

Cómo la conexión de la palabrería trabaja para la presencia del escritorio del agente de la delicadeza

Contenido

[Introducción](#)

[¿Cuál es PALABRERÍA?](#)

[Problema frecuente debido a la desconexión de la PALABRERÍA](#)

[Análisis del registro](#)

[Registros del debug](#)

[Registros de la información](#)

[Registros de Webservices](#)

[Razones comunes para la desconexión de la PALABRERÍA](#)

[Acciones de la sugerencia](#)

Introducción

Este documento describe la arquitectura detrás de las conexiones de la delicadeza que utiliza las Bidireccional-secuencias sobre HTTP síncrono (PALABRERÍA) y cómo los problemas se relacionaron con esto puede ser diagnosticado.

¿Cuál es PALABRERÍA?

La Mensajería y el protocolo extensibles de la presencia (XMPP) es protocolo con estado en un client-server model. Si una aplicación necesita trabajar con XMPP, los problemas múltiples se presentan. Los navegadores no soportan XMPP nativo, así que todo el tráfico XMPP se debe manejar por un programa que se ejecute dentro del navegador.

El primer problema es que el HTTP es protocolo apátrida. Esto significa que cada pedido de HTTP no está relacionado con ninguna otra petición. Sin embargo, este problema se puede abordar por aplicable significa--por ejemplo usando los Cookie/los datos del poste.

El segundo problema es el comportamiento unidireccional del HTTP. Solamente el cliente envía las peticiones, y el servidor puede responder solamente. La incapacidad del servidor para avanzar los datos hace artificial implementar XMPP sobre el HTTP.

Se elimina este problema si el cliente puede hacer las peticiones directas TCP (así eliminando la necesidad del HTTP). Sin embargo, si usted quiere abordar el problema dentro del dominio HTTP (por ejemplo porque el Javascript puede enviar los pedidos de HTTP), hay dos Soluciones posibles. Ambos requieren un Bridge entre el HTTP y XMPP. Las soluciones están sondeando (los pedidos de HTTP relanzados que piden los nuevos datos) e interrogación larga, también conocida como PALABRERÍA.

El propósito de usar la PALABRERÍA es cubrir para arriba el hecho de que el servidor no tiene que responder pues pronto hay una petición. Se retrasa la respuesta (una encuesta larga) hasta

que el servidor tenga datos para el cliente, y entonces se envía como respuesta. Tan pronto como el cliente lo consiga él hace una nueva petición (incluso si él no tiene nada que enviar) y así sucesivamente.

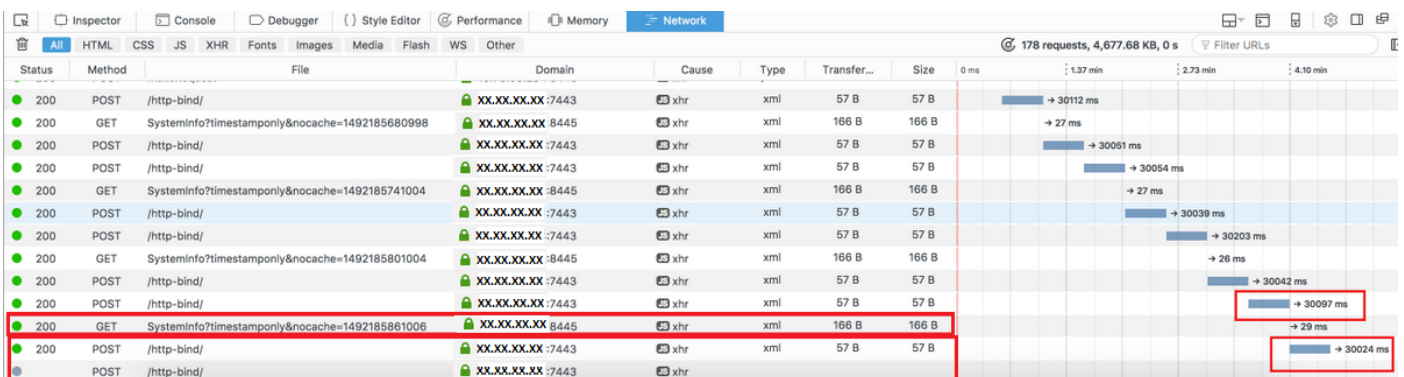
La PALABRERÍA es muy eficiente desde el punto de vista de la carga del servidor y tráfico-sabia.

El cliente de escritorio de la delicadeza (aplicación de Web) establece una conexión aña de la PALABRERÍA cada 30 segundos. Después de 30 segundos, si no hay actualizaciones del servicio de Notificaiton de la delicadeza, el servicio de notificación envía una contestación HTTP con una AUTORIZACIÓN 200 y (casi) un cuerpo vacío de la respuesta. Si el servicio de notificación tiene una actualización en la presencia de un agente o un evento del diálogo (llamada), por ejemplo, los datos se envían inmediatamente al cliente de Web de la delicadeza.

Para resumir:

1. El cliente de Web de la delicadeza siempre tendrá una conexión HTTP aña (HTTP-lazo) configurada al servidor de la delicadeza vía el puerto TCP 7443. Esto se conoce como encuesta larga de la PALABRERÍA.
2. El servicio de notificación de la delicadeza es un servicio de la presencia que fija las actualizaciones con respecto al estado de un agente, de una llamada, de un etc.
3. Si el servicio de Notificaiton tiene una actualización, contestará a la petición del HTTP-lazo con la actualización del estado como mensaje XML en el cuerpo del HTTP de respuesta.
4. Si hay ningún estado pone al día 30 segundos después de recibir la petición del HTTP-lazo, el servicio de Notificaiton contesta sin ningunas actualizaciones del estado para permitir que el cliente de Web de la delicadeza envíe otra petición del HTTP-lazo. Esto sirve como manera para el servicio de notificación de saber que el cliente de Web de la delicadeza puede todavía conectar con el servicio de notificación y que el agente no cerró a su navegador ni puso su ordenador para dormir, el etc.

También, el cliente de Web de la delicadeza envía una petición de SystemInfo que marque el estatus de Tomcat de la delicadeza cada minuto, tal y como se muestra en de la imagen:



Tal y como se muestra en de la imagen, el registro del debug del servicio de notificación (Openfire) muestra el HTTP que ata con el escritorio junto con la dirección IP y el puerto del agente PC.

```
2017.04.14 21:34:21 [Session-1, SSL_NULL_WITH_NULL_NULL] received 0 sent 0
2017.04.14 21:34:21 REQUEST /http-bind/ on org.eclipse.jetty.server.nio.SelectChannelConnector$SelectChannelHttpConnection@2d5226@XXX.XXX.XXX.XX:7443<->XXX.XXX.XXX.XX:49805
2017.04.14 21:34:21 scope null[/http-bind/ @ o.e.j.s.ServletContextHandler{/http-bind,null}]
2017.04.14 21:34:21 context=/http-bind[/ @ o.e.j.s.ServletContextHandler{/http-bind,null}]
2017.04.14 21:34:21 sessionManager=org.eclipse.jetty.server.session.HashSessionManager@176fe4#STARTED
2017.04.14 21:34:21 session=null
2017.04.14 21:34:21 session=null
2017.04.14 21:34:21 servlet /http-bind[/ -> org.jivesoftware.openfire.http.HttpBindServlet-1643193
2017.04.14 21:34:21 chain=null
2017.04.14 21:34:21 HTTPBindLog: HTTP REC(3445afbe): <body sid="3445afbe" rid="164053266"/>
2017.04.14 21:34:21 consumeResponse: org.jivesoftware.openfire.http.HttpSession@47653 status: 3 address: 1001003@XXX.XXX.XXX.XXX.XX.cisco.com/desktop id: 3445afbe presence:
<presence from="1001003@XXX.XXX.XXX.XXX.XX.cisco.com/desktop">
<xmlns="http://jabber.org/protocol/caps" hash="sha-1" node="http://jabber.cisco.com/caxl" ver="VNC6fNuvCxe6FjfdIplryVJRm"/>
</presence> rid: 164053266
2017.04.14 21:34:21 suspended org.eclipse.jetty.server.nio.SelectChannelConnector$SelectChannelHttpConnection@2d5226@XXX.XXX.XXX.XX:7443<->XXX.XXX.XXX.XX:49805
2017.04.14 21:34:24 Launching thread for /127.0.0.1:44667
2017.04.14 21:34:24 Launching thread for /127.0.0.1:44656
```

Tal y como se muestra en de la imagen, el **active más reciente 0 ms** muestra que la sesión es todavía activa.

```
2017.04.14 21:34:26 Exiting since queue is empty for /127.0.0.1:44667
2017.04.14 21:34:26 Exiting since queue is empty for /127.0.0.1:44660
2017.04.14 21:34:26 Session (id=3445afbe) was last active 0 ms ago: 1001003@XXXXXXXX.XXXXXXXXXX.cisco.com/desktop
2017.04.14 21:34:26 time=1492185866851, JID=1001003@XXXXXXXX.XXXXXXXXXX.cisco.com/desktop, msg_sent=4, msg_queue=0, msg_drop=0, bytes_sent=3748
2017.04.14 21:34:26 time=1492185866851, JID=1001003@XXXXXXXX.XXXXXXXXXX.cisco.com/desktop, msg_sent=4, msg_queue=0, msg_drop=0, bytes_sent=3748
2017.04.14 21:34:26 Launching thread for /127.0.0.1:44678
2017.04.14 21:34:26 Exiting since queue is empty for /127.0.0.1:44678
```

Los registros del debug del servicio de notificación del permiso del Centro de contacto unificado expresan (UCCX), tal y como se muestra en de la imagen:

```
[admin:utils uccx notification-service log enable

WARNING! Enabling Cisco Unified CCX Notification Service logging can affect system performance
and should be disabled when logging is not required.

Do you want to proceed (yes/no)? yes

Cisco Unified CCX Notification Service logging enabled successfully.

NOTE: Logging will be disabled automatically if Cisco Unified CCX Notification Service is restarted.
admin:|
```

Registros del debug del servicio de notificación del permiso de la empresa unificada del Centro de contacto (UCCE) (delicadeza independiente), tal y como se muestra en de la imagen:

```
[admin:utils finesse notification logging enable

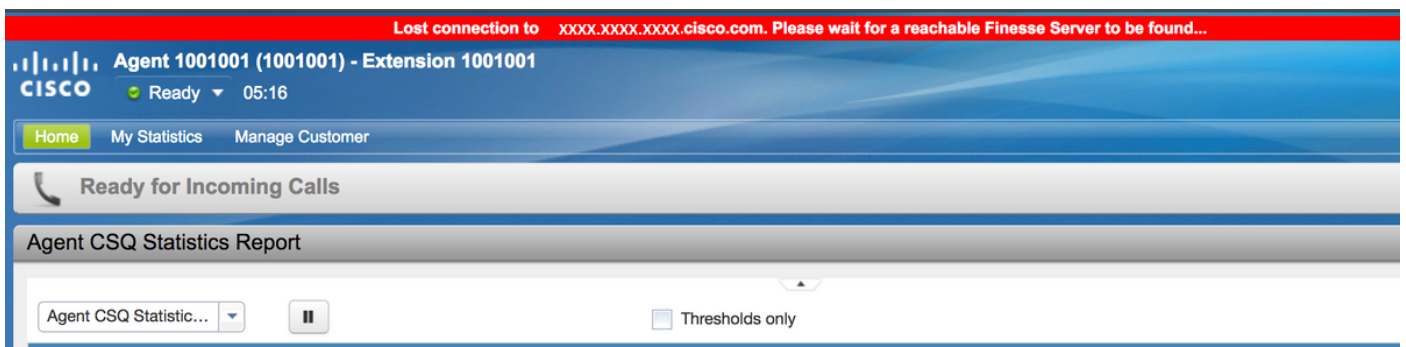
Checking that the Cisco Finesse Notification Service is started...
The Cisco Finesse Notification Service is started.

Cisco Finesse Notification Service logging is now enabled.

WARNING! Cisco Finesse Notification Service logging can affect system performance
and should be disabled when logging is not required.

Note: Logging will be disabled automatically if you restart the Cisco Finesse Notification Service.
```

Problema frecuente debido a la desconexión de la PALABRERÍA



Para UCCX, el temporizador del logout es 60 segundos después de que desconexión del navegador. El agente puede estar en el listo o no el estado Ready (Listo) para que el logout

sucedan.

Para UCCE, la delicadeza toma hasta 120 segundos para detectar cuando un agente cierra al navegador o el navegador causa un crash y la delicadeza espera 60 segundos antes de enviar una petición forzada del logout al CTI Server. Bajo estas condiciones, la delicadeza puede tomar hasta 180 segundos para firmar hacia fuera el agente.

Para más información sobre el comportamiento de escritorio de la delicadeza UCCE, refiera a la sección *de escritorio del comportamiento del capítulo de los mecanismos de la Conmutación por falla de la delicadeza de Cisco* en la [guía de administración de la delicadeza de Cisco](#).

Registre el análisis

Los registros de servicio de la delicadeza y de Notificación se pueden recoger vía RTMT o vía el CLI:

el archivo consigue el activelog /desktop se repite compresión

Registros del debug

Estos registros están en la carpeta de /desktop/logs/openfire y se nombran debug.log.

```
2017.06.17 00:14:34 Session (id=f382a015) was last active 0 ms ago:
```

```
1001003@xxxxxx.xxxx.xxx.cisco.com/desktop
```

```
2017.06.17 00:15:04 Session (id=f382a015) was last active 13230 ms ago:
```

```
1001003@xxxxxx.xxxx.xxx.cisco.com/desktop
```

```
2017.06.17 00:15:34 Session (id=f382a015) was last active 43230 ms ago:
```

```
1001003@xxxxxx.xxxx.xxx.cisco.com/desktop
```

```
2017.06.17 00:16:04 Session (id=f382a015) was last active 63231 ms ago:
```

```
1001003@xxxxxx.xxxx.xxx.cisco.com/desktop
```

```
2017.06.17 00:17:04 Unable to route packet. No session is available so store offline. <message
```

```
from="pubsub.xxxxx.xxxx.xxx.cisco.com" to="1001003@xxxxxx.xxxx.xxx.cisco.com.cisco.com"
```

```
id="/finesse/api/User/1001003__1001003@xxxxxx.xxxx.xxx.cisco.com__o5Aqb"><event
```

```
xmlns="http://jabber.org/protocol/pubsub#event"><items node="/finesse/api/User/1001003"><item
```

```
id="0d78a283-466d-4477-a07e-6e33a856fce388"><notification
```

```
xmlns="http://jabber.org/protocol/pubsub">&lt;Update&gt;
```

Registros de la información

Estos registros están en la carpeta de /desktop/logs/openfire y se nombran info.log.

```
2017.06.17 00:16:04 Closing idle session (id=f382a015): 1001003@xxxxxx.xxxx.xxx.
```

```
cisco.com/desktop after being inactive for more than threshold value of 60
```

```
2017.06.17 00:16:04 A session is being closed for 1001003@xxxxxx.xxxx.xxx.cisco.com/desktop
```

Openfire cerrando a la sesión inactiva indica que logout del agente accionará 1 minuto y éste aparece como código de motivo 255 en los informes

Registros de Webservices

Estos registros están en la carpeta de /desktop/logs/webservices y se nombran Desktop-

webservices.YYYY-MM-DDTHH-MM-SS.sss.log.

```
0000001043: XX.XX.XX.XXX: Jun 17 2017 00:16:04.630 +0530: %CCBU_Smack Listener Processor (1)-6-
PRESENCE_NOTIFICATION_RECIEVED: %[FROM
JID=1001003@xxxxxx.xxxx.xxx.cisco.com/desktop][PRESENCE_TYPE=unavailable]:Finesse received a
presence notificacion
0000000417: XX.XX.XX.XXX: Jun 17 2017 00:16:04.631 +0530: %CCBU_Smack Listener Processor (1)-6-
UNSUBSCRIBE_REQUEST_SUCCESS:
%[NodeId=/finesse/api/User/1001003/ClientLog][user_id=1001003@xxxxxx.xxxx.xxx.cisco.com]:
Sucessfully unsubscribed from a node on the XMPP server
0000001044: XX.XX.XX.XXX: Jun 17 2017 00:16:04.631 +0530: %CCBU_Smack Listener Processor (1)-6-
AGENT_PRESENCE_MONITOR: %[message_string=Adding agent 1001003 into the expiry hash.]:
0000001051: XX.XX.XX.XXX: Jun 17 2017 00:16:35.384 +0530: %CCBU_pool-8-thread-1-6-
AGENT_PRESENCE_MONITOR: %[message_string=[Expired] Removed agent from cache 1001003]:
0000001060: XX.XX.XX.XXX.: Jun 17 2017 00:17:04.632 +0530: %CCBU_CoreImpl-worker12-6-PRESENCE
DRIVEN LOGOUT: %[agent_id=1001003]: Performing CTI Logout on basis of the agents unavailable
presence
0000001061: XX.XX.XX.XXX.: Jun 17 2017 00:17:04.633 +0530: %CCBU_CoreImpl-worker12-6-
MESSAGE_TO_CTI_SERVER: %[cti_message=Invoke id :39 , agentstate : 1, workmode : 0, reason code:
255, forceflag :1, agentcapacity: 1, agenttext: 1001003, agentid: 1001003, supervisorid: null,
ssoFlag=false][cti_message_name=SetAgentStateReq]: Message going to the backend cti server
0000001066: XX.XX.XX.XXX.: Jun 17 2017 00:17:04.643 +0530: %CCBU_CTI_MESSAGE_EVENT_EXECUTOR-0-6-
DECODED_MESSAGE_FROM_CTI_SERVER: %[cti_message=CTIAgentStateEvent [skillGroupState=1 (LOGOUT),
stateDuration=0, skillGroupNumber=-1, skillGroupPriority=0, agentState=1 (LOGOUT),
eventReasonCode=255, numFltSkillGroups=0, CTIClientSignature=null, agentID=1001003,
agentExtension=1001003, agentInstrument=null, agentID_Long=1001003, duration=null,
nextAgentState=null, fltSkillGroupNumberList=[], fltSkillGroupIDList=[],
fltSkillGroupPriorityList=[], fltSkillGroupStateList=[], MRDId=1, agentMode=0]CTI_MESSAGE_BEAN
[invokeID=null, cti_sequence_id=105, msgID=30,
timeTracker={"id":"AgentStateEvent","CTI_MSG_RECEIVED":1497638824642,"CTI_MSG_DISPATCH":14976388
24643}, msgName=AgentStateEvent, deploymentType=CCX]][cti_response_time=1][dispatch_phase=DnD-
CHECKPOINT-3B]: Decoded Message to Finesse from backend cti server
```

Razones comunes para la desconexión de la PALABRERÍA

- Problema de red
- Navegador y/o versión sin apoyo.
- Condición de atascamiento del navegador debido a contentar de la otra lengüeta/ventana.
- Computadora puesta para dormir.
- Problema CPU elevada o de memoria alta en el servidor.
- Gadgets de las de otras compañías que hacen la actividad de los shenanigans en el fondo.
- Conexión perdidosa del motor CTI Server/CCX con el servidor de la delicadeza.
- Problema NTP en el servidor o el cliente.

Acciones de la sugerencia

- Utilice el buscador admitido/la versión y las configuraciones según las guías.

- Pruebe el comportamiento con todos los gadgets de las de otras compañías quitados.
- Entienda el comportamiento y el flujo de trabajo del agente.
- Recomience la delicadeza Tomcat de Cisco y el servicio de notificación.
- Ejecute el **utils diagnostican la prueba** en el servidor, y la validan adelante y búsqueda de DNS reversible del servidor y del cliente ambos.
- Marque el **estatus NTP del utils** en el servidor.
- En los registros del cliente, marque la deriva y la latencia de red. Los registros del cliente se pueden ver de los registros de la consola Web del navegador o presionando el **informe de error del envío** en la página de la delicadeza y recogiendo los registros de la delicadeza. Los registros están situados en **/desktop/logs/clientlogs**.

Note: Los valores del temporizador pudieron cambiar en el futuro según el requerimiento de producto.