

Troubleshooting de la memoria del CVP con Jconsole

Contenido

[Introducción](#)

[prerrequisitos](#)

[Requisitos](#)

[Componentes Utilizados](#)

[Utilidad de Jconsole](#)

Introducción

Este documento describe cómo utilizar la herramienta de la consola Java (jconsole) para resolver problemas los problemas de fuga de memoria del Cisco Unified Customer Voice Portal (CVP).

Prerrequisitos

Requisitos

Cisco recomienda que usted tiene conocimiento de estos temas

- Cisco Unified Customer Voice Portal (CVP)
- Utilidad de la consola Java

Componentes Utilizados

La información en este documento se basa en la versión 9.0(X) y posterior del CVP.

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 la red está funcionando, asegúrese de haber comprendido el impacto que puede tener cualquier comando.

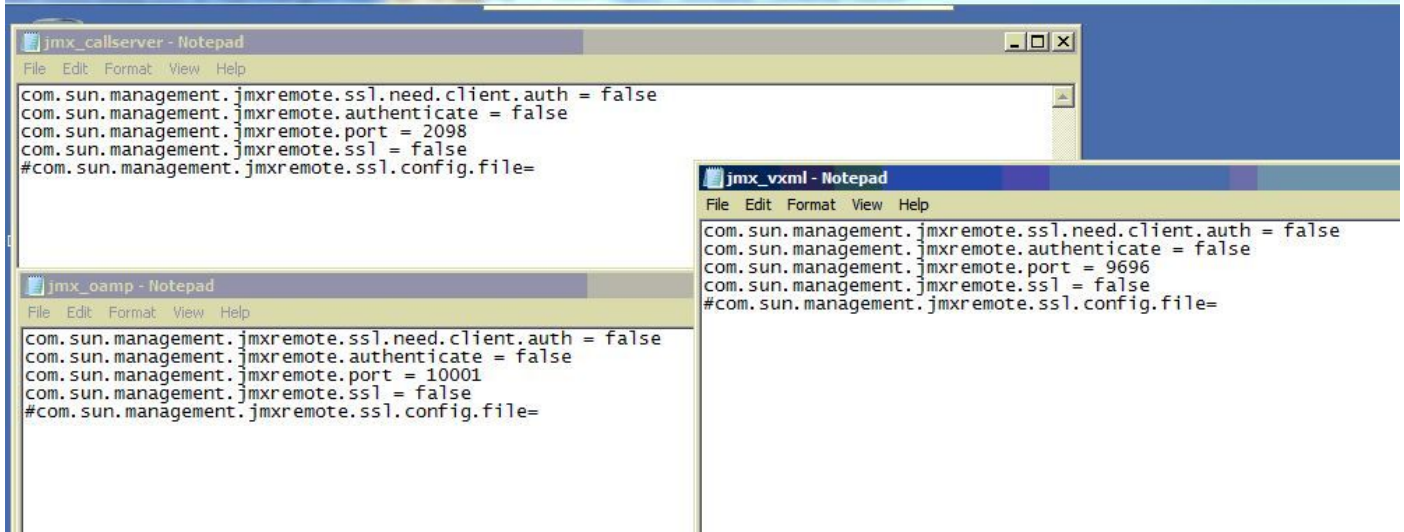
Utilidad de Jconsole

Para resolver problemas el servidor de la llamada, fuga de memoria o los asuntos relacionados con el rendimiento del servidor VXML es a menudo necesario dar vuelta encima de la traza del volcado del montón de las Javas en la utilidad del jconsole. Esto se hace generalmente después de estrecharse abajo de un problema de recurso a un servicio o a los servicios específicos en el servidor vía el visor de eventos, el administrador de tareas, y/o las herramientas de los registros del perfmon. El montón del volcado de las utilidades es un archivo de traza bajo y se recomienda ser dado vuelta para arriba encima del nivel de traza del Troubleshooting del CVP.

Jconsole por abandono está situado en la trayectoria "C:\Cisco\CVP\jre\bin" de los servidores del

servidor, VXML, y OAMP de la llamada y iniciar sesión en los archivos de configuración de la consola que usted debe primero configurar a los portes detalles en `jmx_callserver.conf`, `jmx_vxml.conf`, y `jmx_oamp.conf` de estos servidores respectivamente

- Puerto 2098 del servidor JMX de la llamada
- Puerto 9696 del servidor JMX VXML
- Puerto 10001 del servidor JMX OAMP

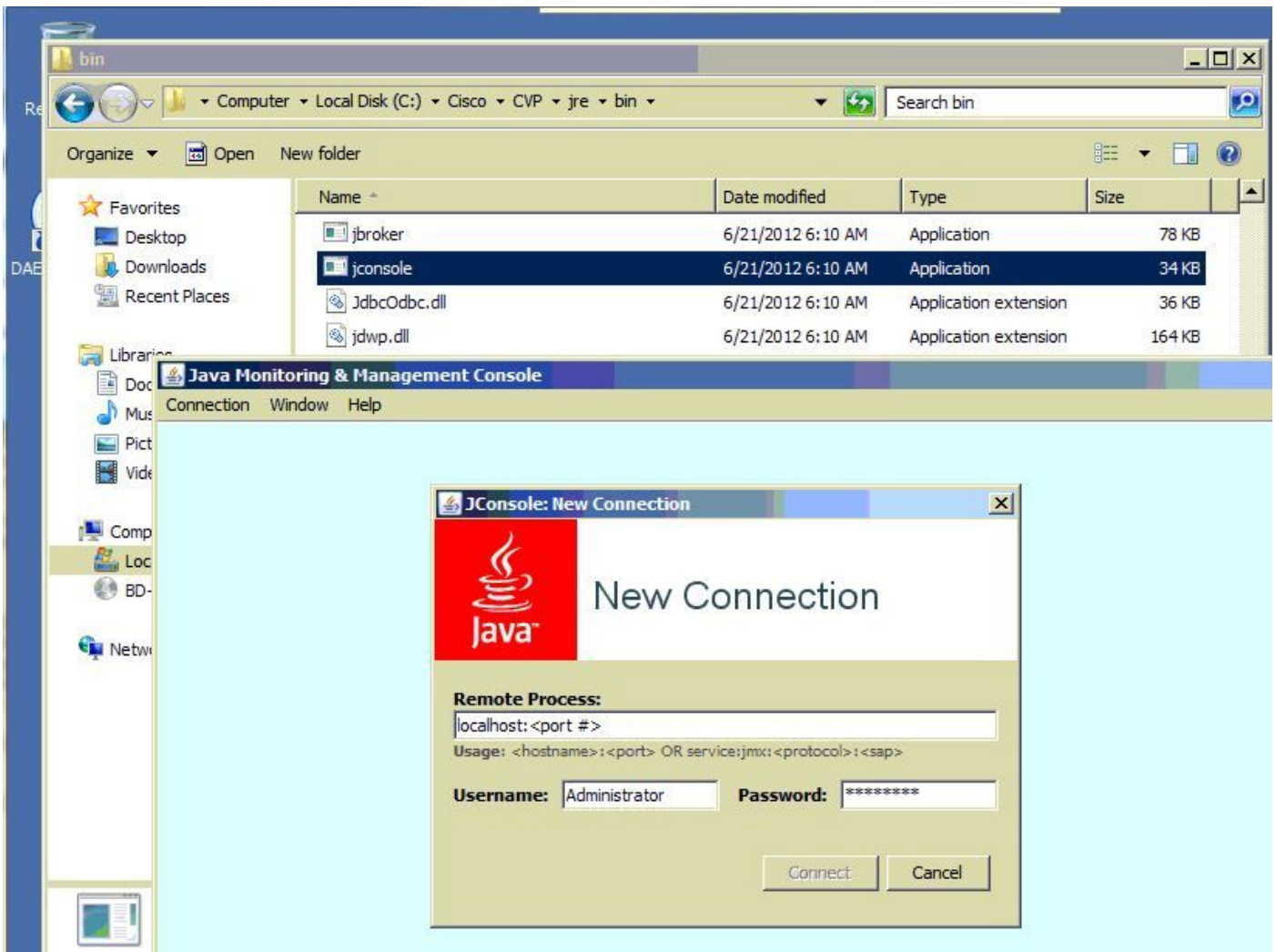


Después de que el archivo de configuración sea actualizado, ejecute el `jconsole` utilitario como se explica en estos pasos

Step1. Abra `%CVP_HOME%/CVP/jre/bin/jconsole.ex3`.

Step2. Conecte con el `localhost`, y especifique el número del puerto correspondiente al puerto del servidor JMX de Callserver, VXML u OAMP.

Step3. Inicie sesión con los credenciales del login del oamp del CVP o los credenciales Admin.



Step4. Abra la lengüeta de MBeans.

Step5. Abra com.sun.management - HotSpotDiagnostics - Operaciones - dumpheap.

Step6. En el p0 ingrese un nombre del archivo y una trayectoria a los cuales usted quiera dirigir el vaciado. Deje el p1 como verdad. Haga clic en el botón del dumpHeap y recoja el archivo generado.

- +
 - +
 - +
 - +
 - +
 -
 -
 - +
 -
 - +
 - +
- Catalina
 JImplementation
 Users
 com.sun.management
 HotSpotDiagnostic
 Attributes
 Operations
 dumpHeap
 getVMOption
 setVMOption
 java.lang
 java.util.logging

Operation invocation

```
void dumpHeap ( p0 String , p1 true )
```

MBeanOperationInfo

Name	Value
Operation:	
Name	dumpHeap
Description	dumpHeap
Impact	UNKNOWN
ReturnType	void
Parameter-0:	
Name	p0
Description	p0
Type	java.lang.String
Parameter-1:	
Name	p1
Description	p1
Type	boolean

Descriptor

Name	Value
Operation:	
openType	javax.management.openmbean.SimpleType(name=java.lang.Void)
originalType	void
Parameter-0:	
openType	javax.management.openmbean.SimpleType(name=java.lang.String)
originalType	java.lang.String
Parameter-1:	
openType	javax.management.openmbean.SimpleType(name=java.lang.Boolean)
originalType	boolean