

High Availability Automation Setup Installation Guide for EPP 8.3 Standalone Setup and Migration

Fresh EPP Standalone HA Installation Guide

Overview: This document will help users to configure the High Availability setup for EPP 8.3 console within the network. High availability focuses on ensuring maximum availability, regardless of disruptions or events that may occur.

Applicable Version: EPP 8.3

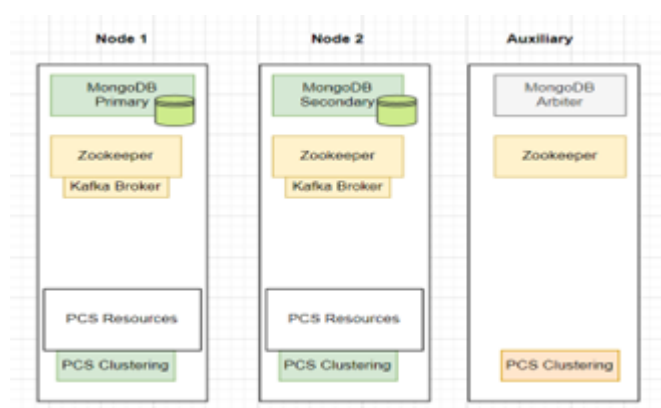
Following are the pre-requisites and steps to configure HA Setup.

Prerequisite:

- **Product Key:** Enable the HA flag on the key. Note that the HA can only be enabled after the key is activated.
- 1 Floating IP in the same network (it should be a reserved IP not allocated to any machine)
- 3 HA Ubuntu 22.04 Machines - 2 with the required normal hardware setup as per EPP licence and 1 with lower configuration for Auxiliary node (for example, 2CPU/4GB RAM).

***Keep fresh server snapshot if possible – Optional Step

High Level Architecture:



1. To set up Node 1 and Node 2,

- Run the following commands in sequence:

- sudo apt update
- apt --fix-broken install

- Install EPP setup and check application login post activation.

- After completing the EPP installation on both the nodes, Run the following commands in sequence again to avoid any library or broken installation issues

- sudo apt update
- Please install ssh if not pre-installed and run command: **apt install sshpass -y**
- **apt --fix-broken install** - run this command if prompted during above install
- Reboot Node 1 and Node 2 and check if login is working in standalone.

2. To Setup Auxiliary Node

- Steps to Install Ansible.

- sudo apt update
- apt --fix-broken install
- sudo add-apt-repository ppa:ansible/ansible-2.10
- sudo apt install ansible
- Please install ssh if not pre installed and run command: **apt install sshpass -y**

To re validate the installation of sshpass and Ansible, execute: **apt install sshpass ansible -y** & Below shown output is expected.

```
ansible is already the newest version (2.10.7+merged+base+2.10.8+dfsg-1).
sshpass is already the newest version (1.09-1).
0 upgraded, 0 newly installed, 0 to remove and 410 not upgraded.
```

- ansible --version

```
ansible 2.10.8
config file = /etc/ansible/ansible.cfg
configured module search path = ['/root/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
ansible python module location = /usr/lib/python3/dist-packages/ansible
executable location = /usr/bin/ansible
python version = 3.10.6 (main, Mar 10 2023, 10:55:28) [GCC 11.3.0]
```

- **Execute below to change locale encoding** - Ansible requires the locale encoding to be UTF-8,

- sudo nano /etc/default/locale

Make below changes mentioned:

LANG=en_US.UTF-8

LC_ALL=en_US.UTF-8

Or

sudo update-locale LANG=en_US.UTF-8 LC_ALL=en_US.UTF-8

- Validate changes after above command is executed, less /etc/default/locale
- **Reboot** the Auxiliary Node to implement the above changes.
- To check ansible Installation, execute: **ansible --version**

```

ansible [core 2.15.2]
  config file = /etc/ansible/ansible.cfg
  configured module search path = ['/root/.ansible/plugins/modules', '/usr/share/ansible/plug
  ansible python module location = /usr/lib/python3/dist-packages/ansible
  ansible collection location = /root/.ansible/ansible/ansible/collections:/usr/share/ansible/collections
  executable location = /usr/bin/ansible
  python version = 3.10.6 (main, Mar 10 2023, 10:55:28) [GCC 11.3.0] (/usr/bin/python3)
  jinja version = 3.0.3
  libyaml = True

```

3. Make the following changes in /etc/ansible/ansible.cfg in Auxiliary Node:

***If above folder and file is not present , please create following above

- set `host_key_checking` to `False` (**Uncomment the line if commented**)
- set `log_path` to `/var/log/ansible.log` (**Uncomment the line if commented**)

Add below:

[defaults]

log_path = /var/log/ansible.log

host_key_checking = False

Execute: **export ANSIBLE_HOST_KEY_CHECKING=False**

4. Setup for HA Automation Script in Auxiliary Node:

Copy HA build zip file from: [/EPS8.2_HA_Config.tar.gz](#). to /root directory.

To extract, run: **tar -xzvf EPS8.2_HA_Config.tar.gz**.

5. Ansible Configuration details:

- Ansible playbooks and inventories are located within the **ha_automation/ansible** directory.
- Navigate to the inventories directory and ensure to update the **hosts.ini** and **vars.yml** files with the necessary parameters as needed.

- In the **hosts.ini** file, update the IP addresses of the nodes for a 3-node deployment. However, avoid making changes to any existing host groups such as [all_nodes] or [eps_nodes], as modifying them might result in playbook failures.

```
[all_nodes]
epsnode1 ansible_hostname=epsnode1 ansible_host=17.10.1.18 epsnode_id=1
epsnode2 ansible_hostname=epsnode2 ansible_host=17.10.1.1 epsnode_id=2
epsnode3 ansible_hostname=epsnode3 ansible_host=17.10.1.7 epsnode_id=3

[eps_nodes]
epsnode1
epsnode2

[arbiter_nodes]
epsnode3

[primary_node]
epsnode1
```

- In the vars.yml file, ensure that you provide the accurate floating IP address along with the corresponding CIDR or subnet mask. You can validate the CIDR by executing the command ip a. It is important to ensure that the subnet mask matches the physical IP addresses of the nodes to prevent any issues with the assignment of the floating IP.

```
VIP: "172.10.110.10"
CIDR: "23"
#####MongoDB#####
db_cls_username: "dbclusteradmin"
db_cls_password: "StrongPassword"
db_replica_set: "rs-0"
#####PCS CLUSTER#####
pcm_k_cluster_name: hacluster
pcm_k_user: hacluster
pcm_k_password: haadmincluster
#####NGINX#####
max_temp_file_size: "0"
arbiter: "epsnode3"
```

6. To avoid lock errors on all three nodes, follow these steps:

- Rerun to check if any new updates
 - apt-get update
 - apt --fix-broken install
 - apt -get -y purge unattended-upgrades
 - systemctl disable --now apt-daily.timer
 - systemctl disable --now apt-daily-upgrade.timer
 - systemctl daemon-reload

Note: Wait for 15 to 20 minutes to allow all machines to stabilize. During this time, various processes such as updates, snapd updates, and unattended-upgrades will run on the auxiliary node. You can monitor these processes using the top command or by checking CPU usage.

After the 15- to 20-minutes, use the top command to verify CPU usage and ensure that no processes are holding any locks.

- Execute to check lock: **sudo lsof /var/lib/dpkg/lock**
- **Run HA Automation Script:** To configure HA using Ansible, execute the following command with the root user privileges from **ha_automation/ansible** directory.

Execute HA Automation: **ansible-playbook site.yml -i inventories/hosts.ini --ask-pass**

*****NOTE:** skip **--ask-pass** if **passwordless-ssh** connection between controller and remote nodes have been established.

The above command should run the automation script with necessary changes.

```
PLAY RECAP *****
epsnode1      : ok=242  changed=173  unreachable=0    failed=0    skipped=31    rescued=0    ignored=6
epsnode2      : ok=196  changed=146  unreachable=0    failed=0    skipped=52    rescued=0    ignored=4
epsnode3      : ok=75   changed=51   unreachable=0    failed=0    skipped=52    rescued=0    ignored=4
```

- To verify the cluster's status after the setup:

Execute: **pcs status**

```
Cluster name: hacluster
Cluster Summary:
 * Stack: corosync
 * Current DC: epsnode2 (version 2.1.2-ada5c3b36e2) - partition with quorum
 * Last updated: Wed Aug 23 09:49:06 2023
 * Last change: Wed Aug 23 09:48:07 2023 by root via crm_resource on epsnode1
 * 3 nodes configured
 * 6 resource instances configured

Node List:
 * Online: [ epsnode1 epsnode2 epsnode3 ]

Full List of Resources:
 * Resource Group: eps_pcs_group:
 * virtual_ip      (ocf:heartbeat:IPaddr2):      Started epsnode1
 * lsyncd_service (systemd:lsyncd):      Started epsnode1
 * redis_service  (systemd:redis-server): Started epsnode1
 * wildfly_service (systemd:wildfly):      Started epsnode1
 * consumer_service (systemd:consumer):    Started epsnode1
 * nginx_service  (ocf:heartbeat:nginx):   Started epsnode1

Daemon Status:
 corosync: active/enabled
 pacemaker: active/enabled
 pcsd: active/enabled
```

7. Setup Alert: Execute below in Node1 or Node 2

/opt/Seqrite_EndPoint_Security/eps_alert/eps_alert.sh --enable

Use **--disable** to disable the alerts

- The email recipient in the alerts is dynamically selected by Ansible based on the recipient configured by the Quick Heal EPP 8.3 installer for the health check scripts during start-up. The source email address is **eps@seqrite.com**.

```
Alerts:
Alert: pcs_alert (path=/var/lib/pacemaker/alert_sntp.sh)
Options: email_sender=eps@seqrite.com
Recipients:
Recipient: pcs_alert-recipient (value=perf@yopmail.com)
```

Existing client from 7.6 to 8.3 migration

Migration with HA needs some specific steps, below are the details:

1. Setup HA, and make sure the to check the PC cluster:

- Run Command in Terminal (any server node) – “**pcs status**”
- For more information, please check HA documentation.
- Make sure HA’s pre-requisites are followed.

2. Start the Migration process by downloading the tool from the HA setup(server)

- Download the Export Tool from the HA setup to migrate data for 7.6 and import the same via the 8.3 HA console.
- Download the client Migration tool [SSP] from HA setup (Path: Deployment>>EPP 7.6 Migration Tab) and run this SSP on the 7.6 server for client migration.

Note:

Please make sure Win7 client Machines are IE 11 Browser is installed [EPP 8.2 Server Client] before running the SSP on 7.6 server.