Ignite checks some JVM and Operation system parameters and outputs performance-suggestions at its startup.

Some checks are described below.

**JVM configuration suggestions:**

* “Enable server mode for JVM (add '-server' to JVM options)”

The aim is to improve performance by using standard optimizations of Java Server Virtual Machine.

This suggestion will be outputted if your JVM was switched in client mode, for example: if you use an x32-version of JVM.

To resolve this issue just to add the '-server' parameter to your JVM options.

* “Switch to the most recent 1.8 JVM version”

Suggestion will be outputted if Ignite has not started under JVM version 1.8.

Please use the most recent version of JVM 1.8 to provide better performance.

* “Enable G1 Garbage Collector (add '-XX:+UseG1GC' to JVM options)”

This suggestion will be outputted if your JVM options doesn’t consists the parameter '-XX:+UseG1GC'.

It is recommended to use G1 garbage collector since it has been being constantly improved.

To resolve this issue just to add the parameter '-XX:+UseG1GC ' to your JVM options.

* “Specify JVM heap max size (add '-Xmx<size>[g|G|m|M|k|K]' to JVM options)”

This suggestion will be outputted if your JVM options doesn’t consists the key '-Xmx '.

It is recommended to specify directly the maximum size of the allocation pool to be sure in the size which you expect.

Just add the key with predefined value to your JVM options, for example '-Xmx10g'.

* “Specify New I/O max total size (add '-XX:MaxDirectMemorySize=<size>[g|G|m|M|k|K]' to JVM options)”

This suggestion will be outputted if your JVM options doesn’t consists the key '-XX:MaxDirectMemorySize '.

It is recommended to specify directly the maximum total size of java.nio direct buffer allocations.

Just add the key with predefined value to your JVM options, for example '-XX:MaxDirectMemorySize=1g'.

* “Enable the thread-local allocation blocks (add '-XX:+UseTLAB' to JVM options)”

This suggestion will be outputted if your JVM options consists the parameter '-XX:-UseTLAB'.

It is recommended to use thread-local object allocation blocks, because it improves concurrency by reducing contention on the shared heap lock.

To resolve this issue just to remove the parameter '-XX:-UseTLAB' from to your JVM options or replace it with '-XX:+UseTLAB'.

* “Disable processing of calls to System.gc() (add '-XX:+DisableExplicitGC' to JVM options)”

This suggestion will be outputted if your JVM options doesn’t consist the parameter '-XX:+DisableExplicitGC '.

It is recommended to disable calls to System.gc() to avoid not necessary calls to a garbage collector by user code.

To resolve this issue just to add the '-XX:+DisableExplicitGC' parameter to your JVM options.

**OS configuration suggestions (at the moment it is only for RedHat family OS):**

Please, consult with your IT department before making changes at the Linux kernel level in production!

* “Speed up flushing of dirty pages by OS (alter vm.dirty\_writeback\_centisecs and vm.dirty\_expire\_centisecs parameters by setting to 500)”

This suggestion will be outputted if the parameters exist in your OS and are differed from “500”.

It is recommended increase pages flushing to 5 seconds by setting the parameter to “500”.

* “Reduce pages swapping ratio (set vm.swappiness=10)”

This suggestion will be outputted if the parameter exists 'vm.swappiness' in your OS and is differed from “10”.

It is recommended to set this parameter to “10” to protect heap and anonymous memory.

* “Disable NUMA memory reclaim (set vm.zone\_reclaim\_mode=0)”  
  This suggestion will be outputted if the parameter 'vm.zone\_reclaim\_mode' exists in your OS and is differed from “0”.

It is recommended to turn off NUMA zone-reclaim optimization by setting the parameter to “0”.

* “Avoid direct reclaim and page allocation failures (set vm.extra\_free\_kbytes=1240000)”

This suggestion will be outputted if the parameter 'vm.extra\_free\_kbytes' exists in your OS and is differed from “1240000”.

It is recommended to set this parameter to “1240000” to get over long GC pauses caused by direct page memory reclaim on Linux.