

了解和排除故障Live数据报告关于UCCX

目录

[简介](#)

[在Live数据上的变化与SocketIO](#)

[技术概要](#)

[WebSocket与长轮询机制](#)

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

[长?](#)

[运行长?的客户端支票号码](#)

[Live数据流](#)

[SocketIO服务器房间&主题](#)

[制造商](#)

[调度程序](#)

[客户端](#)

[验证](#)

[Live数据跟踪](#)

[排除故障Live数据](#)

[基本Healthcheck](#)

[摘要](#)

[验证浏览器版本](#)

[验证CPU和内存使用](#)

[验证Live数据流来源联机](#)

[验证NTP。](#)

[验证DNS](#)

[运行诊断测试](#)

[验证主机名](#)

[验证VM配置](#)

[验证UCCX证书](#)

[验证支持的大小配置](#)

[已知问题说明](#)

[Live数据日志跟踪](#)

[引擎日志](#)

[SocketIO日志\(1\)](#)

[SocketIO日志\(2\)](#)

[客户端Logs\(1\)](#)

[SocketIO日志](#)

[客户端日志\(2\)](#)

简介

本文描述Live排除故障检查点的数据报告在Unified Contact Center Express (UCCX)解决方案的从版本11.x开始和概述与支持的日志分析一起。条款也概述与他们的应急方案的多种已知问题说明。

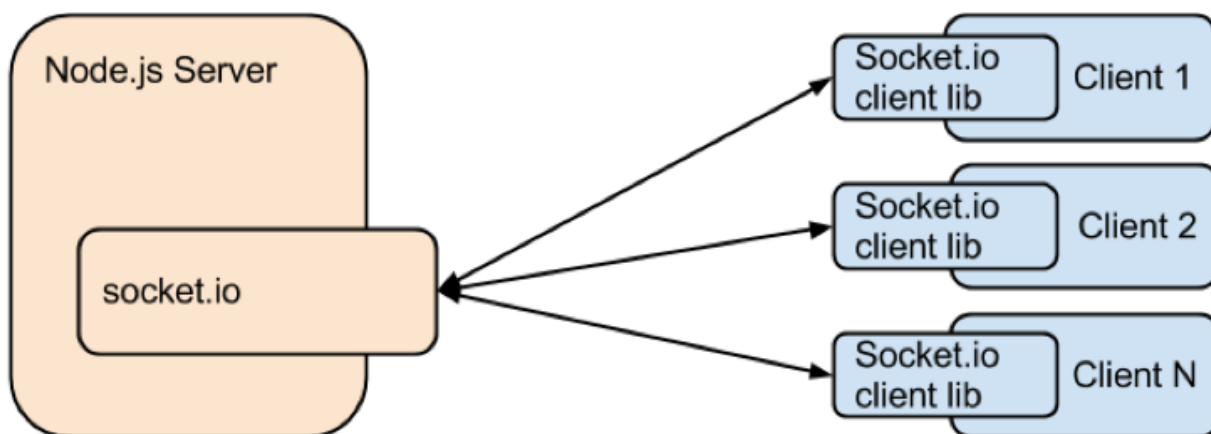
在Live数据上的变化与SocketIO

SocketIO根据Live数据设计替换开火XMPP服务器作为向前报告从UCCX 11.x的默认消息基础设施它是根据Node.js impementation的一事件驱动的可扩展和高效解决方案，与一笔减少的通信开销通过使用WebSocket传送实时信息的协议请求注释6455 (RFC 6455)对浏览器(客户端)。

及早在UCCX 10.x，有单个**思科CCX通知服务**，根据开火使用两精良、聊天、电子邮件通知以及Live数据报告关于精良代理程序小配件和CUIC的XMPP实施)。

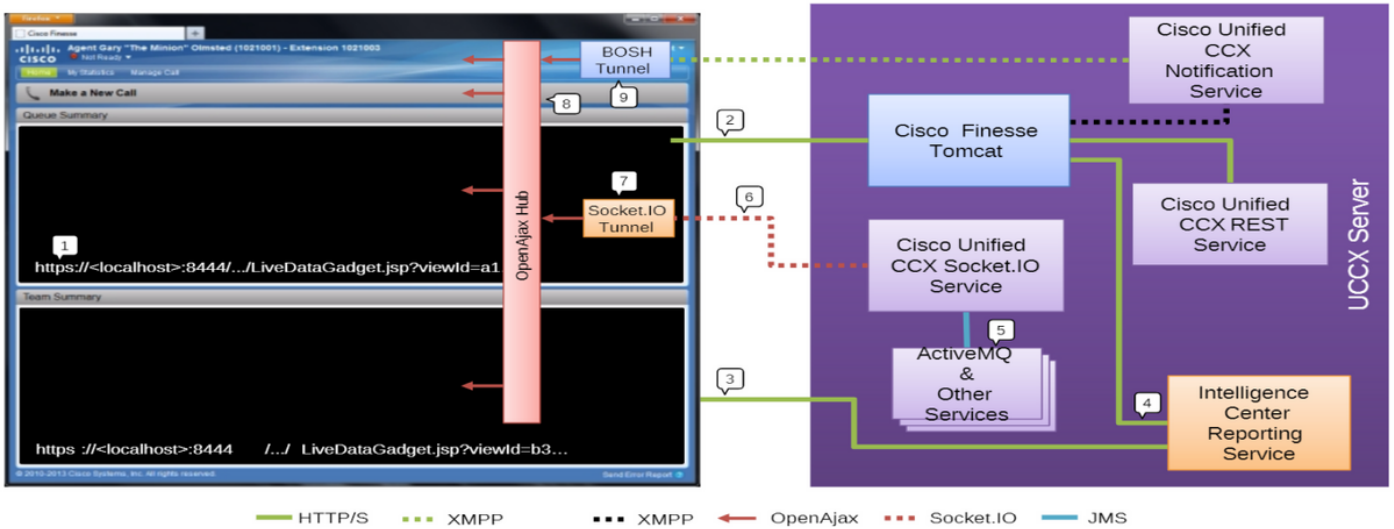
从与SocketIO的介绍的UCCX 11.x，Live现场报告的-精良客户聊天和电子邮件继续使用**思科CCX通知服务**，即开火作为XMPP服务器，但是报告模块的Live数据使用新的CiscoUnified **CCX Socket.IO服务服务**。

这减少报告开销的Live数据，并且使SocketIO独自地负责对草拟和调度Live数据更新对客户端浏览器，如SocketIO服务器所显示镜像。



技术概要

此镜像是SocketIO根据Live数据架构概述。



从一个高层次方面，数据为Live数据报告在Javascript对象符号(JSON)发送从UCCX引擎的实时数据管理器(RTDM)模块，通过Java信息服务(JMS)总线(即使用Apache实现此的ActiveMQ，活动留言队列)，到在UCCX主机的Socket.IO服务器(显示通过Unified CCX Socket.IO服务)然后获取此信息，处理(消耗)他们到屋子然后分配(分派)此给使用在TCP的客户端HTTPS WebSocket连接。

ActiveMQ是实现编程Interface(API)的JMS应用程序的中间件，并且使用发送在两个组件、UCCX Live数据(UCCX引擎)和SocketIO服务器之间的消息。

WebSocket与长轮询机制

使用WebSocket(WS)/WebSocket Secure(WSS)技术，通过使用长轮询机制，有浏览器能订阅接收Live数据更新的两种方式，或者。

WebSocket (RFC 6455)

- WebSocket协议(RFC 6455)是一份独立基于TCP的协议。其唯一的对HTTP的关系是其握手由HTTP服务器解释作为Upgrade请求和使用建立一持久连接用连续更新的服务器。
- 在成功的WebSocket握手以后，客户端和服务在此规格(JSON对象)反复传递数据指的更加小的概念性单元消息。因为一个被分段的消息可能由中间，联合或拆分WebSocket消息不一定对应于特定网络网络层帧。
- 此进程从发送一个正常HTTP请求的客户端开始对服务器。
- 升级报头在通知服务器的此请求包括客户端希望建立WebSocket连接。即HTTP/1.1101交换协议升级：**websocket连接：升级**。
- 在那以后，打开屋子，并且消息不断地被调度到客户端/浏览器。
- WebSocket知道减少处理在发布服务器的开销，并且提供安全，并且动态居住更新。

长？

- 它是传统?技术的变化并且允许信息推送的仿真从服务器的给客户端。

- 使用长?，客户端要求从服务器的信息在此类似对一正常投票。连接保持开放在网络客户端和Web服务器之间，以便，当服务器有最新信息时能推送它对客户端，并且连接然后关闭。
- 如果服务器没有客户端的任何有用的资料，而不是发送一空答复，服务器保持请求并且等待一些信息是可用的。
- 一旦信息变得可用(或在一适当的超时以后)，一完整答复被发送给客户端。
- 客户端通常然后立即再请求从服务器的信息，因此服务器总是有能使用提供数据以回应事件的一可用的等待的请求。
- 长?导致在请求对于此信息的服务器的另外的开销。

运行长?的客户端支票号码

- 对于UCCX Live数据，**WebSocket**在传统龙牌?机制**是有利和推荐的**。
- 要检查客户端是否使用WebSocket或长轮询机制实际数据，您能分析SocketIO调试和看到消息。

步骤：

- 出现Socket IO调试成水平调试(从默认请警告) -导航对**UCCX维护性> Trace**。
- 盯梢最新的SocketIO服务器日志：

您看见此：

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): ...

Live数据流

Live数据流可以汇总如下：**CCX引擎(ICD_RTDM, SS_RMCM) → JMS ActiveMQ → SocketIO服务器→客户端小配件(Web浏览器)**

这些是在UCCX引擎需要启用的日志设备：

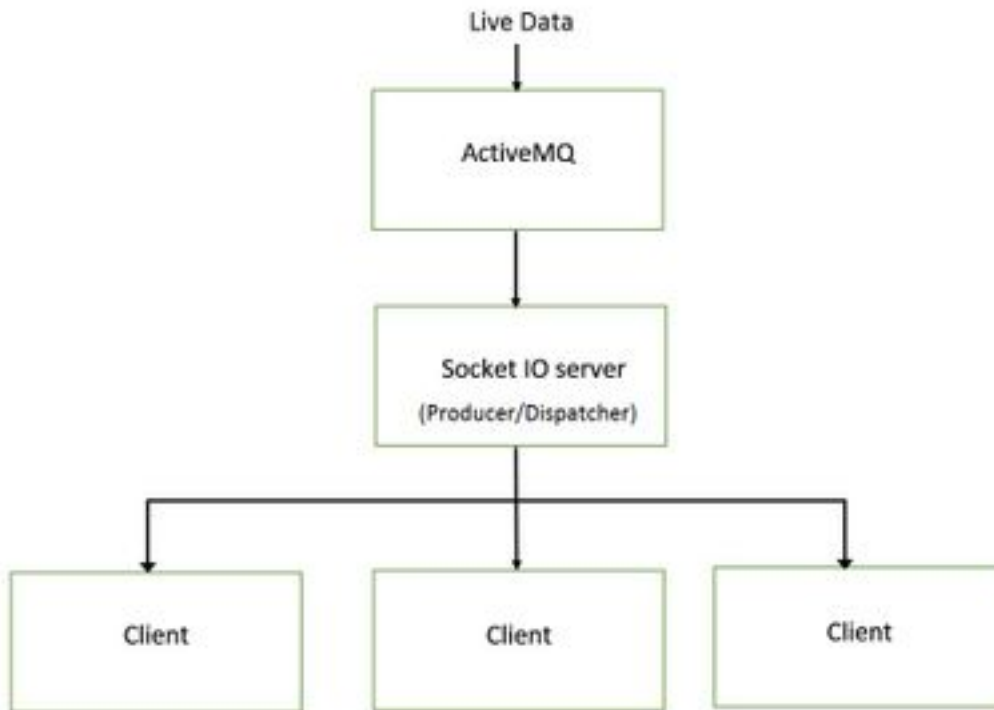
ICD_RTDM =智能呼叫分配的实时数据管理器

SS_RMCM =子系统资源管理器联络管理器软件

JMS ActiveMQ = Java通信业务活动消息传送队列实施

Live数据SocketIO服务器主要设计获得从实际数据源的信息(例如JMS)和通过WebSocket发布它，居住数据客户端(类似在精良或Cisco Unified智能中心(CUIC) Web应用程序的一个客户端小配件。

这些逻辑概要描述从UCCX引擎(Live数据)的数据流到ActiveMQ里，Socket.IO服务器然后获取信息并且分配它给客户端，如SocketIO所显示镜像根据客户端订阅：



Live数据通过ActiveMQ发布在批的事件在主题。

SocketIO服务器订阅对这些主题并且接受事件批。

SocketIO服务器房间&主题

SocketIO服务器配置与在/opt/cisco/uccx/socketioserver/conf/socketioservice.properties文件或主题定义的多房间。

所有Live数据报告被映射对JMS主题，是有用的，当解析日志时。

JMS主题更新在每3秒发生，肯定报告这些是基于的事件(更新被张贴，只有当事件被触发时，类似代理程序状态更改)。

他们在报告里表列出对JMS主题映射的：

报表名称	JMS主题名称
代理程序CSQ统计报告	AgentCSQStats --->事件根据主题
座席状态日志报告	AgentStateDetailStats --->事件根据主题

座席统计报告	ResourceIAQStats
座席小组汇总报告	ResourceIAQStats
团队状态报告	ResourceIAQStats
团队汇总报告	ResourceIAQStats
语音CSQ详细报表	VoiceCSQDetailsStats
语音 CSQ – 汇总	VoiceIAQStats
聊天CSQ汇总报告	ChatQueueStatistics --->事件根据主题
聊天代理程序统计报告	ChatAgentStats --->事件根据主题
电子邮件代理程序统计报告	EmailAgentStats --->事件根据主题
电子邮件CSQ汇总报告	EmailQueueStatistics --->事件根据主题

客户端类似CUIC服务器连接到Socket.IO服务器订阅到以上所列的屋子为了接收从SocketIO服务器的生活数据事件。

当报告从CUIC，很快启动订阅请求从CUIC发送。例如启动语音关于CUIC的CSQ汇总报告触发订阅请求对对应的屋子/主题(VoiceIAQStats)到SIO服务器。

制作商

1. 接受从ActiveMQ的事件批。
2. 分成他们下来他们的离散事件。
3. 在获取事件的id的每个事件的偷看。
4. 路由事件到根据主题的屋子一起与获取的id。

调度程序

调度程序接收名为从制作商的事件的分离屋子并且发送这些事件给订阅到该空间的所有客户端。

客户端

1. 基于Web的客户端(CUIC或精良代理程序/Supervisor小配件)连接到在Web socket的SocketIO服务器。
2. 一旦连接，这些客户端订阅到Socket IO服务器的房间。
3. 几个客户端可能订阅到同一间屋子。
4. 客户端每次接收一个事件。

示例：

```
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
```

SocketIO服务器然后开始发布事件给客户端通过此订阅。

验证

Socket.IO服务器需要在发送数据前验证连接的客户端(类似精良Live数据小配件或CUIC)。

1. Socket.IO服务器报价使用标记基于验证。
2. Live数据小配件执行递交他们一验证标记的最初的认证请求。
3. 这进一步合格打开对验证WSS (WebSocket服务器)连接的创建的Socket.IO服务器。对于此，称API UCCX带有Socket.IO从小配件获得的标记。
4. 使用一个令牌的字符串，所有验证执行。客户端验证在Socket.IO服务中启用通过Client.Authentication.Rest.URL属性。
5. 启用的所有连接客户端必须提供此字符串作为参数在连接URL。
6. Socket.IO服务器的客户端连接，订阅，取消预订，并且断开。初始连接能是连接或也包括订阅。

```
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
```

ws = websocket

安全wss =的websocket

包含WSS的keystore (UCCX相关的证书使用默认：思科Tomcat应用程序证书)

密码短语的位置WSS的(默认：系统的Tomcat)

Live数据跟踪

对于记录，有在UCCX Live数据的3关键流点使用在11.x (引擎MIVR_RTDM的SocketIO体系结构，引擎SS_RMCM > JMS ActiveMQ > SocketIO服务器)

- UCCX引擎记录(与RTDM的MIVR和RMCM设置为Xdebugging5)发送消息对JMS (如在RM子系

统调试中看到的每3秒在引擎(MIVR) SS_RMCM)

- SocketIO日志(SocketIO必须设置服务记录日志为JMS传送信息给SocketIO用户，收到引擎发布的所有消息)的全双工调试。
- 浏览器控制台日志(F12)：处理流入的更新，创建每个主题的适当的房间，然后发送这些的SocketIO对客户端Web浏览器通过消息调度程序。(这些在Web浏览器Console/F12日志被看到)。

这些跟踪必须启用为了发现在日志messaging的详细。

1. UCCX引擎：ICD_RTDM，与Xdebugging级别5的SS_RMCM
2. SocketIO服务记录日志：调试级别。
3. 浏览器检查流入Live数据更新的控制台日志(F12)。

SocketIO日志只告诉主题状况类似已连接/的调试断开了并且订阅/取消预订。

详细的跟踪必须启用为了看到在主题的特定事件。SocketIO制作商和调度程序日志是可用的，在启用详细的跟踪之后(启用详细的跟踪的联系方式Cisco TAC)。

排除故障Live数据

基本Healthcheck

往故障排除Live数据问题的第一步将设立支持的配置基准和执行系统的一个基本健康检查。

在此配置检查点找到的问题能的Live数据转换导致问题在精良小配件或在CUIC网页。

您应该在继续进行排除所有这些点日志基于故障排除前。

摘要

- 浏览器版本应该兼容。
- CPU和内存使用在UCCX必须适当地是。
- Live数据数据来源一定是联机在CUIC数据来源页。
- NTP和DNS在UCCX一定是正在运行的，与同步的NTP和DNS转发/反向查找按正确顺序。
- 必须正确提供VM卵与vRAM和vCPU需求根据思科卵指南。
- NIC在使用中VM的应该是VMXNET3。

验证浏览器版本

验证Live数据行为在不同的浏览器和浏览器版本间的。

UCCX当前支持开始11.5的以下浏览器版本

: [http://docwiki.cisco.com/wiki/Unified_CCX_Software_Compatibility_Matrix_for_11.5\(1\)#Supporte](http://docwiki.cisco.com/wiki/Unified_CCX_Software_Compatibility_Matrix_for_11.5(1)#Supporte)

[d Browsers](#)

因为那不为Live数据，支持请切记禁用在IE的兼容模式。当在长如此轮询此是一重要的考虑因素时，IE下跌回到长轮询机制。

验证CPU和内存使用

使用RTMT或CLI命令，检查CPU使用情况和内存使用在服务器。

即：RTMT >

CLI：show process负载cpu

CLI：show process负载内存

请检查这些进程，如果在CUIC的查看的Live数据：UCCX_Engine，socketioservice，cuicreporting。

验证Live数据流来源联机

为了验证Live数据流来源联机，请导航对CUIC网页>System >数据来源

验证NTP。

验证NTP可访问，并且UCCX同步对在层<=5的Ntp server

CLI：使用情况ntp状态

```
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
```

同步对Ntp server (xx.xx.xx.xx)在层2

验证DNS

验证DNS解决。

CLI：使用情况网络主机<hostname>

本地解决方法：

hostname.domainname解决本地对xx.xx.xx.xx

外部解决方法：

hostname.domainname有地址xx.xx.xx.xx

运行诊断测试

您能从CLI运行诊断测试识别其他潜在问题用服务器(即Tomcat延迟、网络问题等等)

CLI : 使用情况诊断测验

验证主机名

UCCX主机名必须我用小写。

请参考安装/升级指南和此缺陷。

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

验证VM配置

验证正确开放虚拟化设备(卵)模板根据这些链路使用 :

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>

UCCX VM网络适配器(NIC)设置必须是VMXNET3。

如果正确适配器那么没有选择UCCX不发布实时数据在平稳对估计。

如果VM网络适配器设置不是然后VMXNET3跟随此链路更正适配器设置

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

验证UCCX证书

证书不匹配或 , 当证书没有接受时能造成实际数据流来源是断开/脱机。

- 接受证书 , 当提示。
- 保证第三方证书没有超时。
- 验证在证书的主机名正确。否则 , 请遵从下面的步骤。

在CCX侧 :

1. 编辑此文件/usr/local/platform/conf/platformConfig.xml并且更正主机名并且确保DNS条目是根据在此文件的名称。
2. 使用CLI命令 , 在验证在UCCX集群的上述详细信息以后删除并且重新生成证书 :

CLI : 设置cert regen Tomcat

3. 重新启动在UCCX的Cisco Tomcat服务。

CLI : utils service restart Cisco Tomcat

更多在证书 :

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

验证支持的大小配置

保证最大配置限额被遵守在设置。对于此，为最大服务器配置和限额请参考UCCX解决方案级别设计指南：

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...

要确认此您能运行这些CLI命令，验证大小时指南，在设置获得确切的数字。

需要的INFO

托塔尔不Supervisor分配到团队

共计不已配置的Supervisor
共计不已配置的代理程序
共计不已配置的团队

不代理程序每个团队

不Supervisor每个团队

用户编号由角色(CUIC)的
已配置的CSQ编号

查询

运行uccx SQL db_cra挑选计数(*)从(从激活='t')的Supervisor的挑选明显
resourceloginid

从资源运行(*)的uccx SQL db_cra挑选计数active='t'和resourcetype=2

从资源运行(*)的uccx SQL db_cra挑选计数active='t'和resourcetype=1

从团队运行(*)的uccx SQL db_cra挑选计数active='t

从团队在resource.active='t'和team.active='t'组由teamname负责的

team.teamid=resource.assignedTeamID的内联资源uccx SQL db_cra挑

team.teamname, count(resource.resourceid) AS NumOfAgents

从Supervisor在Supervisor.active='t'和team.active='t'组由teamName负责

Supervisor.managedTeamID=team.teamid的内联团队uccx SQL db_cra

team.teamName, count(Supervisor.recordID) AS NumOfSupervisors

运行SQL挑选g.name作为组名，从cuic_data计数(*)和groupsize : cuicg,

cuic_data : cuicgroupmember gm由g.name的g.id = gm.groupid组

从contactservicequeue运行(*)的uccx SQL db_cra挑选计数active='t

已知问题说明

[CSCva13838](#)：在CUIC的Live数据延迟

[CSCus17605](#)：迪莱由于额外的报告的权限对Supervisor (高负载)

[CSCux33949](#)：Live数据问题由于增强版许可证。

[CSCvb67761](#)：导致高CPU的座席状态日志报告影响Live数据。

[CSCva95411](#)：SocketIO服务高CPU由于使用额外的堆内存的Namespace对象。

[CSCvb75279](#)：脱机在对不同步MIVR Keystore密码期间的Live数据流来源

[CSCvc45189](#)：Live数据流来源脱机由于备份与恢复操作

[CSCut04158](#)：Live数据问题由于不正确NIC适配器

注意：要确认或应急方案上述警告，通过打开服务请求请介入Cisco TAC。

Live数据记录跟踪

引擎日志

- Live数据通过激活MQ发布在批的事件在主题。
- SocketIO服务器订阅对这些主题并且接受事件批。

MIVR日志：Live数据从引擎发送通过在主题的一个JMS消息(即：等等VoiceCSQDetailsStats、VoiceIAQStats，ResourceIAQStats)

对于根据的事件主题(ChatAgentStats、ChatQueueStatistics AgentCSQStats，AgentStateDetailStats)引擎在每3秒之后传送空JMS信息保持SocketIO屋子运行。

这些跟踪在启用这些跟踪之后areseen：[SocketIO选派了跟踪](#)

```
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
```

SocketIO日志(1)

SocketIO接受从激活MQ的事件批

当批事件到达，SocketIO服务器执行此：

- 分成他们下来他们的离散事件。
- 在获取事件的id的每个事件的偷看。
- 路由事件到根据主题的屋子一起与获取的id

与空间名称的离散事件在调度程序排队

注意：有JMS监听程序/用户的SocketIO制造商从在主题JMS总线得到数据通过ActiveMQ并且发送它到排队给调度程序的SocketIO屋子。

这些跟踪在启用在SocketIO (启用同样的联系方式Cisco TAC的详细跟踪之后被看到)。

示例 1：VoiceIAQStats主题。

```
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": "01FB2C0110000133771FC3C33F57F543", "isKeyField": true}] report-layer.js:1414:11
```

示例 2 : VoiceCSQDetailsStats主题。

```
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
```

示例 3 : ResourceIAQStats主题

```
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
```

SocketIO日志(2)

Socket IO调度程序：从制作商的获得事件和传送给客户端

调度程序接收名为从制作商的事件的分离屋子并且发送这些事件给订阅到该空间的所有客户端。

这些跟踪在启用在SocketIO (启用同样的联系方式Cisco TAC的详细跟踪之后被看到)。

示例 1 : VoicelAQStats主题。

```
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
```

示例 2 : VoiceCSQDetailsStats主题。

```
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
```

示例 3 : ResourceIAQStats主题

```
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
```

客户端Logs(1)

客户端记录日志显示连接，订阅和接收事件。

1. 基于Web的客户端连接到服务器作为WebSocket或长?客户端。
2. 一旦连接，或者作为连接这些客户端一部分请订阅到Socket IO服务器的房间。
 - 几个客户端可能订阅到同一间屋子。
 - 每个空间只将获得该类型/receive事件
3. 客户端每次将接收1个事件。

运行报告含义订阅对从SIO服务器的事件。

启动CUIC报告触发订阅请求对主题与报告相应到SocketIO服务器。

示例1：发送订阅请求的CUIC在送语音CSQ汇总报告以后(主题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
```

示例2：发送订阅请求的CUIC在送语音CSQ代理程序详细报表以后(主题VoiceCSQDetailsStats)

```
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
```

示例3：发送订阅请求的CUIC在送代理程序统计报告以后(主题：ResourceIAQStats)

```
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
```

SocketIO服务器接收订阅请求并且加入屋子。

SocketIO日志

示例1：SIO在接收订阅请求以后(主题VoiceIAQStats)的服务器加入

```
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
```

示例2 : SIO在接收订阅请求以后(主题VoiceCSQDetailsStats)的服务器加入

```
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
```

示例3 : SIO在接收订阅请求以后(主题ResourceIAQStats)的服务器加入

```
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
```

客户端日志(2)

在接收订阅请求以后的SocketIO服务器，通过消息调度程序开始发送事件。下面在浏览器接收的事件更新。

浏览器日志：更新在浏览器接收(请在浏览器的F12发现更新)

示例1 : CUIC浏览器收到在浏览器(主题VoiceIAQStats)的消息

```
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
```

示例2 : CUIC浏览器收到在浏览器(主题VoiceCSQDetailsStats)的消息

```
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
```

示例2 : CUIC浏览器收到在浏览器(主题ResourceIAQStats)的消息

```
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
```

精良接收在代理程序/Supervisor Desktop的同一次更新。(请按在浏览器的F12发现更新)

示例1 :

```
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
```