



## Purpose

This document describes how to manage your Nagios Log Server Snapshots and Maintenance.

## Target Audience

This document is intended for use by Nagios Log Server Administrators. It describes how Administrators can create and manage snapshots and snapshot repositories.

## Snapshot Overview

Backing up a database is always something that is important to administrators to be able to preserve data for pin point analysis and many other reasons. This is no different in Nagios Log Server especially when something critical happens in your network infrastructure. Making snapshots in Nagios Log Server can guarantee that your log data will be saved in case of database corruption or servers going down hard.

This documentation focuses on using a snapshot repository to backup your log data. System backups and config snapshots are explained in the [Backing Up And Restoring Nagios Log Server](#) documentation.

## Snapshots

Snapshots are point in time backups of your log data that exists in the Elasticsearch database.

- Snapshots are stored in a **Snapshot Repository**
- The repository needs to be accessible by all nodes in your Nagios Log Server cluster
  - Usually a NFS or CIFS network share mounted to a path like `/mnt/snapshot_repository`
  - The mounted path needs to be identical on all nodes
  - It needs to be writable by the `nagios` user/group

The snapshot is performed on the entire cluster. During the snapshot and maintenance job, a node will run the commands to create a new snapshot. Because the snapshot is of indexes that have shards allocated to different instances, you need an NFS or CIFS share so that those instances can store their data in the snapshot being created.

This documentation does not provide the steps for mounting a network path, please refer to the following documentation:

### [Snapshot Repository Considerations](#)

It is advisable however to set the correct permissions, the following commands are an example:

```
chown -R nagios:nagios /mnt/snapshot_repository
chmod -R 775 /mnt/snapshot_repository
```

## Snapshots & Maintenance Location

Navigate to **Admin > System > Snapshots & Maintenance**.

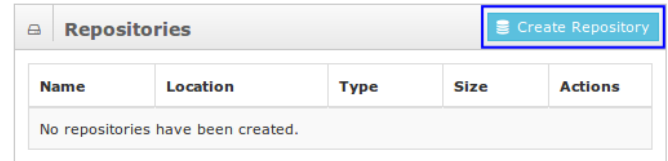
The screenshot shows the Nagios Log Server Admin interface. The top navigation bar includes Home, Dashboards, Alerting, Configure, Help, and Admin (highlighted). A search bar for logs is also present. The left sidebar contains a menu with categories: Reports (Audit Log), System (Cluster Status, Instance Status, Index Status, Snapshots & Maintenance (highlighted), System Status, Command Subsystem), Management (User Management, LDAP/AD Integration), and General (Global Settings, Mail Settings, License Information, Proxy Configuration). The main content area is titled 'Snapshots & Maintenance' and has two tabs: 'Snapshots' and 'Maintenance and Repository Settings' (selected). The 'Maintenance Settings' tab contains a table of settings:

Maintenance Settings	
Optimize indexes older than ?	<input type="text" value="2"/> days
Close indexes older than ?	<input type="text" value="5"/> days
Delete indexes older than ?	<input type="text" value="0"/> days
Repository to store snapshots in ?	You must first create a repository on the right.
Delete snapshots older than ?	You must first create a repository on the right.
Enable Maintenance and Snapshots ?	<input checked="" type="radio"/> Yes <input type="radio"/> No
Last modified ?	Thu, 11 Oct 2018 16:17:26 -0500

Below the settings table is a green 'Save Settings' button. To the right of the maintenance settings is a 'Repositories' section with a 'Create Repository' button. It contains a table with columns: Name, Location, Type, Size, and Actions. The table is currently empty, with the message 'No repositories have been created.' below it.

## Create Snapshot Repository

To create a new repository click the **Create Repository** button.

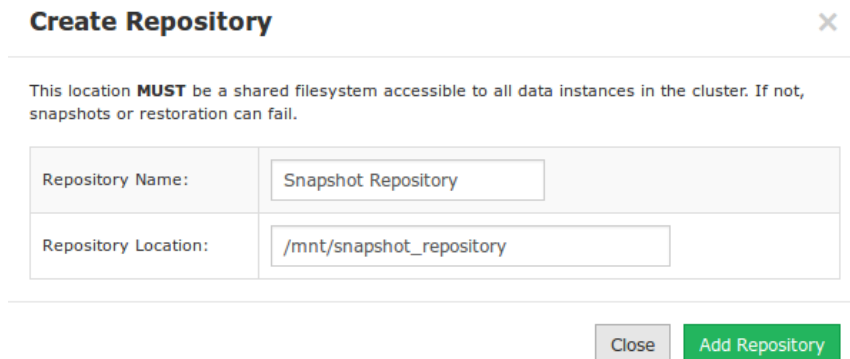


Repositories				
Name	Location	Type	Size	Actions
No repositories have been created.				

This will present the **Create Repository** modal.

Populate the `Name` and `Location` fields, the following screenshot provides an example:

Click the **Add Repository** button to create the repository.



**Create Repository**

This location **MUST** be a shared filesystem accessible to all data instances in the cluster. If not, snapshots or restoration can fail.

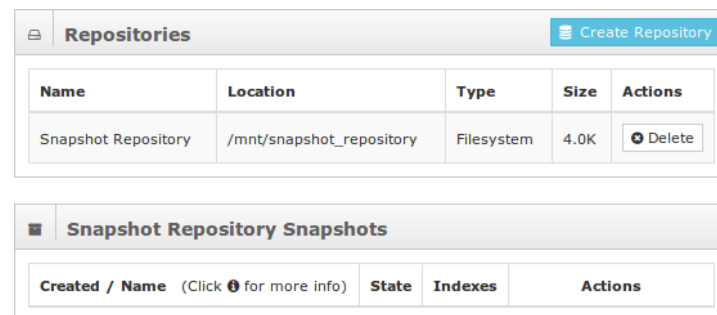
Repository Name:	Snapshot Repository
Repository Location:	/mnt/snapshot_repository

Close Add Repository

Now that you have created your snapshot repository you will see the repository in the Repositories tab and a new snapshot table for the repository (the table will be empty initially).

The new repository is listed with the following details:

- Name
- Location (/mnt/snapshot\_repository in this example)
- Type of Repository
- Size
- Actions
  - Allows you to delete a repository (cannot undo deletions)



Repositories				
Name	Location	Type	Size	Actions
Snapshot Repository	/mnt/snapshot_repository	Filesystem	4.0K	Delete

Snapshot Repository Snapshots				
Created / Name	(Click ⓘ for more info)	State	Indexes	Actions

After creating the repository you will need to update the maintenance settings to use this new repository, this is covered in the next section.

## Maintenance Settings

Maintenance is how Nagios Log Server performs tasks automatically on Indexes and Repositories. It is very simple to configure and once you set it up and save the settings it will work without any other interaction.

In the screenshot to the right you can see that the newly added repository called **Snapshot Repository** has been selected. This is the repository that will be used by Nagios Log Server for snapshots.

Maintenance Settings	
Optimize Indexes older than ?	<input type="text" value="2"/> days
Close indexes older than ?	<input type="text" value="30"/> days
Delete indexes older than ?	<input type="text" value="0"/> days
Repository to store snapshots in ?	Snapshot Repository ▾
Delete snapshots older than ?	<input type="text" value="720"/> days
Enable Maintenance and Snapshots ?	<input checked="" type="radio"/> Yes <input type="radio"/> No
Last modified ?	Tue, 31 Oct 2017 17:09:57 +1100
<input type="button" value="Save Settings"/>	

The other settings are explained as follows.

### Optimize Indexes older than:

This will use a Lucene forceMerge on an index that will not accept or ingest any new data.

Set this to 0 to disable this functionality.

### Close indexes older than:

Marks indexes older than this value as closed

Closed indexes do not take any system resources other than disk space, however they cannot be searched unless re-opened

Set to 0 to disable.

### Delete indexes older than:

Deletes indexes older than this value, freeing resources

This is permanent, the only way to restore a deleted index is from an archived snapshot

Set to 0 to disable.

Repository to store snapshots in:

This configures the maintenance worker to save snapshots to the repository that you select from this list  
You will need to create a new repository first to be able to select a repository

Delete snapshots older than:

Number of days before snapshots are deleted

The default is 720, but you can change this at any time

Enable Maintenance and Snapshots:

Enable or disable processing of all scheduled maintenance jobs

These jobs are also responsible for creating snapshots so you will want to make sure this is set to **Yes** if you want to have snapshots of your repository

After making any changes click the **Save Settings** button.

## Repository Snapshots

This table will show you the the indices that have had snapshots taken of them. If you have just created a new snapshot repository you will need to be patient, it may take up to a day before snapshots show up in the table.

Each index will have the following status and information:

- Name
  - The name of the index that has been saved
- Most Recent State
  - If the last snapshot for this index was successful it will be labeled as SUCCESS
- Most Recent Snapshot Time
  - The beginning and ending timestamps for the last snapshot to save this index
- Elasticsearch Version

- Shows the version of the most recent snapshot which contains this index.
- Versions
  - This shows a number of how many snapshots store this index
  - Each version can be seen by clicking the “Restore” button to the right of the table entry.
  - The number listed per index should not exceed the setting for "Close indexes older than"
- Actions
  - This allows you to restore from snapshots
  - Restoring a snapshot allows you to restore closed indexes that have not yet been deleted
    - Once you restore a closed index you can re-open it again via **Admin > System > Index Status**
    - Re-opening an index allows the data to be searched using a query
  - Snapshots can be deleted from the bottom-right corner of each table

### Select Indexes to Restore ✕

Restoring **logstash-2018.10.11** from time:

**Choose one:**

<input type="radio"/>	Mon Oct 15 2018 16:21:28 GMT-0500 (Central Daylight Time)
<input type="radio"/>	Sun Oct 14 2018 16:21:28 GMT-0500 (Central Daylight Time)
<input type="radio"/>	Sat Oct 13 2018 16:21:28 GMT-0500 (Central Daylight Time)
<input type="radio"/>	Fri Oct 12 2018 16:21:28 GMT-0500 (Central Daylight Time)

Index	Status	Time Range	Version	Versions	Actions
018.10.11	SUCCESS	2018-10-15 16:21:28 - 2018-10-15 16:21:30	1.7.6	4	<input type="button" value="Restore"/>

→



## Disk Space Usage

The amount of disk space consumed by snapshots will vary depending on several factors:

- The amount of log data received each day
- The frequency age at which you choose to delete old snapshots

You will want to observe your disk space usage patterns over time. If you start collecting log data from new sources then this will have an impact on disk space consumption. It is recommended to use Nagios XI to monitor the disk space usage of your snapshot repository so you can be alerted if you are running out of disk space.

## Snapshot Frequency

Snapshots are configured to run once a day as a system job. By default the time they are run is based on when you installed the first node in your Nagios Log Server cluster. Navigate to **Admin > System > Command Subsystem** and you will find the `snapshots_maintenance` system job.

**Nagios LS** Home Dashboards Alerting Configure Help **Admin** Search logs ... nagiosadmin Logout

**Command Subsystem**

The command subsystem runs all the jobs that are scheduled for backup, maintenance, and checks. It also runs occasional jobs that are required by other sections of the program. Other jobs use the command subsystem to run but are not listed here. System jobs that are in **waiting** status are normal.

System Jobs

Job ID	Job Status	Last Run Status	Last Run Time	Frequency	Next Run Time	Type	Actions
cleanup_cmdsubsys	Waiting	SUCCESS	10/31/2017 16:51:16	1 hour	10/31/2017 17:51:16	System	<a href="#">Edit</a> <a href="#">Run</a>
backups	Waiting	SUCCESS	10/31/2017 16:49:23	1 day	11/01/2017 16:49:23	System	<a href="#">Edit</a> <a href="#">Run</a>
snapshots_maintenance	Waiting	SUCCESS	10/31/2017 17:16:37	1 day	11/01/2017 17:16:37	System	<a href="#">Edit</a> <a href="#">Run</a>
run_all_alerts	Waiting	SUCCESS	10/31/2017 17:27:51	20 seconds	10/31/2017 17:28:11	System	<a href="#">Edit</a> <a href="#">Run</a>
run_update_check	Waiting	SUCCESS	10/31/2017 16:50:11	1 day	11/01/2017 16:50:11	System	<a href="#">Edit</a> <a href="#">Run</a>

From here you can change the frequency of the job using the **Edit** link or initiate one to run now using the **Run** link.

## Finishing Up

This completes the documentation on managing snapshots and maintenance in Nagios Log Server.

If you have additional questions or other support related questions, please visit us at our Nagios Support Forums:

<https://support.nagios.com/forum>

The Nagios Support Knowledgebase is also a great support resource:

<https://support.nagios.com/kb>