

Procedimento para gerenciar o nó intermediário no conjunto de réplicas do CPS

Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Componentes Utilizados](#)

[Informações de Apoio](#)

[Problema](#)

[Procedimento para Gerenciar o Árbitro em um Conjunto de Réplicas](#)

Introduction

Este documento descreve o procedimento para gerenciar o nó Arbiter no conjunto de réplicas do Cisco Policy Suite (CPS).

Prerequisites

Requirements

A Cisco recomenda que você tenha conhecimento destes tópicos:

- Linux
- CPS
- MongoDB

Observação: a Cisco recomenda que você tenha acesso raiz privilegiado à CLI do CPS.

Componentes Utilizados

As informações neste documento são baseadas nestas versões de software e hardware:

- CPS 20.2
- Unified Computing System (UCS)-B
- MongoDB v3.6.17 e v3.4.16

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. Se a rede estiver ativa, certifique-se de que você entenda o impacto potencial de qualquer comando.

Informações de Apoio

O CPS usa o MongoDB para constituir sua estrutura básica de banco de dados (DB). Possui

vários conjuntos de réplicas para várias finalidades: ADMIN, Subscriber Profile Repository (SPR), BALANCE, SESSION, REPORTING e AUDIT.

Um conjunto de réplicas no MongoDB é um grupo de processos mongod que mantêm o mesmo conjunto de dados. Os conjuntos de réplicas fornecem redundância e alta disponibilidade (HA). Com várias cópias de dados em diferentes servidores de banco de dados, ele permite operações de leitura de compartilhamento de carga.

Em algumas circunstâncias (como se você tivesse uma instância primária e uma secundária, mas as restrições de custo proibissem a adição de outra secundária), você pode optar por adicionar uma instância mongod a um conjunto de réplicas como um intermediário para votar nas eleições. Um árbitro tem exatamente 1 voto eleitoral. Por padrão, um intermediário tem prioridade 0.

Os arbitradores são instâncias mongod que fazem parte de um conjunto de réplicas, mas não contêm dados (o que significa que eles não fornecem redundância de dados). Podem, no entanto, participar nas eleições. Um intermediário participa das eleições para o principal, mas um intermediário não tem uma cópia do conjunto de dados e não pode se tornar um principal.

Os árbitros têm requisitos mínimos de recursos e não exigem hardware dedicado. Você pode implantar um intermediário em um servidor de aplicativos ou em um host que apenas monitora a rede.

Um intermediário não armazena dados, mas até que o processo intermediário mongod seja adicionado ao conjunto de réplicas, o intermediário age como qualquer outro processo mongod e inicia com um conjunto de arquivos de dados e um diário de tamanho completo.

Este é um exemplo de conjunto de réplicas, ou seja, **set07**.

```
| SET NAME - PORT : IP ADDRESS - REPLICA STATE - HOST NAME - HEALTH - LAST SYNC -PRIORITY
|-----|
| SESSION:set07 |
| Status via arbitervip:27727 sessionmgr01:27727 sessionmgr02:27727 |
| Member-1 - 27727 : - SECONDARY - sessionmgr01 - ON-LINE - 0 sec - 2 |
| Member-2 - 27727 : 192.168.10.146 - ARBITER - arbitervip - ON-LINE - ----- - 0 |
| Member-3 - 27727 : - PRIMARY - sessionmgr02 - ON-LINE - ----- - 3 |
|-----|
```

Problema

Suponha que haja um problema com um intermediário ou um requisito para alterar o intermediário em um conjunto de réplicas, então você deve remover o intermediário atual e adicionar um novo intermediário ao conjunto de réplicas.

Procedimento para Gerenciar o Árbitro em um Conjunto de Réplicas

Etapa 1. Verifique a versão do shell mongo no CPS e o novo intermediário. Execute este comando a partir do sessionmgr primário no conjunto de réplicas e no novo nó intermediário.

Exemplo de saída do sessionmgr:

```
[root@sessionmgr02 ~]# mongo --version
MongoDB shell version v3.6.17
```

Se a versão do shell mongo for a mesma tanto no sessionmgr primário como no novo intermediário ou se a nova versão do shell mongo do intermediário for superior, navegue para o Passo 6.

Caso contrário, se a nova versão de shell do mongo intermediário for mais baixa, você deverá definir `featureCompatibilityVersion` como o valor mais baixo no banco de dados admin do conjunto de réplicas com as próximas etapas.

Exemplo de caso em que a nova versão de shell do arbiter mongo é inferior à do CPS sessionmgr:

```
[root@pcrfclient02 ~]# mongo --version
MongoDB shell version v3.4.16
```

Etapa 2. Faça login na instância primária mongo do conjunto de réplicas.

Command template:

```
#mongo --host <sessionmgrXX> --port <Replica Set port>
```

Sample command:

```
#mongo --host sessionmgr02 --port 27727
```

Etapa 3. Execute este comando para exibir a `featureCompatibilityVersion` no banco de dados admin do conjunto de réplicas.

```
set07:PRIMARY> db.adminCommand( { getParameter: 1, featureCompatibilityVersion: 1 } )
{
  "featureCompatibilityVersion" : {
    "version" : "3.6"
  },
  "ok" : 1,
  "operationTime" : Timestamp(1663914140, 1),
  "$clusterTime" : {
    "clusterTime" : Timestamp(1663914140, 1),
    "signature" : {
      "hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA="),
      "keyId" : NumberLong(0)
    }
  }
}
```

Etapa 4. Execute este comando para `setfeatureCompatibilityVersion` como 3.4 no banco de dados admin do conjunto de réplicas.

```
set07:PRIMARY> db.adminCommand( { setFeatureCompatibilityVersion: "3.4" } )
{ "ok" : 1 }
set07:PRIMARY>
```

Etapa 5. Execute este comando para verificar se `featureCompatibilityVersion` foi alterado para 3.4 no banco de dados admin do conjunto de réplicas.

```
set07:PRIMARY> db.adminCommand( { getParameter: 1, featureCompatibilityVersion: 1 } )
```

```
{ "featureCompatibilityVersion" : { "version" : "3.4" }, "ok" : 1 }
set07:PRIMARY>
```

Etapa 6. Efetue login no Cluster Manager e modifique

O `/var/qps/config/deploy/csv/AdditionalHosts.csv` arquivo com novos detalhes do intermediário.

```
#vi /var/qps/config/deploy/csv/AdditionalHosts.csv
```

Provide new arbiter details in this format:

Host Alias IP Address

```
new-arbiter new-arbiter xx.xx.xx.xx
```

Passo 7. Importe a configuração CSV.

```
#!/var/qps/install/current/scripts/import/import_deploy.sh
```

Etapa 8. Verificar `/etc/hosts` que foram atualizados com as informações dos novos árbitros.

```
#cat /etc/hosts | grep arbiter
```

Etapa 9. Execute este comando para sincronizar `/etc/hosts`.

```
#!/var/qps/bin/update/synchosts.sh
```

Syncing to following QNS Servers:

```
lb01 lb02 sessionmgr01 sessionmgr02 qns01 qns02 pcrfclient01 pcrfclient02
```

Do you want to Proceed? (y/n):y

```
lb01
```

```
lb02
```

```
sessionmgr01
```

```
sessionmgr02
```

```
qns01
```

```
qns02
```

```
pcrfclient01
```

```
pcrfclient02
```

Etapa 10. Verifique se os scripts `mon_db` são interrompidos em VMs `pcrfclient`.

```
#monsum | grep mon_db_for
```

Se parado, esta é a saída:

```
mon_db_for_lb_failover Not monitored Program
```

```
mon_db_for_callmodel Not monitored Program
```

Se não for interrompido, esta é a saída:

```
mon_db_for_lb_failover OK Program
```

```
mon_db_for_callmodel OK Program
```

Observação: se os scripts `mon_db` não forem interrompidos, execute esses comandos nas respectivas VMs `pcrfclient` para interrompê-los manualmente.

```
#monit stop mon_db_for_lb_failover
```

```
#monit stop mon_db_for_callmodel
```

Etapa 11. Execute este comando em pcrfclient01 para remover o intermediário atual do conjunto de réplicas (set07 é um exemplo nesta etapa).

```
#build_set.sh --session --remove-members --setname set07
```

Please enter the member details which you going to remove from the replica-set

Member:Port -----> arbitervip:27727

arbitervip:27727

Do you really want to remove [yes(y)/no(n)]: y

Etapa 12. Execute este comando no Gerenciador de Cluster para verificar se o intermediário foi removido do set07, a saída de set07 não pode ter o árbitro atual nele.

```
#diagnostics.sh --get_replica_status
```

Expected output:

```
-----|
|-----|
| SESSION:set07 |
| Status via sessionmgr01:27727 sessionmgr02:27727 |
| Member-1 - 27727 : - SECONDARY - sessionmgr01 - ON-LINE - 0 sec -|
| Member-2 - 27727 : - PRIMARY - sessionmgr02 - ON-LINE - ----- -|
|-----|
-----|
```

Etapa 13. Atualize o mongoConfig.cfg para ter o intermediário adequado no conjunto de réplicas modificado. Substitua o intermediário atual (ARBITER=arbiter) pelo novo intermediário (ARBITER=new-arbiter). Execute este comando a partir do Gerenciador de Cluster.

```
#vi /etc/broadhop/mongoConfig.cfg
```

Configuração atual:

```
[SESSION-SET2]
SETNAME=set07
OPLOG_SIZE=5120
ARBITER=arbitervip:27727
ARBITER_DATA_PATH=/var/data/sessions.7
MEMBER1=sessionmgr02:27727
MEMBER2=sessionmgr01:27727
DATA_PATH=/var/data/sessions.1/2
[SESSION-SET2-END]
```

Configuração necessária:

```
[SESSION-SET2]
SETNAME=set07
OPLOG_SIZE=5120
ARBITER=new-arbiter:27727
ARBITER_DATA_PATH=/var/data/sessions.7
MEMBER1=sessionmgr02:27727
MEMBER2=sessionmgr01:27727
DATA_PATH=/var/data/sessions.1/2
[SESSION-SET2-END]
```

Etapa 14. Copiar o arquivo atualizado mongoConfig.cfg a todas as VMs. Execute este comando a partir do Gerenciador de cluster.

```
#copytoall.sh /etc/broadhop/mongoConfig.cfg /etc/broadhop/mongoConfig.cfg
```

Etapa 15. Adicione um novo membro intermediário a set07. No Gerenciador de Cluster, execute `/var/qps/install/current/scripts/build/build_etc.sh` para criar o `/etc/directory`.

Etapa 16. Verifique se o novo membro intermediário foi adicionado ao conjunto de réplicas depois de executar o comando `build_etc.sh`, agora você deve aguardar que o servidor AIDO crie/atualize os conjuntos de réplicas com o novo intermediário.

```
#diagnostics.sh --get_replica_status
```

Expected Output:

```
| SET NAME - PORT : IP ADDRESS - REPLICA STATE - HOST NAME - HEALTH - LAST SYNC -PRIORITY  
|-----|  
|-----|  
| SESSION:set07 |  
| Status via arbitervip:27727 sessionmgr01:27727 sessionmgr02:27727 |  
| Member-1 - 27727 : - SECONDARY - sessionmgr01 - ON-LINE - 0 sec - 2 |  
| Member-2 - 27727 : xx.xx.xx.xx - ARBITER - new-arbiter - ON-LINE - ----- - 0 |  
| Member-3 - 27727 : - PRIMARY - sessionmgr02 - ON-LINE - ----- - 3 |  
|-----|
```

Observação: se o novo membro intermediário não for adicionado, continue com as próximas etapas. Navegue até a Etapa 18.

Etapa 17. Execute este comando a partir do gerenciador de cluster para adicionar um novo membro intermediário à força.

```
#build_set.sh --DB_NAME --add-members --setname Setxxx --force
```

Etapa 18. Se a porta do intermediário ainda não estiver ativa, execute este comando a partir do novo nó intermediário para iniciar o mesmo.

Command syntax:

```
#/etc/init.d/sessionmgr-XXXXXX start
```

Sample command:

```
#/etc/init.d/sessionmgr-27727 start
```

Etapa 19. Verifique se o novo Arbiter foi adicionado com êxito.

```
#diagnostics.sh --get_replica_status
```

Etapa 20. Execute este comando a partir do Gerenciador de Cluster para atualizar a prioridade do BD adequadamente.

```
# cd /var/qps/bin/support/mongo/  
# ./set_priority.sh --db session  
# ./set_priority.sh --db spr  
# ./set_priority.sh --db admin  
# ./set_priority.sh --db balance  
# ./set_priority.sh --db audit  
# ./set_priority.sh --db report
```

Etapa 21. Execute este comando no Gerenciador de Cluster para verificar as alterações no

conjunto de réplicas.

```
#diagnostics.sh --get_replica_status
```

Expected Output:

```
| SET NAME - PORT : IP ADDRESS - REPLICA STATE - HOST NAME - HEALTH - LAST SYNC -PRIORITY  
|-----  
|-----|  
| SESSION:set07 |  
| Status via arbitervip:27727 sessionmgr01:27727 sessionmgr02:27727 |  
| Member-1 - 27727 : - SECONDARY - sessionmgr01 - ON-LINE - 0 sec - 2 |  
| Member-2 - 27727 : xx.xx.xx.xx - ARBITER - new-arbiter - ON-LINE - ----- - 0 |  
| Member-3 - 27727 : - PRIMARY - sessionmgr02 - ON-LINE - ----- - 3 |  
|-----|
```

Etapa 22. Verifique se os scripts mon_db são restaurados em VMs pcrfclient. Caso contrário, você deverá iniciá-los manualmente.

```
#monsum | grep mon_db_for
```

Para habilitar o script mon_db, faça login em todas as VMs pcrfclient e execute estes comandos:

```
# monit start mon_db_for_lb_failover  
# monit start mon_db_for_callmodel
```

Sobre esta tradução

A Cisco traduziu este documento com a ajuda de tecnologias de tradução automática e humana para oferecer conteúdo de suporte aos seus usuários no seu próprio idioma, independentemente da localização.

Observe que mesmo a melhor tradução automática não será tão precisa quanto as realizadas por um tradutor profissional.

A Cisco Systems, Inc. não se responsabiliza pela precisão destas traduções e recomenda que o documento original em inglês ([link fornecido](#)) seja sempre consultado.