Configuring database cluster

Once the database replica is setup, it's important to guarantee transactianality rules. To achive it, one database instance must be acting as the master and other as the slave.

Using corosync and pacemaker, you can configure a floating IP address that will mark which one is the active one at each moment.

Node 1	Node 2
Install Corosync and Pacemaker. It is recommended to use **apt** or **yum** because these programs will handle dependencies for you, making the process much easier.	Install Corosync and Pacemaker.
Cluster nodes need a key in order to authenticate the packages sent between them by corosync. *sudo corosync-keygen* Once the key has been generated, copy it to the other nodes: *sudo scp /etc/corosync/authkey <user>@<other-cluster-node>:/home/<user>*	
	Once the key has been copied, move the copied key from the */home/<user>* route to */etc/corosync/authkey*
Now we need to tell Corosync which IP to use to communicate with other nodes in the cluster. Open */etc/corosync/corosync.conf* and edit the bindnetaddr field. Set the right IP and save the file. We need to do this in every node in the cluster, although you can use the same file if you set the right name in your hosts file.	
	Configure Corosync with the right IP binding as done in node 1.
Configure the */etc/default/corosync* file to enable Corosync changing START to yes "START=yes". Then we can start Corosync using *sudo service corosync start*.	
	Enable Corosync and start it as in node 1.

Node 1	Node 2
Allow the nodes a few seconds to start, then you can monitor the cluster nodes using sudo crm_mon. The result should be similar to this: ============ *Last updated: Mon Mar 31 14:05:23 2015* *Stack: corosync* *Current DC: yourDC - partition with quorum* *Version: 1.x.x-yourversion* *2 Nodes configured, 2 expected votes* *0 Resources configured.* ============ *Online: [node1 node2]*	
	Check the nodes with sudo crm_mon
Corosync is ready, now we will tell Pacemaker which resources we want it to handle in HA. These will be the database and a virtual IP (VIP) we will use to address the cluster. Add the VIP to the node, and then use this to create the resource: *sudo crm configure primitive FAILOVER-ADDR ocf:heartbeat:IPaddr2 params ip="your.virtual.IP" nic="your.network.device" op monitor interval="10s" meta is-managed="true"* You can check the result using *sudo crm status*, which should look something like: *Last updated: Wed Jan 18 10:21:12 2017 Last change: Tue Jan 17 13:08:25 2017 by hacluster via crmd on nodename* *Stack: corosync* *Current DC: nodename(version 1.1.14-70404b0) - partition with quorum* *2 nodes and 2 resources configured* *Online: [node1 node2]* *Full list of resources:* *Resource Group: my_cluster* *FAILOVER-ADDR (ocf::heartbeat:IPaddr2): Started node2*	
Now we will add the database. We will use: *sudo crm configure primitive FAILOVER-MARIADB lsb::mysql op monitor interval=15s*	

Revision #4
Created 17 March 2021 09:32:55
Updated 1 June 2022 14:30:39