Performance Tuning for JBoss or Wildfly

Last Updated February 15, 2023

The fine-tuned configurations that we recommend for optimizing the performance of the JBoss or Wildfly application server are:

Recommended JVM Settings

Set the JVM settings:

- **Windows**

  1. Navigate to <JBOSS_HOME>\bin.
  2. Open **standalone.conf.bat** for editing.

3. Set the JVM parameters in JAVA_OPTS with the following recommended values:

- **JVM Minimum Heap Size:** 512 MB
- **JVM Maximum Heap Size:** 4 GB

> If more than 4 GB is required, use the recommended settings described in Fine Tuning Garbage Collection for more aggressive garbage collection.

- **Generic JVM Arguments:** -XX:+UseG1GC -XX:+UseStringDeduplication -XX:+UseCompressedOops -Djava.awt.headless=true -Djava.net.preferIPv4Stack=true

Example:

```
JAVA_OPTS=-Xms512m -Xmx4096m -XX:+UseG1GC -XX:+UseStringDeduplication -XX:+UseCompressedOops -Djava.awt.headless=true -Djava.net.preferIPv4Stack=true -Djava.security.egd=file:/dev/./urandom -Dcom.sun.jersey.server.impl.cdi.lookupExtensionInBeanManager=true"
```

Open Files

On all JBoss or WildFly servers, set the Open Files (ulimit –n) to at least 50000 for the user that runs the JBoss or WildFly process.

Maximum User Processes

On all JBoss or WildFly servers, set the User Process (ulimit –u) to at least 131072 for the user

Set the value of the TCP_KEEPALIVE_PROBES parameter to **5 seconds**, which is the recommended value. The default value is **9 seconds**

```
echo 5  > /proc/sys/net/ipv4/tcp_keepalive_probes
```

Allocate Huge Pages for Java Virtual Machine (JVM) Heap

Set the following three settings in **/etc/sysctl.conf** file.

- vm.nr_hugepages = 2048
- vm.nr_hugepages_mempolicy = 2048
  Calculation: 2048 = 8192 (Max Heap Size) /4 (Page Size)
- Ensure that the following parameter is at least set to the following value (if already set to higher number, even better):
  kernel.shmmax = 8689934592
  Calculation: 8689934592 = 8192 * 1048576 bytes/MB + 100000000 bytes
- kernel.shmall = 2172483648
  Calculation: 2172483648 = 8689934592 (shmmax)) / 4 (Page Size)

Increase Linux kernel Entropy

On all CA Single Sign-On servers, increase the kernel entropy by adding the following rngd daemon:

```
-Xms8g -Xmx8g -XX:+UseG1GC -XX:ConcGCThreads=12 -XX:ParallelGCThreads=22
-XX:MaxGCPauseMillis=1000 -XX:InitiatingHeapOccupancyPercent=40
-XX:G1HeapWastePercent=2 -XX:G1ReservePercent=15
-XX:+UnlockExperimentalVMOptions -XX:G1OldCSetRegionThresholdPercent=15
-XX:G1MixedGCLiveThresholdPercent=90 -XX:G1NewSizePercent=20
-XX:G1MaxNewSizePercent=25
```

- **-XX:ConcGCThreads=12**
  Sets the number of parallel marking threads. Sets n to approximately 1/4 of the number of parallel garbage collection threads (ParallelGCThreads).
- **-XX:ParallelGCThreads=22**
  Sets the value of the STW worker threads. Sets the value of n to the number of logical processors. The value of n is the same as the number of logical processors up to a value of 8.
- **-XX:MaxGCPauseMillis=1000**
  Sets a target value for desired maximum pause time. The default value is 200 milliseconds. The specified value does not adapt to your heap size.
- **-XX:InitiatingHeapOccupancyPercent=40**
  Sets the Java heap occupancy threshold that triggers a marking cycle. The default occupancy is 45 percent of the entire Java heap.
- **-XX:G1HeapWastePercent=2**
  Sets the percentage of heap that you are willing to waste. The Java HotSpot VM does not initiate the mixed garbage collection cycle when the reclaimable percentage is less than the