

Dépannez la panne de validation de capacité de passerelle du rappel de courtoisie CVP (CCB)

Contenu

[Introduction](#)

[Conditions préalables](#)

[Conditions requises](#)

[Composants utilisés](#)

[Informations générales](#)

[Symptômes](#)

[Dépannez](#)

[Solution](#)

[Solution permanente](#)

[Solution de finale de test](#)

Introduction

Ce document décrit comment dépanner un problème du Customer Voice Portal (CVP) CCB quand l'appelant n'obtient pas une offre CCB parce que la capacité de passerelle de joncteur réseau a dépassé.

Conditions préalables

Conditions requises

Cisco vous recommande de prendre connaissance des rubriques suivantes :

- CVP
- Rappel de courtoisie de Cisco CVP

[Composants utilisés](#)

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

- Serveur 10.5 CVP
- Unified Contact Center Enterprise (UCCE) 10.5

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.

[Informations générales](#)

Avant que le problème de capacité de passerelle soit dépanné, il est important de comprendre le processus de validation de joncteur réseau dans CCB. Fondamentalement, le processus détermine d'abord le nombre d'appels à partir de la table de **Callback_current** avec **EventTypeID** dedans (21,22,23) ; En suspens, Inprogress, expérimentaux pour les passerelles spécifiques et les emplacements.

En second lieu, à partir de la même table de **Callback_current**, déterminez, le nombre d'appels terminés avec la cause connectée : **EventTypeID** = 24 (terminé), et **CauseID** = 27 (connecté).

Enfin le processus ajoute ces deux valeurs et rivalise avec le nombre de joncteurs réseau configurés sous le service Survivability.tcl.

Si le résultat est au-dessus du seuil de joncteurs réseau configuré, le processus renvoie une panne (le retour 1), autrement renvoie l'ok (retour 0).

En résumé, la formule pour valider les joncteurs réseau utilisés pour CCB est :

Joncteurs réseau CCB < (table de Callback_current avec EventTypeID dedans (21,22,23) ; En suspens, Inprogress, expérimentaux pour les passerelles spécifiques) + table de Callback_current d'EventTypeID = 24 (terminé), et CauseID = 27 (connecté)

Si la valeur de joncteurs réseau CCB est inférieure la validation échoue.

Symptômes

Un appel d'arrivée n'obtient pas l'offre CCB. L'appel va directement aligner sans se soucier le temps d'attente prévu (EWT)

Dépannez

Étape 1. Collectez les journaux d'activité de l'application de CallbackEntry du serveur du langage XML de Voix (VXML).

Étape 2. Recherche dans les journaux d'activité pour tout appel où la validation n'en est aucune :

```
Validate_02,data,result,none
```

Ce qui signifie que la validation n'a pas passé. Obtenez le GUID pour cet appel. Filtrez l'appel par le callid d'activité et recherchez un callid comme cet exemple :

```
start,parameter,callid=BBBBAAAACCCDDDDDEEEFFFFFAAAABBBB
```

Étape 3. Collectez les logs d'enregistrement CVP pour le serveur d'enregistrement. Trouvez le même callid dans les logs d'enregistrement CVP.

```
ValidateHandler:ValidateHandler.exec: ValidateHandler GUID=BBBBAAAACCCDDDDDEEEFFFFFAAAABBBB  
results:none validation status bitmask=0x00000103
```

Étape 4. Convertissez le nombre de bitmask en binaire. Utilisez une calculatrice de programmeur :
0001 00000011

Étape 5. Vérifiez le bitmask de guide d'enregistrement CVP pour des tables CCB. Vous devriez voir que la validation échoue en raison de « EXCEED_CAPACITY_GW ».

```

OK 00000000 00000001
00000000 00000010 ICM_NO_SCHEDULED_ALLOWED
00000000 00000100 ICM_NO_PREEMPTIVE_ALLOWED
00000000 00001000 NOT_IN_QUEUE
00000000 00010000 TOD
00000000 00100000 EWT
00000000 01000000 PROBE_FAILED_NO_RESPONSE
00000000 10000000 PROBE_FAILED_NO_CONFIG
00000001 00000000 EXCEED_CAPACITY_GW
00000010 00000000 EXCEED_CAPACITY_QUEUE

```

Remarque: ICM_NO_SCHEDULED_ALLOWED et le bit CORRECT sont toujours placés

Étape 6. Rétrécissez la question vers le bas à une file d'attente spécifique. Vérifiez le CCB Servlet du serveur d'enregistrement CVP afin de déterminer s'il y a n'importe quelles files d'attente spécifiques où CCB n'est pas offert. Ouvrez un navigateur Web et tapez.

<http:// {IP Address}:8000/cvp/CallbackServlet?method=Diag> de serveur d'enregistrement

C'est un exemple d'une file d'attente où CCB est offert :

Queue Name	ICM EWT	DQRate	Reconnect Time	SLA Time	Pending Callbacks	Percent Callbacks	<rw-ewt>
billing	2000	2000	30	60	0	0	2.67

Position	Number	Callback	Remaining Wait Time	In Queue State	GUID	Entrance Date	Retry	Validation Status	Start Date	Updated	Recording
1	""	false	1986	INQUEUE	9977E70000010000000000612C6C90A	10:26:53 09/01/2017	false	0x0003	10:26:53 09/01/2017	10:26:53 09/01/2017	""

C'est un exemple d'une file d'attente où CCB n'est pas offert

Queue Name	ICM EWT	DQRate	Reconnect Time	SLA Time	Pending Callbacks	Percent Callbacks	<rw-ewt>
billing	2000	2000	30	60	0	0	2.67

Étape 7. Vérifiez si les files d'attente sont servies par une passerelle spécifique. Vérifiez la configuration de passerelle (paramètres d'application de capacité de survie).

```

application
service new-call flash:bootstrap.vxml
!
service survivability flash:survivability.tcl
paramspace callfeature med-inact-det enable
param ccb id:10.201.198.21;loc:CALO;trunks:512

```

Étape 8. Si la configuration est correcte, vérifiez l'information enregistrée dans la base de données du serveur d'enregistrement (Informix) pour déterminer le nombre de faire appel à ces passerelle

et emplacement spécifiques. Vous pouvez vérifier par l'id CCB (10.201.198.21 dans ce cas) ou le locattion (CALO dans cet exemple).

Étape 9. Sur le serveur d'enregistrement, base de données Informix d'accès.

Ouvrez une demande CMD et tapez : **dbacces**

Naviguez vers la **connexion > se connectent**

Exemple choisi de **cvp**

cvp_dbadmin de nom d'utilisateur de type

mot de passe de type

base de données choisie de **callback@cvp**

quittez et naviguez vers des langages de requête

Étape 10. Exécutez la requête :

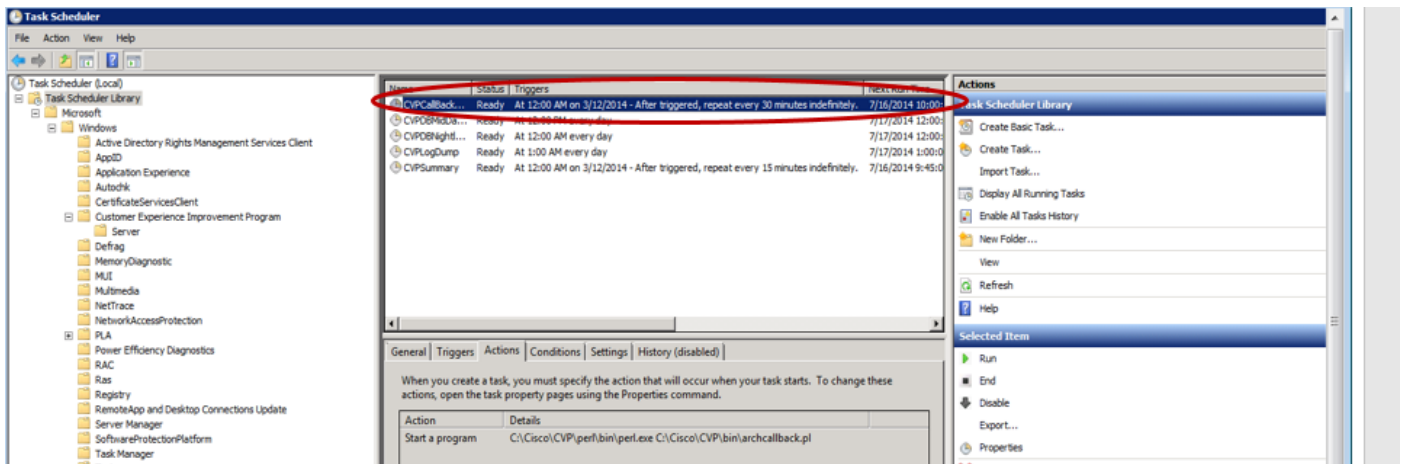
Sélectionnez le compte (*) de callback_current où le == « CALO » d'emplacement ;

Étape 11. Si la valeur est le même ou supérieur à la valeur de joncteur réseau configurée dans la passerelle pour les emplacements, c'est la raison pour laquelle le validation échoue, puisque les nombres maximaux de joncteurs réseau permis ont été atteints dans la table de Callback_Current.

Remarque: Comme référencé dans le guide d'enregistrement CVP, la table de rappel est une vue de deux tables : Callback_Current et Callback_Historical. Les deux tables sont identiques. Toutes les 30 minutes, des données pour des appels terminés sont tirées de Callback_Pending et déplacées à Callback_Historical.

Étape 12. Si la valeur de joncteur réseau par emplacement a atteint ses limites dans la table de Callback_Current et il n'y a aucun rappel dans la file d'attente que ceci indique qu'il y a un problème en déplaçant les enregistrements de rappel de Callback_Current à la table de Callback_Historical.

Étape 13. Assurez-vous que CVPCallbackArchive s'exécute sous les tâches de programme (serveur d'enregistrement CVP). Naviguez **pour commencer - > programme - > des accessoires - > des outils système - > tâche programmée.**



Étape 14. Si cette tâche **CVPCallbackArchive** se termine assurez que code de sortie est (0x0).

Name	Status	Triggers	Next Ru...	Last Run Ti...	Last Run Result	Author
CVPCallBack...	Ready	At 12:00 AM on 1/31/2017 - ...	8/30/20...	8/30/2017 4...	The operation completed successfully. (0x0)	Administrator
CVPDBMid...	Ready	At 12:00 PM every day	8/31/20...	8/30/2017 1...	The operation completed successfully. (0x0)	Administrator
CVPDBNight...	Ready	At 12:00 AM every day	8/31/20...	8/30/2017 1...	The operation completed successfully. (0x0)	Administrator
CVPLogDump	Ready	At 1:00 AM every day	8/31/20...	8/30/2017 1...	The operation completed successfully. (0x0)	Administrator
CVPSummary	Ready	At 12:00 AM on 1/31/2017 - ...	8/30/20...	8/30/2017 4...	The operation completed successfully. (0x0)	Administrator
GoogleHeda	Ready	Multiple triggers defined	8/30/20...	8/30/2017 3...	The operation completed successfully. (0x0)	

Étape 15. Si étape 13 et 14 sont bien, mais toujours aucune données dans la table de **Callback_Historical**, vous devrez déterminer pourquoi les informations ne sont pas ajoutées dans la base de données. Vérifiez l'intégrité de l'information enregistrée dans le courant et la table historique. Exécutez cette requête sur la fenêtre des dbaccess CMD d'informix :

```
Select count (*) from callback_current where surrogateid in (select surrogateid from callback_historical);
```

Étape 16. Si le compte est 1 ou plus élevé, il signifie que la clé primaire sur la table en cours existent déjà dans la table historique et les informations ne sont pas ajoutées dans la base de données. Dans la plupart de ces scénarios, enregistrements d'un doublon de causes de condition de compétitivité à entrer dans la table **callback_current**.

GUID au mappage de **surrogateid** se produit sur la table de file d'attente. Dans les situations où l'appel se déplace de l'attente de rappel au script de file d'attente de rappel, il semble y a une fenêtre où le travail d'archives déplace les enregistrements du courant à l'historique et l'application entre dans un nouveau record dans la table en cours avec le même **surrogateid**. Cette question est liée à ce CDETS [CSCuq86400](#)

Solution

Étape 1. Base de données Informix d'Access. Ouvrez une demande CMD et tapez : **dbaccess**

Étape 2. Naviguez vers la **connexion > connectent** l'exemple choisi de **cvp**. **Cvp_dbadmin** de nom d'utilisateur de type et mot de passe de type

Étape 3. La sortie choisie de base de données de **callback@cvp** et naviguent vers des **langages de requête**

Étape 4. Exécutez ces commandes :

effacement de callback_current où surrogateid dedans (surrogateid choisi de callback_historical) ;

S'il y a une erreur de table provisoire faites :

t1 de table de baisse ;

Étape 5. Exécutez la procédure SP qui déplace les informations du courant à la table historique de rappel des dbaccess de fenêtre de langage d'interrogation.

EXÉCUTEZ le sp_arch_callback() de PROCÉDURE ;

Étape 6. Vérifiez qu'il n'y a pas autant d'enregistrements dans la table en cours en tant qu'avant.

Sélectionnez le compte (*) de callback_current où le == « CALO » d'emplacement ;

Solution permanente

Étape 1. Naviguez vers **Cisco \ CVP \ informix_frag** et ouvrez **sp_arch_callback.sql** dans un éditeur de texte.

Étape 2. Uncomment cette ligne au début du fichier : **--sp_arch_callback de procédure de baisse ;** (retirez -- au début de la ligne).

Étape 3. Ajoutez cette ligne : **effacement de callback_current où substitué dedans (surrogateid choisi de callback_historical) ;** ensuite

créez la ligne de sp_arch_callback() de procédure.

Étape 4. Sauvegardez le fichier.

Étape 5. C'est un exemple sur la façon dont la première pièce du fichier devrait ressembler à.

```
{*****
Stored procedure to move completed calls out of the active table into the
historical table.
*****}
drop procedure sp_arch_callback;
create procedure sp_arch_callback()

DEFINE p_ageoff INTEGER;

-- delete any duplicates found in current table.

delete from callback_current where surrogateid in (select surrogateid from callback_historical);
```

Solution de finale de test

Étape 1. Ouvrez une demande CMD et exécutez la commande : **dbschema**

dbschema - rappel d - sp_arch_callback f

Remarque: Si vous avez une question d'autorisation en exécutant la commande de dbschema, la procédure de connexion comme cvp_dbadmin dans le serveur d'enregistrement et l'essai une fois de plus.

Étape 2. De la sortie, assurez-vous que l'effacement de la commande est exécuté.

```
C:\Users\Administrator>dbschema -d callback -f sp_arch_callback
DBSCHEMA Schema Utility          INFORMIX-SQL Version 12.10.FC3

create procedure "Administrator".sp_arch_callback()
DEFINE p_ageoff INTEGER;
-- delete any duplicates found in current table.
delete from callback_current where surrogateid in (select surrogateid from callb
ack_historical);
SELECT surrogateid
FROM Callback_current
WHERE EventTypeID in (24,29) -- Completed, Too many callbacks
AND CauseID in (27,28) -- Connected, Cancelled
INTO TEMP t1 WITH NO LOG;
```