Uptime Infrastructure Monitor Measurement Tuning

In some cases, you can make measurement adjustments to Uptime Infrastructure Monitor's default values. Changes can be made to the following:

- the number of threads allocated to service monitors
- Java heap size
- status thresholds in the **Resource Scan** and **Global Scan** dashboards
- how often performance and status are checked for monitored hosts

Service Monitor Thread Counts

By default, the number of Java threads allocated to service and performance monitors is 100. This can be modified with the following `uptime.conf` parameter:

```
serviceThreads=100
```

Topics on this page

- Service Monitor Thread Counts
- Java Heap Size
- Status Thresholds
- Platform Performance Gatherer Check Intervals

Java Heap Size

By default, the JVM's heap memory is to a maximum of 1 GB. If your monitoring deployment has a lot of service monitors running or reports to generate, you can the increase the amount of Java heap memory (for example, to 1.5 GB) to improve performance.

> When increasing the Java heap size, ensure your Monitoring Station resources can support the new setting. If the OS does not have the desired amount of memory available exclusively for Uptime Infrastructure Monitor, the up.time Core service may become unstable and crash, despite starting up successfully.

Adjusting the Java Heap Size

The amount of memory allocated to the JVM can be adjusted by modifying one of the following parameters, depending on your Monitoring Station platform:

On Linux, edit the `<uptimeInstallDir>/uptime.jcnf` file and modify the following:

```
-Xmx1G
```

On Windows, edit the `<uptimeInstallDir>\UptimeDataCollector.ini`file and modify the following, which relates to the Java -Xmx option:

```
vm.heapsize.preferred=1024m
```

Note that the default heap size is measured in gigabytes in the Linux configuration file, and megabytes in the Windows configuration file.

Status Thresholds

The Global Scan threshold settings determine when a cell on the **Global Scan** dashboard changes state to reflect a host's status change: green represents normal status, yellow represents Warning status, and red represents Critical.

The Resource Scan threshold settings determine the size of the gauge ranges on the **Resource Scan** view: green represents normal status, yellow represents Warning status, and red represents Critical status.

You can change the thresholds used to determine status by manually inputting settings in the **Uptime Configuration** panel, as outlined in Modifying Uptime Config Panel Settings.

Changing Global Scan Threshold Settings

You can modify the **Global Scan** threshold settings through the following parameters (default values are shown):

`globalscan.cpu.warn=70`	Warning-level status is reported when CPU usage is at 70% or greater
`globalscan.cpu.crit=90`	Critical-level status is reported when CPU usage is at 90% or greater
`globalscan.diskbusy.warn=70`	Warning-level status is reported when a disk on the host is busy for 70% or more of a five-minute time frame
`globalscan.diskbusy.crit=90`	Critical-level status is reported when a disk on the host is busy for 90% or more of a five-minute time frame
`globalscan.diskfull.warn=70`	Warning-level status is reported when 70% or more of the disk space on the host is used
`globalscan.diskfull.crit=90`	Critical-level status is reported when 90% or more of the disk space on the host is used
`globalscan.swap.warn=70`	Warning-level status is reported when 70% or more of the swap space on a disk is in use
`globalscan.swap.crit=90`	Critical-level status is reported when 90% or more of the swap space on a disk is in use

Changes to **Global Scan** thresholds are not retroactively applied to all Elements; only Elements added after threshold changes will reflect those changes.

Resource Scan Threshold Settings

You can modify the **Resource Scan** threshold settings through the following parameters (default values are shown):

`resourcescan.cpu.warn=70`	the Warning-level range in the **CPU Usage** gauge begins at this value (70%), and ends at the Critical-level range
`resourcescan.cpu.crit=90`	the Critical-level range in the **CPU Usage** gauge is between this value (90%) and 100%
`resourcescan.memory.warn=70`	the Warning-level range in the **Memory Usage** gauge begins at this value (70%), and ends at the Critical-level range
`resourcescan.memory.crit=90`	the Critical-level range in the **Memory Usage** gauge is between this value (70%) and 100%
`resourcescan.diskbusy.warn=70`	the Warning-level range in the **Disk Busy** gauge begins at this value (70%), and ends at the Critical-level range
`resourcescan.diskbusy.crit=90`	the Critical-level range in the **Disk Busy** gauge is between this value (70%) and 100%
`resourcescan.diskcapacity.warn=70`	the Warning-level range in the **Disk Capacity** gauge begins at this value (70%), and ends at the Critical-level range
`resourcescan.diskcapacity.warn=90`	the Critical-level range in the **Disk Capacity** gauge is between this value (70%) and 100%

Platform Performance Gatherer Check Intervals

The Platform Performance Gatherer is a core performance monitor that resides on all agent-based Elements.

By default, the Platform Performance Gatherer checks the host Elements' performance levels every 300 seconds. You can change the interval by manually inputting settings in the **Uptime Configuration** panel, as outlined in Modifying Uptime Config Panel Settings.

Changing the Performance Monitor Check Interval

You can modify the Platform Performance Gatherer check interval through the following **Uptime Configuration** parameter (the default value is shown):

```
performanceCheckInterval=300
```

A change to the Platform Performance Gatherer check interval is not retroactively applied to all Elements; only Elements added after an interval change will reflect that change.