

Foires aux questions pour le Cisco MediaSense

Contenu

[Introduction](#)

[Conditions préalables](#)

[Conditions requises](#)

[Composants utilisés](#)

1. [Comment est-ce que vous corrélerez-vous des id d'appel de référence pour différents scénarios d'appel, sous l'Unified Communications Manager téléphonez bifurquer ?](#)
 - 1.1. [Caractéristique d'association de recherche de MediaSense et d'appel de jeu](#)
 - 1.2. [Scénario d'Attente/Reprise d'agent](#)
 - 1.3. [Scénario d'Attente/Reprise de client](#)
 - 1.4. [Transfert d'agent à un autre scénario d'agent](#)
 - 1.5. [Conférences d'agent avec un autre scénario d'agent](#)
2. [Comment corrélons-nous des id d'appel de référence pour différentes sessions d'appel, sous bifurquer d'Unified Border Element ?](#)
 - 2.1. [modification de codecs de Mid-appel](#)
 - 2.2. [Consultez les transferts](#)
 - 2.3. [Consultez la détection d'appels](#)
 - 2.4. [Consultez la détection d'appels de plusieurs participants](#)
 - 2.5. [Résumé](#)
3. [Comment associez-vous des appels dans le Cisco MediaSense avec leurs apparences dans d'autres composants de la solution ?](#)
 - 3.1. [Tableau de corrélation d'identifiant](#)
4. [Comment déterminez-vous quelle piste a l'appelant et quelle piste a l'appelé ?](#)
 - 4.1. [Pour des appels a bifurqué par le CUBE](#)
 - 4.2. [Pour des appels a bifurqué par des téléphones d'Unified CM](#)
5. [Quelles sont les causes possibles pour un état de session de CLOSED ERROR ?](#)
6. [Quelle est la différence entre les sessions taillées et supprimées ?](#)
 - 6.1. [Avec la requête de getAllPrunedSessions](#)
 - 6.2. [Avec la requête de getSessions](#)
 - 6.3. [Pourquoi la différence de comportement en sessions taillées et supprimées ?](#)
7. [Comment configurer une passerelle TDM pour bifurquer de medias ?](#)
8. [Comment capturer le téléphone réel de destination en utilisant un groupe de recherche ?](#)
9. [Pourquoi enregistrement Fondé\(e\) sur le réseau d'Unified Communications Manager recommandé en tant que bifurquer préféré est-il le mécanisme ?](#)
10. [Pourquoi est-ce qu'un noeud prend plus long pour améliorer à MediaSense 10.5 ?](#)
11. [Quelle est l'incidence des modifications russes de fuseau horaire sur l'application de recherche et de jeu de MediaSense ?](#)
12. [Quels sont les langages pris en charge par MediaSense ?](#)
13. [Comment surveiller la performance du système de MediaSense ?](#)
14. [Comment configurer un navigateur pour exécuter le lecteur de dans-navigateur dans MediaSense ?](#)

Introduction

Ce document décrit des Foires aux questions pour le serveur de sens de medias de Cisco.

Conditions préalables

Exigences

Cisco vous recommande de prendre connaissance des rubriques suivantes :

- [Cisco MediaSense](#)
- Gestionnaire de Cisco Unified Communications (CUCM)

Composants utilisés

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

Cisco MediaSense 10.5

1. Comment est-ce que vous corrélerez-vous des id d'appel de référence pour différents scénarios d'appel, sous l'Unified Communications Manager téléphonez bifurquer ?

Dans le Cisco MediaSense, les méta-données pour chaque appel fournissent seulement le xRefCi (ID d'appel de référence) et la référence de périphérique (extension) du périphérique bifurquant et du périphérique final (peut être une passerelle de conférence ou tout autre téléphone).

Le paramètre de xRefCi est l'identifiant de la Communication Manager unifiée pour un flux multimédia particulier. Ils ne correspondent pas toujours 1:1 aux pistes enregistrées.

1.1. Caractéristique d'association de recherche de MediaSense et d'appel de jeu

MediaSense génère des plusieurs sessions pour un appel qui est enregistré en cas d'attente/de reprise ou de transfert, qui le rendent difficile d'identifier toutes les sessions enregistrées dans un appel. Afin de pouvoir associer ces sessions enregistrées dans un appel unique, MediaSense introduit une nouvelle caractéristique nommée comme association d'appel. Par cette caractéristique, tous les appels fortement associés avec une valeur commune de xRefci sont groupés ensemble. MediaSense 10.5 prend en charge la caractéristique d'association d'appel pour des enregistrements de Construire-dans-passerelle.

1.2. Scénario d'Attente/Reprise d'agent

1. L'agent A (extn 1000) et le C d'appelant (extn 2000) se sont appelés et dans l'état parlant.
2. L'agent A met invitent l'attente.
3. Appel de reprises de l'agent A.

Il y a deux sessions enregistrées pour ce scénario :

- La session avec le sessionId = le S1 avec ces deux pistes, parce que le délai prévu /segment avant agent met invitent l'attente :
 trackNumber = 0 avec le participant A (deviceRef = 1000, xRefCi = aaaa)
 trackNumber = 1 avec le participant B (deviceRef = 2000, xRefCi = cccc)
- La session avec le sessionId = le S2 avec ces deux pistes, pour le délai prévu/segment après agent reprend l'appel.
 trackNumber = 0 avec le participant A (deviceRef = 1000, xRefCi = aaaa)
 trackNumber = 1 avec le participant B (deviceRef = 2000, xRefCi = cccc)

MediaSense n'enregistre pas le segment de l'appel tandis que l'agent a mis l'appel sur l'attente.

1.3. Scénario d'Attente/Reprise de client

1. L'agent A (extn 1000) et le C d'appelant (extn 2000) allé et dans l'état parlant.
2. Le C de client met invitent l'attente.
3. Appel de reprises de C de client.

L'appel entier est enregistré en une session pour ce scénario :

- Session avec le sessionId = le S1 avec ces deux pistes :
 trackNumber = 0 avec le participant A (deviceRef = 1000, xRefCi = aaaa)
 trackNumber = 1 avec le participant B (deviceRef = 2000, xRefCi = cccc)

Dans ce scénario MediaSense enregistre également le segment de l'appel tandis que le client a mis l'appel sur l'attente.

1.4. Transfert d'agent à un autre scénario d'agent

1. Le C d'appelant (extn 2000) appelle l'agent A (extn 1000)
2. L'agent A (extn 1000) consulte avec l'agent B (3000)
3. L'agent A (extn 1000) se termine le transfert.
4. L'agent B (extn 3000) raccroche.

Avec l'Unified Communications Manager 9.x et plus tôt, voici les résultats :

le C 1.Caller (extn 2000) appelle l'agent A (extn 1000)

La session S1 COMMENCÉE, la piste 0 est A (extn 1000), la piste 1 est C (extn 2000).

les transferts 2.Agent A (extn 1000) appellent à un autre agent B (extn 3000). Des périphériques A et B sont installés pour bifurquer.

le C 3.Caller (extn 2000) entend la musique d'attente (MoH).

4.Agent A (extn 1000) parle à B (extn 3000).

- Session S1 FINIE
- La session S2 COMMENCÉE, la piste 0 est A (extn 1000), la piste 1 est B (extn 3000)
- La session S3 COMMENCÉE, la piste 0 est B (extn 3000), la piste 1 est A (extn 1000)

Considérations :

- Les extrémités de la session S1 dues au téléphone de l'agent A a placé le C d'appelant sur l'attente.

- Les sessions S2 et S3 existent parce que les deux téléphones sont configurés pour bifurquer.
- Les participants et le xRefCi pour les deux participants à S2 et à S3 sont identiques, mais en positions inverses entre eux.
- des valeurs de xRefCi du S1 ne sont pas reflétées dans S2 ou S3, puisque la consultation est considérée un appel indépendant.

5.Agent A (extn 1000) se termine le transfert.

6.C (extn 2000) parlant à B (extn 3000).

7.A (extn 1000) a déconnecté.

Session S2 FINIE.

La session S3 MISE À JOUR, et la piste 0 est B (l'extn 3000) et la piste 1 est C (extn 2000).

Considérations :

- Un transfert d'extrémité déclenche une MISE À JOUR de la session enregistrée existante.
- Le participant d'éloigné change à celui du S1.
- Le nouveau xRefCi de l'éloigné S3 apparie le xRefCi de l'éloigné S1.
- L'agent B (extn 3000) raccroche.
- Le C (extn 2000) et le B (extn 3000) sont déconnectés.
- Session S3 FINIE

Note: Un transfert d'extrémité a comme conséquence une mise à jour de la session existante. Le téléphone bifurquant demeure le seul participant dans la piste 0, mais le participant aux modifications de la piste 1 au nouvel interlocuteur.

En cas d'Unified Communications Manager 10.0 et plus tard, voici le résultat :

1. Le C d'appelant (extn 2000) appelle l'agent A (extn 1000).

Le C (extn 2000) parle à l'agent A (extn 1000). Session S1 - COMMENCÉ - La piste 0 est A (extn 1000), la piste 1 est le C (2000)

2. L'agent A (extn 1000) consulte l'agent B (extn 3000).

3. Le C (extn 2000) entend le MoH. A (extn 1000) parle à B (extn 3000).

- Session S1 - FINI
- Session S2 - COMMENCÉ - La piste 0 est A (extn 1000), la piste 1 est B (extn 3000)
- Session S3 - COMMENCÉ - La piste 0 est B (extn 3000), la piste 1 est A (extn 1000)

Considérations :

- La session S1 finit parce que le téléphone de l'agent A a placé le C d'appelant sur l'attente.
- Les sessions S2 et S3 chacun des deux existent parce que les deux téléphones sont configurés pour bifurquer.
- Les participants aussi bien que le xRefCi pour les deux participants à S2 et à S3 sont identiques mais dans des positions renversées entre eux.
- Des valeurs du xRefCi S1 ne sont pas reflétées dans S2 ou S3, puisque la consultation est

considérée un appel indépendant

4. L'agent A (extn 1000) se termine le transfert.

5. Le C (extn 2000) parle à B (extn 3000).

6. A (extn 1000) a déconnecté.

- Session S3 - FINI
- Session S2 - FINI
- Session S4 - COMMENCÉ - La piste 0 est B (extn 3000), la piste 1 est C (extn 2000)

Considérations :

- Un transfert d'extrémité déclenche l'extrémité d'une session enregistrée et le début d'une autre session enregistrée.
- Bien que les nouvelles sessions starts, ses valeurs de xRefCi appartiennent aux sessions précédentes.
- Le xRefCi de l'éloigné S4 apparie le xRefCi de l'éloigné S1, et le xRefCi de l'extrémité proche S4 apparie le xRefCi de l'extrémité proche S3.

7. L'agent B (extn 3000) raccroche.

8. Le C (extn 2000) et le B (extn 3000) ont déconnecté.

- Session S4 - FINI

Note: Un transfert d'extrémité a comme conséquence l'extrémité d'une session enregistrée et le début d'une autre session enregistrée.

1.5. Conférences d'agent avec un autre scénario d'agent

1. Le C d'appelant (extn 2000) appelle l'agent A (extn 1000).

2. L'agent A (extn 1000) consulte l'agent B (extn 3000).

3. L'agent A (extn 1000) se termine la conférence.

4. L'agent A (extn 1000) relâche de la conférence.

5. L'agent B (extn 3000) raccroche.

En cas d'Unified Communications Manager 9.x et plus tôt, voici le résultat :

1. Le C d'appelant (extn 2000) appelle l'agent A (extn 1000).

2. Le C (extn 2000) parle à A (extn 1000).

Session S1 COMMENCÉE - La piste 0 est A (extn 1000), la piste 1 est C (extn 2000).

3. L'agent A (extn 1000) consulte l'agent B (extn 3000).

4. Le C (extn 2000) entend des entretiens MoH A (extn 1000) à B (extn 3000)

- Session S1 FINIE
- Session S2 COMMENCÉE - La piste 0 est A (extn 1000), la piste 1 est B (extn 3000)
- Session S3 COMMENCÉE - La piste 0 est B (extn 3000), la piste 1 est A (extn 1052)

Considérations :

- La session S1 finit parce que le téléphone de l'agent A a placé le C d'appelant sur l'attente.
- Les sessions S2 et S3 existent parce que les deux téléphones sont configurés pour bifurquer.
- Les participants aussi bien que le xRefCi pour les deux participants à S2 et à S3 sont identiques mais dans des positions renversées entre eux.
- Des valeurs du xRefCi S1 ne sont pas reflétées dans S2 ou S3, puisque la consultation est considérée un appel indépendant.

5. L'agent A (extn 1000) se termine la conférence.

6. C (extn 2000) parlant à A (extn 1000) et à B (extn 3000).

- Session S2 FINIE
- Session S3 MISE À JOUR - La piste 0 est B (ext. 3000), la piste 1 est passerelle de conférence
- Session S4 COMMENCÉE - La piste 0 est A (extn 1000), la piste 1 est passerelle de conférence

Considérations :

Un transfert d'extrémité déclenche une MISE À JOUR de la session enregistrée existante.

La fin d'une conférence est mise en application :

Pendant la consultation :

- Le téléphone de consultation fait inviter un primaire l'attente et un active consultant l'appel.
- Le téléphone consultant a seulement un appel actif (l'appel de consultation).

Quand la conférence est terminée (tous les interlocuteurs connectés) :

- Le téléphone de consultation consultant l'appel se termine.
- L'appel téléphonique primaire de consultation obtient un transfert d'extrémité à la passerelle de conférence.
- Le téléphone consultant obtient un transfert d'extrémité à la passerelle de conférence.

En conséquence :

- S2 se termine, parce qu'il représente le téléphone de consultation consultant l'appel, qui se termine également.
- Débuts S4 ; il représente le transfert de suite et d'extrémité de l'appel primaire d'A, mais l'original S1 ne peut pas être MIS À JOUR parce qu'était précédemment en raison terminé pour se tenir plus tôt.
- S3 obtient MIS À JOUR parce que l'extrémité du b simplement est transférée d'A vers la passerelle de conférence.
- La valeur de xRefCi de l'extrémité proche S4 appariera la valeur de xRefCi de l'extrémité proche S1.

7. L'agent A (extn 1000) relâche de la conférence.

8. A (extn 1000) a déconnecté. C (extn 2000) parlant à la session S3 B (extn 3000) MISE À JOUR - la piste 0 est B (extn 3000), la piste 1 est C (extn 2000).

Considérations :

- La désescalade d'une conférence dans un appel bipartite normal est mise en application en tant qu'extrémité de les deux téléphones restants transférant entre eux
- Un transfert d'extrémité déclenche une MISE À JOUR de la session enregistrée existante.

- S3 et S1 auront des valeurs assorties de xRefCi d'extrémité proche. Notez que seulement une session demeure active parce que le C d'appelant n'a pas bifurquer activé.

9. L'agent B (extn 3000) raccroche.

10. Le C (extn 2000) et le B (extn 3000) ont déconnecté.
Session S4 FINIE.

Considérations :

- Un transfert d'extrémité a comme conséquence une mise à jour de la session existante. Le téléphone bifurquant demeure le seul participant dans la piste 0, mais le participant aux modifications de la piste 1 au nouvel interlocuteur.
- Une conférence est créée avec le transfert de tous les téléphones à la passerelle de conférence. Par conséquent, une conférence agit juste comme un ensemble de transferts. Des sessions existantes sont mises à jour dans ces sessions que le téléphone bifurquant demeure le seul participant dans la piste 0, mais dans le participant aux modifications de la piste 1 à la passerelle de conférence.
- Une fois le tiers chute de la conférence, les interlocuteurs sont transférés entre eux. Ceci met à jour les sessions existantes de nouveau, le téléphone bifurquant demeure le seul participant dans la piste 0, mais le participant aux modifications de la piste 1 à l'autre interlocuteur.
- Si un quatrième interlocuteur est ajouté à la passerelle de conférence, il n'y a aucune indication dans les métadonnées, à moins que le quatrième interlocuteur fasse également activer son propre bifurquer.

En cas d'Unified Communications Manager 10.x et plus tard, voici le résultat :

1. Le C d'appelant (extn 2000) appelle l'agent A (extn 1000).
2. Le C (extn 2000) parlant à la session S1 A (extn 1000) - COMMENCÉ - la piste 0 est A (extn 1000), la piste 1 est C (extn 2000).
3. L'agent A (extn 1000) consulte l'agent B (extn 3000).
4. Entendre MoH A (extn 1000) de C (extn 2000) parlant à B (extn 3000).
 - Session S1 - FINI
 - Session S2 - COMMENCÉ - La piste 0 est A (extn 1000), la piste 1 est B (extn 3000)
 - Session S3 - COMMENCÉ - La piste 0 est B (extn 3000), la piste 1 est A (extn 1000)

Considérations :

- La session S1 finit parce que le téléphone de l'agent A a placé le C d'appelant sur l'attente.
 - Les sessions S2 et S3 existent parce que les deux téléphones sont configurés pour bifurquer.
 - Les participants aussi bien que le xRefCi pour les deux participants à S2 et à S3 sont identiques mais dans des positions renversées entre eux.
 - Des valeurs du xRefCi S1 ne sont pas reflétées dans S2 ou S3, puisque la consultation est considérée un appel indépendant
5. L'agent A (extn 1000) se termine la conférence.
 6. C (extn 2000) parlant à A (extn 1000) et à B (extn 3000)

- Session S2 - FINI
- Session S3 - FINI
- Session S4 - COMMENCÉ - La piste 0 est A (extn 1000), la piste 1 est passerelle de conférence
- Session S5 - COMMENCÉ - La piste 0 est B (extn 3000), la piste 1 est passerelle de conférence

Considérations :

Un transfert d'extrémité déclenche l'extrémité d'une session enregistrée et le début d'un autre enregistrement. La fin d'une conférence est mise en application l'a répertorié ici :

- Pendant la consultation : Le téléphone de consultation fait inviter un primaire l'attente et un active consultant l'appelLe téléphone consultant a seulement un appel actif (l'appel de consultation)
- Quand la conférence est terminée (tous les interlocuteurs connectés) :
 - Le téléphone de consultation consultant l'appel se termine
 - L'appel téléphonique primaire de consultation obtient un transfert d'extrémité à la passerelle de conférence
 - Le téléphone consultant obtient un transfert d'extrémité à la passerelle de conférence
- En conséquence : Deux nouvelles sessions sont créées parce que l'agent A et l'agent B ont bifurquer activéLa valeur de xRefCi de l'extrémité proche S4 et le xRefCi de l'extrémité proche S1 évaluent des correspondancesLa valeur de xRefCi de l'extrémité proche S5 et le xRefCi de l'extrémité proche S3 évaluent des correspondancesLes valeurs de xRefCi d'extrémité pour S4 et S5 ne s'assortit pas, quoique chacun des deux soient connectés à la même passerelle de conférence

7. L'agent A (extn 1000) relâche de la conférence

8. A (extn 1000) a déconnecté. C (extn 2000) parlant à B (extn 3000)

- Session S4 - FINI
- Session S5 - FINI
- Session S6 - COMMENCÉ - La piste 0 est B (extn 3000), la piste 1 est C (extn 2000)

Considérations :

- La désescalade d'une conférence dans un appel bipartite normal est mise en application en tant qu'extrémité de les deux téléphones restants transférant entre eux
- Un transfert d'extrémité déclenche l'extrémité d'une session enregistrée et le début d'une autre session enregistrée
- S6 et S5 auront des valeurs assorties de xRefCi d'extrémité proche. Notez que seulement une session demeure active parce que le C d'appelant n'a pas bifurquer activé

9. L'agent B (extn 3000) raccroche

10. Le C (extn 2000) et le B (extn 3000) ont déconnecté

- Session S6 - FINI

Considérations :

- Un transfert d'extrémité a comme conséquence la fin d'une session et le début des autres
- Une conférence est créée avec le transfert de tous les téléphones à la passerelle de conférence. Par conséquent, une conférence agit juste comme un ensemble de transferts. Des sessions existantes sont finies et de nouvelles sessions sont créées entre les téléphones bifurquants et la passerelle de conférence
- Une fois le tiers chute de la conférence, les interlocuteurs sont transférés entre eux. Ceci finit les sessions qui ont inclus la passerelle de conférence et commencent de nouvelles sessions entre les deux points finaux restants
- Si un quatrième interlocuteur est ajouté à la passerelle de conférence, il n'y a aucune indication dans les métadonnées, à moins que le quatrième interlocuteur fasse également activer son propre bifurquer

2. Comment corrélons-nous des id d'appel de référence pour différentes sessions d'appel, sous bifurquer d'Unified Border Element ?

Avec Unified Border Element bifurquant, très peu de situations causent un appel d'être coupé en plusieurs sessions enregistrées. L'Attente/Reprise, le transfert et les exécutions de conférence ne commencent pas de nouvelles sessions enregistrées dans la plupart des cas. Dans peu des cas où de nouvelles sessions sont créées, il y ont une valeur commune, CCID (ID de corrélation d'appel). Cette valeur est commune à toutes les sessions dans l'appel. CCID est la forme décimale de Cisco-GUID, une seule clé d'appel qui est générée par des Routeurs de Voix de Cisco. Le premier routeur qui reçoit un appel génère cette clé, et la passe en bas de la ligne à tous les périphériques ultérieurs comprenant le Cisco MediaSense.

Unified Border Element lui-même ne génère pas des valeurs de xRefCi, mais pour créer la similitude avec des appels téléphoniques bifurquants d'Unified Communications Manager, le Cisco MediaSense synthétise également une paire de valeurs de xRefCi pour chaque appel d'Unified Border Element. Ceux-ci peuvent être vus dans les métadonnées au niveau de piste, avec CCID, qui apparaît au niveau de session.

Ces situations causent des enregistrements d'Unified Border Element d'être coupés en plusieurs sessions :

2.1. modification de codecs de Mid-appel

Si un transfert, la conférence, la baisse de conférence, ou toute autre exécution fait renégocier les interlocuteurs leurs codecs, le Cisco MediaSense finit la session enregistrée en cours et commence un neuf. Les deux sessions partagent le même CCID et les mêmes paires de valeurs de xRefCi.

2.2. Consultez les transferts

Un transfert de consultation est un transfert à partir d'un agent à l'autre, dans lequel les deux agents parlent entre eux tandis que l'appelant d'origine attend sur l'attente. Le tronçon de consultation de l'appel est lié d'une certaine façon à l'appel global, et il est possible de configurer l'Unified Communications Manager tels que consultez les appels traversent le CUBE. Cependant, Unified Border Element et le Cisco MediaSense ne savent pas que ces appels sont connexes, et ils créent un nouveau CCID et une nouvelle paire de valeurs de xRefCi pour cette session.

Ces appels peuvent être associés entre eux avec comparant des champs de deviceRef et d'horodateur de participant. Considérez ce scénario :

1. Le C d'appelant (extn 2000) appelle l'agent A (extn 1000) (sessionId = S1, CCID = C1)
2. L'agent A consulte avec l'agent B (extn 3000) (sessionId = S2, CCID = C2)
3. Les baisses de l'agent A, et le C d'appelant parle avec l'agent B (sessionId = S1, CCID = C1)

L'alerte dans ce scénario est dans l'étape 2. Au cours de cette période, l'agent A (deviceRef 1000) est un participant à deux sessions enregistrées immédiatement :

- Session = S1/CCID = C1 et
- Session = S2/CCID = C2

Par conséquent, le S1 est lié à S2 et à C1 est lié au C2.

2.3. Consultez la détection d'appels

D'abord, nous avons besoin d'une définition précise de consultons l'appel :

Tout appel secondaire qui est fait par un participant en cours à une session existante à un point final qui est l'extérieur qui session et qui exclut les autres participants à cette session.

Dans la théorie, ce scénario pourrait inclure un agent place l'appelant sur l'attente pour vérifier avec son patron pour la rupture de luch, ou même un agent a mis l'appelant sur l'attente pour recevoir un appel de son épouse, mais nous ignorons ces possibilités pour l'instant.

Il est possible pour application cliente de détecter un appel de consultation en temps réel par la piste du flot d'événement de Cisco MediaSense. Si le client observe un événement COMMENCÉ par session contient un deviceRef donné, avec un autre événement COMMENCÉ par session contient le même deviceRef sans l'événement FINI par session intervenant, il peut conclure que les sessionIds et le CCIDs trouvés dans les deux événements COMMENCÉS par session sont associés.

Historiquement, un client en peut vérifier consultent les appels qui sont associés avec un appel primaire donné, avec le Cisco MediaSense API. Supposez que le client connaît cet extn utilisé par A 1000 d'agent, dans CCID <C1>. Ce les instructions de trouver tout associé consultent des appels :

Step1. Récupérez les métadonnées de session pour l'appel primaire en émettant getSessionByCCID(<C1>).

Step2. Extrayez le sessionStartDate (appel il <Ta>), et le sessionDuration.

Step3. Calculez le sessionEndDate (appel il <Tb>) en ajoutant le sessionDuration au <Ta>.

Step4. Exécutez cette demande API :

IP

address:8443/ora/queryService/query/getSessionsByDeviceRef?value=1000&minSessionStartDate=<Ta>&maxSessionStartDate=<Tb> de https://Mediasense

Cette requête peut renvoyer plus d'une session. S'il fait, alors on peut assumer que tous sont

associés avec le même appel.

2.4. Consultez la détection d'appels de plusieurs participants

Toute la procédure mentionnée dedans consultent la section de détection d'appel, trouve consultent des appels faits à partir du périphérique qui a reçu l'appel téléphonique initial. Cependant, ce qui si là consultent des appels sont faits à partir d'un périphérique vers lequel l'appel a été ultérieurement transféré ?

Considérez cette procédure :

1. L'appelant appelle l'agent 1
2. L'agent 1 consulte avec l'agent 2, puis des baisses
3. L'appelant parle avec l'agent 2
4. L'agent 2 consulte avec l'agent 3, puis des baisses
5. L'appelant parle avec l'agent 3

Cette procédure n'attrape pas l'appel de consultation entre l'agent 2 et l'agent 3.

Puisque c'est un appel d'Unified Border Element, nous pouvons nous servir du fait que toutes les connexions entre l'appelant et chacun des agents sont incluses en même session enregistrée, et du fait que tous les agents impliqués sont répertoriés comme participants à la même session à un moment ou un autre. Ainsi, des métadonnées primaires de session, nous pouvons collecter une liste de tous les deviceRefs qui étaient impliqués. Pour trouver ces sessions, nous pouvons faire une gamme des appels au `getSessionsByDeviceRef`, spécifie la plage de temps de la session primaire, ainsi qu'un deviceRef par demande.

Alternativement, le processus peut être simplifié avec une demande simple de `getSessions` de ce type :

```
{
  "requestParameters": [
    {
      "fieldName": "deviceRef",
      "fieldConditions": [
        {
          "fieldOperator": "equals",
          "fieldValues": [
            "1000"
          ],
          "fieldConnector": "OR"
        },
        {
          "fieldOperator": "equals",
          "fieldValues": [
            "2000"
          ],
          "fieldConnector": "OR"
        },
        {
          "fieldOperator": "equals",
          "fieldValues": [
            "3000"
          ],
          "fieldConnector": "OR"
        }
      ]
    }
  ]
}
```

```

        {
            "fieldOperator": "equals",
            "fieldValues": [
                "4000"
            ]
        }
    ],
    "paramConnector": "AND"
},
{
    "fieldName": "sessionStartDate",
    "fieldConditions": [
        {
            "fieldOperator": "between",
            "fieldValues": [
                <Ta>, // session start time
                <Tb> // session end time
            ]
        }
    ]
}
]
}
]
}
}

```

Cette requête renvoie tous les appels de consultation associés avec l'appel primaire d'origine et tous ses transferts.

Cette procédure moule réellement le net trop largement. Si par exemple, l'agent au deviceRef 4000 conduisait et terminait un appel complètement indépendant qui s'est avéré justement commencer *après que* <Ta> et *avant qu'il* a été ajouté à l'appel en question, cette procédure inclut qu'indépendant appelez dans le positionnement. Ce problème peut être résolu cependant, avec les informations disponibles dans les métadonnées de la session primaire. Les informations de chaque participant incluent le décalage de temps auquel il a joint la session et la durée de sa tenue. Code client pourrait l'utiliser que les informations pour supprimer simplement les sessions indépendantes de la liste qu'elle a reçue en haut. Ou, il pourrait formuler une gamme de getSession ou de requêtes directs de getSessionByDeviceRef qui encadrent correctement les délais prévus lesoù chaque agent était dans l'appel primaire. Nous partons que comme exercice au lecteur.

2.5. Résumé

Dans la section précédente, nous avons présenté des precedures pour la récupération de toutes les sessions associées avec une session enregistrée donnée de Cisco MediaSense. Cependant, nous avons également vu qu'un appel donné peut être divisé en plus d'une session, comme dans le cas d'un codec de mid-appel changeons.

Comment récupérons-nous tous les enregistrements (aussi bien que consulte) associés avec *toutes les* sessions connectées à l'interaction de l'appelant ?

La réponse est d'étendre le ce des instructions pour le detectiion du multiple consultant des appels. D'abord, nous collectons toutes les sessions qui partagent le CCID de la session primaire en question. Puis, nous établissons notre liste des participants de *tous* ces enregistrements de session. Ensuite, nous calculons les plages de temps comme sessionStartDate de la session la plus tôt par la fin de la dernière session. pour finir, nous pouvons exécuter la requête de getSession affichée.

Comme avant, nous pouvons finir par avec la capture de trop d'enregistrements, ainsi nous

pourrions exécuter une étape de post-traitement pour supprimer ces sessions indépendantes de sa liste.

3. Comment associez-vous des appels dans le Cisco MediaSense avec leurs apparences dans d'autres composants de la solution ?

3.1. Tableau de corrélation d'identifiant

Ces deux tables — une pour des appels d'Unified Communications Manager et une pour des appels d'Unified Border Element. Chaque colonne représente un composant de la solution ou le protocole, avec la première colonne représente le Cisco MediaSense. Chaque ligne représente un type particulier d'identifiant.

Pour lire les tables, commencez par une cellule qui représente l'élément de donnée que vous connaissez, et puis regarde horizontalement à la colonne représente le composant de la solution en lequel vous voulez trouver l'appel. L'entrée en cette cellule indique par quel nom le précis le même élément de donnée est connu dans le composant de cible. Si le composant de cible a une cellule vide dans cette ligne, alors cet élément de donnée n'est pas connu à ce composant. Vous pouvez à la place rechercher une colonne intervenante où vous pouvez croiser verticalement dans une autre ligne où cette cellule n'est pas vide dans la colonne du composant de cible.

Par exemple, avec un appel d'Unified Communications Manager, supposez que vous connaissez le GED-188 CallReferenceID et que vous voulez trouver l'appel dans le Cisco MediaSense. Vérifiez gauche de la colonne GED-188, vous voient qu'il n'y a aucune valeur dans la colonne de MediaSense, ainsi vous ne peut pas tracer directement à elle.

Cependant, il y a une colonne où vous pouvez zigzaguer à travers des lignes : Unified Communications Manager CDR. Un client peut sélectionner l'enregistrement approprié de l'Unified Communications Manager CDR en recherchant un dans lequel l'IncomingProtocolCallRef apparie le GED-188 CallReferenceID. Que l'enregistrement contient une valeur ont appelé destLegCallIdentifier, qui est identique que le xRefCi de MediaSense NearEnd, et peuvent donc être utilisés pour trouver l'enregistrement correspondant dans le Cisco MediaSense.

Des enregistrements de l'Unified Communications Manager CDR ne sont pas écrits jusqu'à une certaine heure après que la fin de l'appel se termine, cependant, ainsi cette méthode peut seulement être utilisée historiquement.

Il y a un autre chemin aussi bien. Regardez vers le bas du GED-188 CallReferenceID. Il s'avère que vous pouvez également employer l'AlertingDevice et l'AnsweringDevice pour apparier le champ de deviceRef dans MediaSense. Cette méthode fonctionne également en temps réel.

Call Correlation for Calls Forked by a Unified CM IP Phone

MediaSense	Ingress Gateway or CUBE	AAA RADIUS CDR	UCM CDR	TAPI/JTAPI field	UCCE Database	UCCE Script	CTIOS	GED-188
(1)	Cisco-GUID		IncomingProtocolCallRef	CiscoConnection.UniqueID	TCD.CallGUID	Call.CallGUID		CallReferenceID
NearEnd xRefCi			destLegCallIdentifier	Terminal.ConnectionID				
FarEnd xRefCi			origLegCallIdentifier	Terminal.ConnectionID				
			global_CallID_call-ManagerId + global_CallID_callId (A.K.A. UCM GCID)	CiscoCall.CallID	TCD.PeripheralCallKey			
deviceId							Agent.AgentInstrument	
deviceRef					TCD.InstrumentPortNumber (2)		Agent.AgentExtension or Agent.Extension	AlertingDevice or AnsweringDevice

Call Correlation for Calls Forked by CUBE

MediaSense	Ingress Gateway or CUBE	AAA RADIUS CDR	UCM CDR	TAPI/JTAPI field	UCCE Database	UCCE Script	CTIOS	GED-188
CCID (3)	Cisco-GUID	Cisco-GUID	IncomingProtocolCallRef		TCD.CallGUID	Call.CallGUID		CallReferenceID
deviceRef	Called or calling party extn				TCD.InstrumentPortNumber (2)		Agent.AgentExtension or Agent.Extension	AlertingDevice or AnsweringDevice

Considérations :

1. Dans les enregistrements des appels d'Unified Communications Manager, le Cisco MediaSense reçoit en fait Cisco-GUID d'UCM, mais il n'est pas le même qui est capturé par les autres périphériques de solution. MediaSense donc n'enregistre pas même cette valeur.
2. Pour des appels d'agent à agent, TCD.InstrumentPortNumber est l'extension de l'agent de destination. L'extension de l'agent appelant peut être trouvée dans TCD.ANI.
3. CCID est Cisco-GUID en forme décimale, qui est 4 ensembles trait d'union-séparés de nombres décimaux 10-digit. Ceux-ci peuvent être convertis en forme hexadécimale avec simplement la conversion de chaque nombre décimal 10-digit à un nombre hexadécimal à 8 chiffres, et retirent les traits d'union. Là où Cisco-GUID est utilisé dans UCCE, il est sous sa forme hexadécimale.

4. Comment déterminez-vous quelle piste a l'appelant et quelle piste a l'appelé ?

4.1. Pour des appels a bifurqué par le CUBE

Pour le CUBE appelle, dépiste 0 trace toujours au flux multimédia de tronçon d'ancre. Le tronçon d'ancre est le cadran-pair que le profil d'enregistrement de medias est configuré. Le deuxième tronçon de non-ancre de cartes de piste.

Si vous avez le profil d'enregistrement de medias activé sur le dialpeer *d'arrivée*, alors le tronçon d'ancre devient le dans-tronçon. En d'autres termes, l'appelant apparaît dans la piste 0, et l'appelé apparaît dans la piste 1.

Si vous avez le profil d'enregistrement de medias activé sur le dialpeer *sortant*, alors le tronçon d'ancre devient le -tronçon. Dans cela le cas l'appelant apparaît dans la piste 1 et l'appelé apparaît dans la piste 0.

4.2. Pour des appels a bifurqué par des téléphones d'Unified CM

Pour l'Unified CM bifurquant, dans les scénarios simples d'appel vous pouvez employer les champs de xRefCi dans les métadonnées pour déterminer quel interlocuteur est dans quels medias dépistez. Le xRefCi numériquement plus petit se rapporte habituellement à la piste de l'appelant. La piste de l'appelé est numériquement plus grande (habituellement par une, mais eux pourraient être plus sous un système raisonnablement chargé). Cependant, ces valeurs de xRefCi s'enveloppent par la suite autour à zéro. Ainsi, si vous constatez qu'une valeur est un nombre élevé et l'autre est un petit nombre, vous supposez que leurs positions sont renversées.

Dans des scénarios plus compliqués, cet algorithme ne fonctionne pas toujours. Si des services supplémentaires sont appelés, comme des transferts et des conférences, et la batterie de gestionnaire UC se compose de plus d'un noeud, alors les valeurs de xRefCi ne sont pas nécessairement générées séquentiellement, et vous ne pouvez pas supposer que leur commande a n'importe quelle signification du tout. Un moyen direct de déterminer si l'ordre de commande d'une paire particulière de valeurs de xRefCi mettent en boîte sont de confiance est de regarder le premier octet des valeurs de xRefCi. Cet octet représente l'ID de noeud de gestionnaire UC sur lequel cet identifiant particulier a été créé. Si les premiers octets des deux valeurs de xRefCi sont identiques, alors leur commande est correcte. S'ils sont différents, alors la commande ne pourrait pas être correcte.

Pour ces cas, la seule manière de déterminer la direction d'appel en temps réel est de saisir les informations de n'importe quelle autre source, telle que le flux d'événement JTAPI. Une fois l'appel a fini et quelques minutes se sont écoulées, vous pouvez toujours déterminer les données CDR du gestionnaire UC de direction et de contrôle d'appel pour l'appel. Spécifiquement, le champ origLegCallIdentifier dans l'enregistrement CDR représente toujours l'appelant.

5. Quelles sont les causes possibles pour un état de session de CLOSED_ERROR ?

Les causes possibles pour un état de session de CLOSED_ERROR incluent :

1. Le serveur de Contrôle d'appel a reçu une réponse d'erreur du serveur de medias (enregistrement) pour la demande ouverte ou étroite.
2. Le serveur de Contrôle d'appel a détecté une erreur de signalisation de SIP, par exemple des disparus ACK.
3. La session a été avec succès clôturée, mais TOUTES LES pistes ont la taille zéro.

Quand une session est dans l'état active, il est normal qu'il n'y ait aucune durée dans les métadonnées, parce que la durée n'est pas connue jusqu'à ce que la session soit fermée.

Pour une session qui est dans l'état CLOSED_ERROR, si les champs de session ou de durée de piste ne sont pas présents dans l'événement ou les données de getSessions, puis le support pour cette piste n'est pas disponible.

6. Quelle est la différence entre les sessions taillées et supprimées ?

Considérez ces deux requêtes :

6.1. Avec la requête de getAllPrunedSessions

Cette requête renvoie un ensemble de sessions, tous ce que de la session des états SONT SUPPRIMÉS :

```
https://Mediaserver IP  
address:8443/ora/queryService/query/getAllPrunedSessions?minSessionStartDate=1301788800000&maxSe  
ssionStartDate=1312329599000
```

6.2. Avec la requête de getSessions

Cette requête ne renvoie aucune session :

```
https://MediaServer IP address:8443/ora/queryService/query/getSessions  
{  
  "requestParameters":  
  [{  
    "fieldName" : "sessionState",  
    "fieldConditions":  
    [{  
      "fieldOperator" : "equals",  
      "fieldValues" : [ "DELETED" ]  
    }],  
    "paramConnector" : "AND"  
  },  
  {  
    "fieldName" : "sessionStartDate",  
    "fieldConditions":  
    [{  
      "fieldOperator" : "between",  
      "fieldValues" : [ "1301788800000", "1312329599000" ]  
    }]  
  }]  
}
```

6.3. Pourquoi la différence de comportement en sessions taillées et supprimées ?

La différence de comportement est par conception. Référez-vous à ceci des sections dans la documentation de MediaSense :

- La description du paramètre API : La description des getAllPrunedSessions API :

Employez cet API pour rechercher tous les enregistrements taillés... que le terme taillé se rapporte aux enregistrements qui sont supprimés par le système de Cisco MediaSense. Si vous avez explicitement supprimé n'importe quel enregistrement utilisant les deleteSessions API, alors ces enregistrements supprimés ne sont pas considérés en tant qu'enregistrements taillés.

- Le MediaSense SRND sous la gestion du stockage proactive de section :

Quand des sessions sont taillées, les métadonnées qui est associé avec ces sessions demeure dans la base de données, même après que ces sessions sont marquées en tant que « taillé ». Ce les métadonnées ne prend pas un grand nombre d'espace de stockage comparé aux enregistrements eux-mêmes mais il prend un certain espace et devrait être périodiquement retiré. Pour faciliter cette activité, les clients peuvent périodiquement émettre une demande API des sessions taillées, ou les clients peuvent choisir de recevoir des événements taillés par session et de supprimer explicitement ces événements dont les clients n'ont besoin plus.

Pour clarifier, les deux requêtes sont entièrement différentes. En fait, la deuxième requête (le lwhich recherche toutes les sessions dont l'état EST SUPPRIMÉ) renvoie toujours un positionnement vide. Les requêtes de jour en jour normales filtrent des sessions avec les états SUPPRIMÉS, même si c'est ce qui est demandé. La seule exception est des **getAllPrunedSessions**. Cette exception est destinée pour aider l'application pour trouver des sessions taillées de sorte que l'application puisse demander que ces sessions soient supprimées.

Une fois que vous utilisez les **deleteSessions** API sur la liste de sessions taillées que vous obtenez des **getAllPrunedSessions**, ces sessions n'apparaît plus dans le résultat des **getAllPrunedSessions**. De telles sessions est complètement retirées des métadonnées immédiatement.

Une autre manière de regarder ceci est que les sessions taillées ne sont pas la même chose que des sessions supprimées :

1. Des sessions taillées ont été marquées pour la suppression par un algorithme dans le système de MediaSense. Aucune personne n'a été impliquée dans la décision de tailler ces sessions. Ainsi quoique ces sessions soient déplacées à l'état SUPPRIMÉ, ces sessions ne sont pas retirées réellement des métadonnées. L'intervention humaine (ou application) est exigée. Puisque ces sessions sont dans l'état SUPPRIMÉ, ces sessions ne sont pas visibles à la plupart des requêtes. Cependant, ces sessions sont visibles à la requête API de **getAllPrunedSessions**. En outre, si des fichiers mp4 étaient générés pour ces sessions, ces fichiers mp4 continuent à être présents sur le disque et à continuer à être disponibles pour le téléchargement jusqu'à ce que les sessions taillées SOIENT SUPPRIMÉES réellement.
2. Des sessions supprimées sont marquées en appelant explicitement les **deleteSessions** API. Ce marquage peut être fait aux sessions qui sont déjà taillées ou aux sessions qui n'ont pas été encore supprimées. Une fois qu'une session a été supprimée par les **deleteSessions** API, cette session n'est plus visible à n'importe quelle requête. Ceci inclut les **getAllPrunedSessions** API. Ces sessions supprimées sont retirées des métadonnées immédiatement de sorte que l'espace disque puisse être repris.

7. Comment configurer une passerelle TDM pour bifurquer de medias ?

Quand vous avez une passerelle PSTN par laquelle appelle les écoulements, et vous voulez enregistrer ces appels. Ces appels sont des appels de TDM-à-SIP. Cependant, bifurquer de medias est seulement disponible aux appels de Sip-à-SIP.

Ces appels peuvent être enregistrés. Ces appels mettent en boîte dirigé par le routeur une deuxième fois. Des conseils de configuration et d'autres détails peuvent être trouvés en [ce livre blanc](#).

8. Comment capturer le téléphone réel de destination en utilisant un groupe de recherche ?

Quand vous utilisez des medias bifurquant du CUBE, les métadonnées de MediaSense contient normalement l'extension de l'appelé. Cependant, si le nombre appelé est un numéro pilote de groupe de recherche de gestionnaire de transmissions, puis par défaut les métadonnées contient seulement ce numéro pilote. Il ne contient pas l'extension du téléphone qui a répondu réellement à l'appel.

Il y a une configuration de gestionnaire de transmissions qui peut changer ceci. À la page de configuration de recherche/pilote, trouvez la section autorisée des **transformations connectées d'interlocuteur**. La ligne d'affichage de configuration **DN de membre du groupe en tant qu'interlocuteur connecté** doit être activée.

Cette capacité est disponible dans le gestionnaire de transmissions 9.0(1) et plus tard.

9. Pourquoi enregistrement Fondé(e) sur le réseau d'Unified Communications Manager recommandé en tant que bifurquer préféré est-il le mécanisme ?

Avec l'enregistrement Fondé(e) sur le réseau d'Unified Communications Manager (NBR), vous pouvez utiliser une passerelle pour enregistrer des appels. NBR permet à l'Unified Communications Manager pour conduire des appels d'enregistrement, indépendamment du périphérique, de l'emplacement, ou de la zone géographique. Avec NBR, les medias d'enregistrement d'appels peuvent être originaires du téléphone IP ou d'une passerelle qui est connectée à l'Unified Communications Manager au-dessus d'un joncteur réseau de SIP. L'Unified Communications Manager sélectionne dynamiquement la bonne source de medias basée sur l'écoulement et les participants de l'appel d'appel.

NBR offre une Construire-dans-passerelle automatique de retour (bavoir) quand les Integrated Services Router (ISR) sont indisponibles car aucune configuration de enregistrement distincte n'est exigée. C'est utile dans les cas où les clients veulent inclure l'agent-agent consultant des appels dans les stratégies d'enregistrement car Unified Border Element ne peut pas enregistrer consultant des appels, ainsi le bavoir doit être activé séparément.

NBR et appels de bavoir peuvent être corrélés utilisant le xRefci, qui est fourni par l'Unified Communications Manager JTAPI. CISCO-GUID n'est pas nécessaire, qui signifie que ni le serveur CTI ni les connexions CTIOS ne sont priés. Car il y a un identifiant simple de corrélation, la corrélation à travers des composants est plus forte et peut être faite dans un indépendant uniforme de manière de l'écoulement d'appel.

Le NBR, direct-étant composé aussi bien que des appels sortants numéroteur-initiés peuvent être corrélés avec leur apparence dans d'autres composants de la solution.

Avec NBR, l'enregistrement de passerelle TDM est automatiquement utilisé sans le fractionnement de la capacité du routeur. Actuellement, l'enregistrement de passerelle TDM n'est pas pris en charge avec MediaSense 10.5.

10. Pourquoi est-ce qu'un noeud prend plus long pour améliorer

à MediaSense 10.5 ?

Un noeud peut prendre plusieurs heures pour améliorer dépend du nombre et de la taille d'enregistrements qu'elle se tient. Pour MediaSense 10.5, quand vous améliorez un noeud avec les postes de données très grands, cela prend environ 90 minutes supplémentaires par 1 million d'enregistrements.

11. Quelle est l'incidence des modifications russes de fuseau horaire sur l'application de recherche et de jeu de MediaSense ?

Les utilisateurs du MediaSense les recherchent et l'application de jeu est affectée s'ils se trouvent dans des fuseaux horaires affectés l'un des ou s'ils sélectionnent un fuseau horaire affecté dans les critères de recherche. Les Produits de partenaire de tiers qui se connectent par interface à MediaSense sont affectés pareillement jusqu'à ce qu'ils mettent à jour leurs tables respectives de fuseau horaire.

Le contournement est de sélectionner un fuseau horaire qui apparie le décalage correct du GMT même si ville n'est plus correcte.

12. Quels sont les langages pris en charge par MediaSense ?

Voici les langages pris en charge par MediaSense :

- Arabe
- Danois
- Néerlandais
- L'anglais (Etats-Unis)
- Finlandais
- Français
- Allemand
- Italien
- Japonais
- Coréen
- Norvégien
- Polonais
- Portugais (brésiliens)
- Russe
- Chinois simplifié
- Espagnol
- Suédois
- Chinois traditionnel
- Turc

13. Comment surveiller la performance du système de MediaSense ?

Pour surveiller la performance du système de MediaSense, analysez les valeurs de ces

indicateurs de performances de clé (ICP) avec l'outil principal d'assurance d'outil RTMT ou de Collaboration de Cisco.

Pour plus d'informations sur l'outil principal d'assurance d'outil RTMT ou de Collaboration de Cisco, référez-vous les sections Administration principales unifiées d'assurance de gestion RTMT et de Collaboration de Cisco du [guide utilisateur de Cisco MediaSense](#).

ICP et leurs valeurs seuil

Indicateur de performances de clé	Compteurs RTMT	Valeurs seuil suggérées
Taux de réussite des appels	Service de Contrôle d'appel de MediaSense > nombre de sessions enregistrées sans erreurs Service de Contrôle d'appel de MediaSense > nombre de sessions enregistrées avec des erreurs	99.99% Taux de réussite des appels = nombre de sessions enregistrées sans erreurs / (nombre de sessions enregistrées sans erreurs + nombre de sessions enregistrées avec des erreurs) * 100
Temps de réponse API moyenne pour l'API	Temps de réponse de requête de service > de moyen du Cisco MediaSense API	60 sec
Retard d'installation de moyen de début d'enregistrement	Retard d'installation de service > de moyen de Contrôle d'appel de Cisco MediaSense	3 sec
Utilisation de moyen CPU	Processeur > % de temps- CPU	90%
Moyenne utilisation de mémoire	Mémoire > %Mem utilisé	70%
Perte de paquets RTP/UDP	Interface réseau > Rx relâché > eth0 Interface réseau > erreurs de Rx > eth0	0 0

14. Comment configurer un navigateur pour exécuter le lecteur de dans-navigateur dans MediaSense ?

Basé sur le navigateur, exécutez ces étapes pour exécuter le lecteur de dans-navigateur :

Internet Explorer 9

1. En recherche et jeu de **MediaSense**, cliquez sur l'icône de jeu d'une session enregistrée.

Internet Explorer 11

Conditions préalables : En cas de frais installez de MediaSense 11.0, s'assurent que des Noeuds de MediaSense sont ajoutés à la batterie utilisant le nom de domaine complet respectif (FQDN).
En cas de mise à jour à MediaSense 11.0, assurez-vous que les Noeuds de MediaSense qui ont été précédemment ajoutés utilisant

Mozilla Firefox

1. Ajoutez le certificat auto-sig d'un noeud de MediaSense port 8446 de mp4url dans l'au de confiance de Mozilla Firefox

l'adresse Internet, devraient maintenant être affichés par le FQDN respectif. Vérifiez la liste de serveur de MediaSense dans la fenêtre de **configuration du serveur de MediaSense (gestion de Cisco MediaSense > système > configuration du serveur de MediaSense)**.

Exécutez les étapes suivantes :

1. Placez le *hostnameformediaurl* CLI de *positionnement* en tant que « vrai » pour faire MediaSense préparer le mp4url et le reste des mediaurls utilisant le FQDN seulement.

```
admin:set useHostNameForMediaURL
admin:set useHostNameForMediaURL true
```

2. Redémarrez le service de configuration pour lancer la propriété.

```
admin:utils service restart Cisco
MediaSense Configuration Service
```

Remarque: Si le service n'a pas redémarré correctement, exécutez la même commande de nouveau.

2. Clic **oui** pour faire confiance au certificat.

Remarque: Vérifiez que le certificat auto-signé offert est du noeud visé de MediaSense en validant le FQDN dans les détails techniques du certificat.

3. Après que les reprises de service de configuration, se déconnectent et se connectent à la **recherche de MediaSense et les lisent**.

Limite : Au cas où les Noeuds de MediaSense étaient précédemment ajoutés utilisant l'IPS, puis les Noeuds continuent à être affichés par l'IPS seulement même après une mise à jour à MediaSense 11.0. le lecteur de Dans-navigateur ne travaille pas à l'Internet Explorer 11, indépendamment de la valeur de la commande CLI de *hostnameformediaurl*. Dans ce cas, l'il est recommandé que la commande CLI de *hostnameformediaurl* ne devrait pas être placé en tant que « rectifient. »

Exécutez les étapes suivantes pour ajouter le certificat auto-signé par MediaSense à Windows a fait confiance à l'autorité.

1. **Recherche et jeu** ouverts de **MediaSense**.

Une fenêtre externe de Security

2. Pour ajouter le certificat auto-signé, cliquez sur l'icône de **téléchargement d'une** session enregistrée et sélectionnez m Cette **connexion est** fenêtre externe **non approuvée** appar

3. Cliquez sur l'icône de **jeu** correspondant à l'enregistrement sélectionné.

Le lecteur de dans-navigateur lit la session d'enregistrement sélectionné.

2. Pour ajouter le certificat auto-signé, cliquez sur l'icône de **téléchargement d'une** session enregistrée et sélectionnez m **Cette connexion est** fenêtre externe **non approuvée** appar **Remarque:** Vérifiez que le ce

Certificate apparaît.

2. Le clic **continuent**.

La fenêtre de **recherche et de jeu de MediaSense** apparaît.

3. Dans la barre d'adresses, cliquez sur l'icône d'**erreur de certificat**.

4. **Certificats de vue de clic**.

La fenêtre externe de certificat apparaît.

5. Le clic **installent le certificat**.

L'assistant d'importation de certificat apparaît.

6. Cliquez sur Next.

7. Dans la fenêtre de **mémoire de certificat**, sélectionnez l'**endroit tous les Certificats dans la** case d'option suivante de **mémoire** et le clic **parcourent**.

La boîte de dialogue **choisie de mémoire de certificat** apparaît.

8. Cochez la case **physique de mémoires d'exposition** et sélectionnez le répertoire **de confiance de Certificationhorities de racine**.

9. Cliquez sur OK et **ensuite**.

10. Cliquez sur Finish pour se terminer l'importation de certificat.

Une fenêtre externe d'alerte de sécurité semble confirmer l'installation du certificat.

11. Clic **oui**.

Le message suivant apparaît.

```
The import was successful.
```

12. Cliquez sur **OK**.

13. Cliquez sur OK sur la fenêtre externe de **certificat**.

14. La fin et ouvrent le navigateur.

15. **Recherche et jeu** ouverts de **MediaSense**.

L'issxvi de Security Certificate persiste toujours.

16. Allez aux **outils > aux options**

Internet > a avancé, décoche l'**avertissement au sujet de la Sécurité de boxviiunder de contrôle de non-concordance d'adresse de certificat**.

17. Cliquez sur Apply et **APPROUVEZ**.

18. Redémarrez le navigateur et la **recherche** ouverte de **MediaSense** et le lisez.

auto-signé offert est du noeud de MediaSense en validant le FQDN dans les détails techniques du certificat.

Assurez-vous que le serveur de MediaSense est accessible par le FQDN sur le navigateur. Sinon, naviguez vers C:\Windows\System32\drivers\etc, ouvrez les **hôtes** classent dans Notepad et ajoutent l'adresse IP du serveur de MediaSense et de son FQDN au bas du fichier. le lecteur de Dans-navigateur commence fonctionner sur l'Internet Explorer 11.

Remarque: Si un enregistrement est présent sur un noeud différent de MediaSense de la batterie, vous êtes incité à ajouter le certificat de ce noeud de MediaSense dans l'autorité de confiance.

3. Le clic **I comprennent le lien** **risques.**

4. Cliquez sur Add l'**exception**
La fenêtre externe d'**exception**
Sécurité d'ajouter apparaît.

5. Le clic **confirment l'exceptio**
Sécurité.

Le certificat auto-signé du noe
particulier de MS du port 8446
obtient ajouté à l'autorité de
confiance du navigateur.