

Finesse Agent/Supervisor Queue/SkillGroup Stats Update Problem.

Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Components Used](#)

[Conventions](#)

[Background Information](#)

[Problem Symptoms:](#)

[Finesse CTI Messaging and Finesse Queue Stats Buffer](#)

[Possible Causes for Finesse Queue Stats Buffer Overrun](#)

[Relevant Finesse Trace](#)

[Log Analyses:](#)

[Workaround:](#)

Introduction

The document describes the troubleshooting method for problem identification on Queue or Skillgroup stats update issues observed in Finesse agent desktop environment, specifically caused by message delays between the Computer-Telephony integration (CTI) servers and Finesse servers. The article provides log analyses, and it concludes with a workaround to improve Finesse server capability in handling these Stats Update messages in a sub-optimal delayed network.

Contributed by Tian Lei Xia, Cisco TAC Engineer.

Prerequisites

Requirements

Cisco recommends you to have the prior knowledge of these topics

- Cisco UCCE CTI server
- Cisco Finesse server

Components Used

UCCE Agent Peripheral Gateway with CTI Server (CTISVR) installed.

Finesse Server Cluster.

Conventions

Refer to the [Cisco Technical Tips Conventions](#) for information on document conventions.

Background Information

Finesse servers which subscribe to CTI SEVER as CTI clients, provide agent functions which are similar to what a Computer Telephony Integration Object Server (CTIOS) and a Cisco Agent Desktop (CAD) can offer. Finesse agents/supervisors can also experience some of the problems faced by CTIOS and CAD agents.

One of such problems is Queue or Skillgroup real time stats not updating to the Finesse agent/supervisor desktop. In the CTIOS and CAD environment, engineers usually check the design guidelines and verify if the configuration limits such as skillgroup per agent, total skillgroups per peripherals and teams per supervisor etc. have been over-subscribed. Engineers would also check number of concurrent cti client connections on the CTISVR, Refer to [Cisco Unified Contact Center Enterprise Design Guide, Release 10.0\(1\)](#)

Relavant information can also be referenced from design guides for version 10.5(x), and version 11.

Troubleshooting of this type of problems on Finesse agents/superisors also starts with above mentioned design limit verifications. However, Finesse agents can be impacted by additional limitations which are exclusively found on Finesse servers.

Problem Symptoms:

Queue or Skillgroup stats not updating issue is typically found in following scenario:

- Current agent states are not reflected in individual skills/queues gadget on the Finesse agent desktop. However, checking the agent states using **opctest** with **la** command on the agent Peripheral Gateway (PG), indicates the agent states are correct with skillgroups
- Number of agents are in TALKING state for a while, however, Finesse agent or supervisor desktops still show 0 in skills/queues for talking time.
- Restart of Finesse server would allow the system to work temporarily, but usually the same problem resurface within minutes or hours.

Finesse CTI Messaging and Finesse Queue Stats Buffer

Finesse agent Queue stats or Skill stats updates are carried out through the exchanges of following pairs of request and response CTI messages in Finesse servers.

getQueryQueueStatisticsReq() message request by Finesse and the **QuerySkillGroupStatisticsConf** message as the result of CTISVR responses.

By default, Finesse can process 751 Skill Group Requests within the designated 10 second stats refresh interval. Requests that are not processed will be buffered in a message queue to be processed at a later time. Finesse by default is initialized with this Message Buffer Queue to hold 5000 REQUEST messages.

However, if the buffer fills up and is overwhelmed, some of these queue stats request messages will be timed out and dropped.

Possible Causes for Finesse Queue Stats Buffer Overrun

1. Design/Configuration over-subscription. eg skills per agents, total skillgroups per peripherals and teams per supervisor etc. Refer to [Cisco Unified Contact Center Enterprise Design Guide](#) for recommended configuration limits. Over-subscription can lead to excessive CTI messagings on Stats updates, and hence overruns the Finesse Queue Stat Request buffer.
2. Exceeding max allowed concurrent CTI client connection including All Events connections and Monitored Mode connections. CTISVR resource depletion which lead to significant slow down on CTI message processing speed.
3. PG performance eg. CPU, Memory, and Disk I/O etc..
4. Not enough Network bandwidth to support the CTI messageing delays allowed for Finesse Application, ie *62ms*.

Finesse bandwidth calculator provided in the link below with the current design specs, to allocate recommended network bandwidth.

<http://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-technical-reference-list.html>

Finess Queue Stats Buffer Overrun Due to CTI message delay.

Based on this Finesse limitation on the REQUEST message processing speed and the message buffer, the default value of max average request/response delay is *62ms* for average Finesse deployments. If the average delays significantly exceed the benchmark of *62ms*, eg. CTI message delay around *100ms*. Then buffered ***getQueryQueueStatisticsReq()*** cti messages can never be sent to CTISVR and be responded with ***QuerySkillGroupStatisticsConf*** messages fast enough within that 10 second refresh interval. And the remaining timed out ***getQueryQueueStatisticsReq()*** messages will be dropped from the Queue Stats buffer

Relavent Finesse Trace

Queue Stats messages can be found in *webservices* finesse log. It requires DEBUG trace level to reveal detailed queue stats messages.

For steps to turn up DEBUG trace level for webservices, please refer to following docwiki link.

Log Analyses:

When the Queue Buffer Overrun takes place following log snippets can be observed from the webservices log.

Look for the beginning of a round of 751 queue stat request update, at the beginning of 10 second refresh interval.

eg.

```
Sep 22 2014 14:34:59.878 -0700: %CCBU_pool-21-thread-1-6-
QUEUE_STATISTICS_REQUEST: %[count=751]: Starting new round of querying
active queue statistics
```

Between this and the next round of 751 requests which occurs after 10 seconds, filter and verify with a text tool eg. Notepad++, if there are matching 751 **QuerySkillGroupStatisticsConf** messages.

eg

```
Sep 22 2014 14:34:59.888 -0700: %CCBU_CTIMessageEventExecutor-0-6-
DECODED_MESSAGE_FROM_CTI_SERVER:
%[cti_message=CTIQuerySkillGroupStatisticsConf[peripheralID=5000,
skillGroupNumber=28353, routerCallsQNow=0, longestRouterCallQNow=0,
agentsNotReady=0, agentsReady=0, agentsTalkingIn=0, agentsTalkingOut=0,
agentsTalkingOther=0, agentsWorkNotReady=0,
agentsWorkReady=0]CTIMessageBean [invokeID=112223, msgID=115,
timeTracker={"id":"QuerySkillGroupStatisticsConf","CTI_MSG_NOTIFIED":141
1536082977,"CTI_MSG_RECEIVED":1411536082976},
msgName=QuerySkillGroupStatisticsConf,
deploymentType=CCE]][cti_response_time=1]: Decoded Message to Finesse
from backend cti server
```

For example, if there are only 329 **QuerySkillGroupStatisticsConf** messages processed by Finesse for this round, in another word, there must be 422 messages being queued in the buffer. Obviously, if round 400 messages are to be queued every 10 seconds then the buffer can reach its threshold of 5000 messages within 3 minutes.

Search for the first occurrence of polling error happens within 3 minutes, that is the sign of Buffer Overrun.

```
Sep 22 2014 14:37:29.883 -0700: %CCBU_pool-21-thread-1-3-QUEUE
STATISTICS POLLING ERROR: %[ERROR_DESCRIPTION= maximum pool and queue
capacity reached so discarding execution][error_message=Thread pool
saturated, discarding execution ]: Error during queue statistics polling
```

Illustration of CTISVR processing delay.

- Finesse sends *getQueryQueueStatisticsReq()* request to CTISVR - tracking **invokeId=112223, queueId=28353**

```
Sep 23 2014 22:21:22.875 -0700: %CCBU_pool-19-thread-4-7-
CTIWriter.getQueryQueueStatisticsReq(): {Thrd=pool-19-thread-4}
params : invokeId=112223, queueId=28353
```

- CTISVR received the request -
match **InvokeID:0x1b65f** with **invokeId=112223** in the Finesse request
and **SkillGroupNumber:28353** with **queueId** in the Finesse request

```
22:21:22:921 cg1A-ctisvr SESSION 9:
MsgType:QUERY_SKILL_GROUP_STATISTICS_REQ (InvokeID:0x1b65f)
```

PeripheralID:5000

22:21:22:921 cglA-ctisvr SESSION 9:

SkillGroupNumber:28353

SkillGroupID:N/A)

- CTISVR response

22:21:22:999 cglA-ctisvr SESSION 9:

MsgType:QUERY_SKILL_GROUP_STATISTICS_CONF (**InvokeID:0x1b65f**

PeripheralID:5000

22:21:22:999 cglA-ctisvr SESSION 9:

SkillGroupNumber:28353

SkillGroupID:9431 AgentsLoggedOn:0 AgentsAvail:0 AgentsNotReady:0

22:21:22:999 cglA-ctisvr SESSION 9: AgentsReady:0

AgentsTalkingIn:0 AgentsTalkingOut:0 AgentsTalkingOther:0

22:21:22:999 cglA-ctisvr SESSION 9: AgentsWorkNotReady:0

AgentsWorkReady:0 AgentsBusyOther:0 AgentsReserved:0 AgentsHold:0

22:21:22:999 cglA-ctisvr SESSION 9: AgentsICMAvailable:0

AgentsApplicationAvailable:0 AgentsTalkingAutoOut:0

22:21:22:999 cglA-ctisvr SESSION 9: AgentsTalkingPreview:0

AgentsTalkingReservation:0 RouterCallsQNow:0

- Finesse received the CTISVR response, and formed **QuerySkillGroupStatisticsConf** message

Sep 23 2014 22:21:22.977 -0700: %CCBU_CTIMessageEventExecutor-0-6-

DECODED_MESSAGE_FROM_CTI_SERVER:

%[cti_message=CTIQuerySkillGroupStatisticsConf[peripheralID=5000,
skillGroupNumber=28353, routerCallsQNow=0, longestRouterCallQNow=0,

agentsNotReady=0, agentsReady=0, agentsTalkingIn=0,

agentsTalkingOut=0, agentsTalkingOther=0, agentsWorkNotReady=0,

agentsWorkReady=0]CTIMessageBean [**invokeID=112223**, msgID=115,

timeTracker={"id":"QuerySkillGroupStatisticsConf","CTI_MSG_NOTIFIED":
1411536082977,"CTI_MSG_RECEIVED":1411536082976},

msgName=QuerySkillGroupStatisticsConf,

deploymentType=CCE]][cti_response_time=1]: Decoded Message to Finesse
from backend cti server

Notice it took over 100ms seconds for Finesse to receiving the matching **QuerySkillGroupStatisticsConf** message, If this is an average response time. Finesse can run into the Buffer Queue Overrun issue.

Workaround:

There is a property in *aws.properties* which sets the refresh interval on the Finesse server side. This is basically the interval between two rounds of queue stats requests (one round being 751 queue stats requests in this deployment) from Finesse to CTISVR. Finesse by default requests it every 10 seconds. This property could potentially be changed to something higher in value which would mean that Finesse will have a little more time (eg: 20sec instead of 10sec) to process one round of queue stats requests. It also effectively extends the benchmark CTI skill stats request/response delay from 62ms to 124ms.

- Obtained root access to the Finesses OS platform.
- VI to the property file `/opt/cisco/desktop/conf/webservices/aws.properties`

- Modified the following property value from 10 to 20
`com.cisco.cc.webservices.reporting.core.queue_statistics_refresh_interval`

Note: Refresh interval for queue statistics in seconds

- save the `aws.properties` file
- restarted the Finesse Tomcat Service
- The same steps are to be carried out on all Finesse nodes within the Finesse cluster