GlusterFS + Heketi [Ubuntu 18.04]

> Requirement to this guide : Having an empty / unused partition available for configuration on all bricks. Size does not really matter, but it needs to be the same on all nodes.

Configuring your nodes

Configuring your **/etc/hosts** file :

```
## on gluster00 :
127.0.0.1 localhost localhost.localdomain glusterfs00
10.1.1.3 gluster01
10.1.1.4 gluster02

## on gluster01
127.0.0.1 localhost localhost.localdomain glusterfs01
10.1.1.2 gluster00
10.1.1.4 gluster02

## on gluster02
127.0.0.1 localhost localhost.localdomain glusterfs02
10.1.1.2 gluster00
10.1.1.3 gluster01
```

Installing glusterfs-server on your bricks (data nodes). In this example, on gluster00 and gluster01 :

```
apt update
apt upgrade
apt-get install software-properties-common
add-apt-repository ppa:gluster/glusterfs-7
apt-get install glusterfs-server
```

Enable/Start GLuster

```
systemctl enable glusterd
systemctl start glusterd
```

Connect on either node peer with the second host. In this example I'm connected on gluster00 and allow peer on the other hosts using the hostname :

```
gluster peer probe gluster01
```

Should give you something like this :

```
Number of Peers: 1

Hostname: gluster01
Uuid: 6474c4e6-2957-4de7-ac88-d670d4eb1320
State: Peer in Cluster (Connected)
```

> If you are going to use Heketi skip the volume creation steps

Creating your storage volume

Now that you have both of your nodes created and in sync, you will need to create a volume that your clients will be able to use.

Syntax :

```
gluster volume create $VOL_NAME replica $NUMBER_OF_NODES transport tcp
$DOMAIN_NAME1:/path/to/directory $DOMAIN_NAME2.com:/path/to/directory force

## actual syntax in for our example

gluster volume create testvolume replica 2 transport tcp glusterfs00:/gluster-volume
glusterfs01:/gluster-volume force
```

Start the volume you have created :

```
gluster volume start testvolume
```

Configuring your client(s)

```
apt-get install software-properties-common
add-apt-repository ppa:gluster/glusterfs-7
apt install glusterfs-client
```

Once completed, you will need to mount the storage that you previously created. First, make sure you have your mount point created :

```
mkdir /gluster-data
```

Mount your volume to your newly created mount point :

```
mount -t glusterfs gluster00:testvolume /gluster-data
```

Adding / Removing a brick from production

Once your node is ready with the proper packages and updates...
Make sure to edit its /etc/hosts and update every other nodes as well with your new entry :

```
echo "10.1.1.5 gluster03" >> /etc/hosts
```

Adding a new brick

Once you've completed the above points, simply connect on a node already part of the cluster :

```
gluster peer probe gluster03
```

And connect it to the volumes you want the new node to be connected to :

```
gluster volume add-brick testvolume replica 3 gluster03:/gluster-volum
```

Removing a clustered brick
Re-adding a node that has been previously removed

Install Heketi on **one** of the nodes

> Requirement : Already existing GlusterFS install

Download Heketi bin

```
wget https://github.com/heketi/heketi/releases/download/v9.0.0/heketi-
v9.0.0.linux.amd64.tar.gz
tar -zxvf heketi-v9.0.0.linux.amd64.tar.gz
```

Copy bin

```
chmod +x heketi/{heketi,heketi-cli}
cp heketi/{heketi,heketi-cli} /usr/local/bin
```

Check heketi is working

```
heketi --version
heketi-cli --version
```

Add a user/group for heketi

```
groupadd --system heketi
useradd -s /sbin/nologin --system -g heketi heketi
```

Create dir for heketi

```
mkdir -p /var/lib/heketi /etc/heketi /var/log/heketi
```

```
vim /etc/heketi/heketi.json
```

> Make sure you replace the "key" values with proper passwords

```
{
  "_port_comment": "Heketi Server Port Number",
  "port": "8080",

"_enable_tls_comment": "Enable TLS in Heketi Server",
"enable_tls": false,

"_cert_file_comment": "Path to a valid certificate file",
"cert_file": "",

"_key_file_comment": "Path to a valid private key file",
"key_file": "",



  "_use_auth": "Enable JWT authorization. Please enable for deployment",
  "use_auth": false,

  "_jwt": "Private keys for access",
  "jwt": {
    "_admin": "Admin has access to all APIs",
    "admin": {
      "key": "KEY_HERE"
```

```
    },
    "_user": "User only has access to /volumes endpoint",
    "user": {
      "key": "KEY_HERE"
    }
  },

  "_backup_db_to_kube_secret": "Backup the heketi database to a Kubernetes secret when running
in Kubernetes. Default is off.",
  "backup_db_to_kube_secret": false,

  "_profiling": "Enable go/pprof profiling on the /debug/pprof endpoints.",
  "profiling": false,

  "_glusterfs_comment": "GlusterFS Configuration",
  "glusterfs": {
    "_executor_comment": [
      "Execute plugin. Possible choices: mock, ssh",
      "mock: This setting is used for testing and development.",
      "      It will not send commands to any node.",
      "ssh:  This setting will notify Heketi to ssh to the nodes.",
      "      It will need the values in sshexec to be configured.",
      "kubernetes: Communicate with GlusterFS containers over",
      "            Kubernetes exec api."
    ],
    "executor": "ssh",

    "_sshexec_comment": "SSH username and private key file information",
    "sshexec": {
      "keyfile": "/etc/heketi/heketi_key",
      "user": "root",
      "port": "22",
      "fstab": "/etc/fstab"
    },

    "_db_comment": "Database file name",
    "db": "/var/lib/heketi/heketi.db",

     "_refresh_time_monitor_gluster_nodes": "Refresh time in seconds to monitor Gluster
nodes",
```

```
    "refresh_time_monitor_gluster_nodes": 120,

    "_start_time_monitor_gluster_nodes": "Start time in seconds to monitor Gluster nodes when
the heketi comes up",
    "start_time_monitor_gluster_nodes": 10,

    "_loglevel_comment": [
      "Set log level. Choices are:",
      "  none, critical, error, warning, info, debug",
      "Default is warning"
    ],
    "loglevel" : "debug",

    "_auto_create_block_hosting_volume": "Creates Block Hosting volumes automatically if not
found or exsisting volume exhausted",
    "auto_create_block_hosting_volume": true,

    "_block_hosting_volume_size": "New block hosting volume will be created in size mentioned,
This is considered only if auto-create is enabled.",
    "block_hosting_volume_size": 500,

    "_block_hosting_volume_options": "New block hosting volume will be created with the
following set of options. Removing the group gluster-block option is NOT recommended.
Additional options can be added next to it separated by a comma.",
    "block_hosting_volume_options": "group gluster-block",

    "_pre_request_volume_options": "Volume options that will be applied for all volumes
created. Can be overridden by volume options in volume create request.",
    "pre_request_volume_options": "",

    "_post_request_volume_options": "Volume options that will be applied for all volumes
created. To be used to override volume options in volume create request.",
    "post_request_volume_options": ""
  }
}
```

Load all Kernel modules that will be required by Heketi.

```
for i in dm_snapshot dm_mirror dm_thin_pool; do
  sudo modprobe $i
```

```
done
```

Create ssh key for the API to connect to the other hosts

```
ssh-keygen -f /etc/heketi/heketi_key -t rsa -N ''
chown heketi:heketi /etc/heketi/heketi_key*
```

Send key to all hosts

```
for i in gluster00 gluster01 gluster02; do
  ssh-copy-id -i /etc/heketi/heketi_key.pub root@$i
done
```

Create a systemd file

```
vim /etc/systemd/system/heketi.service
```

```
[Unit]
Description=Heketi Server

[Service]
Type=simple
WorkingDirectory=/var/lib/heketi
EnvironmentFile=-/etc/heketi/heketi.env
User=heketi
ExecStart=/usr/local/bin/heketi --config=/etc/heketi/heketi.json
Restart=on-failure
StandardOutput=syslog
StandardError=syslog

[Install]
WantedBy=multi-user.target
```

Reload systemd and enable new heketi service

```
systemctl daemon-reload
systemctl enable --now heketi
```

Allow heketi user perms on folders

```
chown -R heketi:heketi /var/lib/heketi /var/log/heketi /etc/heketi
```

Create topology

```
vim /etc/heketi/topology.json
```

```
{
  "clusters": [
    {
      "nodes": [
                    {
          "node": {
            "hostnames": {
              "manage": [
                "gluster00"
              ],
              "storage": [
                "10.1.1.2"
              ]
            },
            "zone": 1
          },
          "devices": [
            "/dev/vdc","/dev/vdd","/dev/vde"
          ]
        },            {
          "node": {
            "hostnames": {
              "manage": [
                "gluster01"
              ],
              "storage": [
                "10.1.1.3"
              ]
            },
            "zone": 1
          },
          "devices": [
            "/dev/vdc","/dev/vdd","/dev/vde"
          ]
        },            {
          "node": {
```

```
            "hostnames": {
              "manage": [
                "gluster02"
              ],
              "storage": [
                "10.1.1.4"
              ]
            },
            "zone": 1
          },
          "devices": [
            "/dev/vdc","/dev/vdd","/dev/vde"
          ]
        }
      ]
    }
  ]
}
```

Load topology

(note you can make changes and the load it again in the future if you want to add more drives)

```
heketi-cli topology load --json=/etc/heketi/topology.json
```

Check connection to other devices work

```
heketi-cli cluster list
```

Notes

Mount all volumes

```
for i in `gluster volume list`
do mkdir -p /etc/borg/gluster_backup/$i && \
mount -t glusterfs 127.0.0.1:$i /mnt/$i
done
```

Revision #10
Created 17 January 2020 08:53:34 by Dave

Updated 16 March 2020 09:07:50 by Dave