

Comprenant et le dépannage des données vivantes rend compte d'UCCX

Contenu

[Introduction](#)

[Changements des données vivantes avec SocketIO](#)

[Aperçu technique](#)

[WebSocket contre le long mécanisme de sondage](#)

[WebSocket \(RFC 6455\)](#)

[Longue interrogation](#)

[Nombre de contrôle de clients exécutant la longue interrogation](#)

[Flux de données vivant](#)

[Salles serveur et thèmes de SocketIO](#)

[Producteur](#)

[Répartiteur](#)

[Clients](#)

[Authentification](#)

[Suivi vivant de données](#)

[Dépannage des données vivantes](#)

[Healthcheck de base](#)

[Résumé](#)

[Vérifiez la version de navigateur](#)

[Vérifiez l'utilisation de CPU et mémoire](#)

[Vérifiez la source vivante de transmission en continu est en ligne](#)

[Vérifiez le NTP.](#)

[Vérifiez les DN](#)

[Exécutez un test de diagnostic](#)

[Vérifiez l'adresse Internet](#)

[Vérifiez la configuration VM](#)

[Vérifiez les Certificats UCCX](#)

[Vérifiez la configuration de classement par taille prise en charge](#)

[Mises en garde connues](#)

[Suivi vivant de log de données](#)

[Logs d'engine](#)

[Logs de SocketIO \(1\)](#)

[Logs de SocketIO \(2\)](#)

[Client Logs\(1\)](#)

[Logs de SocketIO](#)

[Logs de client \(2\)](#)

Introduction

Ce document décrit l'enregistrement vivant de données dans la solution d'Unified Contact Center Express (UCCX) à partir de la version 11.x et les contours dépannant des points de reprise avec prendre en charge l'analyse de log. L'article trace les grandes lignes également de diverses mises en garde connues avec leurs contournements.

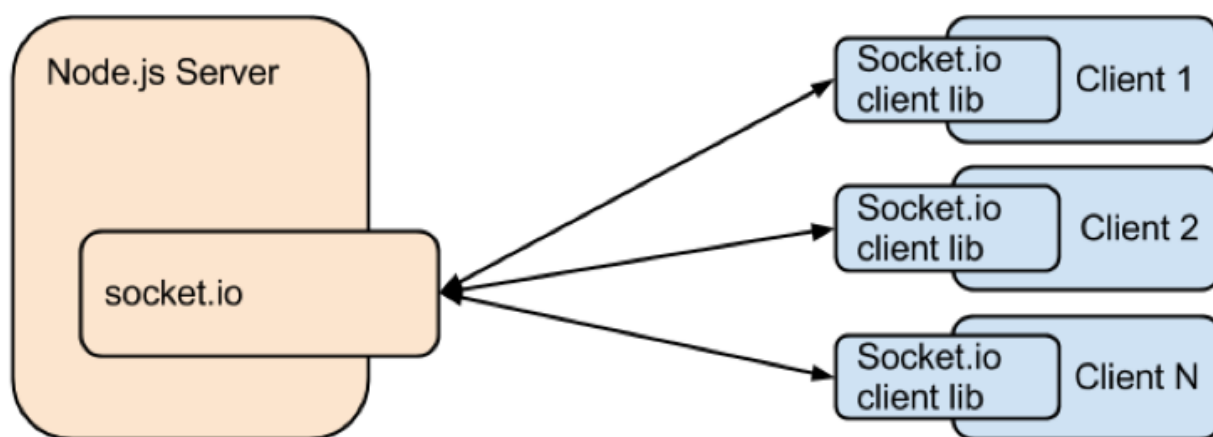
Changements des données vivantes avec SocketIO

SocketIO a basé la conception de données vivante remplace Openfire serveur XMPP comme infrastructure par défaut de message d'enregistrement d'**UCCX 11.x** en avant C'est une solution extensible et efficace entraînée par les événements qui est basée sur l'implémentation Node.js, avec un temps système réduit de transmission à l'aide du Request For Comments 6455 (RFC 6455) de protocole de WebSocket pour envoyer les messages en temps réel au navigateur (clients).

Plus tôt dans UCCX 10.x, il y avait un **service** simple de **notification de Cisco CCX**, basé sur l'implémentation OpenFire XMPP) qui a été utilisée pour des les deux finesse, conversation, notifications électroniques aussi bien qu'états vivants de données sur des instruments d'agent de finesse et CUIC.

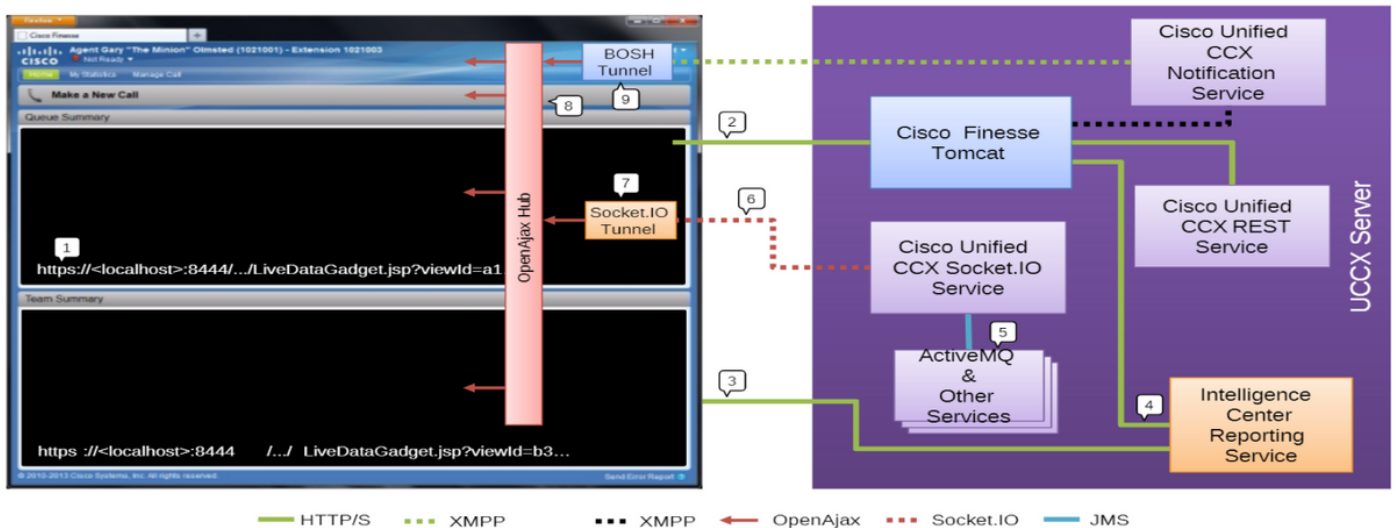
D'UCCX 11.x, avec l'introduction de SocketIO pour des états Live - la conversation et l'email de client de finesse continuent à utiliser le **service de notification de Cisco CCX**, c.-à-d. Openfire en tant que serveur XMPP, mais le module vivant d'enregistrement de données utilise le nouveau **service du service de CiscoUnified CCX Socket.IO**.

Ceci réduit l'enregistrement vivant de données au-dessus, et rend SocketIO seulement responsable de dessiner et d'acheminer les mises à jour vivantes de données au navigateur de client, suivant les indications de l'image du serveur de SocketIO.



Aperçu technique

Cette image est de SocketIO a basé l'aperçu architectural de données vivantes.



D'un point de vue de haut niveau, les données pour des états de données Live sont introduites la notation d'objet de Javascript (JSON) du module en temps réel du gestionnaire des données de l'engine UCCX (RTDM), par l'intermédiaire du bus du service de messagerie de Javas (JMS) (utilisation Apache ActiveMQ d'implémenter ceci, c.-à-d. file d'attente active de Message), au serveur Socket.IO hébergé sur UCCX (exposé par l'intermédiaire du service d'Unified CCX Socket.IO) qui récupère alors ces informations, les transforme (consomme) en salles et puis distribuent (des répartitions) ceci aux clients utilisant des connexions HTTPS WebSocket au-dessus de TCP.

ActiveMQ est un middleware qui implémente l'application JMS programmant Interface(API) et il est utilisé pour envoyer message entre deux composants, UCCX vivent serveur de données (engine UCCX) et de SocketIO.

WebSocket contre le long mécanisme de sondage

Il y a deux manières que les navigateurs peuvent s'abonner pour recevoir les mises à jour vivantes de données, utilisant la technologie WebSocket(WS)/WebSocket Secure(WSS), ou à l'aide du long mécanisme de sondage.

WebSocket (RFC 6455)

- Le WebSocket Protocol (RFC 6455) est un protocole basé sur TCP indépendant. Ses seules relations au HTTP sont que sa prise de contact est interprétée par des serveurs HTTP car une demande de mise à jour, et est utilisée pour établir une connexion persistante avec le serveur pour les mises à jour continues.
- Après une prise de contact réussie de WebSocket, les clients et serveurs transfèrent des données dans les deux sens dans de plus petites unités conceptuelles (objets JSON) désignées dans cette spécification sous le nom des **messages**. Le message de WebSocket ne correspond pas nécessairement à un tramage particulier de couche réseau, car un message fragmenté peut être fusionné ou séparé par un intermédiaire.
- Ce process starts avec le client envoyant une demande de HTTP régulière au serveur.
- Une en-tête de mise à jour est incluse dans cette demande qui informe le serveur que le client

souhaite établir une connexion de WebSocket. c.-à-d. **mise à jour de protocoles de la commutation HTTP/1.1101 : connexion de websocket : Mise à jour.**

- Après ce, une salle est ouverte et des messages sont acheminés continuellement au client/au navigateur.
- WebSocket est connu pour réduire traiter le temps système sur le serveur de édition, et fournit les modifications direct sécurisées et dynamiques.

Longue interrogation

- C'est une variation de la technique traditionnelle d'interrogation et permet l'émulation d'un pousser de l'information d'un serveur à un client.
- Avec la longue interrogation, le client demande les informations du serveur d'une manière semblable à un balayage normal. Une connexion est jugée ouverte entre le client web et le web server de sorte que quand le serveur a les nouvelles informations elle puisse les pousser au client et la connexion soit alors fermée.
- Si le serveur n'a aucune informations disponibles pour le client, au lieu de l'envoi une réponse vide, le serveur tient la demande et attend quelques informations pour être disponible.
- Une fois que les informations deviennent disponibles (ou après un délai d'attente approprié), une réponse complète est envoyée au client.
- Les re-demands de client normalement puis immédiatement les informations du serveur, de sorte que le serveur ait toujours une demande attendante disponible qu'elles peuvent employer pour fournir des données en réponse à un événement.
- La longue interrogation entraîne le temps système supplémentaire sur le serveur qui est prié pour ces informations.

Nombre de contrôle de clients exécutant la longue interrogation

- Pour UCCX vivent les données, **WebSocket est favorable et recommandé** au-dessus du mécanisme traditionnel de long-interrogation.
- Pour vérifier si les clients utilisent WebSocket OU long mécanisme de sondage pour des données vivantes, vous pouvez analyser le SocketIO met au point et voit les messages.

Étapes :

- Le tour vers le haut du socket E/S met au point pour niveler le DEBUG (du par défaut AVERTISSEZ) - naviguent vers l'**utilité > le suivi UCCX.**
- Suivez les derniers journaux du serveur de SocketIO :

Vous voyez ceci :

E.g: IE compatibility mode -

TimeStamp <LOG message> (POLLING): ...

```
0000000135: 10.78.91.238: Nov 28 2016 20:19:44.297 +0530: %CCBU_nioEventLoopGroup-5-1-6-
MessageDispatcher: %[address=/10.107.11.107:51809(WEBSOCKET)] [message=Subscribe-
{AgentCSQStats=[Email_CSQ]}} [socket_io_server_type=WSS]: Subscription Request
```

E.g: Firefox/Chrome/IE Native with HTML5/websocket enabled:

TimeStamp <LOG message> (WEBSOCKET): ...

Flux de données vivant

Le flux de données vivant peut être récapitulé en tant que : **CCX instruments de clients de → de serveur de SocketIO de → du → JMS ActiveMQ de l'engine (ICD_RTDM, SS_RMCM) (navigateur Web)**

Ceux-ci se connectent les équipements qui doivent être activés sur l'engine UCCX :

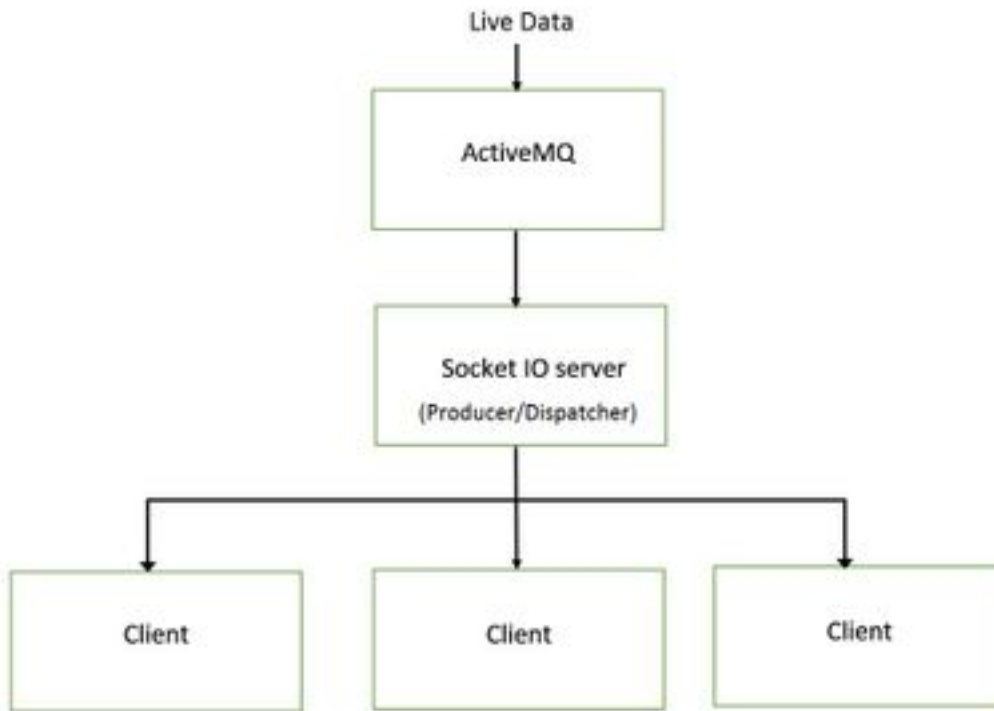
ICD_RTDM = gestionnaire en temps réel intelligent des données de distribution d'appel

SS_RMCM = gestionnaire de contacts de gestionnaire de ressources de sous-système

JMS ActiveMQ = implémentation active de file d'attente de Messagerie de service de messagerie de Javas

Le serveur vivant de SocketIO de données est conçu principalement pour recevoir les informations du point d'émission de données vivant (tel que JMS) et pour les éditer, par l'intermédiaire de WebSocket, pour vivre des clients de données (comme un instrument de client sur la finesse ou l'application Web de centre d'intelligence de Cisco Unified (CUIC).

Ces le schéma logique décrit le flux de données de l'engine UCCX (données vivantes) dans ActiveMQ, où le serveur Socket.IO puis récupère les informations et les distribue aux clients, suivant les indications de l'image de SocketIO a basé des abonnements de client :



Les données vivantes éditent des événements dans les lots au-dessus des thèmes par l'intermédiaire d'ActiveMQ.

Le serveur de SocketIO s'abonne à ces thèmes et reçoit les lots d'événement.

Salles serveur et thèmes de SocketIO

Le serveur de SocketIO est configuré avec de diverses salles ou les thèmes définis dans `/opt/cisco/uccx/socketioserver/conf/socketioservice.properties` classent.

Tous les états vivants de données sont tracés à un thème JMS, qui est utile tout en analysant les logs.

La mise à jour de thème JMS se produit en toutes les 3 secondes, parce que certains signaux de ces derniers sont basés sur des événements (une mise à jour est signalée seulement quand un événement est déclenché, comme une modification de l'état de l'agent).

Ils sont répertoriés dans la table de l'état au mappage de thème JMS :

Nom du rapport	Nom de thème
État de statistiques de l'agent CSQ	AgentCSQStats -----> l'événement basé sur le thème
État de log d'état de l'agent	AgentStateDetailStats -----> l'événement basé sur le thème
État de statistique de l'agent	ResourceIAQStats
Compte rendu succinct d'équipe d'agent	ResourceIAQStats

État d'état d'équipe	ResourceIAQStats
Compte rendu succinct d'équipe	ResourceIAQStats
État de détail de la Voix CSQ	VoiceCSQDetailsStats
Résumé de la Voix CSQ	VoicelAQStats
Compte rendu succinct de la conversation CSQ	ChatQueueStatistics -----> l'événement a basé le thème
État de statistique de l'agent de conversation	ChatAgentStats -----> l'événement a basé le thème
État de statistique de l'agent d'email	EmailAgentStats -----> l'événement a basé le thème
Compte rendu succinct de l'email CSQ	EmailQueueStatistics -----> l'événement a basé le thème

Les clients comme le serveur CUIIC se connectent au serveur Socket.IO pour s'abonner aux salles répertoriées ci-dessus afin de recevoir les événements vivants de données du serveur de SocketIO.

La demande d'abonnement est envoyée du CUIIC pendant que bientôt un état est lancé du CUIIC. Par exemple lancer le **compte rendu succinct de la Voix CSQ** sur le CUIIC déclenche une demande d'abonnement à la salle/au thème correspondants (VoicelAQStats) au serveur SIO.

Producteur

1. Reçoit des lots d'événement d'ActiveMQ.
2. Les divise vers le bas en leurs événements discrets.
3. Coups d'oeil sur chaque événement pour obtenir l'id de l'événement.
4. Conduit l'événement à la salle basée sur le thème combiné avec l'id obtenu.

Répartiteur

Le répartiteur reçoit la salle discrète nommée des événements du producteur et envoie ces événements à tous les clients qui s'abonnent à cette pièce.

Clients

1. Les clients basés sur le WEB (CUIIC OU des instruments d'agents/superviseur de finesse) se connectent au serveur de SocketIO au-dessus du socket de Web.
2. Une fois que connectés, ces clients s'abonnent aux salles dans le serveur du socket E/S.
3. Plusieurs clients peuvent s'abonner à la même salle.
4. Les clients reçoit un événement à la fois.

Exemple :

```
1/18/2017, 10:14:07 PM=INFOC8E2DB0C10000140000000A40A4E5E6B= Subscribing for rooms:  
<VoiceIAQStats=cssCsq> report-layer.js:1414:11
```

```
1/18/2017, 10:14:07 PM=INFOC8E2DB0C10000140000000A40A4E5E6B= ReportManager : filterString  
[{"fieldId":"720298FB10000140000000A10A4E5E6F", "fieldType":"VALUELIST", "name":"VoiceIAQStats.esd  
Name", "operator":"SetValues", "value":[{"key":"cssCsq", "desc":"cssCsq"}], "valuelistId":"01FB2C011  
0000133771FC3C33F57F543", "isKeyField":true}] report-layer.js:1414:11
```

Le serveur de SocketIO commence alors des événements de édition au client par cet abonnement.

Authentification

Le serveur Socket.IO doit authentifier les clients se connectants (comme l'instrument de données de finesse OU le CUIC vivant) avant d'envoyer les données.

1. Le serveur Socket.IO propose d'utiliser l'authentification basée par jeton.
2. Les instruments vivants de données effectuent une première demande d'authentification qui leur remet un jeton authentique.
3. Ceci est encore passé en fonction au serveur Socket.IO pour authentifier la création d'une connexion WSS (serveur de WebSocket). Pour ceci, Socket.IO appelle l'API équipé par UCCX de jeton qu'il a obtenu des instruments.
4. Toute l'authentification est faite utilisant une chaîne de symboles. L'authentification client est activée dans le service Socket.IO par l'intermédiaire de la propriété Client.Authentication.Rest.URL.
5. Tous connecter des clients activés doivent fournir cette chaîne comme paramètre sur l'URL de connexion.
6. Les clients au serveur Socket.IO se connectent, s'abonnent, se désabonnent, et déconnectent. La connexion initiale peut être juste une connexion ou également inclure l'abonnement.

Base connection URL

```
ws://xx.xx.xx.xx:12007  
wss://xx.xx.xx.xx:12008
```

Connection URL with subscription

```
ws://xx.xx.xx.xx:12007?action=subscribe&agent=7900,7901
```

Connection URL with token

```
ws://xx.xx.xx.xx:12007?token=sometokenstring
```

Connection URL with token and subscription

```
ws://xx.xx.xx.xx:12007?token=sometokenstring&action=subscribe&agent=7900,7901
```

le WS = le websocket

wss = websocket sécurisé

Le keystore qui contient les Certificats associés pour WSS (UCCX utilise le par défaut : Certificats d'application de Cisco Tomcat)

L'emplacement du mot de passe pour WSS (par défaut : Tomcat du système)

Suivi vivant de données

Pour se connecter, il y a 3 points principaux d'écoulement dans des données vivantes UCCX utilisant l'architecture de SocketIO sur 11.x (**engine MIVR_RTDM, engine SS_RMCM > JMS ActiveMQ > serveur de SocketIO**)

- L'engine UCCX se connecte (MIVR avec le positionnement RTDM et RMCM à Xdebugg5) envoyer message à JMS (toutes les 3 secondes comme vues dans le sous-système de RM met au point sur l'engine (MIVR) SS_RMCM)
- Logs de SocketIO (se connecter de service de SocketIO doit être placé à la pleine élimination des imperfections où JMS l'envoie message au consommateur de SocketIO, qui reçoit tous les messages édité par l'engine).
- Messages de console de navigateur (F12) : SocketIO traitant les mises à jour entrante, créant les salles appropriées pour chaque thème, et puis envoyant ces derniers aux navigateurs Web de clients par l'intermédiaire du répartiteur de message. (Ceux-ci sont vus dans les logs du navigateur Web Console/F12).

Ces suivis doivent être activés afin de voir le détaillé messaged dans les logs.

1. Engine UCCX : **ICD_RTDM, SS_RMCM** avec le niveau 5 de Xdebugging
2. Se connecter de service de SocketIO : Niveau de DEBUG.
3. Messages de console de navigateur (F12) pour vérifier les mises à jour vivantes entrantes de données.

SocketIO mettent au point des logs disent seulement le statut des thèmes comme connecté/l'ont déconnecté et se sont abonné/désabonné.

Le suivi détaillé doit être activé afin de voir des événements spécifiques sur un thème. Les logs de producteur et de répartiteur de SocketIO sont disponibles seulement après activer le suivi détaillé (contact Cisco TAC pour activer le suivi détaillé).

Dépannage des données vivantes

Healthcheck de base

La première étape vers les questions vivantes de données de dépannage est d'établir une spécification de base prise en charge de configuration et d'exécuter une vérification de l'intégrité de base du système.

Les questions trouvées dans des points de reprise de cette configuration peuvent entraîner des questions dans le rendu vivant de données sur l'instrument de finesse OU sur la page Web CUIC.

Vous devriez éliminer tous ces points avant de commencer avec le dépannage basé par log.

Résumé

- La version de navigateur devrait être compatible.
- L'utilisation de CPU et mémoire sur l'UCCX doit être dans les limites.
- Le point d'émission de données vivant de données doit être en ligne à la page de points d'émission de données CUIC.
- Le NTP et les DN sur l'UCCX doivent être en service, avec le NTP dans le sync, et les DN en avant/consultations d'inverse dans l'ordre approprié.
- Les OVULES VM doivent correctement provisionnés avec des conditions requises de vRAM et de vCPU selon des instructions d'OVULES de Cisco.
- Le NIC en service pour la VM devrait être VMXNET3.

Vérifiez la version de navigateur

Vérifiez le comportement des données Live à travers différents navigateurs et versions de navigateur.

UCCX prend en charge actuellement les versions de navigateur suivantes commençant 11.5 : [http://docwiki.cisco.com/wiki/Unified_CCX_Software_Compatibility_Matrix_for_11.5\(1\)#Supported_Browsers](http://docwiki.cisco.com/wiki/Unified_CCX_Software_Compatibility_Matrix_for_11.5(1)#Supported_Browsers)

Souvenez-vous POUR DÉSACTIVER le mode compatible sur l'IE puisque cela n'est pas pris en charge pour des données Live. L'IE retombe au long mécanisme de sondage quand dans long le vote ainsi de ceci est une importante considération.

Vérifiez l'utilisation de CPU et mémoire

Vérifiez l'utilisation du CPU et l'utilisation de mémoire sur le serveur utilisant des commandes RTMT ou CLI.

Par exemple : RTMT >

CLI : affichez la CPU de chargement de processus

CLI : affichez la mémoire de chargement de processus

Vérifiez ces processus si des données vivantes de visionnement sur CUIC : UCCX_Engine, socketioservice, cuicreporting.

Vérifiez la source vivante de transmission en continu est en ligne

Afin de vérifier que la source vivante de transmission en continu est en ligne, naviguez vers la page **Web CUIC > le système > les points d'émission de données**

Vérifiez le NTP.

Vérifiez que le NTP est accessible et UCCX est synchronisé au serveur de NTP à la strate <=5

CLI : état de ntp d'utilis

```
ntpd (pid 7152) is running...

remote          refid          st t when poll reach  delay  offset  jitter
=====
*xx.xx.xx.xx   .GPS.          1 u  161 1024  377   22.451    0.269   0.958
+xx.xx.xx.xx   xx.xx.xx.xx    2 u  642 1024  377   22.604    4.439   4.414
synchronisé au serveur de NTP (xx.xx.xx.xx) au strate 2
```

Vérifiez les DN

Vérifiez que les DN les résout.

CLI : <hostname> d'hôte réseau d'utilis

Résolution locale :

résolutions hostname.domainname localement à xx.xx.xx.xx

Résolution externe :

hostname.domainname a l'adresse xx.xx.xx.xx

Exécutez un test de diagnostic

Vous pouvez exécuter un test de diagnostic du CLI pour identifier n'importe quel autre éventuel problème avec le serveur (par exemple latence de Tomcat, problèmes de réseau, etc.)

CLI : les utils diagnostiquent le test

Vérifiez l'adresse Internet

L'adresse Internet UCCX doit j'en minuscules.

Référez-vous s'il vous plaît installent/guide de mise à jour et ce défaut.

<http://cdets.cisco.com/apps/dumpcr?identifieur=CSCva75058&content=summary&format=html>

Vérifiez la configuration VM

Vérifiez cela le modèle ouvert que correct d'appareils de virtualisation (OVULES) est utilisé selon ces liens :

https://www.cisco.com/c/dam/en/us/td/docs/voice_ip_comm/uc_system/virtualization/virtualization-cisco-unified-contact-center-express.html#11.5

<https://software.cisco.com/download/type.html?mdfid=286287033&flowid=76362>

La configuration de l'adaptateur réseau VM UCCX (NIC) doit être **VMXNET3**.

Si l'adaptateur correct n'est pas sélectionné alors UCCX n'édite pas les données en temps réel à régulier évalué.

Si la configuration d'adaptateur réseau VM n'est pas VMXNET3 alors suivent ce lien pour corriger les configurations d'adaptateur :

http://docwiki.cisco.com/wiki/Virtualization_for_Cisco_Unified_Contact_Center_Express

Vérifiez les Certificats UCCX

La non-concordance de certificat ou quand des Certificats ne sont pas reçus peut faire disparaître la source vivante de transmission en continu déconnectée/off-line.

- Recevez le certificat une fois incité.
- Assurez-vous que des Certificats de tiers ne sont pas expirés.
- Vérifiez que l'adresse Internet dans le certificat est correcte. Sinon, suivez la procédure ci-dessous.

CCX du côté :

1. Éditez ce fichier `/usr/local/platform/conf/platformConfig.xml` et corrigez l'adresse Internet et assurez-vous que les entrées DNS sont selon le nom dans ce fichier.
2. Supprimez et régénérez les Certificats après avoir vérifié les détails ci-dessus sur la batterie UCCX utilisant les commandes CLI :
CLI : placez le chat du CERT REGEN
3. Redémarrez le service de Cisco Tomcat sur UCCX.
CLI : reprise Cisco Tomcat de service d'utilis

Plus sur des Certificats :

<http://www.cisco.com/c/en/us/support/docs/customer-collaboration/unified-contact-center-express/118855-configure-uccx-00.html>

Vérifiez la configuration de classement par taille prise en charge

Assurez-vous que les limites de configuration maximale sont adhérentes à sur l'installation. Pour ceci, référez-vous le guide de conception de niveau de solution UCCX pour la configuration du serveur et les limites maximum :

UCCX 11.5

: http://www.cisco.com/c/en/us/td/docs/voice_ip_comm/cust_contact/contact_center/crs/express_11_5/desi..

UCCX 11.0 : http://www.cisco.com/c/en/us/td/docs/voice_ip_comm/cust_contact/contact_center/crs/express_11_0/desi..

Pour confirmer ceci vous pouvez exécuter ces commandes CLI, vérifiant l'instruction de dimensionnement, sur l'installation pour obtenir le numéro exact.

L'INFORMATION REQUISE

Total non des superviseurs assignés aux équipes

Total non des superviseurs configurés

Total non des agents configurés

Total non des équipes configurées

Non des agents par équipe

Non des superviseurs par équipe

Nombre d'utilisateurs par le rôle (CUIC)

Nombre de CSQ configurés

REQUÊTE

exécutez le compte choisi de db_cra de l'uccx SQL (*) de (resourcelogin distinct choisi de superviseur où active = « t »)

exécutez le compte CHOISI de db_cra de l'uccx SQL (*) de la ressource active='t' et resourcetype=2

exécutez le compte CHOISI de db_cra de l'uccx SQL (*) de la ressource active='t' et resourcetype=1

exécutez le compte CHOISI de db_cra de l'uccx SQL (*) de l'équipe où active='t'

exécutez le db_cra team.teamname CHOISI de l'uccx SQL, count(resource.resourceid) COMME NumOfAgents de ressource en JOINDRE INTÉRIEUR d'équipe SUR

team.teamid=resource.assignedTeamID où GROUPE resource.active= team.active='t' PAR le teamname

exécutez le db_cra team.teamName CHOISI de l'uccx SQL, count(Supervisor.recordID) COMME NumOfSupervisors d'équipe de JOINDRE INTÉRIEUR de superviseur SUR

Supervisor.managedTeamID=team.teamid où GROUPE Supervisor.act et team.active='t' PAR le teamName

exécutez SQL g.name choisi comme groupname, comptez (*) comme groupsize du cuic_data : cuicgroup g, cuic_data : GM de cuicgroupmem où groupe g.id = gm.groupid par g.name

exécutez le compte choisi de db_cra de l'uccx SQL (*) du contactservicequeue où active='t'

Mises en garde connues

[CSCva13838](#) : Retard vivant de données sur CUIC

[CSCus17605](#) : Retardez en raison des autorisations excessives d'enregistrement aux superviseurs (la charge élevée)

[CSCux33949](#) : Questions vivantes de données dues aux permis améliorés.

[CSCvb67761](#) : État de log d'état de l'agent entraînant la CPU de haute affectant des données vivantes.

[CSCva95411](#) : La CPU de haute de service de SocketIO due à l'espace de noms objecte prenant la mémoire excessive de segment de mémoire.

[CSCvb75279](#) : Source vivante de transmission en continu allant off-line pendant à hors du mot de passe du sync MIVR Keystore

[CSCvc45189](#) : Source vivante de transmission en continu off-line due à l'exécution de sauvegarde et de restauration

Remarque: Pour confirmer ou le contournement les mises en garde ci-dessus, impliquent s'il vous plaît Cisco TAC en ouvrant une demande de service.

Suivi vivant de log de données

Logs d'engine

- Les données vivantes éditent des événements dans les lots au-dessus des thèmes par l'intermédiaire de MQ actif.
- Le serveur de SocketIO s'abonne à ces thèmes et reçoit les lots d'événement.

Logs MIVR : Des données vivantes sont envoyées de l'engine par un message JMS au-dessus des thèmes (par exemple : VoiceCSQDetailsStats, VoiceIAQStats, ResourceIAQStats etc...) Pour l'événement basé l'engine de thèmes (ChatAgentStats, ChatQueueStatistics AgentCSQStats, AgentStateDetailStats) envoie les messages vides JMS après toutes les 3 secondes pour maintenir la salle de SocketIO active.

Ces suivis areseen seulement après activer ces suivis : [SocketIO a détaillé le suivi](#)

```
558468: Jan 18 22:27:00.003 EST %MIVR-ICD_RTDM-7-UNK:LiveData: Sending JMS message :
[{"id":"cssCsq","operation":"UPDATE","VoiceIAQStats":{"id":2,"esdId":2,"esdName":"cssCsq","nResourcesLoggedIn":1,"nTotalContacts":7,"nHandledContacts":7,"nAbandonedContacts":0,"nDequeuedContacts":0,"avgTalkDuration":503030,"longestTalkDuration":3225646,"avgWaitDuration":17959,"longestWaitDuration":30383,"longestCurrentlyWaitingDuration":0,"nAvailResources":0,"nInSessionResources":0,"nUnavailResources":1,"nWorkResources":0,"nSelectedResources":0,"nWaitingContacts":0,"nAverageHandlingTimeLowThreshold":0.0,"nAverageHandlingTimeHighThreshold":0.0,"nAverageWaitingTimeLowThreshold":0.0,"nAverageWaitingTimeHighThreshold":0.0,"nAbandonedCallsLowThreshold":0,"nAbandonedCallsHighThreshold":0,"nDequeuedCallsLowThreshold":0,"nDequeuedCallsHighThreshold":0,"nSLAPercentageLowThreshold":0.0,"nSLAPercentageHighThreshold":0.0}}]
```

```
558493: Jan 18 22:27:00.006 EST %MIVR-ICD_RTDM-7-UNK:LiveData: Sending JMS message :
[{"id":"agent7901","operation":"UPDATE","VoiceCSQDetailsStats":{"agentId":"agent7901","agentName":"agent7901","agentState":"Not Ready","skillGroup":"","agentStateDuration":148093,"reasonCode":0,"AgentVoiceCSQNames":[{"agentVoiceCSQName":"cssCsq"}]}]
```

```
558501: Jan 18 22:27:00.008 EST %MIVR-ICD_RTDM-7-UNK:LiveData: Sending JMS message :
[{"id":"agent7901","operation":"UPDATE","ResourceIAQStats":{"resourceId":"agent7901","resourceName":"agent7901","resourceState":2,"durationInStateMillis":148093,"nHandledContacts":0,"nPresentedContacts":0,"avgTalkDuration":0,"longestTalkDuration":0,"avgHoldDuration":0,"longestHoldDuration":0,"avgHandleDuration":0,"avgWorkDuration":0,"totalTalkTime":0,"totalHoldTime":0,"maxReadyTime":2547,"avgReadyTime":2547,"totalReadyTime":2547,"maxNotReadyTime":100760,"avgNotReadyTime":58701,"totalNotReadyTime":265495,"maxWorkTime":0,"totalWorkTime":0,"logonDuration":268042,"totalReservedTime":0,"otherTime":0,"avgSpeedOfAnswer":0,"rsrcCurrentStateReasonCode":0,"nOutboundOfferedContacts":0,"nOutboundActiveOfferedContacts":0,"nOutboundTimedOutContacts":0,"nOutboundVoiceContacts":0,"nOutboundAcceptedContacts":0,"nOutboundActiveVoiceContacts":0,"nOutboundActiveAcceptedContacts":0,"nOutboundRejectedContacts":0,"nOutboundClosedContacts":0,"avgOutboundTalkDuration":0,"activeOutboundTalkDuration":0,"avgOutboundHoldDuration":0,"activeOutboundHoldDuration":0,"longestOutboundTalkDuration":0,"longestOutboundHoldDuration":0,"totalOutboundTalkTime":0,"totalOutboundHoldTime":0,"avgOutboundWorkTime":0,"maxOutboundWorkTime":0,"totalOutboundWorkTime":0,"avgOutboundTalkWindow1":0,"avgOutboundHoldWindow1":0,"avgOutboundTalkWindow2":0,"avgOutboundHoldWindow2":0,"nOutboundHandledContacts":-
```

```
1, "avgTalkingWindow1":0, "avgHoldWindow1":0, "avgTalkingWindow2":0, "avgHoldWindow2":0, "strResource
State":"Not Ready"}}]
```

```
//Engine sends empty heartbeat
```

```
558509: Jan 18 22:27:01.274 EST %MIVR-SS_RMCM-7-UNK:HeartBeatAction : Sending JMS message : []
for destinations [topic://ChatAgentStats, topic://ChatQueueStatistics, topic://AgentCSQStats,
topic://AgentStateDetailStats]
```

```
558510: Jan 18 22:27:01.279 EST %MIVR-SS_RMCM-7-UNK:HeartBeatAction : Sending JMS message : []
SUCCESS
```

Logs de SocketIO (1)

SocketIO reçoit des lots d'événement de MQ actif

Pendant que les séries d'événements arrivent, le serveur de SocketIO fait ceci :

- Les divise vers le bas en leurs événements discrets.
- Coups d'oeil sur chaque événement pour obtenir l'id de l'événement.
- Conduit l'événement à la salle basée sur le thème combiné avec l'id obtenu

Des événements discrets avec des noms de pièce sont mis en file d'attente sur le répartiteur

Remarque: Le producteur de SocketIO qui a l'auditeur/consommateur JMS obtient les données du bus JMS sur les thèmes par l'intermédiaire d'ActiveMQ et les envoie aux salles de SocketIO qui sont mises en file d'attente au répartiteur.

Ces suivis sont vus seulement après activer les suivis détaillés sur SocketIO (contact Cisco TAC pour activer la même chose).

Exemple 1 : Thème de VoiceIAQStats.

```
CCBU.2017-01-18T22-25-57.006.log 414 0000013032: 10.86.64.208: Jan 18 2017 22:27:00.005 -
0500: %CCBU_Camel (camel-1) thread #6 - JmsConsumer[VoiceIAQStats]-6-MessageProducer:
%[message={"id":"cssCsq", "operation":"UPDATE", "VoiceIAQStats":{"id":2, "esdId":2, "esdName":"cssCs
q", "nResourcesLoggedIn":1, "nTotalContacts":7, "nHandledContacts":7, "nAbandonedContacts":0, "nDeque
uedContacts":0, "avgTalkDuration":503030, "longestTalkDuration":3225646, "avgWaitDuration":17959, "l
ongestWaitDuration":30383, "longestCurrentlyWaitingDuration":0, "nAvailResources":0, "nInSessionRes
ources":0, "nUnavailResources":1, "nWorkResources":0, "nSelectedResources":0, "nWaitingContacts":0, "
nAverageHandlingTimeLowThreshold":0.0, "nAverageHandlingTimeHighThreshold":0.0, "nAverageWaitingTi
meLowThreshold":0.0, "nAverageWaitingTimeHighThreshold":0.0, "nAbandonedCallsLowThreshold":0, "nAba
ndonedCallsHighThreshold":0, "nDequeuedCallsLowThreshold":0, "nDequeuedCallsHighThreshold":0, "nSLA
PercentageLowThreshold":0.0, "nSLAPercentageHighThreshold":0.0}}] [room_name=VoiceIAQStats*cssCsq]
[room_prefix=VoiceIAQStats] [status=RECEIVED] [topic=jms:topic:VoiceIAQStats]: Event Detail Trace
```

```
CCBU.2017-01-18T22-25-57.006.log 415 0000059433: 10.86.64.208: Jan 18 2017 22:27:00.006 -
0500: %CCBU_Camel (camel-1) thread #6 - JmsConsumer[VoiceIAQStats]-6-MessageDispatcher:
%[client_count=-
1] [message={"id":"cssCsq", "operation":"UPDATE", "VoiceIAQStats":{"id":2, "esdId":2, "esdName":"cssC
sq", "nResourcesLoggedIn":1, "nTotalContacts":7, "nHandledContacts":7, "nAbandonedContacts":0, "nDequ
euedContacts":0, "avgTalkDuration":503030, "longestTalkDuration":3225646, "avgWaitDuration":17959, "
longestWaitDuration":30383, "longestCurrentlyWaitingDuration":0, "nAvailResources":0, "nInSessionRe
sources":0, "nUnavailResources":1, "nWorkResources":0, "nSelectedResources":0, "nWaitingContacts":0,
```

```
"nAverageHandlingTimeLowThreshold":0.0,"nAverageHandlingTimeHighThreshold":0.0,"nAverageWaitingTimeLowThreshold":0.0,"nAverageWaitingTimeHighThreshold":0.0,"nAbandonedCallsLowThreshold":0,"nAbandonedCallsHighThreshold":0,"nDequeuedCallsLowThreshold":0,"nDequeuedCallsHighThreshold":0,"nSLAPercentageLowThreshold":0.0,"nSLAPercentageHighThreshold":0.0}}][room_name=VoiceIAQStats*cssCsq][socket_io_server_type=WS/WSS][status=ENQUEUED]: Event Detail Trace
```

Exemple 2 : Thème de VoiceCSQDetailsStats.

```
0000013033: 10.86.64.208: Jan 18 2017 22:27:00.009 -0500: %CCBU_Camel (camel-4) thread #1 - JmsConsumer[VoiceCSQDetailsStats]-6-MessageProducer: %[message={"id":"agent7901","operation":"UPDATE","VoiceCSQDetailsStats":{"agentId":"agent7901","agentName":"agent7901","agentState":"Not Ready","skillGroup":"","agentStateDuration":148093,"reasonCode":0,"AgentVoiceCSQNames":[{"agentVoiceCSQName":"cssCsq"}]}]}][room_name=VoiceCSQDetailsStats*agent7901][room_prefix=VoiceCSQDetailsStats][status=RECEIVED][topic=jms:topic:VoiceCSQDetailsStats]: Event Detail Trace
```

```
0000059438: 10.86.64.208: Jan 18 2017 22:27:00.009 -0500: %CCBU_Camel (camel-4) thread #1 - JmsConsumer[VoiceCSQDetailsStats]-6-MessageDispatcher: %[client_count=-1][message={"id":"agent7901","operation":"UPDATE","VoiceCSQDetailsStats":{"agentId":"agent7901","agentName":"agent7901","agentState":"Not Ready","skillGroup":"","agentStateDuration":148093,"reasonCode":0,"AgentVoiceCSQNames":[{"agentVoiceCSQName":"cssCsq"}]}]}][room_name=VoiceCSQDetailsStats*agent7901][socket_io_server_type=WS/WSS][status=ENQUEUED]: Event Detail Trace
```

Exemple 3 : Thème de ResourceIAQStats

```
0000013034: 10.86.64.208: Jan 18 2017 22:27:00.011 -0500: %CCBU_Camel (camel-3) thread #4 - JmsConsumer[ResourceIAQStats]-6-MessageProducer: %[message={"id":"agent7901","operation":"UPDATE","ResourceIAQStats":{"resourceId":"agent7901","resourceName":"agent7901","resourceState":2,"durationInStateMillis":148093,"nHandledContacts":0,"nPresentedContacts":0,"avgTalkDuration":0,"longestTalkDuration":0,"avgHoldDuration":0,"longestHoldDuration":0,"avgHandleDuration":0,"avgWorkDuration":0,"totalTalkTime":0,"totalHoldTime":0,"maxReadyTime":2547,"avgReadyTime":2547,"totalReadyTime":2547,"maxNotReadyTime":100760,"avgNotReadyTime":58701,"totalNotReadyTime":265495,"maxWorkTime":0,"totalWorkTime":0,"logonDuration":268042,"totalReservedTime":0,"otherTime":0,"avgSpeedOfAnswer":0,"rsrcCurrentStateReasonCode":0,"nOutboundOfferedContacts":0,"nOutboundActiveOfferedContacts":0,"nOutboundTimedOutContacts":0,"nOutboundVoiceContacts":0,"nOutboundAcceptedContacts":0,"nOutboundActiveVoiceContacts":0,"nOutboundActiveAcceptedContacts":0,"nOutboundRejectedContacts":0,"nOutboundClosedContacts":0,"avgOutboundTalkDuration":0,"activeOutboundTalkDuration":0,"avgOutboundHoldDuration":0,"activeOutboundHoldDuration":0,"longestOutboundTalkDuration":0,"longestOutboundHoldDuration":0,"totalOutboundTalkTime":0,"totalOutboundHoldTime":0,"avgOutboundWorkTime":0,"maxOutboundWorkTime":0,"totalOutboundWorkTime":0,"avgOutboundTalkWindow1":0,"avgOutboundHoldWindow1":0,"avgOutboundTalkWindow2":0,"avgOutboundHoldWindow2":0,"nOutboundHandledContacts":-1,"avgTalkingWindow1":0,"avgHoldWindow1":0,"avgTalkingWindow2":0,"avgHoldWindow2":0,"strResourceState":"Not Ready"}}]}][room_name=ResourceIAQStats*agent7901][room_prefix=ResourceIAQStats][status=RECEIVED][topic=jms:topic:ResourceIAQStats]: Event Detail Trace
```

```
0000059443: 10.86.64.208: Jan 18 2017 22:27:00.011 -0500: %CCBU_Camel (camel-3) thread #4 - JmsConsumer[ResourceIAQStats]-6-MessageDispatcher: %[client_count=-1][message={"id":"agent7901","operation":"UPDATE","ResourceIAQStats":{"resourceId":"agent7901","resourceName":"agent7901","resourceState":2,"durationInStateMillis":148093,"nHandledContacts":0,"nPresentedContacts":0,"avgTalkDuration":0,"longestTalkDuration":0,"avgHoldDuration":0,"longestHoldDuration":0,"avgHandleDuration":0,"avgWorkDuration":0,"totalTalkTime":0,"totalHoldTime":0,"maxReadyTime":2547,"avgReadyTime":2547,"totalReadyTime":2547,"maxNotReadyTime":100760,"avgNotReadyTime":58701,"totalNotReadyTime":265495,"maxWorkTime":0,"totalWorkTime":0,"logonDuration":268042,"totalReservedTime":0,"otherTime":0,"avgSpeedOfAnswer":0,"rsrcCurrentStateReasonCode":0,"nOutboundOfferedContacts":0,"nOutboundActiveOfferedContacts":0,"nOutboundTimedOutContacts":0,"nOutboundVoiceContacts":0,"nOutboundAcceptedContacts":0,"nOutboundActiveVoiceContacts":0,"nOutboundActiveAcceptedContacts":0,"nOutboundRejectedContacts":0,"nOutboundClosedContacts":0,"avgOutboundTalkDuration":0,"activeOutboundTalkDuration":0,"avgOutboundHoldDuration":0,"activeOutboundHoldDuration":0,"longestOutboundTalkDuration":0,"longestOutboundHoldDuration":0,"totalOutboundTalkTime":0,"t
```



```
otalOutboundHoldTime":0,"avgOutboundWorkTime":0,"maxOutboundWorkTime":0,"totalOutboundWorkTime":0,"avgOutboundTalkWindow1":0,"avgOutboundHoldWindow1":0,"avgOutboundTalkWindow2":0,"avgOutboundHoldWindow2":0,"nOutboundHandledContacts":-1,"avgTalkingWindow1":0,"avgHoldWindow1":0,"avgTalkingWindow2":0,"avgHoldWindow2":0,"strResourceState":"Not Ready}}}[room_name=ResourceIAQStats*agent7901][socket_io_server_type=WS/WSS][status=ENQUEUED]: Event Detail Trace
```

Logs de SocketIO (2)

Répartiteur du socket E/S : Obtient des événements de producteur et les livre aux clients

Le répartiteur reçoit la salle discrète nommée des événements du producteur et envoie ces événements à tous les clients qui s'abonnent à cette pièce.

Ces suivis sont vus seulement après activer les suivis détaillés sur SocketIO (contact Cisco TAC pour activer la même chose).

Exemple 1 : Thème de VoiceIAQStats.

```
0000059436: 10.86.64.208: Jan 18 2017 22:27:00.007 -0500: %CCBU_pool-19-thread-1-6-MessageDispatcher:
%[client_count=1][message={"id":"cssCsq","operation":"UPDATE","VoiceIAQStats":{"id":2,"esdId":2,"esdName":"cssCsq","nResourcesLoggedIn":1,"nTotalContacts":7,"nHandledContacts":7,"nAbandonedContacts":0,"nDequeuedContacts":0,"avgTalkDuration":503030,"longestTalkDuration":3225646,"avgWaitDuration":17959,"longestWaitDuration":30383,"longestCurrentlyWaitingDuration":0,"nAvailResources":0,"nInSessionResources":0,"nUnavailResources":1,"nWorkResources":0,"nSelectedResources":0,"nWaitingContacts":0,"nAverageHandlingTimeLowThreshold":0.0,"nAverageHandlingTimeHighThreshold":0.0,"nAverageWaitingTimeLowThreshold":0.0,"nAverageWaitingTimeHighThreshold":0.0,"nAbandonedCallsLowThreshold":0,"nAbandonedCallsHighThreshold":0,"nDequeuedCallsLowThreshold":0,"nDequeuedCallsHighThreshold":0,"nSLAPercentageLowThreshold":0.0,"nSLAPercentageHighThreshold":0.0}}}[room_name=VoiceIAQStats*cssCsq][socket_io_server_type=WSS][status=SENT]: Event Detail Trace
```

Exemple 2 : Thème de VoiceCSQDetailsStats.

```
0000059441: 10.86.64.208: Jan 18 2017 22:27:00.010 -0500: %CCBU_pool-19-thread-1-6-MessageDispatcher:
%[client_count=1][message={"id":"agent7901","operation":"UPDATE","VoiceCSQDetailsStats":{"agentId":"agent7901","agentName":"agent7901","agentState":"Not Ready","skillGroup":"","agentStateDuration":148093,"reasonCode":0,"AgentVoiceCSQNames":[{"agentVoiceCSQName":"cssCsq"}]}]}][room_name=VoiceCSQDetailsStats*agent7901][socket_io_server_type=WSS][status=SENT]: Event Detail Trace
```

Exemple 3 : Thème de ResourceIAQStats

```
0000059446: 10.86.64.208: Jan 18 2017 22:27:00.011 -0500: %CCBU_pool-19-thread-1-6-MessageDispatcher:
%[client_count=2][message={"id":"agent7901","operation":"UPDATE","ResourceIAQStats":{"resourceId":"agent7901","resourceName":"agent7901","resourceState":2,"durationInStateMillis":148093,"nHandledContacts":0,"nPresentedContacts":0,"avgTalkDuration":0,"longestTalkDuration":0,"avgHoldDuration":0,"longestHoldDuration":0,"avgHandleDuration":0,"avgWorkDuration":0,"totalTalkTime":0,"totalHoldTime":0,"maxReadyTime":2547,"avgReadyTime":2547,"totalReadyTime":2547,"maxNotReadyTime":100760,"avgNotReadyTime":58701,"totalNotReadyTime":265495,"maxWorkTime":0,"totalWorkTime":0,"logonDuration":268042,"totalReservedTime":0,"otherTime":0,"avgSpeedOfAnswer":0,"rsrcCurrentStateReasonCode":0,"nOutboundOfferedContacts":0,"nOutboundActiveOfferedContacts":0,"nOutboundTimedOutContacts":0,"nOutboundVoiceContacts":0,"nOutboundAcceptedContacts":0,"nOutboundActiveVoiceContacts":0,"nOutboundActiveAcceptedContacts":0,"nOutboundRejectedContacts":0,"nOutboundClosedContacts":0,"avgOutboundTalkDuration":0,"activeOutboundTalkDuration":0,"avgOutboundHoldDuration":0,"activeOutboundHoldDuration":0,"longestOutboundTalkDuration":0,"longestOutboundHoldDuration":0,"totalOutboundTalkTime":0,"totalOutboundHoldTime":0,"avgOutboundWorkTime":0,"maxOutboundWorkTime":0,"totalOutboundWorkTime":0,"avgOutboundTalkWindow1":0,"avgOutboundHoldWindow1":0,"avgOutboundTalkWindow2":0}}
```

```
0, "avgOutboundHoldWindow2":0, "nOutboundHandledContacts":-
1, "avgTalkingWindow1":0, "avgHoldWindow1":0, "avgTalkingWindow2":0, "avgHoldWindow2":0, "strResource
State": "Not
Ready"}}][room_name=ResourceIAQStats*agent7901][socket_io_server_type=WSS][status=SENT]: Event
Detail Trace
```

Client Logs(1)

Client se connectant des expositions connectant, s'abonnant et recevant des événements.

1. Les clients basés sur le WEB se connectent au serveur comme WebSocket ou à de longs clients d'interrogation.
2. Une fois que connecté, ou en tant qu'élément de connecter ces clients abonnez-vous aux salles dans le serveur du socket E/S.
 - Plusieurs clients peuvent s'abonner à la même salle.
 - Chaque pièce seulement obtiendra/reçoit des événements de ce type
3. Les clients recevront 1 événement à la fois.

S'exécuter des états signifie s'abonner aux événements du serveur SIO.

Lancer l'état CUIIC déclenche une demande d'abonnement au thème correspondant à l'état au serveur de SocketIO.

Exemple1 : CUIIC envoyant la demande d'abonnement après avoir exécuté le compte rendu succinct de la Voix CSQ (thème VoiceIAQStats)

```
1/18/2017, 10:14:07 PM=INFOC8E2DB0C10000140000000A40A4E5E6B= Subscribing for rooms:
<VoiceIAQStats=cssCsq> report-layer.js:1414:11
1/18/2017, 10:14:07 PM=INFOC8E2DB0C10000140000000A40A4E5E6B= ReportManager : filterString
[{"fieldId":"720298FB10000140000000A10A4E5E6F", "fieldType":"VALUELIST", "name":"VoiceIAQStats.esd
Name", "operator":"SetValues", "value":[{"key":"cssCsq", "desc":"cssCsq"}], "valuelistId":"01FB2C011
0000133771FC3C33F57F543", "isKeyField":true}] report-layer.js:1414:11
1/18/2017, 10:14:07 PM=INFOC8E2DB0C10000140000000A40A4E5E6B= Snapshot Data :
[[{"id":"cssCsq", "operation":"UPDATE", "VoiceIAQStats":{"id":2, "esdId":2, "esdName":"cssCsq", "nRes
ourcesLoggedIn":1, "nTotalContacts":7, "nHandledContacts":7, "nAbandonedContacts":0, "nDequeuedConta
cts":0, "avgTalkDuration":503030, "longestTalkDuration":3225646, "avgWaitDuration":17959, "longestWa
itDuration":30383, "longestCurrentlyWaitingDuration":0, "nAvailResources":0, "nInSessionResources":
0, "nUnavailResources":1, "nWorkResources":0, "nSelectedResources":0, "nWaitingContacts":0, "nAverage
HandlingTimeLowThreshold":0.0, "nAverageHandlingTimeHighThreshold":0.0, "nAverageWaitingTimeLowThr
eshold":0.0, "nAverageWaitingTimeHighThreshold":0.0, "nAbandonedCallsLowThreshold":0, "nAbandonedCa
llsHighThreshold":0, "nDequeuedCallsLowThreshold":0, "nDequeuedCallsHighThreshold":0, "nSLAPercenta
geLowThreshold":0.0, "nSLAPercentageHighThreshold":0.0}]] report-layer.js:1414:11
```

Exemple2 : CUIIC envoyant la demande d'abonnement après avoir exécuté le rapport détaillé des agents de la Voix CSQ (thème VoiceCSQDetailsStats)

```
1/18/2017, 10:14:36 PM=INFO9A7A14A510000140000000EB0A4E5E6B= Subscribing for rooms:
<VoiceCSQDetailsStats=agent7901,agent7902,agent7903,agent7904,agent7905,agent7900> report-
layer.js:1414:11
1/18/2017, 10:14:36 PM=INFO9A7A14A510000140000000EB0A4E5E6B= ReportManager : filterString
[{"fieldId":"86F96290100001400000000C0A4E5E6B", "fieldType":"VALUELIST", "name":"VoiceCSQDetailsSt
ats.agentId", "operator":"SetValues", "value":[{"key":"agent7900", "desc":"agent7900"}, {"key":"agen
t7901", "desc":"agent7901"}, {"key":"agent7902", "desc":"agent7902"}, {"key":"agent7903", "desc":"age
nt7903"}, {"key":"agent7904", "desc":"agent7904"}, {"key":"agent7905", "desc":"agent7905"}], "valueli
stId":"39C80E70100001400000003660A4E5B1F", "isKeyField":true}] report-layer.js:1414:11
1/18/2017, 10:14:36 PM=INFO9A7A14A510000140000000EB0A4E5E6B= Snapshot Data :
[[{"id":"agent7900", "operation":"UPDATE", "VoiceCSQDetailsStats":{"agentId":"agent7900", "agentNam
e":"agent7900", "agentState":"Not
```

```
Ready", "skillGroup": "", "agentStateDuration": 2624186, "reasonCode": 32760, "AgentVoiceCSQNames": [{"a
gentVoiceCSQName": "cssCsq"}]]] report-layer.js:1414:11
```

Example3 : CUIC envoyant la demande d'abonnement après état courant de statistique de l'agent (thème : ResourceIAQStats)

```
1/18/2017, 10:12:42 PM=INFO67D4370710000140000001060A4E5E6B= Subscribing for rooms:
<ResourceIAQStats=agent7901,agent7902,agent7903,agent7904,agent7905,agent7900> report-
layer.js:1414:11
1/18/2017, 10:12:42 PM=INFO67D4370710000140000001060A4E5E6B= ReportManager : filterString
[{"fieldId": "67C4404410000140000000930A4E5E6B", "fieldType": "VALUELIST", "name": "ResourceIAQStats.
resourceId", "operator": "SetValues", "value": [{"key": "agent7900", "desc": "agent7900"}, {"key": "agent
7901", "desc": "agent7901"}, {"key": "agent7902", "desc": "agent7902"}, {"key": "agent7903", "desc": "agen
t7903"}, {"key": "agent7904", "desc": "agent7904"}, {"key": "agent7905", "desc": "agent7905"}], "valuelis
tId": "39C80E7010000140000003660A4E5B1F", "isKeyField": true}] report-layer.js:1414:11
1/18/2017, 10:12:42 PM=INFO67D4370710000140000001060A4E5E6B= Snapshot Data :
[[{"id": "agent7900", "operation": "UPDATE", "ResourceIAQStats": {"resourceId": "agent7900", "resourceN
ame": "agent7900", "resourceState": 2, "durationInStateMillis": 2512110, "nHandledContacts": 7, "nPresen
tedContacts": 7, "avgTalkDuration": 503030, "longestTalkDuration": 3225646, "avgHoldDuration": 0, "longe
stHoldDuration": 0, "avgHandleDuration": 0, "avgWorkDuration": 0, "totalTalkTime": 3521198, "totalHoldTi
me": 0, "maxReadyTime": 2578942, "avgReadyTime": 235602, "totalReadyTime": 5183249, "maxNotReadyTime": 53
24632, "avgNotReadyTime": 780731, "totalNotReadyTime": 15784543, "maxWorkTime": 0, "totalWorkTime": 0, "l
ogonDuration": 24526009, "totalReservedTime": 0, "otherTime": -
81, "avgSpeedOfAnswer": 5187, "rsrcCurrentStateReasonCode": 32760, "nOutboundOfferedContacts": 0, "nOut
boundActiveOfferedContacts": 0, "nOutboundTimedOutContacts": 0, "nOutboundVoiceContacts": 0, "nOutboun
dAcceptedContacts": 0, "nOutboundActiveVoiceContacts": 0, "nOutboundActiveAcceptedContacts": 0, "nOutb
oundRejectedContacts": 0, "nOutboundClosedContacts": 0, "avgOutboundTalkDuration": 0, "activeOutboundT
alkDuration": 0, "avgOutboundHoldDuration": 0, "activeOutboundHoldDuration": 0, "longestOutboundTalkDu
ration": 0, "longestOutboundHoldDuration": 0, "totalOutboundTalkTime": 0, "totalOutboundHoldTime": 0, "a
vgOutboundWorkTime": 0, "maxOutboundWorkTime": 0, "totalOutboundWorkTime": 0, "avgOutboundTalkWindow1"
: 0, "avgOutboundHoldWindow1": 0, "avgOutboundTalkWindow2": 0, "avgOutboundHoldWindow2": 0, "nOutboundHa
ndledContacts": -
1, "avgTalkingWindow1": 0, "avgHoldWindow1": 0, "avgTalkingWindow2": 0, "avgHoldWindow2": 0, "strResource
State": "Not Ready"}]]] report-layer.js:1414:11
```

Message received on

Le serveur de SocketIO a reçu la demande d'abonnement et JOINT la salle.

Logs de SocketIO

Example1 : Le serveur SIO SE JOIGNENT après réception de la demande d'abonnement (thème VoicelAQStats)

```
0000056387: 10.86.64.208: Jan 18 2017 22:13:59.895 -0500: %CCBU_nioEventLoopGroup-5-3-6-
MessageDispatcher: %[address=/10.86.248.157:62226(WEBSOCKET)][message=Subscribe-
{VoiceIAQStats=[cssCsq]}][socket_io_server_type=WSS]: Subscription Request
0000056388: 10.86.64.208: Jan 18 2017 22:13:59.895 -0500: %CCBU_nioEventLoopGroup-5-3-6-
MessageDispatcher:
 %[address=/10.86.248.157:62226][room_name=VoiceIAQStats*cssCsq][socket_io_server_type=WSS][statu
s=JOIN]: Room Detail Trace
```

Example2 : Le serveur SIO SE JOIGNENT après réception de la demande d'abonnement (thème VoiceCSQDetailsStats)

```
0000056539: 10.86.64.208: Jan 18 2017 22:14:29.047 -0500: %CCBU_nioEventLoopGroup-5-3-6-
MessageDispatcher: %[address=/10.86.248.157:62226(WEBSOCKET)][message=Subscribe-
{VoiceCSQDetailsStats=[agent7901, agent7902, agent7903, agent7904, agent7905,
agent7900]}][socket_io_server_type=WSS]: Subscription Request
0000056540: 10.86.64.208: Jan 18 2017 22:14:29.047 -0500: %CCBU_nioEventLoopGroup-5-3-6-
MessageDispatcher:
 %[address=/10.86.248.157:62226][room_name=VoiceCSQDetailsStats*agent7901][socket_io_server_type=
WSS][status=JOIN]: Room Detail Trace
0000056541: 10.86.64.208: Jan 18 2017 22:14:29.047 -0500: %CCBU_nioEventLoopGroup-5-3-6-
```

```

MessageDispatcher:
%[address=/10.86.248.157:62226][room_name=VoiceCSQDetailsStats*agent7902][socket_io_server_type=WSS][status=JOIN]: Room Detail Trace
0000056542: 10.86.64.208: Jan 18 2017 22:14:29.047 -0500: %CCBU_nioEventLoopGroup-5-3-6-
MessageDispatcher:
%[address=/10.86.248.157:62226][room_name=VoiceCSQDetailsStats*agent7903][socket_io_server_type=WSS][status=JOIN]: Room Detail Trace
0000056543: 10.86.64.208: Jan 18 2017 22:14:29.048 -0500: %CCBU_nioEventLoopGroup-5-3-6-
MessageDispatcher:
%[address=/10.86.248.157:62226][room_name=VoiceCSQDetailsStats*agent7904][socket_io_server_type=WSS][status=JOIN]: Room Detail Trace
0000056544: 10.86.64.208: Jan 18 2017 22:14:29.048 -0500: %CCBU_nioEventLoopGroup-5-3-6-
MessageDispatcher:
%[address=/10.86.248.157:62226][room_name=VoiceCSQDetailsStats*agent7905][socket_io_server_type=WSS][status=JOIN]: Room Detail Trace
0000056545: 10.86.64.208: Jan 18 2017 22:14:29.048 -0500: %CCBU_nioEventLoopGroup-5-3-6-
MessageDispatcher:
%[address=/10.86.248.157:62226][room_name=VoiceCSQDetailsStats*agent7900][socket_io_server_type=WSS][status=JOIN]: Room Detail Trace

```

Example3 : Le serveur SIO SE JOIGNENT après réception de la demande d'abonnement (thème ResourceIAQStats)

```

0000055973: 10.86.64.208: Jan 18 2017 22:12:34.849 -0500: %CCBU_nioEventLoopGroup-5-3-6-
MessageDispatcher: %[address=/10.86.248.157:62226(WEB_SOCKET)][message=Subscribe-
{ResourceIAQStats=[agent7901, agent7902, agent7903, agent7904, agent7905,
agent7900]}][socket_io_server_type=WSS]: Subscription Request
0000055974: 10.86.64.208: Jan 18 2017 22:12:34.850 -0500: %CCBU_nioEventLoopGroup-5-3-6-
MessageDispatcher:
%[address=/10.86.248.157:62226][room_name=ResourceIAQStats*agent7901][socket_io_server_type=WSS]
[status=JOIN]: Room Detail Trace
0000055975: 10.86.64.208: Jan 18 2017 22:12:34.850 -0500: %CCBU_nioEventLoopGroup-5-3-6-
MessageDispatcher:
%[address=/10.86.248.157:62226][room_name=ResourceIAQStats*agent7902][socket_io_server_type=WSS]
[status=JOIN]: Room Detail Trace
0000055976: 10.86.64.208: Jan 18 2017 22:12:34.850 -0500: %CCBU_nioEventLoopGroup-5-3-6-
MessageDispatcher:
%[address=/10.86.248.157:62226][room_name=ResourceIAQStats*agent7903][socket_io_server_type=WSS]
[status=JOIN]: Room Detail Trace
0000055977: 10.86.64.208: Jan 18 2017 22:12:34.850 -0500: %CCBU_nioEventLoopGroup-5-3-6-
MessageDispatcher:
%[address=/10.86.248.157:62226][room_name=ResourceIAQStats*agent7904][socket_io_server_type=WSS]
[status=JOIN]: Room Detail Trace
0000055978: 10.86.64.208: Jan 18 2017 22:12:34.850 -0500: %CCBU_nioEventLoopGroup-5-3-6-
MessageDispatcher:
%[address=/10.86.248.157:62226][room_name=ResourceIAQStats*agent7905][socket_io_server_type=WSS]
[status=JOIN]: Room Detail Trace
0000055979: 10.86.64.208: Jan 18 2017 22:12:34.850 -0500: %CCBU_nioEventLoopGroup-5-3-6-
MessageDispatcher:
%[address=/10.86.248.157:62226][room_name=ResourceIAQStats*agent7900][socket_io_server_type=WSS]
[status=JOIN]: Room Detail Trace

```

Logs de client (2)

Serveur de SocketIO après réception de la demande d'abonnement, débuts envoyant les événements par le répartiteur de message. Sont ci-dessous les mises à jour d'événement qui sont reçues sur le navigateur.

Log de navigateur : La mise à jour est reçue sur le navigateur (presse F12 sur le navigateur pour voir la mise à jour)

Example1 : Le navigateur CUIC reçoit les messages sur le navigateur (thème VoicelAQStats)

Message received on

```
socket::{"id":"cssCsq","operation":"UPDATE","VoiceIAQStats":{"id":2,"esdId":2,"esdName":"cssCsq",
,"nResourcesLoggedIn":1,"nTotalContacts":8,"nHandledContacts":8,"nAbandonedContacts":0,"nDequeue
dContacts":0,"avgTalkDuration":503030,"longestTalkDuration":3225646,"avgWaitDuration":17959,"lon
gestWaitDuration":30383,"longestCurrentlyWaitingDuration":0,"nAvailResources":0,"nInSessionResou
rces":1,"nUnavailResources":0,"nWorkResources":0,"nSelectedResources":0,"nWaitingContacts":0,"nA
verageHandlingTimeLowThreshold":1556,"nAverageHandlingTimeHighThreshold":1556,"nAverageWaitingTi
meLowThreshold":32836,"nAverageWaitingTimeHighThreshold":32836,"nAbandonedCallsLowThreshold":0,"
nAbandonedCallsHighThreshold":0,"nDequeuedCallsLowThreshold":0,"nDequeuedCallsHighThreshold":0,"
nSLAPercentageLowThreshold":0,"nSLAPercentageHighThreshold":0}} socketlib.js:510:3
```

Example2 : Le navigateur CUIC reçoit les messages sur le navigateur (thème VoiceCSQDetailsStats)

Message received on

```
socket::{"id":"agent7901","operation":"UPDATE","VoiceCSQDetailsStats":{"agentId":"agent7901","ag
entName":"agent7901","agentState":"Talking","skillGroup":"cssCsq","agentStateDuration":4538,"rea
sonCode":0,"AgentVoiceCSQNames":[{"agentVoiceCSQName":"cssCsq"}]}} socketlib.js:510:3
```

Example2 : Le navigateur CUIC reçoit les messages sur le navigateur (thème ResourceIAQStats)

Message received on

```
socket::{"id":"agent7901","operation":"UPDATE","ResourceIAQStats":{"resourceId":"agent7901","res
ourceName":"agent7901","resourceState":2,"durationInStateMillis":148093,"nHandledContacts":0,"nP
resentedContacts":0,"avgTalkDuration":0,"longestTalkDuration":0,"avgHoldDuration":0,"longestHold
Duration":0,"avgHandleDuration":0,"avgWorkDuration":0,"totalTalkTime":0,"totalHoldTime":0,"maxRe
adyTime":2547,"avgReadyTime":2547,"totalReadyTime":2547,"maxNotReadyTime":100760,"avgNotReadyTim
e":58701,"totalNotReadyTime":265495,"maxWorkTime":0,"totalWorkTime":0,"logonDuration":268042,"to
talReservedTime":0,"otherTime":0,"avgSpeedOfAnswer":0,"rsrcCurrentStateReasonCode":0,"nOutboundO
fferedContacts":0,"nOutboundActiveOfferedContacts":0,"nOutboundTimedOutContacts":0,"nOutboundVoi
ceContacts":0,"nOutboundAcceptedContacts":0,"nOutboundActiveVoiceContacts":0,"nOutboundActiveAcc
eptedContacts":0,"nOutboundRejectedContacts":0,"nOutboundClosedContacts":0,"avgOutboundTalkDurat
ion":0,"activeOutboundTalkDuration":0,"avgOutboundHoldDuration":0,"activeOutboundHoldDuration":0
,"longestOutboundTalkDuration":0,"longestOutboundHoldDuration":0,"totalOutboundTalkTime":0,"tota
lOutboundHoldTime":0,"avgOutboundWorkTime":0,"maxOutboundWorkTime":0,"totalOutboundWorkTime":0,"
avgOutboundTalkWindow1":0,"avgOutboundHoldWindow1":0,"avgOutboundTalkWindow2":0,"avgOutboundHold
Window2":0,"nOutboundHandledContacts":-
1,"avgTalkingWindow1":0,"avgHoldWindow1":0,"avgTalkingWindow2":0,"avgHoldWindow2":0,"strResource
State":"Not Ready"}} socketlib.js:510:3
```

La finesse reçoit la même mise à jour sur l'appareil de bureau d'agent/superviseur. (Presse F12 sur le navigateur pour voir les mises à jour)

Example1 :

Message received on

```
socket::{"id":"agent7901","operation":"UPDATE","ResourceIAQStats":{"resourceId":"agent7901","res
ourceName":"agent7901","resourceState":2,"durationInStateMillis":148093,"nHandledContacts":0,"nP
resentedContacts":0,"avgTalkDuration":0,"longestTalkDuration":0,"avgHoldDuration":0,"longestHold
Duration":0,"avgHandleDuration":0,"avgWorkDuration":0,"totalTalkTime":0,"totalHoldTime":0,"maxRe
adyTime":2547,"avgReadyTime":2547,"totalReadyTime":2547,"maxNotReadyTime":100760,"avgNotReadyTim
e":58701,"totalNotReadyTime":265495,"maxWorkTime":0,"totalWorkTime":0,"logonDuration":268042,"to
talReservedTime":0,"otherTime":0,"avgSpeedOfAnswer":0,"rsrcCurrentStateReasonCode":0,"nOutboundO
fferedContacts":0,"nOutboundActiveOfferedContacts":0,"nOutboundTimedOutContacts":0,"nOutboundVoi
ceContacts":0,"nOutboundAcceptedContacts":0,"nOutboundActiveVoiceContacts":0,"nOutboundActiveAcc
```

pectedContacts":0,"nOutboundRejectedContacts":0,"nOutboundClosedContacts":0,"avgOutboundTalkDuration":0,"activeOutboundTalkDuration":0,"avgOutboundHoldDuration":0,"activeOutboundHoldDuration":0,"longestOutboundTalkDuration":0,"longestOutboundHoldDuration":0,"totalOutboundTalkTime":0,"totalOutboundHoldTime":0,"avgOutboundWorkTime":0,"maxOutboundWorkTime":0,"totalOutboundWorkTime":0,"avgOutboundTalkWindow1":0,"avgOutboundHoldWindow1":0,"avgOutboundTalkWindow2":0,"avgOutboundHoldWindow2":0,"nOutboundHandledContacts":-1,"avgTalkingWindow1":0,"avgHoldWindow1":0,"avgTalkingWindow2":0,"avgHoldWindow2":0,"strResourceState":"Not Ready"}} concat:7930:3