Sammeln von Heap Dump und Thread Dump vom CVP VXML-Server

Inhalt

Einführung Voraussetzungen Anforderungen Verwendete Komponenten Problem Fehlerbehebung Schritte zum Abrufen von Heap-Dump Schritte zum Abrufen von Thread-Dump

Einführung

In diesem Dokument wird beschrieben, wie Sie Heap Dump und Thread Dump für Tomcat vom Cisco Customer Voice Portal (CVP) Voice eXtensible Markup Language Server (VXML) sammeln.

Voraussetzungen

Anforderungen

Cisco empfiehlt, über Kenntnisse in folgenden Bereichen zu verfügen:

- CVP
- CVP VXML-Server
- CVP VXML-Anwendungen

Verwendete Komponenten

Die Informationen in diesem Dokument basieren auf den folgenden Software- und Hardwareversionen:

- CVP-Version 11.5
- CVP VXML-Server 11.5

Die Informationen in diesem Dokument wurden von den Geräten in einer bestimmten Laborumgebung erstellt. Alle in diesem Dokument verwendeten Geräte haben mit einer leeren (Standard-)Konfiguration begonnen. Wenn Ihr Netzwerk in Betrieb ist, stellen Sie sicher, dass Sie die potenziellen Auswirkungen eines Befehls verstehen.

Problem

Szenario 1. Während Sie die Leistung des VXML-Servers überwachen, haben Sie festgestellt, dass dieser 4 GB Speicher verwendet. Sie möchten die Ursache des Speicherlecks kennen, bevor er abstürzt.

Szenario 2. Während Sie den VXML-Server überwachen, haben Sie entdeckt, dass der VXML-Server hohe Threads (~ 500) verwendet, was ziemlich ungewöhnlich ist. Sie möchten wissen, wie Sie die Threadverwendung analysieren und den Threadzustand und die Komponente/Anwendung ermitteln, die mehr Threads beansprucht.

Beispiel: VXML Tomcat verbraucht 4,5 GB Arbeitsspeicher, wie im Bild gezeigt.

				10	isk Manager				
File Opti	ons View								
Processes	Performance	Users	Details	Services					
Name		PID	Status		User name	CPU	Memory (p	Description	^
tomcat8.exe		8092	Running		UCCE.Install	00	4,385,852 K	Commons Daemon	
tomcat8.exe		9220	Running		SYSTEM	00	728,440 K	Commons Daemon	
tomcat8.exe		7696	Running		SYSTEM	00	301,740 K	Commons Daemon	
notepad.exe		17020	Runn	ing	dutta_1247	00	210,072 K	Notepad	
explorer.exe		1480	Runn	ing	V_256454	00	166,524 K	Windows Explorer	
🍰 java.exe		6936	Runn	ing	SYSTEM	00	162,844 K	Java(TM) Platform S	
splunkd.exe		1676	Running		SYSTEM	00	102,840 K	splunkd service	
svchost.exe		2400	Runn	ing	NETWORK	00	84,012 K	Host Process for Wi	
ServerManager.exe		12840	Running		Gupta_237	00	52,416 K	Server Manager	
-		245.6							

VXML-Speicher bei 4,5 GB

Fehlerbehebung

Schritte zum Abrufen von Heap-Dump

Schritt 1: JDK-Version ähnlich der Version herunterladen, die in CVP installiert ist

C:\Cisco\CVP\jre\bin>java -version java version "1.7.0_51" Java(TM) SE Runtime Environment (build 1.7.0_51-b13) Java HotSpot(TM) Client VM (build 24.51-b03, mixed mode) Schritt 2: Kopieren Sie JDK vom Desktop auf den CVP-Server.

Download windows JDK exe Open with 7-Zip Dump contents into a directory %JDK-EXE% cmd: cd %JDK-EXE%.rsrc\1033\JAVA_CAB10 cmd: extrac32 111 Now have a tools.zip in directory, open it in 7-Zip Extract contents into a new directory %JDK-VERSION% cmd: cd %JDK-VERSION% cmd: for /r %x in (*.pack) do .\bin\unpack200 -r "%x" "%~dx%~px%~nx.jar"

Hinweis: Laden Sie einfach JDK von

<u>http://www.oracle.com/technetwork/java/javaee/downloads/index.html</u> herunter und installieren Sie EXE im lokalen Ordner und kopieren Sie das JDK von Ihrem lokalen Rechner nach CVP .

Schritt 3: Erfassen Sie die Prozess-ID (PID) des VXML-Servers vom Task-Manager.

		Ta	sk Manager				×	10	tomcat7 Properties
File Options View								General Com	patibility Security Details Previous Versions
Processes Performance	Users	Details Services					_		
Name *	PID	Status	User name	CPU	Memory (p	Description	~		tomcat7
svchost.exe	540	Running	NETWORK	00	2,124 K	Host Process for Wi			1000000000
svchost.exe	668	Running	LOCAL SE	00	9,684 K	Host Process for Wi		Type of file:	Application (.exe)
svchost.exe	700	Running	SYSTEM	00	10,896 K	Host Process for Wi		Description:	Commons Daemon Service Runner
svchost.exe	736	Running	LOCAL SE	00	3,316 K	Host Process for Wi			
svchost.exe	844	Running	NETWORK	00	5,104 K	Host Process for Wi		Location:	C:\Cisco\CVP\\XXMLServer\Tomcat\bin
svchost.exe	980	Running	LOCAL SE	00	4,364 K	Host Process for Wi		Gaa:	75.5 KB (77.212 butes)
svchost.exe	552	Running	SYSTEM	00	2,700 K	Host Process for Wi		3420.	10.0 ND (11.012 0908)
svchost.exe	1264	Running	SYSTEM	00	2,836 K	Host Process for Wi		Size on disk:	76.0 KB (77.824 bytes)
svchost.exe	1344	Running	SYSTEM	00	3,048 K	Host Process for Wi			
svchost.exe	1520	Running	NETWORK	00	37,160 K	Host Process for Wi		Created:	Friday, May 5, 2017, 7:53:10 PM
svchost.exe	960	Running	NETWORK	00	784 K	Host Process for Wi		Modfied:	Tuesday, January 17, 2012, 9:53:46 AM
System	4	Running	SYSTEM	00	28 K	NT Kernel & System			
System Idle Process	0	Running	SYSTEM	98	4 K	Percentage of time t		Accessed:	Enday, May 5, 2017, 7:53:10 PM
System interrupts	-	Running	SYSTEM	02	0 K	Deferred procedure		and the second second	
taskhostex.exe	3920	Running	Administra	00	1,904 K	Host Process for Wi		Attributes:	Read-only Hidden Advanced
Taskmgr.exe	7688	Running	Administra	00	9,060 K	Task Manager			
tomcat7.exe	728	Running	SYSTEM	00	156,796 K	Commons Daemon			
tomcat7.exe	1308	Running	SYSTEM	00	446,164 K	Commons Daemon	=		
tomcat7.exe	1368	Running	SYSTEM	00	250,960 K	Commons Daemon			
wininit.exe	356	Running	SYSTEM	00	552 K	Windows Start-Up A		-	OK Const Link
winlogon.exe	384	Running	SYSTEM	00	752 K	Windows Logon Ap			UN Cances Appy
winlogon.exe	3716	Running	SYSTEM	00	808 K	Windows Logon Ap			
wrapper.exe	1076	Running	SYSTEM	00	964 K	wrapper	4		
Constanting							÷		

Schritt 4: Führen Sie diesen Befehl aus, um HeapDump zu sammeln. (Bsp.: jmap - dump:file=vxml.hprof <PID DER TOMCAT-Instanz).

C:\jdk1.7.0_80\jdk1.7.0_80\bin>jmap -dump:file=vxml.hprof 1308 Dumping heap to C:\jdk1.7.0_80\jdk1.7.0_80\bin\vxml.hprof ... Heap dump file created

C:_	jdk1.7.0_	_80_	dk1.7.0_	_80\bin>
and shall be a state of the sta	and the second s	and the second second	C. Lineau C.	and the second sec

Construction of the Assessment of the Assessme										
🌡 l 🗋 🚺 = l		bin					×			
File Home Sha	re View						Y (
💿 • 🕇 📕 •	This PC 🔸 Local Disk (C:) 🔸 jdk1.7.0	_80 ▶ jdk1.7.0_80 ▶ bin			~ 0	Search bin	Q			
🔆 Favorites	Name	Date modified	Туре	Size			-			
E Desktop	vxml.hprof	9/13/2017 12:22 AM	HPROF File	252,628 KB						
🗼 Downloads	appletviewer	6/8/2017 3:23 PM	Application	16 KB						
🔢 Recent places	📰 apt	6/8/2017 3:23 PM	Application	16 KB						
	🗟 beanreg.dli	6/8/2017 3:23 PM	Application extens	40 KB						
🛤 This PC	extcheck	6/8/2017 3:23 PM	Application	16 KB						

Hinweis: Ihr hProf wird erstellt, Sie können auf das lokale System kopieren und offline überwachen.

Schritte zum Abrufen von Thread-Dump

Thread-Dump-Sammlung ist relativ einfach im Vergleich zu Heap-Dump.

Schritt 1: Herstellen einer Verbindung zum CVP VXML-Server auf <CVP VXML-Server>:9696 (9696 ist der Standard-JMX-Port für VXML-Server), der jVisualVM verwendet.

Schritt 2: Klicken Sie mit der rechten Maustaste auf JXM Connection und Collect Thread Dump.



Schritt 3: Hier wird Thread-Dump im Remote-Server abgelegt, wo er als Datei gespeichert und für weitere Analysen verwendet werden kann.