# Nagios Log Server 2026R2 Performance and Storage Walkthrough

## Purpose

This document describes requirements and performance needs such as disk space, memory, and processor speed for Nagios Log Server 2026R2 based on cluster size and the amount of log data read per day, along with details on checking and optimizing performance, and links to related resources.

## Target Audience

This document is intended for use by administrators while planning, creating, and managing Nagios Log Server clusters.

## Minimum Requirements for Nagios Log Server

The base VMware virtual machine running CentOS 9.x with Nagios Log Server installed has the following default hardware resources:

- 4 x CPU Cores
- 8 GB of RAM
- 40 GB of disk space

These are the minimum requirements that we have set to work consistently with 500 MBs of log data a day. This should be used as a baseline since the numbers will easily scale.

Many customers have reported that memory is one of the first resources that needs increasing on a Nagios Log Server instance, 8 GB is the bare minimum of RAM required. We also recommend allocating at least 8 CPU cores to each instance.

## Nagios Log Server Cluster

Nagios Log Server is a clustered application, it consists of one or more instances of Nagios Log Server. An instance is an installation of Nagios Log Server, it participates in the cluster and acts as a location for the received log data to reside. Log data is spread across instances using the OpenSearch database, a special database used by Nagios Log Server.


Copyright © 2026 Nagios Enterprises, LLC. All rights reserved.
Trademarks are the property of their respective owner.

# Nagios Log Server 2026R2 Performance and Storage Walkthrough

We will describe how Nagios Log Server operates using an example system cluster called **nagioslsdev** so we can quantify requirements throughout this document.

| Instance | Total Disk Space | Memory | Processor | Primary Shard | Replica Shard |
|---|---|---|---|---|---|
| nagioslsdev01 | 1.5 TB | 16 GB | 2GHz / 8 cores | 500 GB | 500 GB |
| nagioslsdev02 | 1.5 TB | 16 GB | 2GHz / 8 cores | 500 GB | 500 GB |
| nagioslsdev03 | 1.5 TB | 16 GB | 2GHz / 8 cores | 500 GB | 500 GB |
| nagioslsdev04 | 1.5 TB | 16 GB | 2GHz / 8 cores | 500 GB | 500 GB |

- **Total Disk Space**
  - The actual disk space on each Nagios Log Server Instance
  - Using Solid State Drives (SSD) are recommended, they significantly boost queries and indexing
- **Memory**
  - Relates to log processing power and speed of displaying, querying and filtering dashboards
  - This larger amount of memory will allow more sophisticated filters and queries as the entire database will need to be read to display the requested dashboard(s)
  - Memory is more important than processing power
- **Processor**
  - More CPUs cores will perform better than CPUs with strictly better clock-speeds
- **Primary and Replica Shards**
  - Each Index in a cluster has a primary and a replica shard, a new index is created every day
  - When you have 3 or more instances the data is distributed over all the instances, allowing for better redundancy and can improve query performance


Copyright © 2026 Nagios Enterprises, LLC. All rights reserved.
Trademarks are the property of their respective owner.

Let's assume the following about our example cluster:

- The cluster receives 500 GB of data every day
- There are 4 Instances in our cluster each with a 1.5 TB storage capacity
- Each Storage index will have a 500 GB primary shard and a 500 GB replica shard per day
- 500 GB/day Primary shard and 500 GB/day Replica shards = 1 TB/day
- Over a 30 day period the total data is 30 TB (1 TB a day for each daily index)
- Each system will need extra space for the operating system and other locally run processes

## Performance Balancing

In our example we have 4 instances that will share the overall data storage and because of this they will also share processing and memory power when running the queries on their local primary index shard. At this point each shard has been allocated, you can find out if the shards have been successfully allocated under the **Admin > System > Cluster Status** page.

When running a query for something like `'*error'` in the dashboard, Nagios Log Server will search through each primary shard from each Instance, which processes the request with its native resources. Once each cluster has been searched through, processed and collected the dashboards graphs and tables are drawn with the data query result.

## What Happens When an Instance Goes Down?

There will be times when an instance in a cluster will lose connection or go down. Nagios Log Server can account for this by reallocating shards from the server into Instances that are still working. This allows the Administrator to react to the downed machine without losing functionality.

The Replica shard is designed to replace a primary shard if one of the servers was to go down. For example, let's say that **nagioslsdev02** went down. Other instances that contain the replica shards for the instance that is currently down would use their replica shards to reallocate the missing data from the downed server and distribute them among the healthy servers. This way none of the data from **nagioslsdev02** will be lost since the replica shard recreated the missing primary shards in the cluster and the cluster will operate as normal until the downed server can be fixed.

In Nagios Log Server no single instance is more important than another instance, it is the cluster functionality which provides this flexibility. There is always a "master" instance, however if the master goes down then another one of the instances takes over instantly.


Copyright © 2026 Nagios Enterprises, LLC. All rights reserved.
Trademarks are the property of their respective owner.

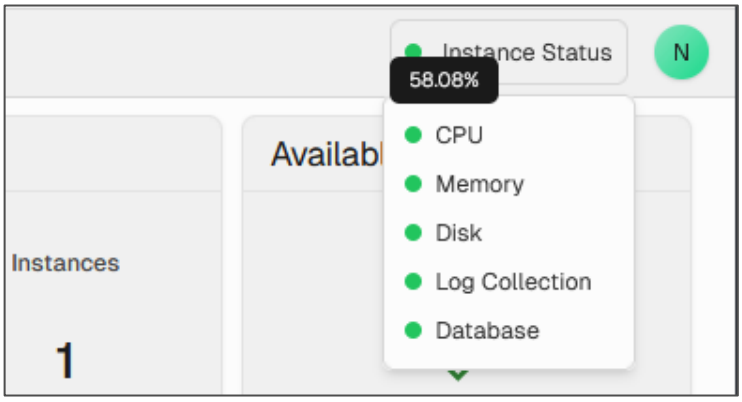
## Checking Instance Performance

Nagios Log Server includes in-UI system performance displays for quick visual reference, and each instance is also pre-loaded with the Nagios Cross Platform Agent (NCPA) on install, along with an NCPA management menu.

NCPA itself also comes pre-loaded with two special Opensearch check plugins: `check_os_cluster_status` and `check_os_jvm_heap`.

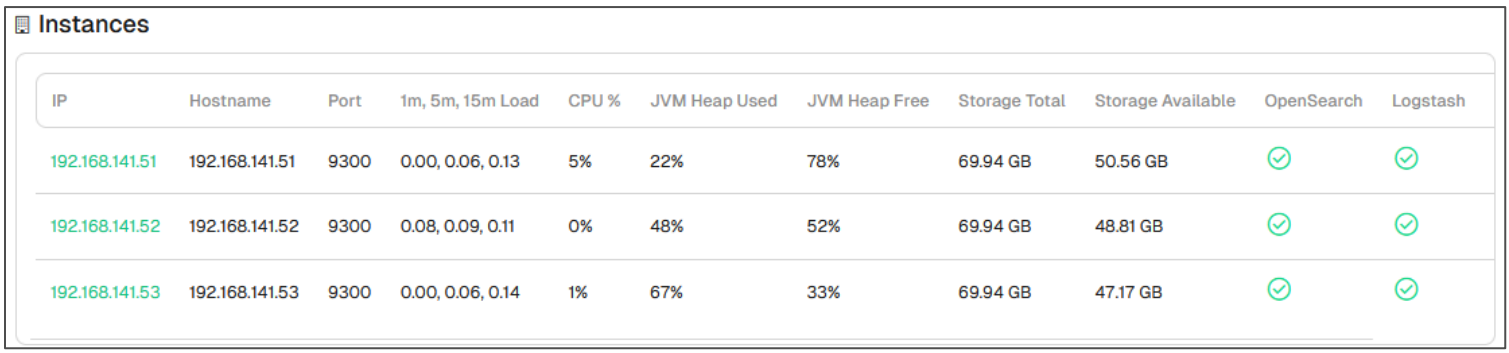
### Viewing Performance

An easy way to check the performance of the instance you're logged into the UI on is the **Instance Status** dropdown on the upper right:




Note that the Memory metric reflects *JVM Heap Used*. Since the Java Virtual Machine (JVM) that Opensearch runs on pre-allocates the majority of available memory, JVM Heap Used is a more useful memory usage metric than raw memory used (which will tend to appear quite high due to the pre-allocation).

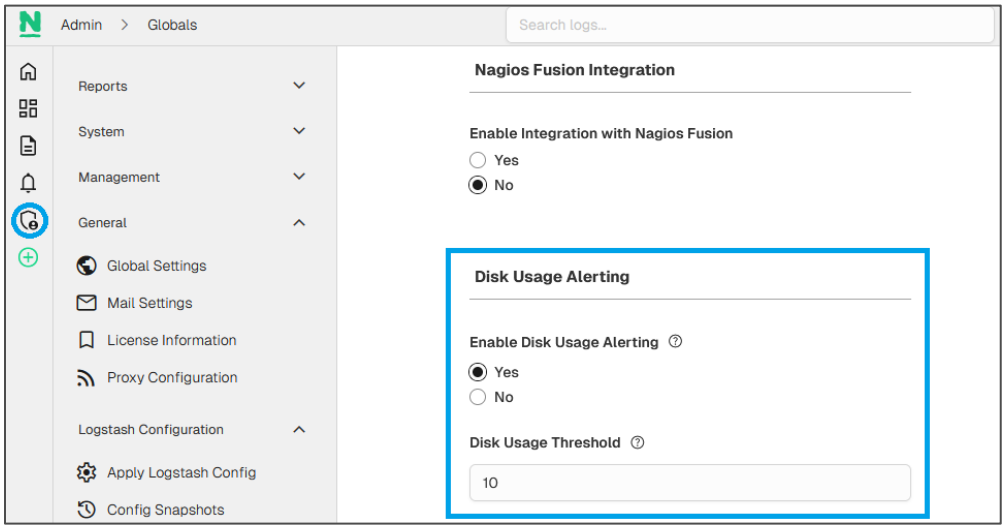
You can also navigate to **Admin > System > Instance Status** to view performance for all instances, and click through on any instance for more comprehensive details.



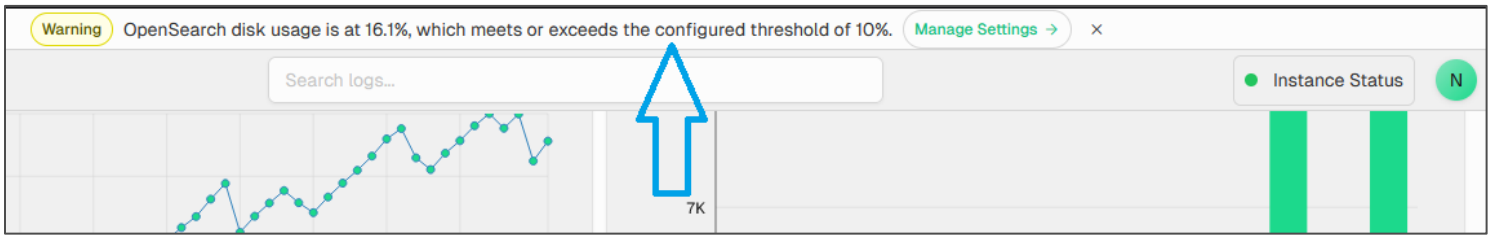

Copyright © 2026 Nagios Enterprises, LLC. All rights reserved.
Trademarks are the property of their respective owner.

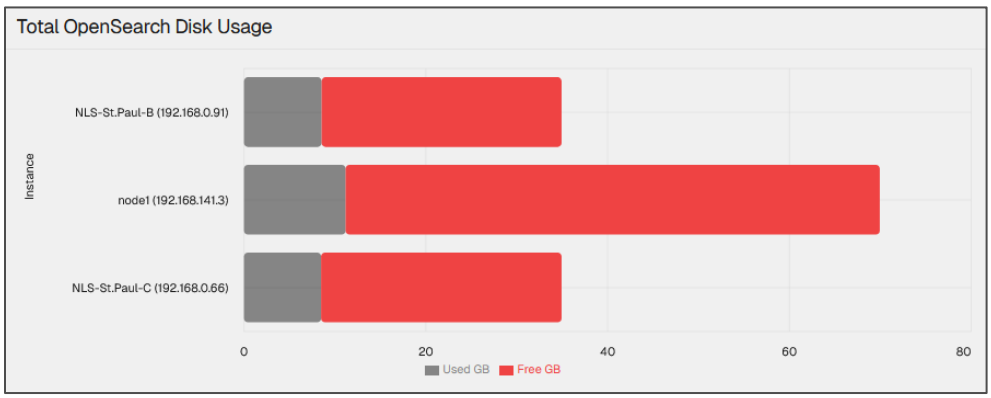
## Opensearch Disk Usage Alerting

Another option is an additional Opensearch Disk Usage visual indicator. To enable it, navigate to **Admin > Global Settings**, and scroll down to the **Disk Usage Alerting** section. Choose your Disk Usage Threshold (the Opensearch disk usage over which you wish to see an alert), then click **Save Settings**.




You will now see a banner at the top of the UI, and the Free GB portion of the Total OpenSearch Disk Usage panel in the Advanced Home page will turn red, when your Disk Usage Threshold is exceeded.



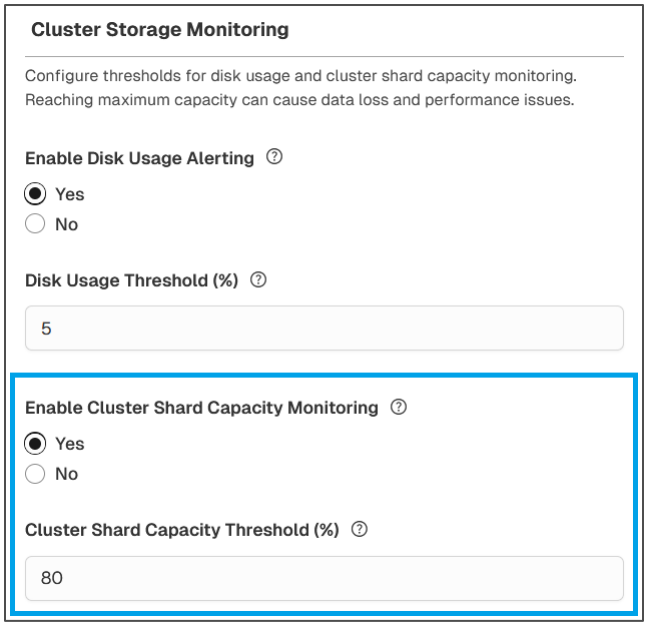





Copyright © 2026 Nagios Enterprises, LLC. All rights reserved.
Trademarks are the property of their respective owner.

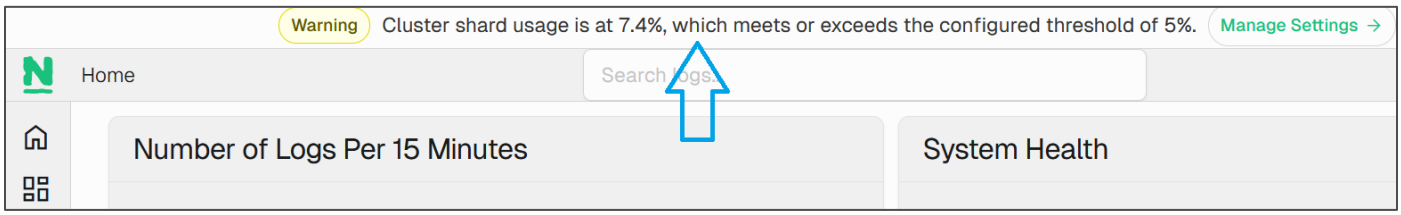
## Cluster Shard Capacity Monitoring

Just below the Opensearch Disk Usage alerting section of **Admin > Global Settings ,** you'll see an option to **Enable Cluster Shard Capacity Monitoring**.




Once enabled, this option will show a visual alert at the top of the UI when Cluster Shard Capacity exceeds the threshold you define here.

This dynamic detection is based on allocated JVM RAM, using a ratio of 1,000 shards per 16 GB RAM.




To **lower shard usage**, reduce the number of open indices. You can find details on managing indices in this guide:

Managing Indices in Nagios Log Server

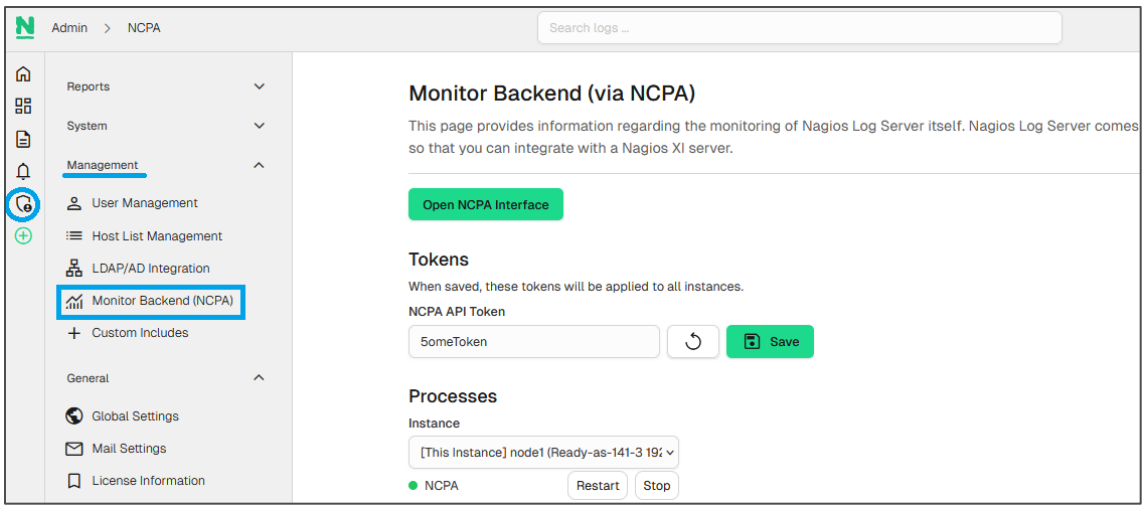
To **raise shard capacity**, add more RAM to one or more of your Data nodes.


Copyright © 2026 Nagios Enterprises, LLC. All rights reserved.
Trademarks are the property of their respective owner.

## Monitoring Peformance

To access the NCPA agent settings, navigate to **Admin > Management > Monitor Backend (NCPA)**.

Here you can manage your NCPA token, and start/stop/restart NCPA on each instance.



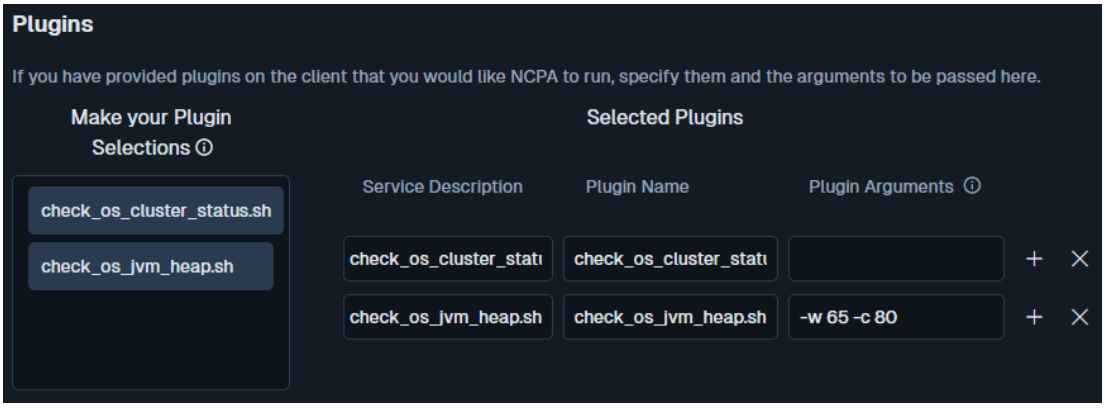

To set up monitoring and alerting on the status of your instances and cluster with Nagios XI using NCPA, you can refer to the following guide. Since NCPA is pre-loaded, you can skip the **Installing NCPA** section and jump right to **Running the NCPA Configuration Wizard**:

Monitoring Devices using NCPA

You'll see the two special plugins mentioned above at the bottom of Step 2 of the wizard.

`check_os_cluster_status` will automatically alert based on the number of instances available in your cluster, and requires no thresholds.

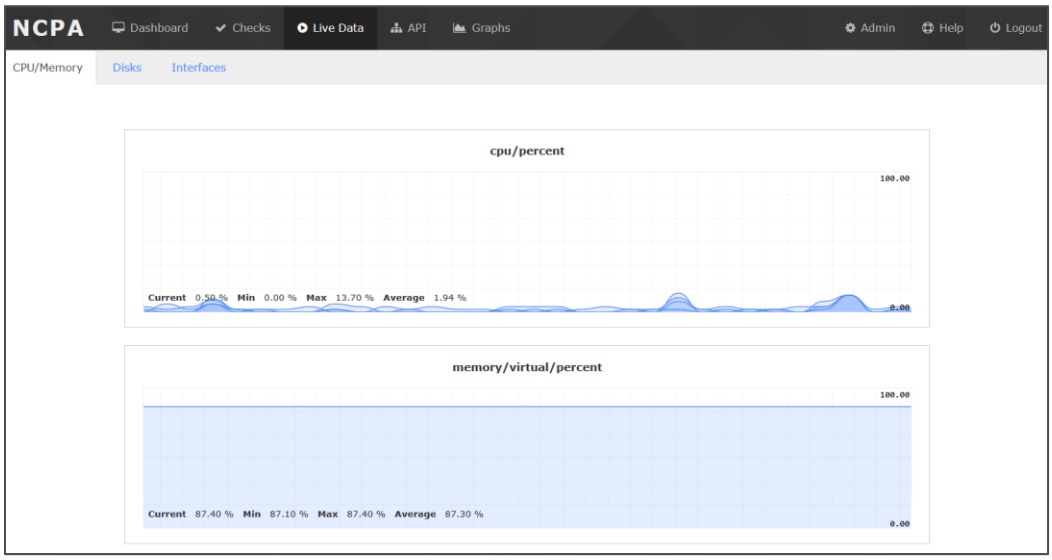
`check_os_jvm_heap` can be used with or without thresholds. To define custom warning and critical thresholds, enter them in the bottom **Plugin Arguments** section as seen here (`-w 65 -c 80`) :




Copyright © 2026 Nagios Enterprises, LLC. All rights reserved.
Trademarks are the property of their respective owner.

Also note that, since NCPA is pre-installed on each of your Log Server instances, you can navigate to the following URL (replacing `NLS.Instance.IP` with the IP of the target instance) to access the native NCPA UI:

```
https://NLS.Instance.IP:5693
```




## Archived Logs – Snapshots

When you are bringing in archived logs (snapshots) from another configuration you will not need to allocate extra space for these logs. Snapshots logs are going to be located on a file system with separate storage, so they won't take any space on your current clusters drives.

You can learn more about snapshots here:

Backing Up and Restoring Nagios Log Server

## Bottlenecks

When you look at the **nagioslsdev** cluster you will observe that they all have the same amount of memory. If, for example, 3 of the instances in the cluster had 16 GB of memory and the 4th instance only had 4 GB of memory, there would be a significant delay for the smaller instance to finish a query over its data. The amount of data will be the same over the 4 instances, but the slower instance will finish its query request 4 times slower than the other 3 instances.


Copyright © 2026 Nagios Enterprises, LLC. All rights reserved.
Trademarks are the property of their respective owner.

In this situation you could remove the lower memory instance, and this would increase the performance of the cluster dramatically. This should be considered when creating or engineering your Nagios Log Server cluster.

## Load Balancing

This document provides high level details on load balancing log event collection with Nagios Log Server, so that log intake continues even in the event of individual cluster instance failures.

Load Balancing Log Collection in Nagios Log Server

## Finishing Up

This completes the documentation on Nagios Log Server 2026 Performance and Storage Walkthrough. If you have additional questions or other support-related questions, please visit the Nagios Support Forum, Nagios Documentation Hub, or Nagios Library:

Visit Nagios Support Forum  Visit Documentation Hub  Visit Nagios Library


Copyright © 2026 Nagios Enterprises, LLC. All rights reserved.
Trademarks are the property of their respective owner.