**How to Generate and Analyse WebSphere Thread Dump?**

As a WebSphere administrator, you should be aware of taking thread dumps and tool to analyze them. Thread dumps are often needed to diagnose the application performance issue like deadlocks, hung threads, and bottlenecks in Java threads.

**Taking Thread Dumps in WebSphere**

It’s recommended to take multiple thread dumps in the interval of 5-8 seconds. You can use any of the following methods to generate it.

**1. Using wsadmin.sh**

* Login into WAS Server
* Go to profile and bin folder
* Execute wsadmin.sh file

[root@localhost bin]# **./wsadmin.sh**

WASX7209I: Connected to process "dmgr" on node localhostCellManager01 using SOAP connector; The type of process is: DeploymentManager

WASX7029I: For help, enter: "$Help help"

wsadmin>

* Set JVM name in a variable

set jvm [$AdminControl completeObjectName type=JVM,process=server1,\*]

**Note:** server1 is for example. Change this to your actual JVM name.

wsadmin>**set jvm [$AdminControl completeObjectName type=JVM,process=server1,\*]**

WebSphere:name=JVM,process=server1,platform=proxy,node=localhostNode01,j2eeType=JVM,J2EEServer=server1,version=8.5.5.0,type=JVM,mbeanIdentifier=JVM,cell=localhostCell01,spec=1.0

wsadmin>$AdminControl invoke $jvm dumpThreads

This will generate thread dump and will be available under profile path.

**2. Using kill**

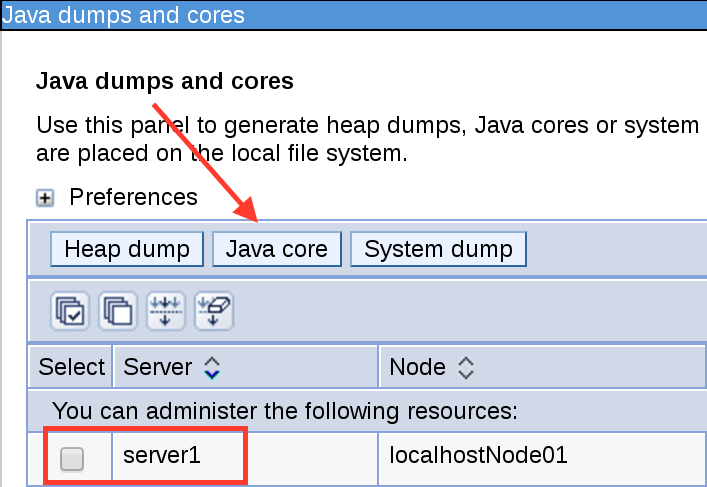
* Find JVM process ID using ps command
* Execute kill -3 $PID

# kill -3 $PID

You will find the dump in profile path.

**3. Using WebSphere Administrative Console**

* Login into DMGR Console
* Navigate to Troubleshooting at left side
* Select the JVM and click on “Java core”



Once you have the thread dump, you can use following tools to analyse them.

**Analysing WebSphere Thread Dumps**

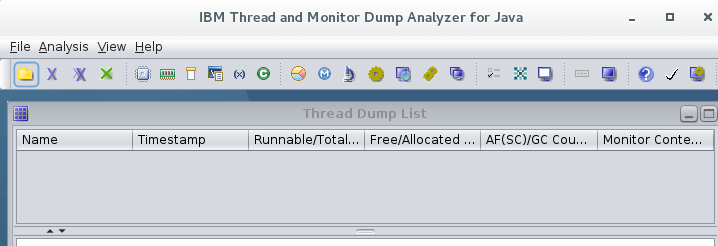
Analysing dumps are always challenging and following FREE tools will help you.

**1. IBM Thread Dump Analyzer (TDA)**

Using IBM TDA, you can identify hangs, deadlocks, and bottlenecks in Java threads.

You can download [IBM TDA from here](https://www.ibm.com/developerworks/community/groups/service/html/communityview?communityUuid=2245aa39-fa5c-4475-b891-14c205f7333c). It’s just the jar file and once downloaded, you can execute following to start the GUI.

java -jar jca457.jar

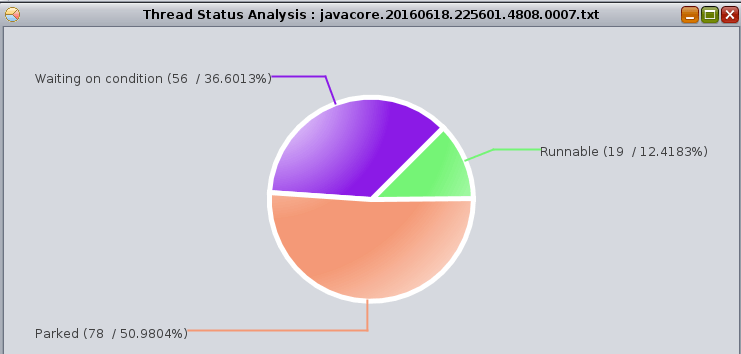


* Click on File >> Open Thread Dumps
* Browse the folder and select the dump you wish to analyze

It will take few seconds and you are all set to do the following analysis

* CPU usage
* Memory segment
* User process resources limit
* Command line argument
* Environment variable
* Shared class cache
* Native memory
* Thread status
* Method

All are available under “Analysis” menu. A quick look at thread status analysis



So go through the analysis you are looking for have fun with IBM TDA.

**2. Samurai**

Another popular analyzer is a samurai, which you can [download from here](http://yusuke.homeip.net/samurai/en/samurai.jar) and execute as following.

java –jar samurai.jar

So now you should be able to generate and analyse the thread dumps for application troubleshooting. There is one more [online thread dump analyser](https://spotify.github.io/threaddump-analyzer/) which you may try.