

# Configurez Cisco rencontrant le groupement de base de données de passerelle d'appel de serveur

## Contenu

[Introduction](#)

[Conditions préalables](#)

[Conditions requises](#)

[Composants utilisés](#)

[Configurez](#)

[Création de certificat de la partie 1.](#)

[Configuration de pont d'appel de la partie 2.](#)

[Diagramme du réseau](#)

[Vérifiez](#)

[Dépannez](#)

[Informations connexes](#)

## Introduction

Ce document décrit les étapes pour configurer la base de données (DB) groupant sur Cisco des passerelles rencontrant du serveur (CMS) ou de l'Acano appel (CB).

## Conditions préalables

### Conditions requises

- Cisco recommande que vous ayez au moins 3 Noeuds de CB à pouvoir créer une batterie viable de DB.

Remarque: Il a recommandé d'avoir un nombre impair de Noeuds de batterie de DB car il est important pour la sélection principale et le mécanisme actif de Basculement. Une autre raison pour ceci est que le noeud principal de DB serait le noeud qui a des connexions au plus du DB dans la batterie. Jusqu'à 8 Noeuds de batterie de DB ont été testés au moment de la publication de ce document, bien qu'il soit techniquement possible de grouper beaucoup plus.

- Port 5432 ouvert sur le Pare-feu.

Remarque: Le maître de batterie de DB écoute sur le port 5432 des connexions des noeuds client, ainsi s'il y a un Pare-feu (FW) entre les Noeuds, s'assure que ce port est ouvert.

### [Composants utilisés](#)

Ce document n'est pas limité à des versions de matériel et de logiciel spécifiques.

Les informations contenues dans ce document ont été créées à partir des périphériques d'un environnement de laboratoire spécifique. Tous les périphériques utilisés dans ce document ont démarré avec une configuration effacée (par défaut). Si votre réseau est opérationnel, assurez-vous que vous comprenez l'effet potentiel de toute commande.

## Configurez

Il y a deux types de Certificats pour le groupement de DB :

1. Client : Le certificat client, comme nom suggérez, est utilisé par les clients de DB pour connecter au DB le serveur (maître). Ce certificat doit contenir la chaîne, des postgres, dans son domaine commun du nom (NC).
2. Serveur : Le certificat de serveur, comme nom suggérez, est utilisé par le serveur de DB pour connecter aux postgres le DB.

### Création de certificat de la partie 1.

1. Connectez à un Protocole Secure Shell (SSH) avec les qualifications d'admin au serveur MMP.
2. Générez la demande de signature de certificat (CSR) :

a. Pour le certificat client de databasecluster :

```
NC de basename> csr <key/cert de PKI : postgres
```

Exemple : **NC databasecluster\_client csr de PKI : postgres**

b. Pour le certificat de serveur de databasecluster :

```
NC de basename> csr <key/cert de PKI : <domainname>
```

Exemple : **NC de databasecluster\_server csr de PKI : vngtpres.aca**

Remarque: La NC pour la partie A Doit être des postgres et la partie B Peut être le nom de domaine de la passerelle d'appel.

### Configuration de pont d'appel de la partie 2.

Sur le CB qui exécute le DB principal, suivez ces étapes :

1. Pour sélectionner l'interface pour utiliser, sélectionnez la commande ci-dessous :

```
localnode a de batterie de base de données
```

Ceci permet à l'interface « a » d'être utilisée pour le groupement de DB.

2. Définissez les Certificats Ca de client, de serveur et de racine aussi bien que les clés privées à utiliser par la batterie de DB avec ces commandes :

```
<ca_crt> de <client_crt> de <client_key> de CERT de batterie de base de données
```

```
<ca_crt> de <client_crt> de <client_key> de <server_crt> de <server_key> de CERT de batterie de base de données
```

Remarque: Les mêmes Certificats de client et serveur peuvent être utilisés sur d'autres Noeuds de CB à grouper en copiant simplement les clés privées et les Certificats à travers sur les autres Noeuds. C'est possible parce que les Certificats ne contiennent aucun SAN les attachant à une passerelle spécifique d'appel. Cependant, il est recommandent d'avoir différents Certificats pour chaque noeud de DB.

3. Initialisez ce DB sur les CB locaux comme maître pour cette batterie de DB :

**la batterie de base de données initialisent**

4. Sur le CallBridges qui ferait partie du DB groupé et deviendrait les esclaves de DB exécutent cette commande après que vous vous terminiez les étapes 1 et 2 pour la partie :

**la batterie de base de données joignent l'adresse IP de CB de <Master >**

Exemple : **la batterie de base de données joignent <10.48.36.61>**

Ceci initie la synchronisation de DB et copie le DB du pair principal.

Remarque: Le DB local qui a existé avant que la **batterie de base de données joignent la** commande a été initié, continue à exister jusqu'à ce que le noeud soit retiré du DB groupé. Pour tout long que le noeud est dans la batterie de DB, son DB local n'est pas utilisé.

## [Diagramme du réseau](#)

## Vérifiez

Référez-vous à cette section pour vous assurer du bon fonctionnement de votre configuration.

Pour vérifier l'état groupé de DB, exécutez cette commande sur les Noeuds l'uns des dans la batterie de DB :

**état de batterie de base de données**

La sortie est semblable au ci-dessous :

```
Status                : Enabled
Nodes:
  10.48.36.61          : Connected Master
  10.48.36.118         : Connected Slave ( In Sync )
  10.48.36.182 (me)   : Connected Slave ( In Sync )
Node in use           : 10.48.36.61
```

```
Interface          : a

Certificates
Server Key         : dbclusterserver.key
Server Certificate  : dbclusterserver.cer
Client Key         : dbclusterclient.key
Client Certificate  : dbclusterclient.cer
CA Certificate      : vngtpRootca.cer
Last command       : 'database cluster join 10.48.36.61' (Success)
```

## Dépannez

Cette section fournit des informations que vous pouvez utiliser pour dépanner votre configuration.

Utilisez cette commande, sur le CLI, de visualiser les logs de courant liés au groupement de DB :

### le Syslog suivent

Le collecteur de log CMS, voit le lien, fournissent une interface utilisateur facile et conviviale (UI) de collecter des logs du serveur CMS :

<http://www.cmslogcollector.com/>

Les sorties de log pour le DB contiennent habituellement la chaîne de postgres, des exemples comme suit :

```
Mar 30 12:39:04 local0.warning DBMaster postgres[20882]: [2-7] #011SQL statement "INSERT INTO
domains(domain_id, domain_name, tenant_id, target, priority, passcode_separator) VALUES
(inp_domain_id, inp_domain_name, inp_tenant_id, existing_target, inp_priority,
inp_passcode_separator)"
Mar 30 12:39:04 local0.warning DBMaster postgres[20882]: [2-8] #011PL/pgSQL function
create_or_update_matching_domain(boolean,uuid,text,boolean,uuid,integer,integer,integer,text)
line 61 at SQL statement
Mar 30 12:39:04 local0.warning DBMaster postgres[20882]: [2-9] #011SQL statement "SELECT * FROM
create_or_update_matching_domain(TRUE, inp_domain_id, inp_domain_name, TRUE, inp_tenant_id,
inp_target_true, 0, inp_priority, inp_passcode_separator)"
Mar 30 12:39:04 local0.warning DBMaster postgres[20882]: [2-10] #011PL/pgSQL function
create_matching_domain(uuid,text,uuid,integer,integer,text) line 3 at SQL statement
```

Voici quelques questions et solutions typiques de DB :

- Erreur de schéma de DB sur un pair de non-maître :

```
Mar 30 12:39:04 local0.warning DBMaster postgres[20882]: [2-7] #011SQL statement "INSERT INTO
domains(domain_id, domain_name, tenant_id, target, priority, passcode_separator) VALUES
(inp_domain_id, inp_domain_name, inp_tenant_id, existing_target, inp_priority,
inp_passcode_separator)"
Mar 30 12:39:04 local0.warning DBMaster postgres[20882]: [2-8] #011PL/pgSQL function
create_or_update_matching_domain(boolean,uuid,text,boolean,uuid,integer,integer,integer,text)
line 61 at SQL statement
Mar 30 12:39:04 local0.warning DBMaster postgres[20882]: [2-9] #011SQL statement "SELECT * FROM
create_or_update_matching_domain(TRUE, inp_domain_id, inp_domain_name, TRUE, inp_tenant_id,
inp_target_true, 0, inp_priority, inp_passcode_separator)"
Mar 30 12:39:04 local0.warning DBMaster postgres[20882]: [2-10] #011PL/pgSQL function
create_matching_domain(uuid,text,uuid,integer,integer,text) line 3 at SQL statement
```

Solution :

1. D'abord, exécutez cette commande d'effacer l'erreur :

## erreur d'espace libre de batterie de base de données

2. Suivi de cette commande d'améliorer le schéma de DB :

### upgrade\_schema de batterie de base de données

3. Vérifiez alors à l'état du DB groupant avec :

### état de batterie de base de données

Les logs devraient afficher la sortie semblable à ceci :

```
Mar 30 11:22:45 user.notice acanosrv05 schema_builder: Upgrading schema with connect line
'connect_timeout=4 user=postgres host=127.0.0.1 port=9899 sslmode=verify-ca
sslcert=/srv/pgsql/client.crt sslkey=/srv/pgsql/client.key sslrootcert=/srv/pgsql/ca.crt 'Mar 30
11:22:45 user.notice acanosrv05 schema_builder: Upgrading schema with connect line
'connect_timeout=4 user=postgres host=127.0.0.1 port=9899 sslmode=verify-ca
sslcert=/srv/pgsql/client.crt sslkey=/srv/pgsql/client.key sslrootcert=/srv/pgsql/ca.crt '
```

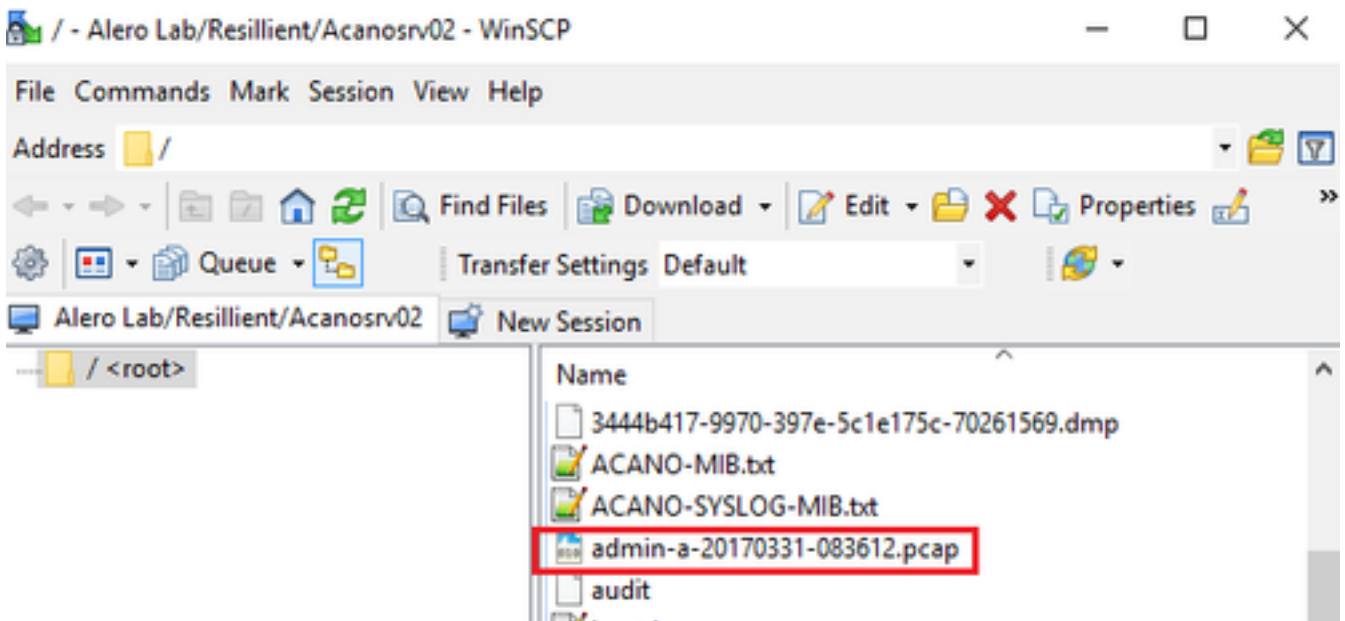
- Noeuds de pair incapables de se connecter au noeud principal de DB :

```
Mar 31 10:16:59 user.info acanosrv02 sfpool: Health check 10.48.54.119: error (up = 1): could
not connect to server: Connection refused|#011Is the server running on host "10.48.54.119" and
accepting|#011TCP/IP connections on port 5432?|
```

Solution :

Employez ces étapes pour collecter des suivis pour dépanner les questions de connexion :

1. Exécutez le **<interface> de pcap de** commande sur le noeud (slave) de non-maître et après quelques minutes, arrêtent la capture avec le **CTRL-C**.
2. Connectez à un client de Protocole SFTP (Secure File Transfer Protocol) au serveur et téléchargez le fichier **.pcap** à partir du répertoire racine :



3. Ouvrez le fichier de capture sur Wireshark et le filtre sur le port 5432 avec **tcp.port==5432** pour vérifier le trafic entre le pair de non-maître et le maître de DB.

4. S'il n'y a aucun trafic de retour du serveur puis il est probable qu'un FW pourrait bloquer le port entre des deux l'emplacement logique serveurs.

Voici une capture typique de paquet d'une connexion fonctionnante entre le client et serveur :

En cela l'exemple l'IP du client est 10.48.54.119 et le serveur est 10.48.54.75.

No.	Time	Source	Destination	Protocol	Length	Info
54	2017-03-31 08:36:13.558867	10.48.54.119	10.48.54.75	TCP	66	35826 → 5432 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 WS=128
55	2017-03-31 08:36:13.558976	10.48.54.75	10.48.54.119	TCP	66	5432 → 35826 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460 SACK_PERM=1 WS=128
56	2017-03-31 08:36:13.559098	10.48.54.119	10.48.54.75	TCP	60	35826 → 5432 [ACK] Seq=1 Ack=1 Win=29312 Len=0
57	2017-03-31 08:36:13.559147	10.48.54.119	10.48.54.75	PGSQL	62	>
58	2017-03-31 08:36:13.559169	10.48.54.75	10.48.54.119	TCP	54	5432 → 35826 [ACK] Seq=1 Ack=9 Win=29312 Len=0
59	2017-03-31 08:36:13.559710	10.48.54.75	10.48.54.119	PGSQL	55	<
60	2017-03-31 08:36:13.559798	10.48.54.119	10.48.54.75	TCP	60	35826 → 5432 [ACK] Seq=9 Ack=2 Win=29312 Len=0
61	2017-03-31 08:36:13.560499	10.48.54.119	10.48.54.75	TLSv1.2	257	Client Hello
62	2017-03-31 08:36:13.560963	10.48.54.75	10.48.54.119	TLSv1.2	2605	Server Hello, Certificate, Certificate Request, Server Hello Done
63	2017-03-31 08:36:13.561060	10.48.54.119	10.48.54.75	TCP	60	35826 → 5432 [ACK] Seq=212 Ack=2553 Win=34304 Len=0
64	2017-03-31 08:36:13.564761	10.48.54.119	10.48.54.75	TLSv1.2	2983	Certificate, Client Key Exchange, Certificate Verify, Change Cipher Spec, Encrypted Handshake Message
65	2017-03-31 08:36:13.564810	10.48.54.75	10.48.54.119	TCP	54	5432 → 35826 [ACK] Seq=2553 Ack=3141 Win=36224 Len=0
66	2017-03-31 08:36:13.568036	10.48.54.75	10.48.54.119	TLSv1.2	1688	New Session Ticket, Change Cipher Spec, Encrypted Handshake Message
67	2017-03-31 08:36:13.568194	10.48.54.119	10.48.54.75	TCP	60	35826 → 5432 [ACK] Seq=3141 Ack=4187 Win=37632 Len=0
68	2017-03-31 08:36:13.568551	10.48.54.119	10.48.54.75	TLSv1.2	124	Application Data
69	2017-03-31 08:36:13.570438	10.48.54.75	10.48.54.119	TLSv1.2	406	Application Data
70	2017-03-31 08:36:13.571070	10.48.54.119	10.48.54.75	TLSv1.2	120	Application Data
71	2017-03-31 08:36:13.571738	10.48.54.75	10.48.54.119	TLSv1.2	382	Application Data

## Informations connexes

Pour plus sur le dépannage des questions avec, et les questions qui associent au DB groupant, référez-vous aux Foires aux questions dans ces liens

- <https://kb.acano.com/content/23/238/en/when-clustering-database-servers-why-do-i-need-to-place-them-on-different-locations.html>
- <https://kb.acano.com/content/8/212/en/we-have-a-database-cluster-and-i-see-a-database-error-or-warning-in-the-log-what-should-i-do.html#content>
- <https://kb.acano.com/content/8/237/en/one-or-more-of-the-database-servers-are-not-connected-or-in-sync-state-what-should-i-do.html#content>
- <https://kb.acano.com/content/23/214/en/what-do-i-do-if-there-is-no-master-database.html>
- <https://kb.acano.com/content/23/75/en/how-do-i-move-the-master-database.html>