

Rebalancing cluster workloads

In this article

About workload balance for a GitHub Enterprise Server cluster

Checking the distribution of cluster jobs

Rebalancing allocations

Scheduling allocation rebalancing

Further reading

You can force your GitHub Enterprise Server cluster to evenly distribute job allocations for workloads on the cluster's nodes.

Who can use this feature

People with administrative SSH access to a GitHub Enterprise Server instance can rebalance cluster workloads on the instance.

GitHub determines eligibility for clustering, and must enable the configuration for your instance's license. Clustering requires careful planning and additional administrative overhead. For more information, see "[About clustering](#)."

About workload balance for a GitHub Enterprise Server cluster

A GitHub Enterprise Server instance in a cluster configuration assigns each task to a node according to the node's role. This assignment is called an allocation.

If a cluster node is unreachable by other nodes due to a hardware or software failure, your instance creates a new allocation to distribute jobs from the unhealthy node to another node that can handle the workload. In some situations, this distribution does not occur automatically, and a single node may run more jobs than expected.

You can manage allocations using the `ghe-cluster-balance` utility, which can display the status of existing allocations or force your instance to balance allocations. For example, you should balance allocations after you add a new node to the cluster. Optionally, you can schedule regular balancing.

You can run the following commands from any node in your cluster using the administrative shell. For more information, see "[Accessing the administrative shell \(SSH\)](#)."

Checking the distribution of cluster jobs

In some cases, such as hardware failure, the underlying software that manages allocations will migrate tasks from the unhealthy node to a healthy node. If the unhealthy node recovers, the task may remain assigned to the recovered node, which can result in unbalanced load. The risk of job failure may increase if allocations are unbalanced and additional nodes fail. You can check the distribution of allocations using the `ghe-cluster-balance status` utility.

- 1 To see a list of allocations, run the following command. The utility displays healthy allocations in green. If any jobs are not properly distributed, the utility displays the allocation's count in red.

Shell



```
ghe-cluster-balance status
```

- 2 If a job is not properly distributed, inspect the allocations by running the following command. Replace JOB with a single job or comma-delimited list of jobs.

Shell



```
ghe-cluster-balance status -j JOB
```

For example, to see the status of allocations for your instance's HTTP server and authorization service, you can run `ghe-cluster-balance status -j github-unicorn,authzd`.

Rebalancing allocations [↗](#)

After you determine which jobs are unbalanced across your cluster's nodes, you can rebalance allocations using the `ghe-cluster-balance rebalance` utility. The utility checks the distribution of existing jobs. If any jobs are unbalanced, the utility displays the jobs and prompts you to continue. If you continue, the utility creates new allocations to redistribute the jobs.

- 1 To perform a dry run and see the result of rebalancing without making changes, run the following command. Replace JOB with a single job or comma-delimited list of jobs.

Shell



```
ghe-cluster-balance rebalance --dry-run -j JOB
```

For example, to perform a dry run of rebalancing jobs for your instance's HTTP server and authorization service, you can run `ghe-cluster-balance rebalance --dry-run -j github-unicorn,authzd`.

- 2 To rebalance, run the following command. Replace JOB with a single job or comma-delimited list of jobs.

Shell



```
ghe-cluster-balance rebalance -j JOB
```

Scheduling allocation rebalancing [↗](#)

You can schedule rebalancing of jobs on your cluster by setting and applying configuration values for your GitHub Enterprise Server instance.

Note: Currently, you can only schedule reallocation of jobs for the HTTP server, `github-unicorn`.

- 1 To configure automatic, hourly balancing of jobs, run the following command.

Shell



```
ghe-config app.cluster-rebalance.enabled true
```

- 2 Optionally, you can override the default schedule by defining a cron expression. For example, run the following command to balance jobs every three hours.

Shell



```
ghe-config app.cluster-rebalance.schedule '0 */3 * * *'
```

- 3 To apply the configuration, run the following command.

Note: During a configuration run, services on your GitHub Enterprise Server instance may restart, which can cause brief downtime for users.

Shell



```
ghe-config-apply
```

- 4 Wait for the configuration run to complete.

Further reading [↗](#)

- ["Command-line utilities"](#)

Legal

© 2023 GitHub, Inc. [Terms](#) [Privacy](#) [Status](#) [Pricing](#) [Expert services](#) [Blog](#)