminé, vous devez voir l'état **Activé** :

```
cms> database cluster status Status: Enabled Nodes: 172.16.85.108: Connected Primary
172.16.85.117 (me): Connected Replica 172.16.85.118: Connected Replica Node in use:
172.16.85.108 Interface: a Certificates Server Key: DBServer.key Server Certificate:
DBServer.cer Client Key: DBClient.key Client Certificate: DBClient.cer CA Certificate:
CA_root.cer Last command: 'database cluster join 172.16.85.108' (Success)
```