

Comment collecter le vidage mémoire principal et fileter le vidage mémoire du serveur CVP VXML

Contenu

[Introduction](#)

[Conditions préalables](#)

[Conditions requises](#)

[Composants utilisés](#)

[Problème](#)

[Dépannez](#)

[Étapes pour obtenir le vidage mémoire de segment de mémoire](#)

[Étapes pour obtenir le vidage mémoire de thread](#)

Introduction

Ce document décrit comment collecter le vidage mémoire de segment de mémoire et fileter le vidage mémoire pour Tomcat du serveur de langage d'extensible markup de Voix du Customer Voice Portal de Cisco (CVP) (VXML).

Conditions préalables

Conditions requises

Cisco vous recommande de prendre connaissance des rubriques suivantes :

- CVP
- Serveur CVP VXML
- Applications CVP VXML

[Composants utilisés](#)

Les informations contenues dans ce document sont basées sur les versions de matériel et de logiciel suivantes :

- Version 11.5 CVP
- Serveur 11.5 CVP VXML

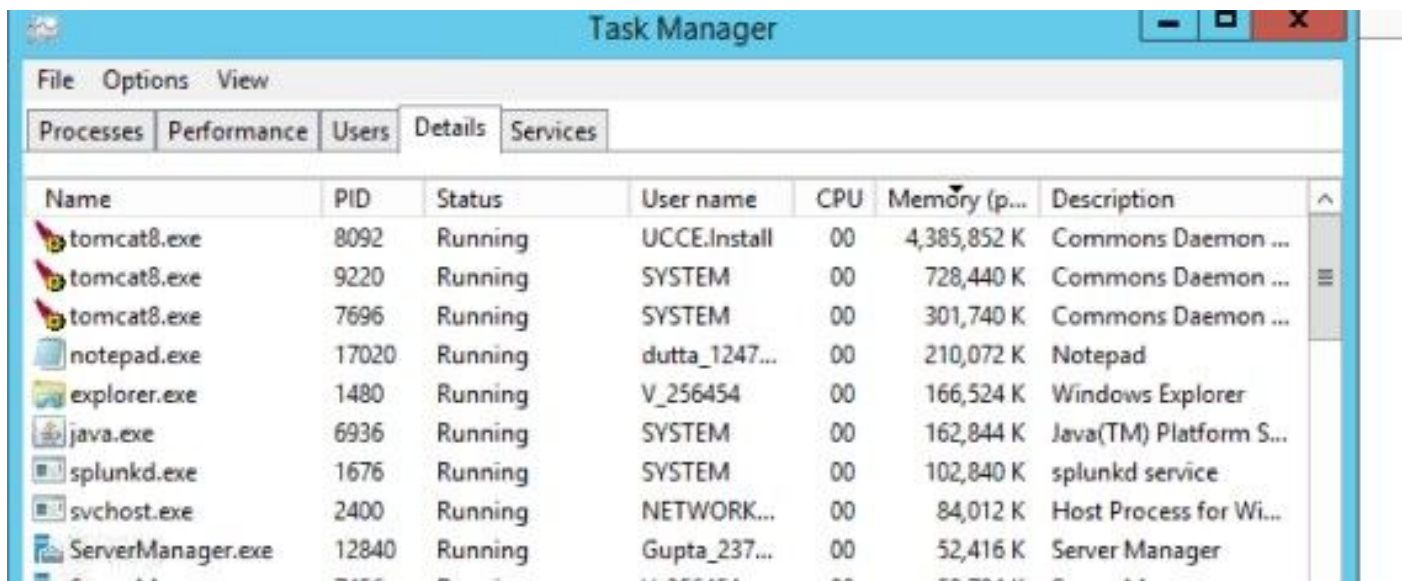
Les informations contenues dans ce document ont été créées à partir des périphériques d'un environnement de laboratoire spécifique. Tous les dispositifs utilisés dans ce document ont démarré par une configuration effacée (par défaut). Si votre réseau est vivant, assurez-vous que vous comprenez l'impact potentiel de n'importe quelle commande.

Problème

Scénario 1. Tandis que vous surveillez la performance des serveurs VXML vous avez découvert qu'elle utilise 4GB de mémoire. Vous voudriez connaître la cause principale de la fuite de mémoire avant qu'elle tombe en panne.

Scénario 2. Tandis que vous surveillez le serveur VXML, vous avez découvert que le serveur VXML utilise les thread élevés (| 500), qui est tout à fait peu commun. Vous voudriez savoir analyser l'utilisation de thread et découvrir l'état de thread et le composant/application qui prend plus de thread.

Par exemple : VXML Tomcat consomme 4.5GB de mémoire totale, suivant les indications de l'image.



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	Running	dutta_1247...	00	210,072 K	Notepad
explorer.exe	1480	Running	V_256454	00	166,524 K	Windows Explorer
java.exe	6936	Running	SYSTEM	00	162,844 K	Java(TM) Platform S...
splunkd.exe	1676	Running	SYSTEM	00	102,840 K	splunkd service
svchost.exe	2400	Running	NETWORK...	00	84,012 K	Host Process for Wi...
ServerManager.exe	12840	Running	Gupta_237...	00	52,416 K	Server Manager

Mémoire VXML à 4.5 Go

Dépannez

Étapes pour obtenir le vidage mémoire de segment de mémoire

Étape 1. Téléchargez la version JDK semblable à la version installée dans CVP

```
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)
```

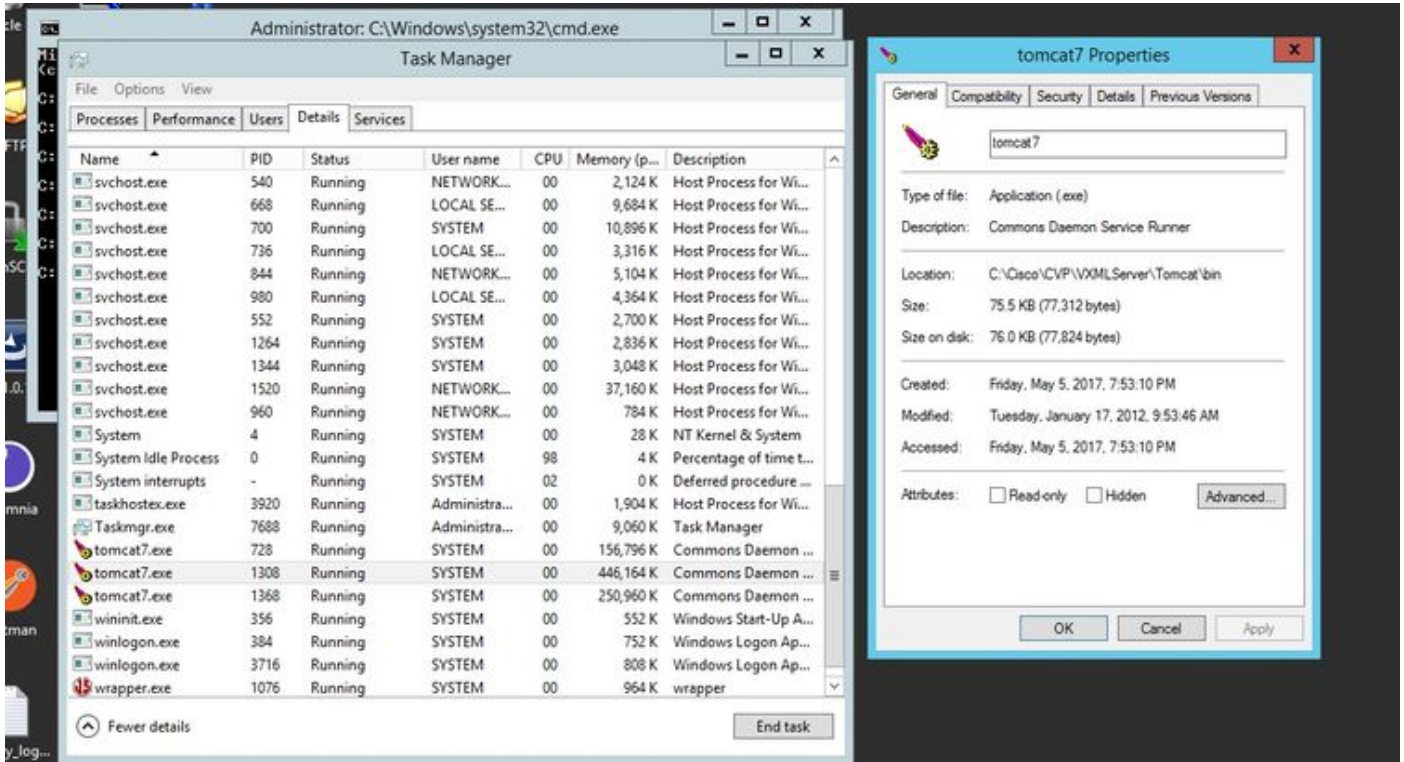
Étape 2. Copiez le JDK de l'appareil de bureau sur le serveur CVP.

```
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"
```

Note: Téléchargez simplement le JDK de <http://www.oracle.com/technetwork/java/javasee/downloads/index.html> et installez EXE sur le répertoire local et copiez le JDK de votre ordinateur local sur CVP.

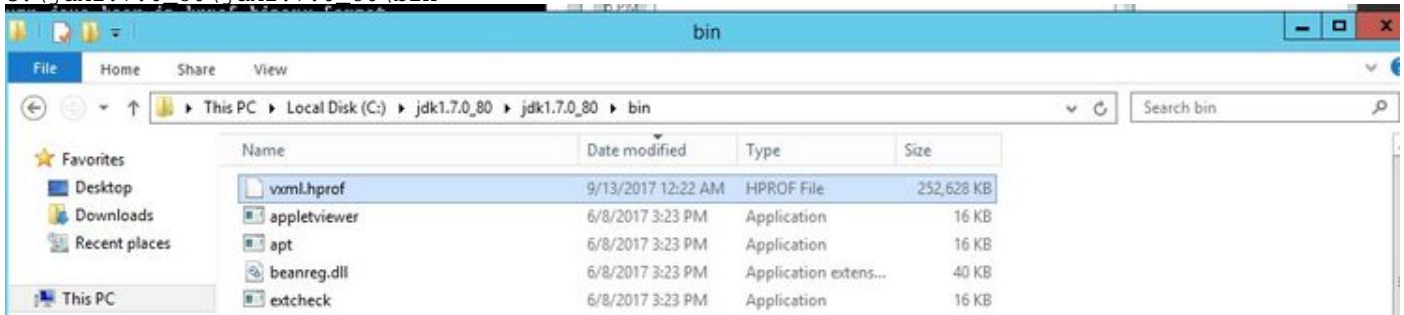
Étape 3. Collectez l'ID de processus (PID) du serveur VXML du gestionnaire de tâches.



Étape 4. Exécutez cette commande afin de collecter HeapDump. (Ex : jmap -dump:file=vxml.hprof <PID d'exemple de TOMCAT).

```
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\jdk1.7.0_80\bin>
```



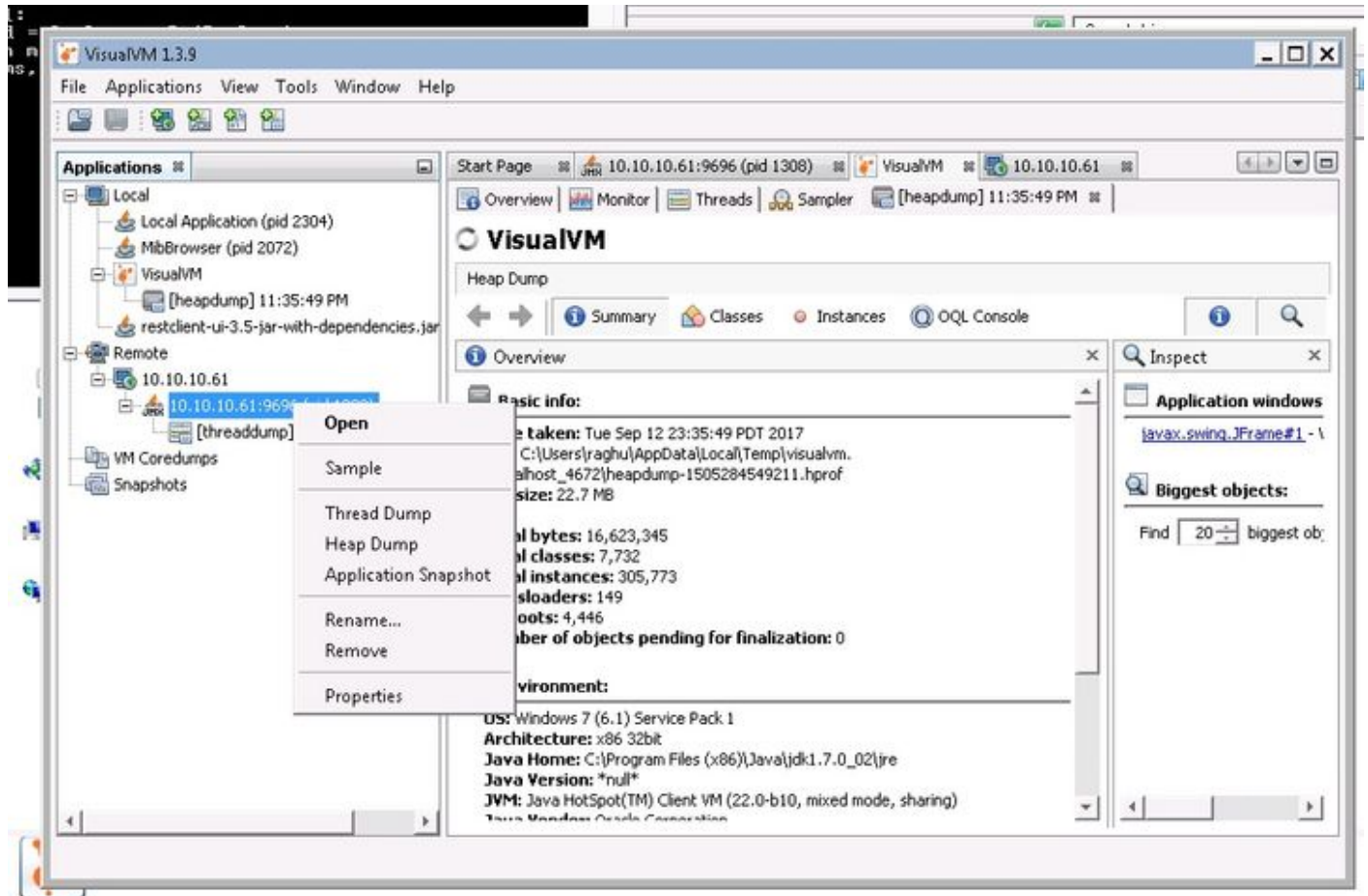
Note: Votre hProf est créé et vous pouvez copier sur le système local et surveiller off-line.

Étapes pour obtenir le vidage mémoire de thread

La collection de vidage mémoire de thread est relativement facile comparée pour amasser le vidage mémoire.

Étape 1. Connectez au serveur CVP VXML sur <CVP VXML Server> : **9696** (9696 est port par défaut JMX pour le serveur VXML) ces utilise le jVisualVM.

Étape 2. Righ-clik sur la connexion JXM et collectent le vidage mémoire de thread.



Étape 3. Filetez ici le vidage mémoire vidé dans le serveur distant, où il peut être enregistré comme fichier et être utilisé pour l'analyse approfondie.