Initializing the cluster

In this article

About initialization of a GitHub Enterprise Server cluster

Installing GitHub Enterprise Server

Configuring the first node

Initializing the cluster

About the cluster configuration file

A GitHub Enterprise Server cluster must be set up with a license and initialized using the administrative shell (SSH).

> 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 initialization of a GitHub Enterprise Server cluster

To deploy a GitHub Enterprise Server cluster in your environment, you must install GitHub Enterprise Server, upload a cluster-enabled license, configure the first node, and initialize the node with a configuration file.

> **Note:** GitHub Enterprise Server clustering must be configured with HTTPS.

Installing GitHub Enterprise Server

To start setting up the cluster, install the GitHub Enterprise Server appliance on each node's virtual machine (VM), then configure an IP address.

1. On each cluster node, provision and install GitHub Enterprise Server. For more information, see "Setting up a GitHub Enterprise Server instance."

2. Using the administrative shell or DHCP, **only** configure the IP address of each node. Don't configure any other settings.

Configuring the first node

On the node that will function as your primary MySQL node, install your GitHub Enterprise Server license.

1. Connect to the node that will be designated as MySQL primary in `cluster.conf`. For more information, see "Initializing the cluster."

2. In your web browser, visit `https://<ip address>:8443/setup/`.

3. At the prompt, upload your license file and set a management console password. For more information, see "Managing your license for GitHub Enterprise."

4. In the Management Console, configure and save your desired settings.

5. The instance will restart automatically.

Initializing the cluster

To initialize the cluster, you need a cluster configuration file (`cluster.conf`). For more information, see "Initializing the cluster".

1. From the first node that was configured, run `ghe-cluster-config-init`. This will initialize the cluster if there are nodes in the cluster configuration file that are not configured.

2. Run `ghe-cluster-config-apply`. This will validate the `cluster.conf` file, apply the configuration to each node file and bring up the configured services on each node.

To check the status of a running cluster use the `ghe-cluster-status` command.

About the cluster configuration file

The cluster configuration file (`cluster.conf`) defines the nodes in the cluster, and what services they run. For more information, see "About cluster nodes."

This example `cluster.conf` defines a cluster with 11 nodes.

- Two nodes called `ghes-front-end-node-\*` run services responsible for responding to client requests.
- Three nodes called `ghes-database-node-\*` run services responsible for storage, retrieval, and replication of database data.
- Three nodes called `ghes-search-node-\*` run services responsible for search functionality.
- Three nodes called `ghes-storage-node-\*` run services responsible for storage, retrieval, and replication of data.

The names of the nodes can be any valid hostname you choose. The names are set as the hostname of each node, and will also be added to `/etc/hosts` on each node, so that the nodes are locally resolvable to each other.

Specify the first cluster node you configured as the MySQL primary via `mysql-server` and `mysql-master`.

```
[cluster]
  mysql-master = ghes-database-node-1
  redis-master = ghes-database-node-1
  primary-datacenter = primary
[cluster "ghes-front-end-node-1"]
  hostname = ghes-front-end-node-1
  ipv4 = 192.168.0.2
  # ipv6 = fd12:3456:789a:1::2
  consul-datacenter = primary
  datacenter = primary
  web-server = true
  job-server = true
  memcache-server = true
```

```
[cluster "ghes-front-end-node-2"]
  hostname = ghes-front-end-node-2
  ipv4 = 192.168.0.3
  # ipv6 = fd12:3456:789a:1::3
  consul-datacenter = primary
  datacenter = primary
  web-server = true
  job-server = true
  memcache-server = true
[cluster "ghes-database-node-1"]
  hostname = ghes-database-node-1
  ipv4 = 192.168.0.4
  # ipv6 = fd12:3456:789a:1::4
  consul-datacenter = primary
  datacenter = primary
  consul-server = true
  mysql-server = true
  redis-server = true
[cluster "ghes-database-node-2"]
  hostname = ghes-database-node-2
  ipv4 = 192.168.0.5
  # ipv6 = fd12:3456:789a:1::5
  consul-datacenter = primary
  datacenter = primary
  consul-server = true
  mysql-server = true
  redis-server = true
[cluster "ghes-database-node-3"]
  hostname = ghes-database-node-3
  ipv4 = 192.168.0.6
  # ipv6 = fd12:3456:789a:1::6
  consul-datacenter = primary
  datacenter = primary
  consul-server = true
  mysql-server = true
  redis-server = true
[cluster "ghes-search-node-1"]
  hostname = ghes-search-node-1
  ipv4 = 192.168.0.7
  # ipv6 = fd12:3456:789a:1::7
  consul-datacenter = primary
  datacenter = primary
  elasticsearch-server = true
[cluster "ghes-search-node-2"]
  hostname = ghes-search-node-2
  ipv4 = 192.168.0.8
  # ipv6 = fd12:3456:789a:1::8
  consul-datacenter = primary
  datacenter = primary
  elasticsearch-server = true
[cluster "ghes-search-node-3"]
  hostname = ghes-search-node-3
  ipv4 = 192.168.0.9
  # ipv6 = fd12:3456:789a:1::9
  consul-datacenter = primary
  datacenter = primary
  elasticsearch-server = true
[cluster "ghes-storage-node-1"]
  hostname = ghes-storage-node-1
  ipv4 = 192.168.0.10
  # ipv6 = fd12:3456:789a:1::10
  consul-datacenter = primary
  datacenter = primary
  git-server = true
  pages-server = true
  storage-server = true
  metrics-server = true
[cluster "ghes-storage-node-2"]
  hostname = ghes-storage-node-2
  ipv4 = 192.168.0.11
  # ipv6 = fd12:3456:789a:1::11
  consul-datacenter = primary
  datacenter = primary
  git-server = true
  pages-server = true
  storage-server = true
  metrics-server = true
[cluster "ghes-storage-node-3"]
  hostname = ghes-storage-node-3
  ipv4 = 192.168.0.12
  # ipv6 = fd12:3456:789a:1::12
  consul-datacenter = primary
  datacenter = primary
  git-server = true
  pages-server = true
  storage-server = true
  metrics-server = true
```

Create the file `/data/user/common/cluster.conf` on the configured first node. For example, using `vim`:

```
ghe-data-node-1:~$ sudo vim /data/user/common/cluster.conf
```

Legal

© 2023 GitHub, Inc. Terms Privacy Status Pricing Expert services Blog