# Comprender y solucionar problemas de implementación de BOSH Finesse

## Contenido


Introducción
Prerequisites
Requirements
Componentes Utilizados
Antecedentes
Comprender la implementación de BOSH Finesse
Comprender XMPP
Ejemplo de mensaje XMPP
Implementación de XMPP con Finesse
Ejemplo de solicitud/respuesta XMPP de Finesse
Comprensión de los mensajes XMPP y los nodos XMPP de Finesse
Ejemplo 1: Uso de Pidgin para ver nodos XMPP de Finesse
Ejemplo 2: Utilice la ficha Red de herramientas de desarrollador de navegadores para ver mensajes HTTP
Troubleshooting de Mensaje de Error de Desconexión BOSH
Análisis de registro
Registros del servicio de notificación de depuración
Registros de Info Notification Service
Registros de Webservices
Razones comunes para la desconexión de BOSH
Problema: los agentes se desconectan en distintos momentos (problema del lado del cliente)
Acciones recomendadas
Problema: todos los agentes se desconectan al mismo tiempo (problema en el servidor)
Acciones recomendadas
Utilizar Fiddler
Problema común de Fiddler
Pasos de configuración de ejemplo
Utilizar Wireshark
Defectos relacionados
Información Relacionada


## Introducción

Este documento describe la arquitectura detrás de las conexiones Finesse que utilizan BOSH y cómo se pueden diagnosticar los problemas de conexión BOSH.

## Prerequisites

### Requirements

Cisco recomienda que tenga conocimiento sobre estos temas:

- Cisco Finesse

- Unified Contact Center Enterprise (UCCE)
- Unified Contact Center Express (UCCX)
- Herramientas de desarrollador del navegador web
- Administración de Windows o Mac

### Componentes Utilizados

La información que contiene este documento se basa en las siguientes versiones de software y hardware.

- Cisco Finesse 9.0(1) - 11.6(1)
- UCCX 10.0(1) - 11.6(2)

La información que contiene este documento se creó a partir de los dispositivos en un ambiente de laboratorio específico. Todos los dispositivos que se utilizan en este documento se pusieron en funcionamiento con una configuración verificada (predeterminada). Si tiene una red en vivo, asegúrese de entender el posible impacto de cualquier comando.

## Antecedentes

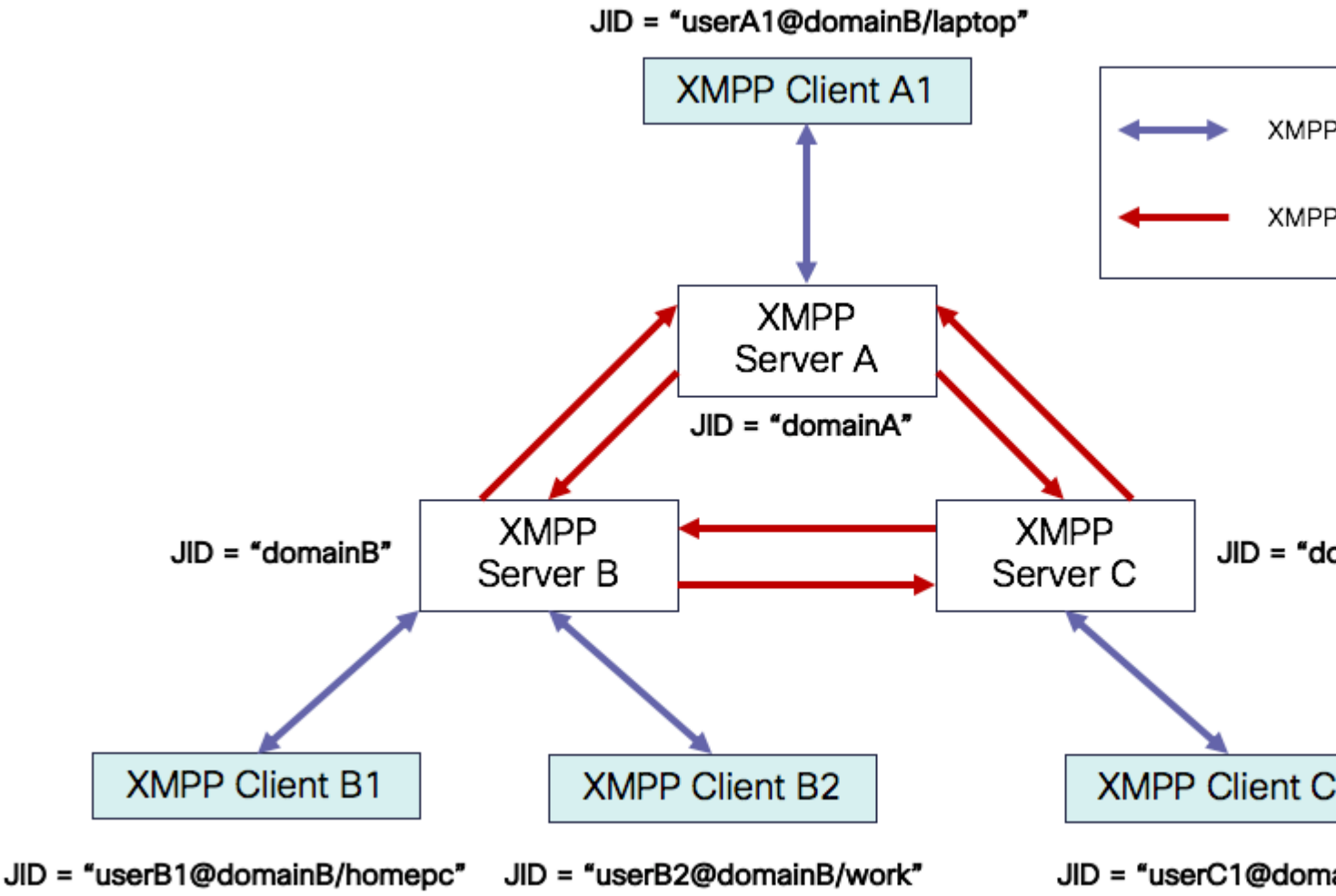
Las conexiones que utilizan secuencias bidireccionales sobre HTTP sincrónico se denominan BOSH.

## Comprender la implementación de BOSH Finesse

### Comprender XMPP

Extensible Messaging and Presence Protocol (XMPP) (también conocido como Jabber) es un protocolo con estado en un modelo cliente-servidor. XMPP permite la entrega rápida de pequeños fragmentos de datos XML (Lenguaje de marcado extensible) estructurados de una entidad a otra. XMPP/Jabber se utiliza ampliamente en aplicaciones de presencia y mensajería instantánea (IM).

Todas las entidades XMPP se identifican mediante su Jabber ID (JID).




Esquema de direccionamiento de JID: user@domain/resource

| | |
|---|---|
| usuario | nombre de usuario del cliente en el servidor XMPP o nombre de la sala de conferencias |
| domain | Nombre de dominio completo (FQDN) del servidor XMPP |
| recurso | identificador de la entidad o terminal específico del usuario (por ejemplo, portátil, smartphone, etc.), un identificador de sesión o un nombre de nodo pubsub |

---

> **Nota**: Los tres componentes JID no se utilizan en todos los casos. Normalmente, un servidor solo lo definiría el dominio, una sala de conferencias definida por user@domain y un cliente por user@domain/resource.

---

Los mensajes XMPP se denominan estrofas. Hay tres estrofas centrales en XMPP:

1. <mensaje>: una dirección, un destinatario

2. <presencia>: una dirección, publicar en muchas

3. <iq>: info/query - request/response

Todas las estrofas tienen direcciones de origen y destino y la mayoría de las estrofas también tienen tipo, id y xml:langattributes.

| **Atributo Stanza** | **Propósito** |
|---|---|
| a | JID de destino |
| desde | JID de origen |
| tipo | propósito del mensaje |
| id | identificador único utilizado para vincular una solicitud con una respuesta para <iq> estrofas |
| xml:lang | define el idioma predeterminado para cualquier XML legible por personas en la estrofa |

**Ejemplo de mensaje XMPP**

```
<message to='person1@example' from='person2@example' type='chat'>
  <subject> Team meeting </subject>
  <body>Hey, when is our meeting today? </body>
  <thread>A4567423</thread>
</message>
```

## Implementación de XMPP con Finesse

Si una aplicación web necesita funcionar con XMPP, surgen varios problemas. Los exploradores no admiten XMPP a través del protocolo de control de transmisión (TCP) de forma nativa, por lo que todo el tráfico XMPP debe gestionarse mediante un programa que se ejecute dentro del explorador. Los servidores y exploradores Web se comunican a través de mensajes HTTP (Protocolo de transferencia de hipertexto), por lo que Finesse y otras aplicaciones Web incluyen mensajes XMPP en los mensajes HTTP.

La primera dificultad de este enfoque es que HTTP es un protocolo sin estado. Esto significa que cada solicitud HTTP no está relacionada con ninguna otra solicitud. Sin embargo, este problema puede solucionarse por medios aplicables, por ejemplo, mediante el uso de cookies/datos de publicación.

La segunda dificultad es el comportamiento unidireccional de HTTP. Sólo el cliente envía solicitudes y el servidor sólo puede responder. La incapacidad del servidor para insertar datos hace que no sea natural implementar XMPP a través de HTTP.

Este problema no existe en la especificación de núcleo XMPP original (RFC 6120), donde XMPP está enlazado a TCP. Sin embargo, si desea solucionar el problema con XMPP enlazado a HTTP, por ejemplo,

debido a que Javascript puede enviar solicitudes HTTP, existen dos soluciones posibles. Ambos requieren un puente entre HTTP y XMPP.

Las soluciones propuestas son:

1. Sondeo (protocolo heredado): solicitudes HTTP repetidas que solicitan nuevos datos definidos en XEP-0025: sondeo HTTP de Jabber

2. El sondeo largo también se conoce como BOSH: protocolo de transporte que emula la semántica de una conexión TCP bidireccional de larga duración entre dos entidades mediante el uso eficiente de múltiples pares de solicitud/respuesta HTTP sincrónicos sin requerir el uso de sondeo frecuente definido en XEP-0124: HTTP Binding y extendido por XEP-0206: XMPP Over BOSH

Finesse implementa BOSH, ya que es bastante eficiente desde el punto de vista de la carga del servidor y del tráfico. La razón para utilizar BOSH es encubrir el hecho de que el servidor no tiene que responder tan pronto como hay una solicitud. La respuesta se retrasa hasta un tiempo especificado hasta que el servidor tiene datos para el cliente y, a continuación, se envía como respuesta. Tan pronto como el cliente obtiene la respuesta, el cliente hace una nueva solicitud y así sucesivamente.

El cliente de escritorio Finesse (aplicación web) establece una conexión BOSH obsoleta a través del puerto TCP 7443 cada 30 segundos. Después de 30 segundos, si no hay actualizaciones del servicio de notificación de Finesse, el servicio de notificación envía una respuesta HTTP con 200 OK y un cuerpo de respuesta (casi) vacío. Si el servicio de notificación tiene una actualización de la presencia de un agente o un evento de diálogo (llamada), por ejemplo, los datos se envían inmediatamente al cliente web de Finesse.

**Ejemplo de solicitud/respuesta XMPP de Finesse**

Este ejemplo muestra la primera respuesta de solicitud de mensaje XMPP compartida entre el cliente Finesse y el servidor Finesse para configurar la conexión BOSH.

```
Finesse client request:
<body xmlns="http://jabber.org/protocol/httpbind" xml:lang="en-US" xmlns:xmpp="urn:xmpp:xbosh" hold="1"

Finesse server response:
<body xmlns="http://jabber.org/protocol/httpbind" xmlns:stream="http://etherx.jabber.org/streams" authid
```

Para resumir:

1. El cliente web Finesse tiene una conexión HTTP obsoleta (http-bind) configurada en el servidor Finesse a través del puerto TCP 7443. Esto se conoce como una encuesta larga de BOSH.

2. El servicio de notificación de Finesse es un servicio de presencia que publica actualizaciones relacionadas con el estado de un agente, una llamada, etc.

3. Si el servicio de notificación tiene una actualización, responde a la solicitud http-bind con la actualización de estado como un mensaje XMPP en el cuerpo de la respuesta HTTP.

4. Si no hay actualizaciones de estado 30 segundos después de recibir la solicitud http-bind, el servicio de notificación responde sin ninguna actualización de estado para permitir que el cliente web Finesse envíe otra solicitud http-bind. Esto sirve como una manera para que el servicio de notificación sepa que el cliente web Finesse todavía puede conectarse al servicio de notificación y que el agente no cerró su navegador o puso su computadora en suspensión, y así sucesivamente.

## Comprensión de los mensajes XMPP y los nodos XMPP de Finesse

Finesse también implementa la especificación XMPP XEP-0060: Publish-Subscribe. El propósito de esta especificación es permitir que el servidor XMPP (servicio de notificación) obtenga información publicada en nodos XMPP (temas) y, a continuación, envíe eventos XMPP a entidades suscritas al nodo. En el caso de Finesse, el servidor de Integración de telefonía y ordenador (CTI) envía mensajes CTI al servicio web de Finesse para indicarle a Finesse las actualizaciones de configuración, como, entre otras, la creación de un agente o una cola de servicio de contacto (CSQ) o información sobre una llamada. Esta información se convierte en un mensaje XMPP que el servicio web Finesse publica en el servicio de notificación Finesse. A continuación, el servicio de notificación de Finesse envía mensajes XMPP sobre BOSH a los agentes suscritos a determinados nodos XMPP.

Algunos de los objetos de la API de Finesse definidos en la Guía del desarrollador de servicios Web de Finesse son nodos XMPP. Los clientes web Finesse de agente y supervisor pueden suscribirse a actualizaciones de eventos para algunos de estos nodos XMPP con el fin de tener información actualizada sobre eventos en tiempo real (como eventos de llamada, eventos de estado, etc.). Esta tabla muestra los nodos XMPP que están habilitados para pubsub.

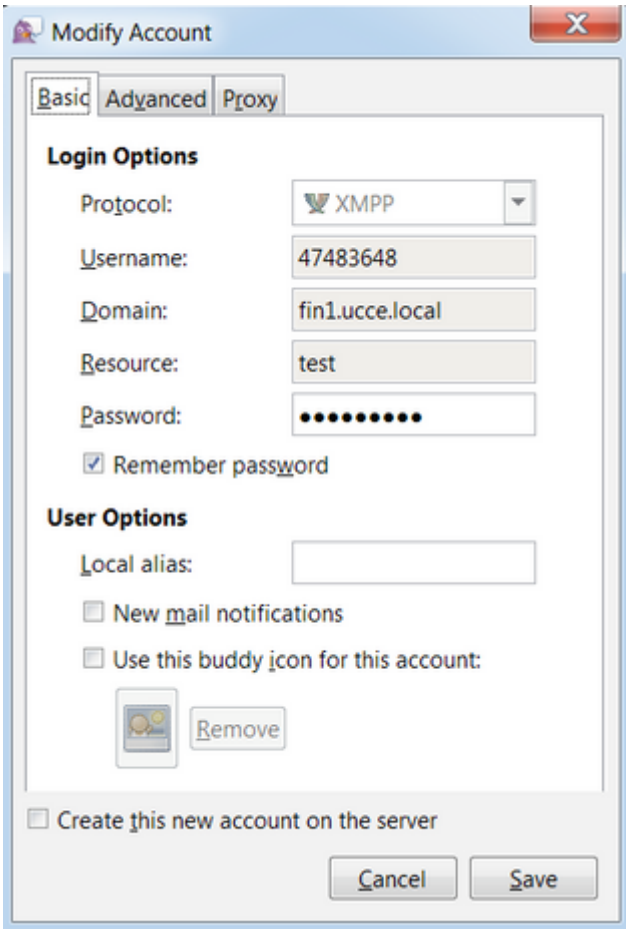
| **Objeto API Finesse** | **Propósito** | **Suscripción** |
|---|---|---|
| /finesse/api/User/<ID de inicio de sesión> | Muestra el estado y la asignación de equipo del agente | Agentes y supervisores |
| /finesse/api/User/<ID de inicio de sesión>/Dialogs | Muestra las llamadas manejadas por el agente. | Agentes y supervisores |
| /finesse/api/User/<ID de inicio de sesión>/ClientLog | Se utiliza para capturar los registros del cliente desde el botón **Enviar informe de errores** | Agentes y supervisores |
| /finesse/api/User/<LoginID>/Queue/<queueID> | Muestra datos de estadísticas de cola (si está habilitado) | Agentes y supervisores |
| /finesse/api/Team/<IdDeEquipo>/Users | Muestra los agentes que pertenecen a un equipo determinado, incluida la información de estado | Supervisores |
| /finesse/api/SystemInfo | Muestra el estado del servidor Finesse. Se utiliza para determinar si se necesita la conmutación por error | Agentes y supervisores |

### Ejemplo 1: Uso de Pidgin para ver nodos XMPP de Finesse

Paso 1. Descargue e instale el cliente XMPP Pidgin.

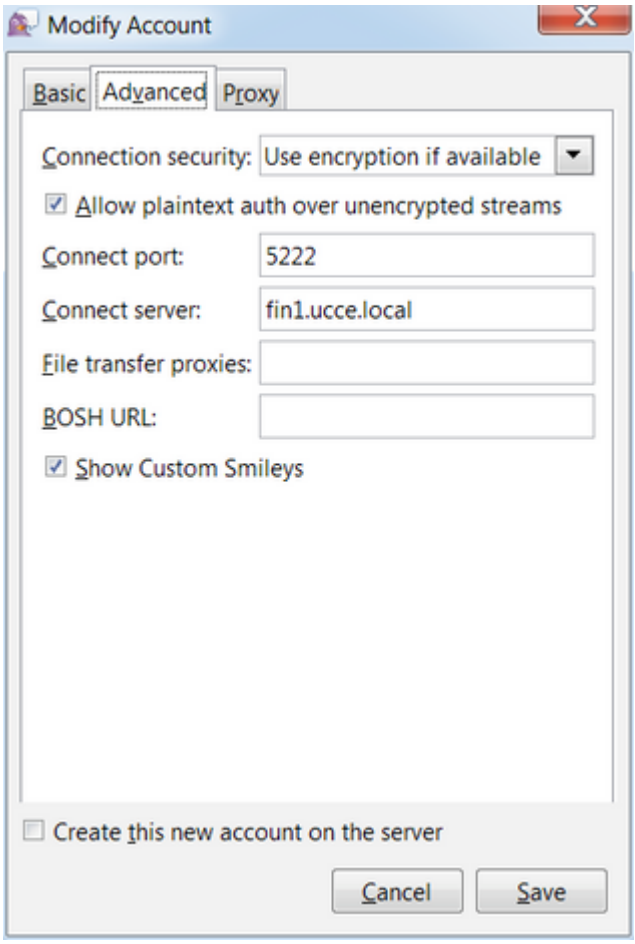
Paso 2. Navegue hasta **Cuentas > Modificar > Básico** y configure las **Opciones de Login**:

- Protocolo: XMPP
- Nombre de usuario: LoginID para cualquier agente
- Dominio: FQDN del servidor Finesse
- Recurso: Marcador de posición: se puede utilizar cualquier valor, por ejemplo, test
- Contraseña: contraseña del agente
- Marque la casilla de verificación **Recordar contraseña**




Paso 3. Navegue hasta **Cuentas > Modificar > Avanzado** y configure:

- Seguridad de la conexión: utilizar cifrado si está disponible
- Verifique el comando **Allow plaintext auth other unencryption streams**.
- Puerto de conexión: 522. Utilice el puerto 522 predeterminado. Este puerto es necesario para los clientes XMPP externos. Los clientes de escritorio Finesse utilizan 7443. No utilice el puerto 7443.
- Servidor de conexión: FQDN de servidor Finesse

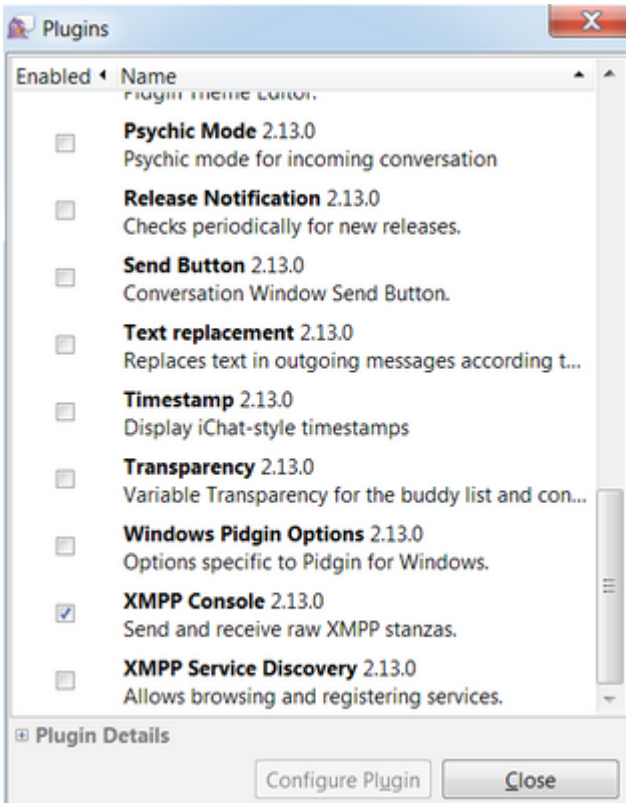



---

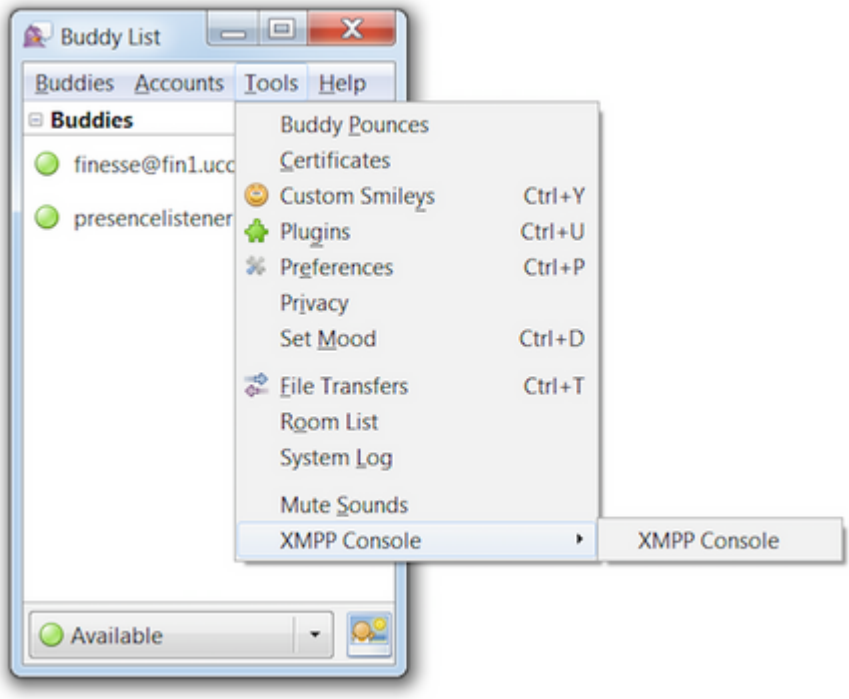
**Nota**: El puerto 5222 se utiliza solamente porque los clientes web de Finesse pueden utilizar el puerto 7443 para conectarse al servicio de notificación.

---

Paso 4. Vaya a **Tools > Plugins** y habilite la Consola XMPP.

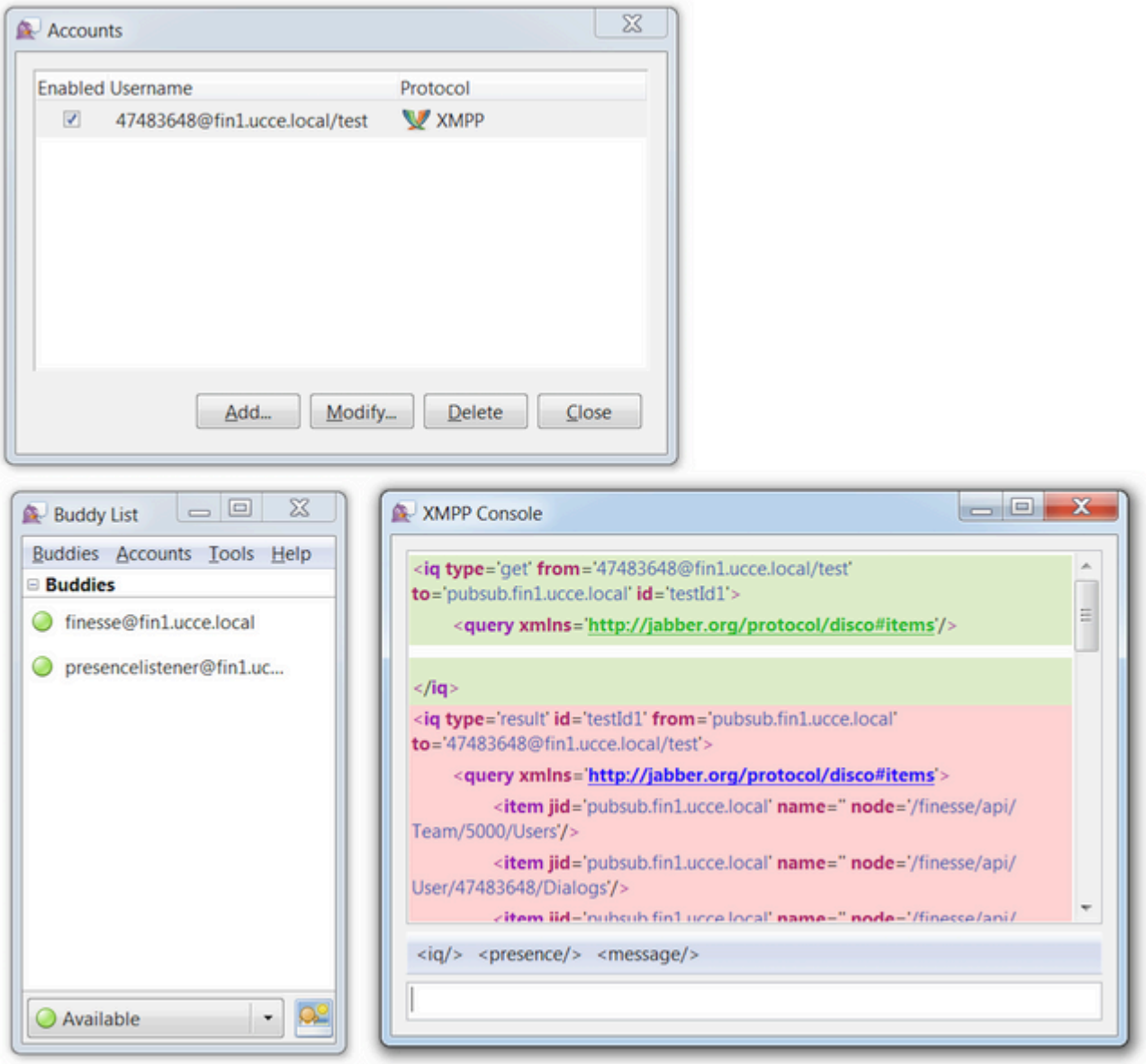



Paso 5. Vaya a **Herramientas > Consola XMPP > Consola XMPP** para abrir la Consola XMPP.




Paso 6. Ejecute este mensaje **<iq>** para ver todos los nodos XMPP que existen.

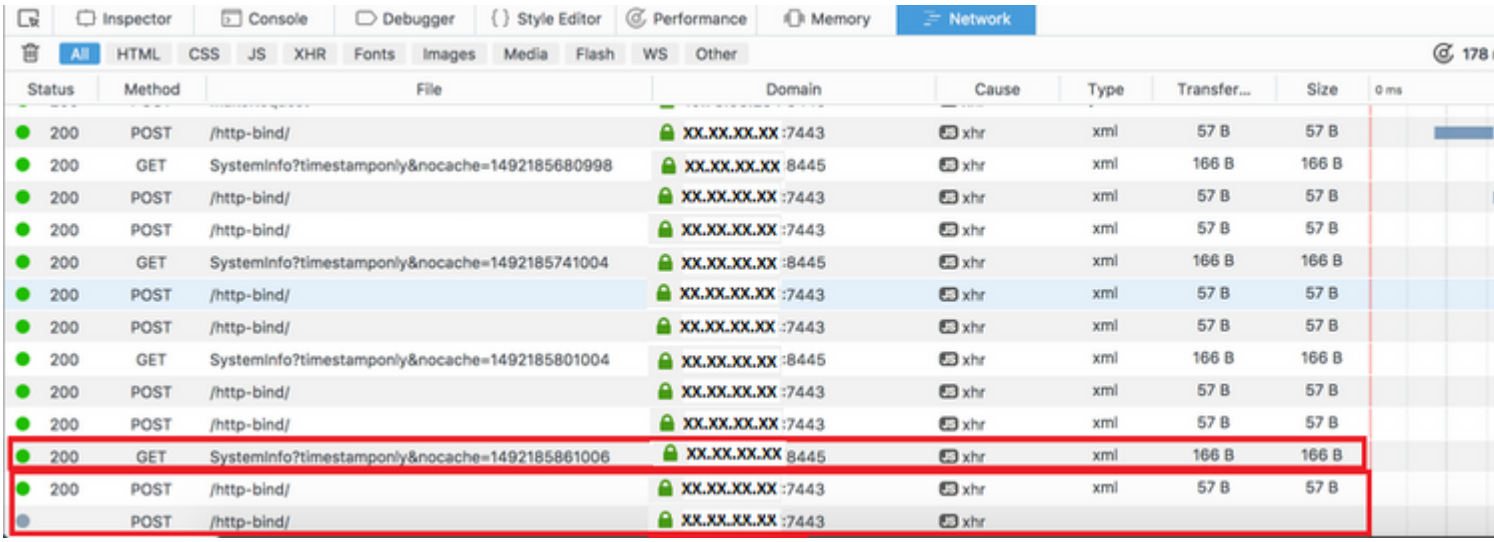
Por ejemplo:




En un entorno de laboratorio con dos agentes y dos colas de servicio de contacto configuradas, este resultado se incluye en la respuesta de Finesse:

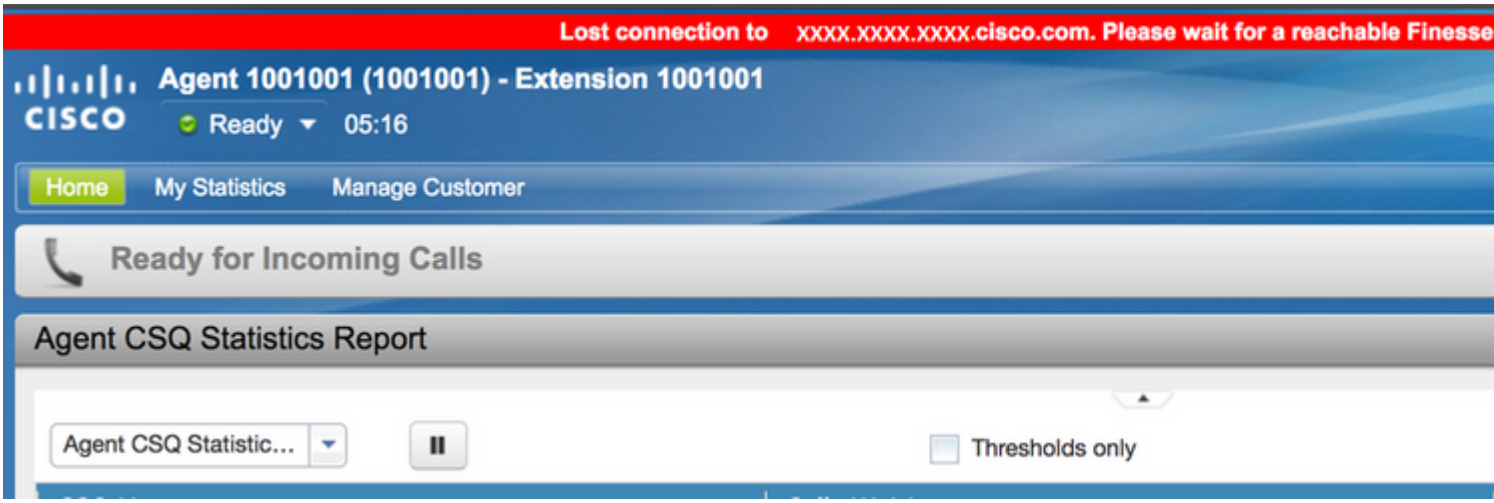
**Ejemplo 2: Utilice la ficha Red de herramientas de desarrollador de navegadores para ver mensajes HTTP**

Cada navegador tiene un conjunto de herramientas de desarrollador. La ficha Red de las herramientas del desarrollador muestra los mensajes HTTP enviados y recibidos por el cliente web Finesse (navegador). Por ejemplo, esta imagen muestra cómo el cliente web Finesse envía una solicitud SystemInfo que comprueba el estado de Tomcat Finesse cada minuto como una comprobación de conmutación por error. Además, también se muestran los mensajes http-bind de la conexión BOSH. El servidor Finesse devuelve una respuesta en 30 segundos si no hay actualizaciones para publicar en los nodos XMPP a los que está suscrito el cliente web.




# Troubleshooting de Mensaje de Error de Desconexión BOSH

Cuando se produce una desconexión BOSH, se produce el error Conexión perdida con {FQDN del servidor Finesse}. Espere a que se encuentre un servidor Finesse accesible... aparece en un banner rojo en la parte superior del escritorio de Finesse.




Este mensaje se muestra porque, en este momento, no se pueden recibir eventos de suscripción XMPP desde el servicio de notificación Cisco Finesse. Por lo tanto, la información de estado y los detalles de llamada no se pueden mostrar en el escritorio del agente.

Para UCCX, 60 segundos después de que el explorador se desconecte, el agente se pondrá en estado Desconectado. El agente puede estar en el estado Preparado o No preparado para que se produzca el cierre de sesión.

En el caso de UCCE, Finesse tarda hasta 120 segundos en detectar si un agente cierra el navegador o éste se bloquea y Finesse espera 60 segundos antes de enviar una solicitud de cierre de sesión forzoso al servidor CTI, lo que provoca que el servidor CTI ponga al agente en estado No preparado. En estas condiciones, Finesse puede tardar hasta 180 segundos en cerrar la sesión del agente. A diferencia de UCCX, el agente pasa al estado No preparado en lugar del estado Desconectado.

---

> **Nota**: la desconexión CTI no está preparada frente a El comportamiento del estado de desconexión en UCCE se controla mediante el parámetro PG /LOAD. Según las notas de la versión de Unified

---

Contact Center Enterprise y Hosted versión 10.0(1), el parámetro /LOAD deja de estar aprobado a partir de UCCE 10.0.

Para obtener más información sobre el comportamiento de UCCE Finesse Desktop, consulte la sección Comportamiento del Escritorio del capítulo Mecanismos de Failover de Cisco Finesse en la Guía de Administración de Cisco Finesse.

**Nota**: Los valores del temporizador pueden cambiar en el futuro según los requisitos del producto.

## Análisis de registro

Los registros del servicio de notificaciones de Finesse y UCCX se pueden recopilar a través de RTMT o de CLI:

**file get activelog /desktop recurs compress**

**Registros del servicio de notificación de depuración**

**Nota**: Establezca los logs de nivel de debug solamente mientras reproduce un problema. Desactive las depuraciones después de que se haya reproducido el problema.

**Nota**: Finesse 9.0(1) no tiene registro de nivel de depuración. El registro de nivel de depuración se introdujo en Finesse 9.1(1). El proceso para habilitar el registro es diferente en 9.1(1) comparado con Finesse 10.0(1) - 11.6(1). Para este proceso, consulte la guía Finesse Administration and Serviceability.

Habilitar los registros de depuración del servicio de notificación de Unified Contact Center Express (UCCX), como se muestra:

<#root>

```
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 can be disabled automatically if Cisco Unified CCX Notification Service is restarted.
```

Habilitar los registros de depuración del servicio de notificación de Unified Contact Center Enterprise (UCCE) (independiente de Finesse), como se muestra:

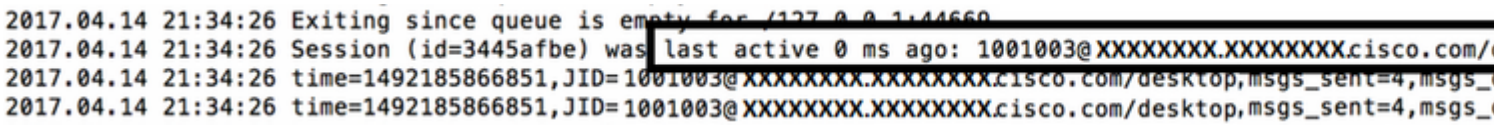
<#root>

```
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 can be disabled automatically if you restart the Cisco Finesse Notification Service
```

Estos registros se encuentran en la carpeta /desktop/logs/openfire y se denominan debug.log.

Como se muestra en la imagen, el debug.log del servicio de notificación (Openfire) muestra el enlace http con el escritorio junto con la dirección IP y el puerto del equipo del agente.


Como se muestra en la imagen, los últimos 0 ms activos muestran que la sesión sigue activa.




El cierre de la sesión inactiva de Openfire indica que el cierre de sesión del agente puede desencadenarse en 60 segundos, donde Finesse puede enviar un cierre de sesión forzado con un código de motivo de 255 al servidor CTI. El comportamiento real del escritorio en estas condiciones depende de la configuración de Desconexión al desconectar el agente (LOAD) en UCCE. En UCCX, este es siempre el comportamiento.

Si el cliente Finesse no envía mensajes http-bind al servidor Finesse, los registros pueden mostrar el tiempo de actividad de la sesión y mostrar el cierre de la sesión.

```
2017.06.17 00:14:34 Session (id=f382a015) was last active 0 ms ago: 1001003@xxxxx.xxxx.xxx.cisco.com/des
2017.06.17 00:15:04 Session (id=f382a015) was last active 13230 ms ago: 1001003@xxxxx.xxxx.xxx.cisco.com
2017.06.17 00:15:34 Session (id=f382a015) was last active 43230 ms ago: 1001003@xxxxx.xxxx.xxx.cisco.com
2017.06.17 00:16:04 Session (id=f382a015) was last active 63231 ms ago: 1001003@xxxxx.xxxx.xxx.cisco.com

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

### Registros de Info Notification Service

Estos registros se encuentran en la carpeta /desktop/logs/openfire y se denominan info.log. Si el cliente Finesse no envía mensajes http-bind al servidor Finesse, los registros pueden mostrar que la sesión se vuelve inactiva.

```
2017.06.17 00:16:04 Closing idle session (id=f382a015): 1001003@xxxxx.xxxx.xxx. cisco.com/desktop
after inactivity for more than threshold value of 60
2017.06.17 00:16:04 A session is closed for 1001003@xxxxx.xxxx.xxx. cisco.com/desktop
```

### Registros de Webservices

Estos registros se encuentran en la carpeta /desktop/logs/webservices y se denominan Desktop-webservices.AAAA-MM-DDTHH-MM-SS.sss.log. Si el cliente Finesse no envía mensajes http-bind al servidor Finesse dentro de la cantidad de tiempo especificada, los registros pueden mostrar que la presencia del agente deja de estar disponible y 60 segundos después, se puede producir un cierre de sesión controlado por la presencia.

```
0000001043: XX.XX.XX.XXX: Jun 17 2017 00:16:04.630 +0530: %CCBU_Smack Listener Processor (1)-6-PRESENCE_
0000000417: XX.XX.XX.XXX: Jun 17 2017 00:16:04.631 +0530: %CCBU_Smack Listener Processor (1)-6-UNSUBSCRI
0000001044: XX.XX.XX.XXX:  Jun 17 2017 00:16:04.631 +0530: %CCBU_Smack Listener Processor (1)-6-AGENT_PR
0000001051: XX.XX.XX.XXX:  Jun 17 2017 00:16:35.384 +0530: %CCBU_pool-8-thread-1-6-AGENT_PRESENCE_MONITO
0000001060: XX.XX.XX.XXX:: Jun 17 2017 00:17:04.632 +0530: %CCBU_CoreImpl-worker12-6-PRESENCE DRIVEN LOG
0000001061: XX.XX.XX.XXX:: Jun 17 2017 00:17:04.633 +0530: %CCBU_CoreImpl-worker12-6-MESSAGE_TO_CTI_SERV
1, workmode : 0, reason code: 255, forceflag :1, agentcapacity: 1, agentext: 1001003, agentid: 1001003,
0000001066:  XX.XX.XX.XXX:: Jun 17 2017 00:17:04.643 +0530: %CCBU_CTIMessageEventExecutor-0-6-DECODED_ME
skillGroupNumber=-1, skillGroupPriority=0, agentState=1 (LOGOUT), eventReasonCode=255, numFltSkillGroups
duration=null, nextAgentState=null, fltSkillGroupNumberList=[], fltSkillGroupIDList=[], fltSkillGroupPri
msgID=30, timeTracker={"id":"AgentStateEvent","CTI_MSG_RECEIVED":1497638824642,"CTI_MSG_DISPATCH":149763
Decoded Message to Finesse from backend cti server
```

## Razones comunes para la desconexión de BOSH

Las conexiones BOSH son configuradas por el cliente web, y el servidor Finesse determina si la presencia del agente no está disponible. Estos problemas son casi siempre problemas del lado del cliente relacionados con el navegador, el equipo del agente o la red, ya que la responsabilidad de iniciar la conexión depende del cliente.

## Problema: los agentes se desconectan en distintos momentos (problema del lado del cliente)

### Acciones recomendadas

Compruebe estos problemas:

1. Problema de red:

- Revisar reglas y registros del firewall: el puerto TCP 7443 no debe bloquearse ni acelerarse

- Utilice un rastreador de tráfico web HTTP como Fiddler® o Wireshark® para confirmar que el navegador envía solicitudes http-bind a través del puerto TCP 7443 y recibe respuestas

- Verifique todos los dispositivos de red/interfaces entre el equipo agente y el servidor Finesse para detectar retrasos excesivos o caídas de paquetes
    - Traceroute puede ser útil para determinar la trayectoria y los retrasos
        - En un PC con Microsoft® Windows®: tracert {Finesse Server IP | FQDN de servidor Finesse}
        - En un Mac®: traceroute {Finesse Server IP | FQDN de servidor Finesse}
        - En el software Cisco IOS®, se pueden verificar las estadísticas de la interfaz: show interfaces
            - Consulte Resolución de Problemas de Caídas de Cola de Entrada y Caídas de Cola de Salida

- Recopilar registros del cliente Finesse para un agente de prueba. Los registros de clientes se pueden recopilar de tres maneras:
    1. Registros de la consola web del navegador
        - Consola web de Firefox
        - Consola web de Microsoft Edge
        - Consola web Chrome
    2. Presione el botón Send Error Report en la página Finesse y recopile los registros del servidor Finesse. Los registros se encuentran en /desktop/logs/clientlogs.
    3. Inicie sesión mediante https://<Finesse-FQDN>/desktop/locallog y recopile los registros después de que se produzca el problema.

Cada minuto, el cliente se conecta al servidor Finesse para calcular la variabilidad y la latencia de red:

```
<PC date-time with GMT offset>: : <Finesse FQDN>: <Finesse server date-time with offset>:
Header : Client: <date-time>, Server: <date-time>, Drift: <drift> ms, Network Latency (round trip): <RT

2019-01-11T12:24:14.586 -05:00: : fin1.ucce.local: Jan 11 2019 11:24:14.577 -0600: Header : Client: 2019
```

En caso de problemas de recopilación de registros, consulte Resolución de problemas de registro persistente de Cisco Finesse Desktop

2. Navegador y/o versión no compatible:

Utilice las configuraciones y la versión del navegador compatibles según las matrices de compatibilidad:

Matriz de compatibilidad de UCCE

Matriz de compatibilidad de UCCX

3. Estado de bloqueo del navegador debido al contenido/procesamiento de otra pestaña/ventana:

Compruebe el flujo de trabajo del agente para ver si:

- Suele tener otras fichas o ventanas activas que ejecutan constantemente otras aplicaciones en tiempo real, como transmisión de música o vídeo, conexiones WebSocket, clientes Web personalizados de gestión de relaciones con los clientes (CRM), etc
- Tienen un gran número de pestañas o ventanas abiertas
- Deshabilitar almacenamiento en caché del explorador

- Han mantenido su navegador en funcionamiento durante mucho tiempo y no lo cierran al final de la jornada laboral

4. Ordenador puesto en suspensión:

Compruebe si el agente pone el equipo en suspensión antes de cerrar sesión en Finesse o si el temporizador de configuración de suspensión del equipo es muy bajo.

5. Uso excesivo de la CPU o problema de memoria alta en la computadora cliente:

- Si el explorador del agente se ejecuta en un entorno compartido como Microsoft Windows Remote Desktop Services, Citrix® XenApp®, Citrix XenDesktop®, determine si el rendimiento del explorador depende del número de usuarios que lo ejecuten al mismo tiempo
    - Asegúrese de que la memoria y los recursos de CPU adecuados estén configurados en función del número de usuarios

- Comprobar problemas de utilización de recursos informáticos:
    - Windows:
        - El comando Windows PowerShell Get-Counter que comprueba % de tiempo de CPU, Megabytes de memoria disponibles y % de memoria en uso cada 2 segundos: Get-Counter -Counter "\Processor(_Total)\% Processor Time","\Memory\Available MBytes","\Memory\% Committed Bytes In Use" -SampleInterval 2 -Continuous
        - Como alternativa al uso de PowerShell para ver los contadores de rendimiento de Windows, se puede utilizar el Monitor de rendimiento de Windows
        - El Administrador de tareas se puede utilizar para ver estadísticas de memoria y CPU en directo de forma global y proceso por proceso
    - Mac:
        - Terminal Top que verifica la CPU y la memoria totales en vivo: top
            - Compruebe los procesos y ordénelos por el uso de la CPU: arriba -o CPU
            - Compruebe los procesos y ordénelos por el uso de la memoria: top -o MEM
        - El Control de actividad se puede utilizar para ver estadísticas de memoria y CPU en directo de forma global y proceso por proceso

6. gadgets de terceros que realizan una actividad problemática e inesperada en segundo plano:

Pruebe el comportamiento del escritorio Finesse con todos los gadgets de terceros eliminados.

7. Problema NTP en el servidor o el cliente:

- Verifique **utils ntp status** en el servidor editor Finesse para asegurarse de que el estrato del servidor NTP sea 4 o inferior
- En los registros del cliente, compruebe la variabilidad y la latencia de red

## Problema: todos los agentes se desconectan al mismo tiempo (problema en el servidor)

### Acciones recomendadas

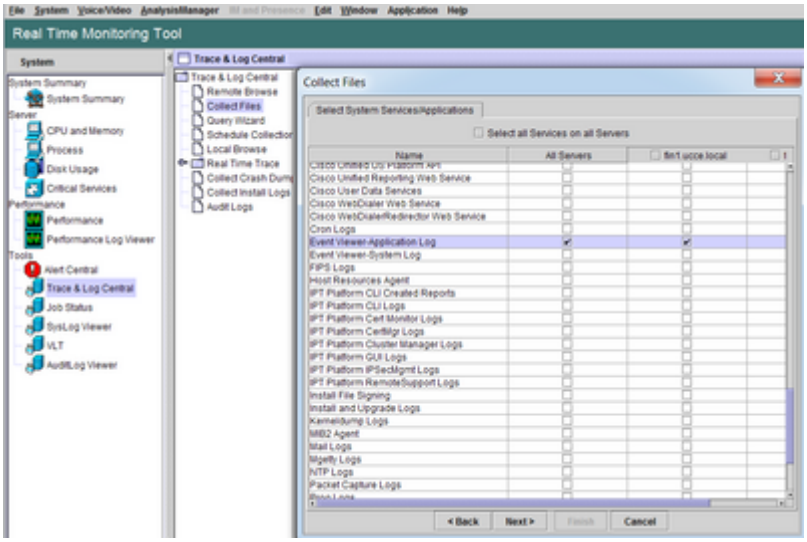
Compruebe estos problemas:

1. Servicio Cisco Unified Communications Manager CTIManager desconectado. Si todos los proveedores de CTIManager para UCCX se apagan o se bloquean, los agentes UCCX verán el error de banner rojo. Los agentes de UCCE no ven el banner rojo si esto sucede, pero las llamadas no se enrutan correctamente a los agentes.

- Compruebe si el servicio Cisco CTIManager se ha iniciado en los servidores CUCM utilizados como

proveedores CTI

- Compruebe si el servicio Cisco CTIManager falló a través del Visor de eventos: la aplicación inicia sesión en RTMT para ver si el servicio Cisco CTIManager falló
    - Para recopilar los registros del visor de eventos en RTMT, vaya a **System > Tools > Trace y Log Central > Collect Files > Select System Services/Applications > Event Viewer-Application Log**.




- Para recopilar los registros de Event Viewer-Application en CLI: file get activelog /syslog/CiscoSyslog* abstime hh:mm:MM/DD/YY hh:mm:MM/DD/YY
- Para ver vaciados de memoria en la CLI: lista activa de núcleo de utils

---

**Nota**: Los nombres de archivo de vaciados de memoria utilizan el formato: core.<ProcessID>.<SignalNumber>.<ProcessName>.<EpochTime>.
Ejemplo: core.24587.6.CTIManager.1533441238
Por lo tanto, el tiempo del accidente se puede determinar a partir del tiempo de la época.

---

2. Finesse/UCCX Notification Service detenido o bloqueado:

- Compruebe los registros de la aplicación del Visor de eventos para ver si hay errores de Servicio de notificación o si el servicio se ha detenido
- Compruebe si el servicio de notificación está activo: lista de servicios de utils
- Compruebe las horas de cierre del servicio de notificación: file search activelog /desktop/logs/openfire "Openfire se detuvo"
- Compruebe las horas en que se inició el servicio de notificación: file search activelog /desktop/logs/openfire "HTTP bind service started"
- Compruebe si hay volcados de memoria del servicio de notificación como resultado de un desperfecto: file list activelog /desktop/logs/openfire/*.hprof
- Verifique si el servicio de notificación está escuchando el tráfico en el puerto TCP 7443: show open ports regexp 7443.*LISTEN
- Verifique si estos defectos son aplicables (estos defectos podrían causar un error de inicio de sesión para los agentes que inician sesión y para los agentes que ya han iniciado sesión, estos agentes verían el mensaje de desconexión de Finesse de banner rojo):
    - ID de bug de Cisco CSCva7280 - Finesse Tomcat y Openfire Crash para caracteres XML no válidos
    - Id. de error de Cisco CSCva72325 - UCCX: Finesse Tomcat y Openfire Crash para caracteres XML no válidos

Reinicie Cisco Finesse Tomcat and Notification Service si se sospecha que se ha producido un crash. Esto solo se recomienda en una situación de inactividad de la red; de lo contrario, estos reinicios desconectan a los agentes del servidor Finesse.

Pasos para UCCE:

- utils service stop Cisco Finesse Tomcat
- utils service stop Servicio de notificación de Cisco Finesse
- utils service start Cisco Finesse Tomcat
- utils service start Servicio de notificación de Cisco Finesse

Pasos para UCCX:

- utils service stop Cisco Finesse Tomcat
- utils service stop Servicio de notificación de Cisco Unified CCX
- utils service start Cisco Finesse Tomcat
- utils service start Servicio de notificación de Cisco Unified CCX

## Utilizar Fiddler

Configurar Fiddler puede ser una tarea un tanto desafiante sin entender los pasos necesarios y entender cómo funciona Fiddler. Fiddler es un proxy web man-in-the-middle que se encuentra entre el cliente Finesse (navegador web) y el servidor Finesse. Debido a las conexiones seguras entre el cliente Finesse y el servidor Finesse, esto añade una capa de complejidad a la configuración de Fiddler para ver los mensajes seguros.

**Problema común de Fiddler**

Dado que Fiddler se encuentra entre el cliente Finesse y el servidor Finesse, la aplicación Fiddler necesita crear certificados firmados para todos los puertos TCP de Finesse que requieren certificados:

Certificados de servicio Tomcat de Cisco Finesse

1. Servidor editor Finesse TCP 8445 (y/o 443 para UCCE)
2. Servidor de suscriptor Finesse TCP 8445 (y/o 443 para UCCE)

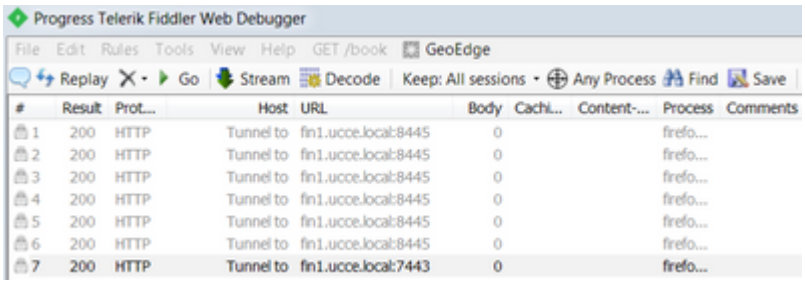
Certificados del servicio de notificación de Cisco Finesse (Unified CCX)

1. Servidor editor Finesse TCP 7443
2. Servidor de suscriptor Finesse TCP 7443

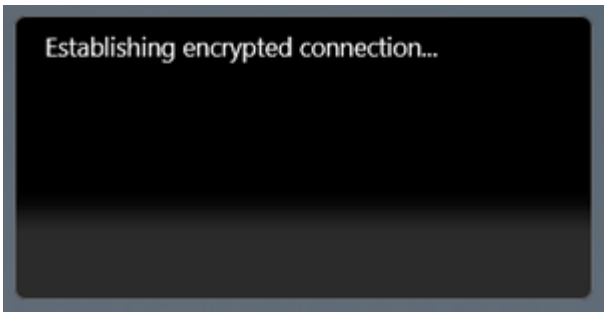
El descifrado HTTPS debe estar habilitado para que Fiddler genere dinámicamente certificados en nombre del servidor Finesse. Esta opción no está activada de forma predeterminada.

Si no se configura el descifrado HTTPS, se ve la conexión de túnel inicial con el servicio de notificación, pero no el tráfico http-bind. Fiddler solo muestra:

```
Tunnel to <Finesse server FQDN>:7443
```




A continuación, el cliente debe confiar en los certificados Finesse firmados por Fiddler. Si estos certificados no son de confianza, no es posible pasar de la fase Establecimiento de la conexión cifrada... del inicio de sesión de Finesse.




En algunos casos, la aceptación de las excepciones de certificado del inicio de sesión no funciona y el explorador debe confiar en los certificados manualmente.

**Pasos de configuración de ejemplo**

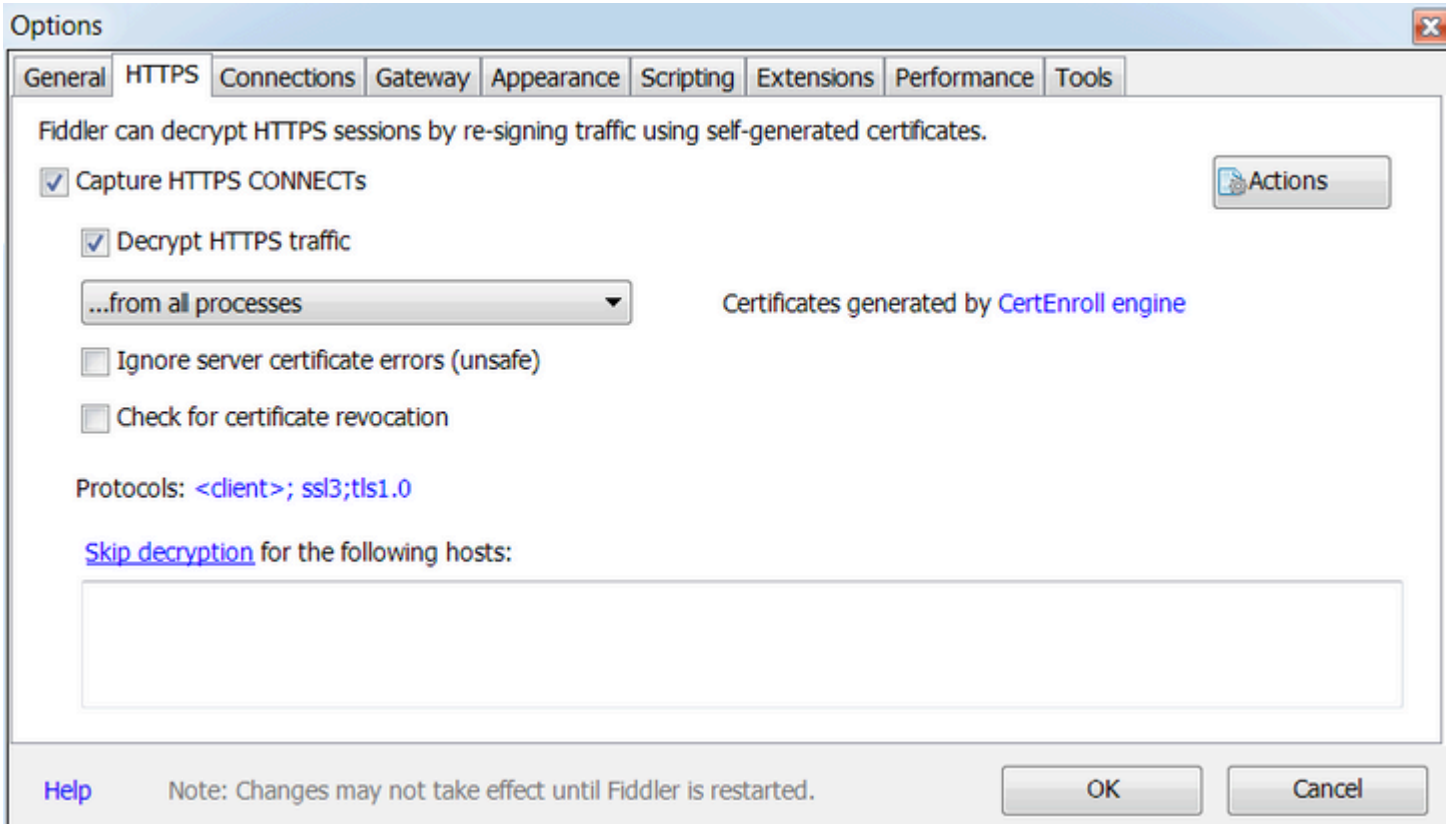
---

> **Precaución**: el ejemplo de configuración proporcionado es para Fiddler v5.0.20182.28034 para .NET 4.5 y Mozilla Firefox 64.0.2 (32 bits) en Windows 7 x64 en un entorno de laboratorio. Estos procedimientos no se pueden generalizar a todas las versiones de Fiddler, a todos los exploradores o a todos los sistemas operativos de los equipos. Si su red está activa, asegúrese de comprender el impacto potencial de cualquier configuración. Consulte la documentación oficial de Fiddler para obtener más información.

---

Paso 1. Descargar Fiddler

Paso 2. Habilite el descifrado HTTPS. Navegue hasta **Herramientas > Opciones > HTTPS** y marque la casilla de verificación **Descifrar tráfico HTTPS**.

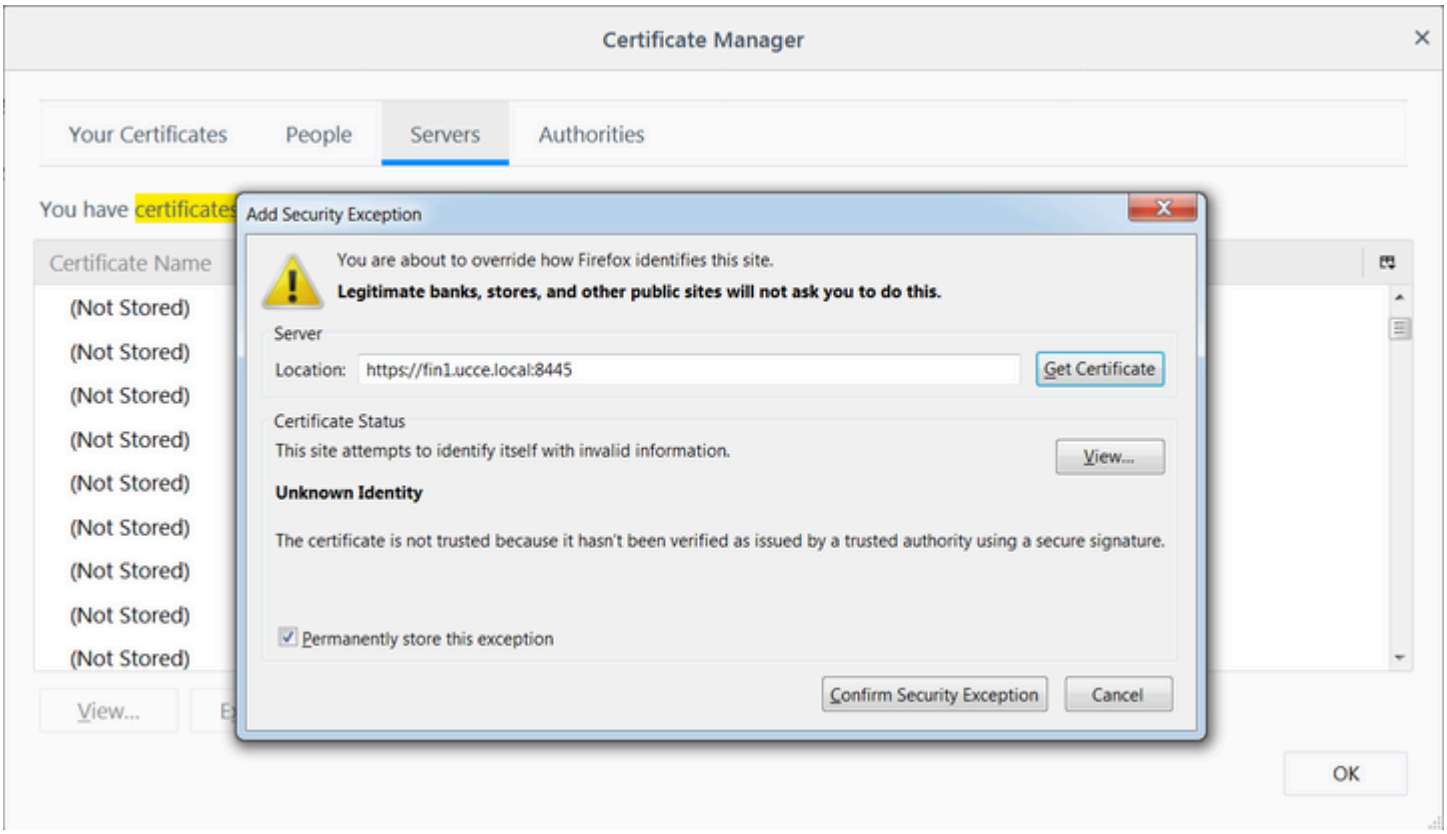



Paso 3. Se abre un cuadro de mensaje de advertencia para solicitar que se confíe en el certificado raíz de Fiddler. Seleccione **Sí**.

Paso 4. Se abre un cuadro de mensaje de advertencia con el mensaje "Va a instalar un certificado de una entidad emisora de certificados (CA) que afirma representar: DO_NOT_TRUST_FiddlerRoot... ¿Desea instalar este certificado?". Seleccione **Sí**.

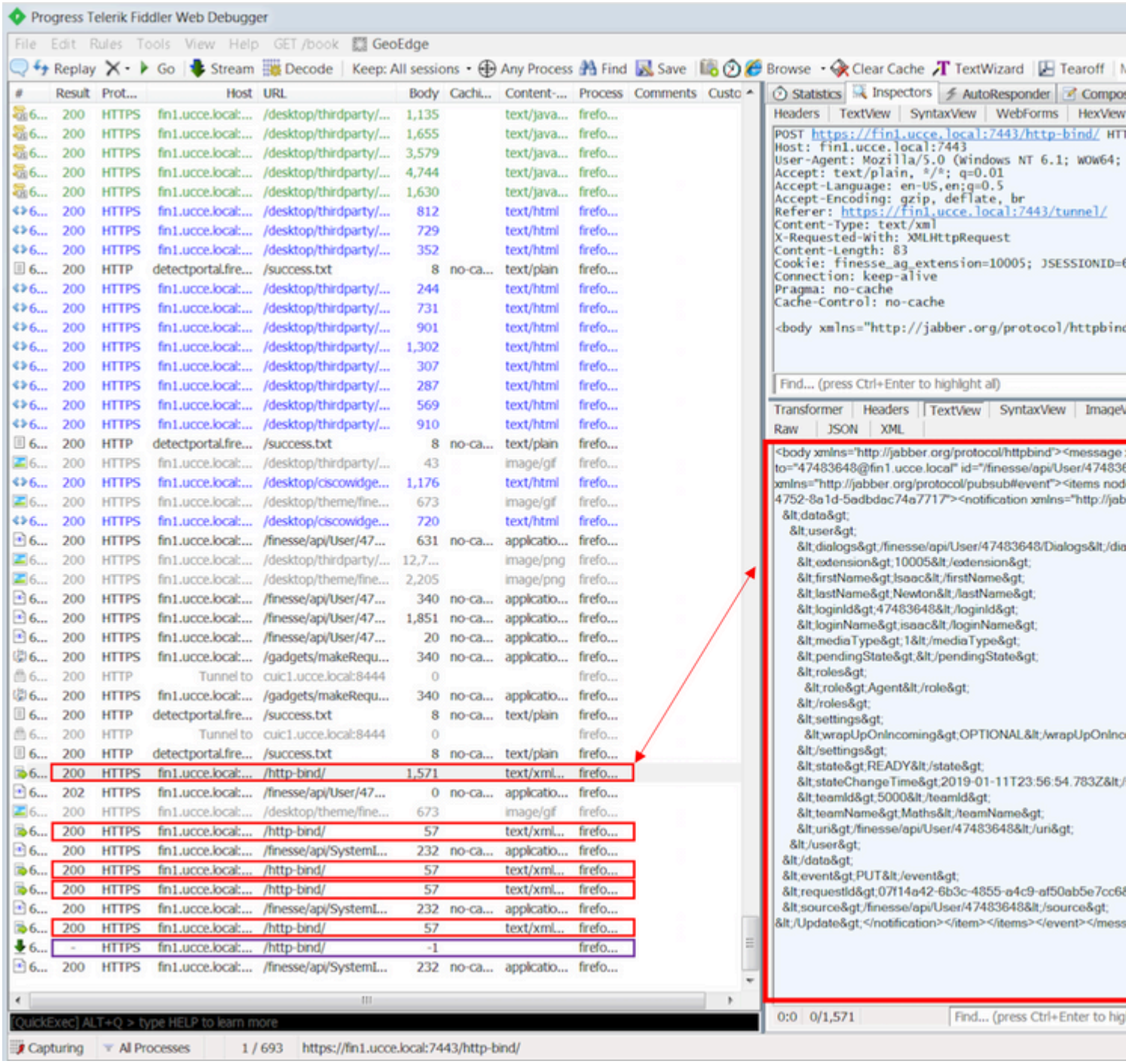
Paso 5. Agregue manualmente los certificados de editor y suscriptor de Finesse al almacén de confianza de certificados del equipo o del explorador. Asegúrese de los puertos 8445, 7443 y (solo para UCCE) 443. Por ejemplo, en Firefox, esto se puede hacer simplemente sin descargar certificados de la página de administración del sistema operativo Finesse:

**Options > Find in Options** (search) **> Certificates > Servers > Add Exception > Location >** Enter https://<Finesse server>:port para los puertos relevantes de ambos servidores Finesse.

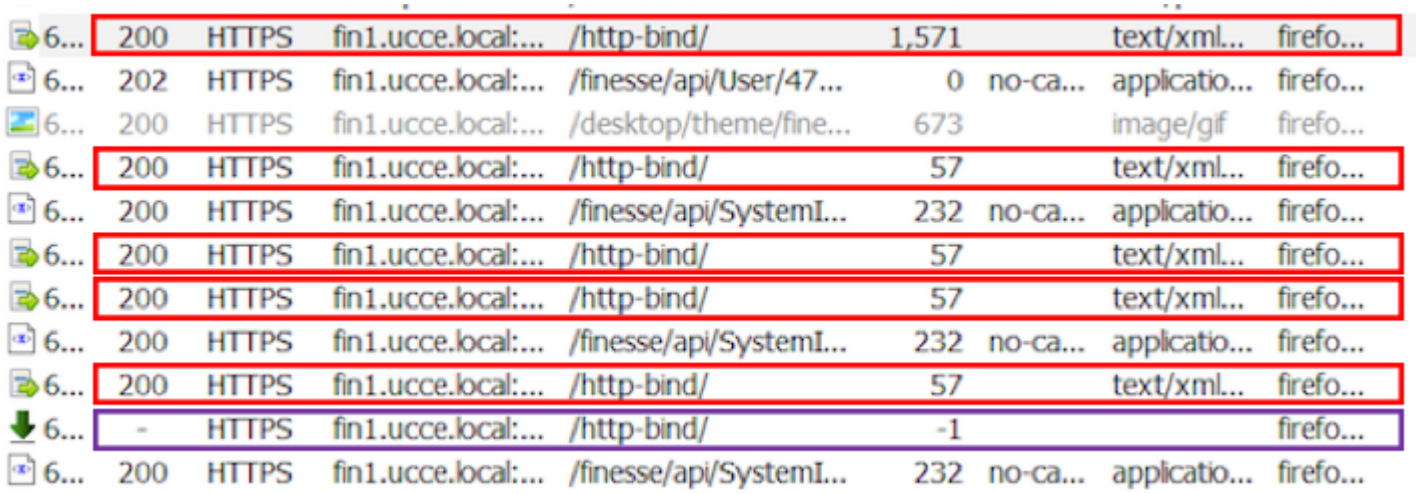



Paso 6. Inicie sesión en Finesse y vea los mensajes http-bind que dejan el cliente Finesse al servidor Finesse a través de Fiddler.

En el ejemplo proporcionado, los primeros 5 mensajes muestran mensajes http-bind que fueron respondidos por el servidor Finesse. El primer mensaje contiene 1571 bytes de datos devueltos en el cuerpo del mensaje. El cuerpo contiene una actualización XMPP relativa a un evento de agente. El mensaje http-bind final ha sido enviado por el cliente Finesse, pero no ha obtenido una respuesta del servidor Finesse. Esto se puede determinar cuando se observa que el resultado HTTP es nulo (-) y el número de bytes del cuerpo de la respuesta es nulo (-1).




Vista más cercana de los datos:



| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 6... | 200 | HTTPS | fin1.ucce.local:... | /http-bind/ | 1,571 | | text/xml... | firefo... |
| 6... | 202 | HTTPS | fin1.ucce.local:... | /finesse/api/User/47... | 0 | no-ca... | applicatio... | firefo... |
| 6... | 200 | HTTPS | fin1.ucce.local:... | /desktop/theme/fine... | 673 | | image/gif | firefo... |
| 6... | 200 | HTTPS | fin1.ucce.local:... | /http-bind/ | 57 | | text/xml... | firefo... |
| 6... | 200 | HTTPS | fin1.ucce.local:... | /finesse/api/SystemI... | 232 | no-ca... | applicatio... | firefo... |
| 6... | 200 | HTTPS | fin1.ucce.local:... | /http-bind/ | 57 | | text/xml... | firefo... |
| 6... | 200 | HTTPS | fin1.ucce.local:... | /http-bind/ | 57 | | text/xml... | firefo... |
| 6... | 200 | HTTPS | fin1.ucce.local:... | /finesse/api/SystemI... | 232 | no-ca... | applicatio... | firefo... |
| 6... | 200 | HTTPS | fin1.ucce.local:... | /http-bind/ | 57 | | text/xml... | firefo... |
| 6... | - | HTTPS | fin1.ucce.local:... | /http-bind/ | -1 | | | firefo... |
| 6... | 200 | HTTPS | fin1.ucce.local:... | /finesse/api/SystemI... | 232 | no-ca... | applicatio... | firefo... |

Cuerpo de la respuesta para el mensaje XMPP:


```
<body xmlns='http://jabber.org/protocol/httpbind'><message xmlns="jabber:client" from="pubsub.fin1.ucce.local"
to="47483648@fin1.ucce.local" id="/finesse/api/User/47483648__47483648@fin1.ucce.local__K7hYF"><event
xmlns="http://jabber.org/protocol/pubsub#event"><items node="/finesse/api/User/47483648"><item id="26a3e421-
4752-8a1d-5adbdac74a7717"><notification xmlns="http://jabber.org/protocol/pubsub">&lt;Update&gt;
 &lt;data&gt;
  &lt;user&gt;
   &lt;dialogs&gt;/finesse/api/User/47483648/Dialogs&lt;/dialogs&gt;
   &lt;extension&gt;10005&lt;/extension&gt;
   &lt;firstName&gt;Isaac&lt;/firstName&gt;
   &lt;lastName&gt;Newton&lt;/lastName&gt;
   &lt;loginId&gt;47483648&lt;/loginId&gt;
   &lt;loginName&gt;isaac&lt;/loginName&gt;
   &lt;mediaType&gt;1&lt;/mediaType&gt;
   &lt;pendingState&gt;&lt;/pendingState&gt;
   &lt;roles&gt;
    &lt;role&gt;Agent&lt;/role&gt;
   &lt;/roles&gt;
   &lt;settings&gt;
    &lt;wrapUpOnIncoming&gt;OPTIONAL&lt;/wrapUpOnIncoming&gt;
   &lt;/settings&gt;
   &lt;state&gt;READY&lt;/state&gt;
   &lt;stateChangeTime&gt;2019-01-11T23:56:54.783Z&lt;/stateChangeTime&gt;
   &lt;teamId&gt;5000&lt;/teamId&gt;
   &lt;teamName&gt;Maths&lt;/teamName&gt;
   &lt;uri&gt;/finesse/api/User/47483648&lt;/uri&gt;
  &lt;/user&gt;
 &lt;/data&gt;
 &lt;event&gt;PUT&lt;/event&gt;
 &lt;requestId&gt;07f14a42-6b3c-4855-a4c9-af50ab5e7cc6&lt;/requestId&gt;
 &lt;source&gt;/finesse/api/User/47483648&lt;/source&gt;
&lt;/Update&gt;</notification></item></items></event></message></body>
```


## Utilizar Wireshark

Wireshark es una herramienta de rastreo de paquetes de uso común que se puede utilizar para rastrear y decodificar el tráfico HTTPS. El tráfico HTTPS es tráfico HTTP protegido mediante la seguridad de la capa de transporte (TLS). TLS proporciona integridad, autenticación y confidencialidad entre dos hosts. Se utiliza

habitualmente en aplicaciones web, pero se puede utilizar con cualquier protocolo que utilice TCP como protocolo de capa de transporte. Secure Sockets Layer (SSL) es la versión anterior del protocolo TLS, que ya no se utiliza porque es inseguro. Estos nombres se utilizan a menudo indistintamente, y el filtro Wireshark utilizado para el tráfico SSL o TLS es ssl.

---

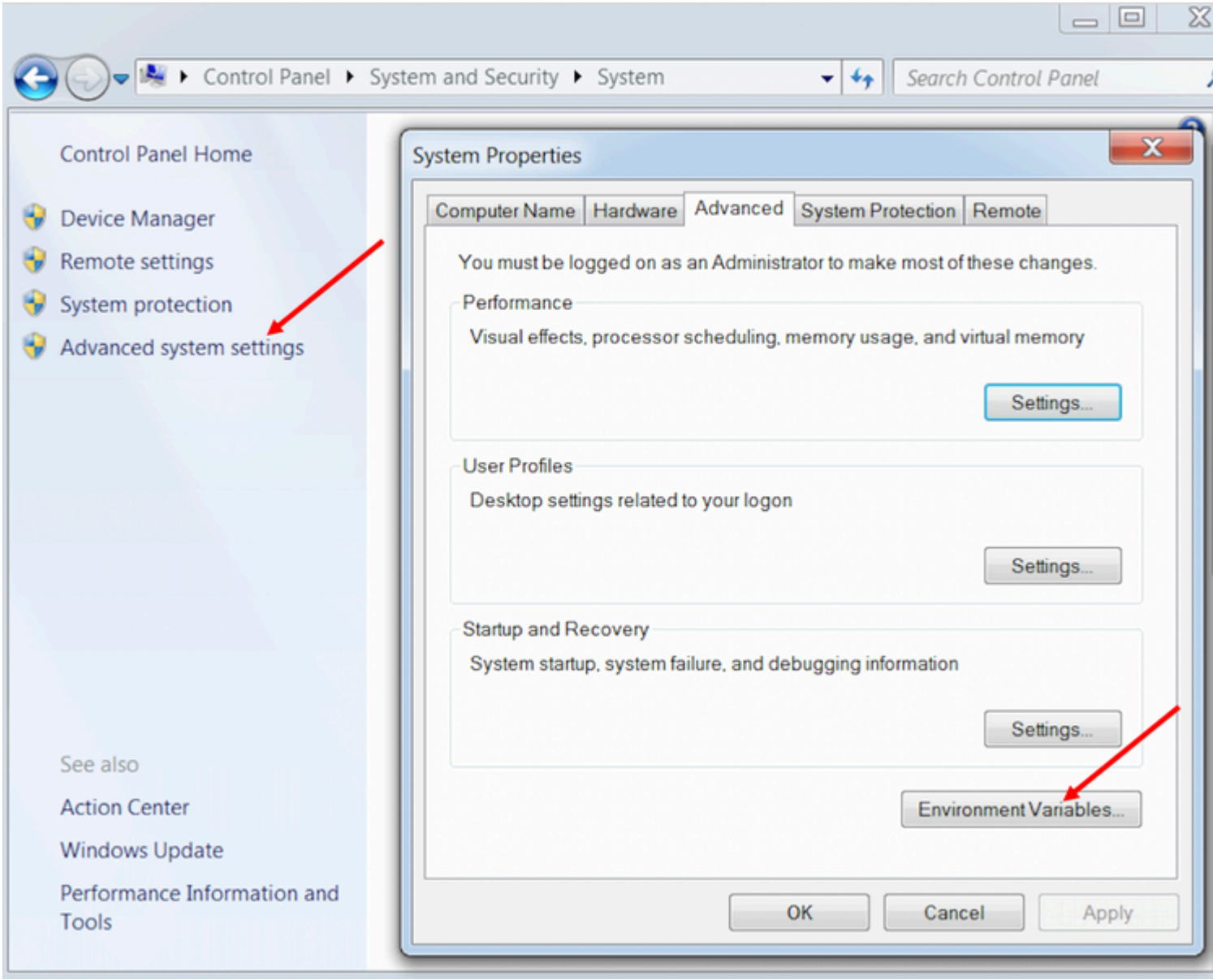
> **Precaución**: el ejemplo de configuración proporcionado es para Wireshark 2.6.6 (v2.6.6-0-gdf942cd8) y Mozilla Firefox 64.0.2 (32 bits) en Windows7 x64 en un entorno de laboratorio. Estos procedimientos no pueden generalizarse a todas las versiones de Fiddler, a todos los navegadores o a todos los sistemas operativos de los equipos. Si su red está activa, asegúrese de comprender el impacto potencial de cualquier configuración. Consulte la documentación oficial de Wireshark SSL para obtener más información. Se requiere Wireshark 1.6 o superior.

---

> **Nota**: Este método solo puede funcionar para Firefox y Chrome. Este método no funciona para Microsoft Edge.

---

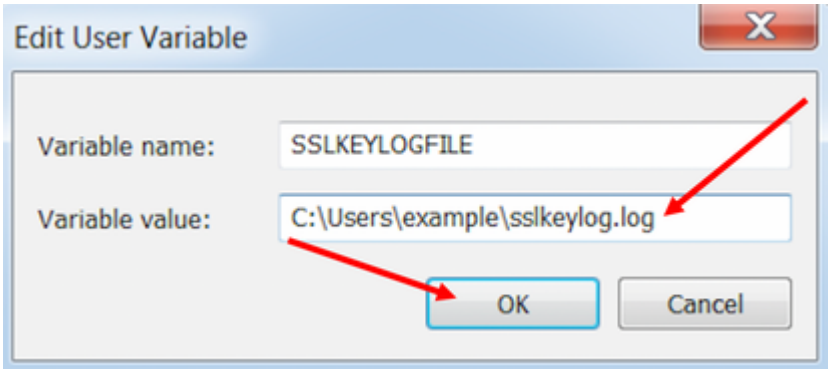
Paso 1. En la PC con Windows del agente, navegue hasta **Panel de control > Sistema y seguridad > Sistema > Configuración avanzada del sistema Variables ambientales...**



Paso 2. Navegue hasta **Variables de usuario para el usuario <username> > Nuevo...**

Cree una variable denominada **SSLKEYLOGFILE.**

Cree un archivo para almacenar el secreto de premaster SSL en un directorio privado: SSLKEYLOGFILE=</path/to/private/directory/with/logfile>



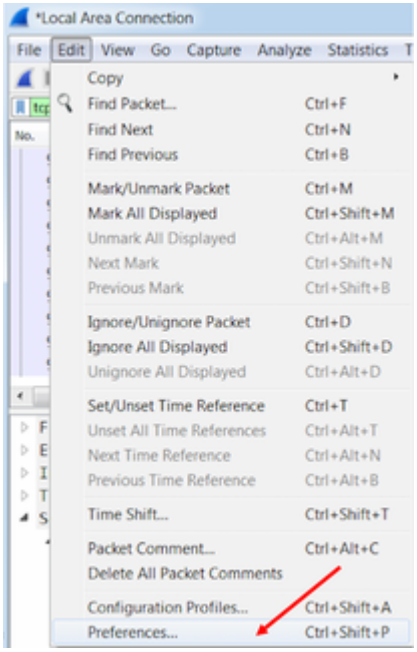

---

> **Nota**: Cree una variable de sistema en lugar de una variable de usuario y/o almacene el archivo en un directorio no privado, pero todos los usuarios del sistema podrán acceder al secreto de premaster, que es menos seguro.

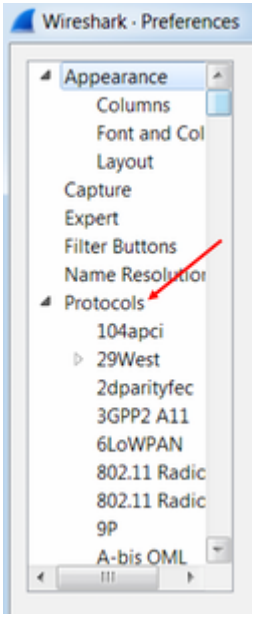
---

Paso 3. Si Firefox o Chrome están abiertos, cierre las aplicaciones. Una vez reabiertos, pueden comenzar a escribir en SSLKEYLOGFILE.

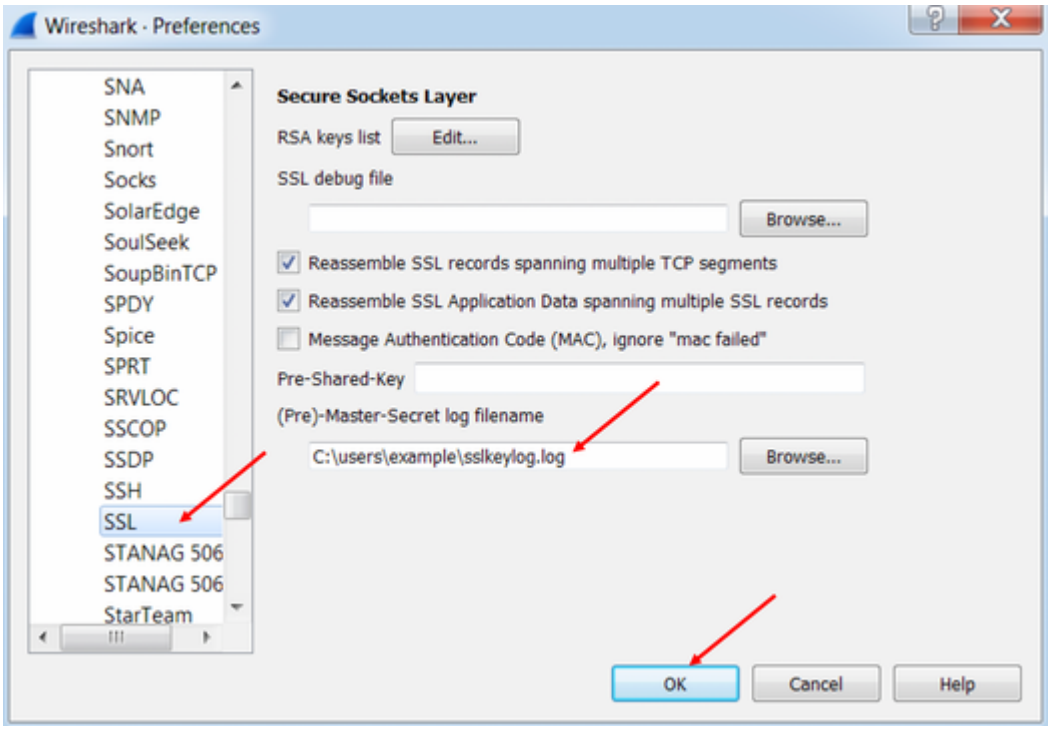
Paso 4. En Wireshark, vaya a **Edición > Preferencias...**



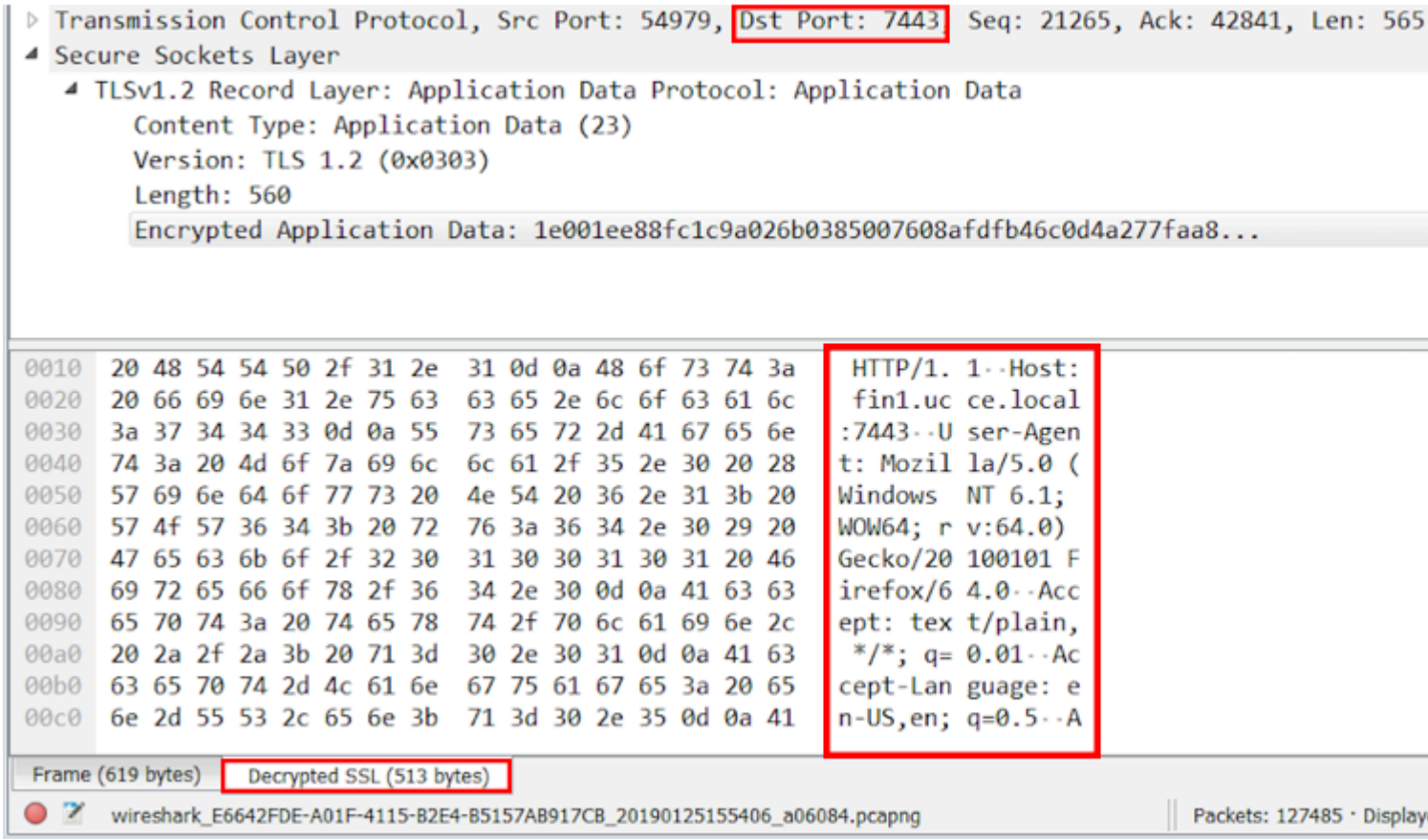

Vaya a **Protocolos > SSL**.




Paso 5. Introduzca la ubicación del nombre de archivo del registro secreto de premaster configurado en el paso 2.




Paso 6. Utilice el filtro Wireshark **tcp.port==7443 && ssl**, la comunicación HTTP segura entre el cliente Finesse y el servidor Finesse (Servicio de notificación) se ve descifrada.




### Defectos relacionados

- ID de bug de Cisco CSCva7280 - Finesse Tomcat y Openfire Crash para caracteres XML no válidos
- ID de error de Cisco CSCva72325 - UCCX: Finesse Tomcat y Openfire Crash para caracteres XML no válidos

## Información Relacionada

- Especificaciones de XMPP
- XEP-0124: BOSH
- XEP-0060: Publicar-Suscribir
- Consola web de Firefox
- Consola web de Microsoft Edge
- Consola web Chrome
- Windows PowerShell
- Monitor de rendimiento de Windows
- Resolución de problemas en los paquetes descartados en las colas de entrada y salida
- Administrador de tareas de Windows
- Terminal Mac
- Mac Activity Monitor
- Descarga de Fiddler
- Configuración de Fiddler
- Descarga de Wireshark
- Descifrado SSL de Wireshark
- Soporte Técnico y Documentación - Cisco Systems

Acerca de esta traducción
Cisco ha traducido este documento combinando la traducción automática y los recursos humanos a fin de ofrecer a nuestros usuarios en todo el mundo contenido en su propio idioma.

Tenga en cuenta que incluso la mejor traducción automática podría no ser tan precisa como la proporcionada por un traductor profesional.

Cisco Systems, Inc. no asume ninguna responsabilidad por la precisión de estas traducciones y recomienda remitirse siempre al documento original escrito en inglés (insertar vínculo URL).