Testing OpenLDAP in High Availability Environments on SLE12

Peter Varkoly, varkoly@suse.com

Februar 2014

A Failover setup for OpenLDAP for testing.

The following sections introduce an example setup for a High Available OpenLDAP Service. It is a ivirtualized two-node setup where both nodes are running OpenLDAP using *MirrorMode* replication to keep the LDAP Data and Configuration synchronized.

1 Preparations

Network Configuration Each node in the cluster has two separate network interfaces. One is purely used for the cluster's internal communication (LDAP replication traffic and internal communication of the different cluster components). The other one is used for the external traffic, i.e. mainly the LDAP clients accessing the LDAP service. In this setup the cluster internal interfaces will use IP Addresses from the `192.168.100.0/24` subnet while the external/public interfaces are on the `192.168.101.0/24` network.

- Setup a virtualisation server with IP `192.168.101.1`.
- Configure `eth0` on `node1` with IP Address `192.168.100.2`.
- Configure `eth0` on `node2` with IP Address `192.168.100.3`.
- Configure `eth1` on `node1` with IP Address `192.168.101.2`.
- Configure `eth1` on `node2` with IP Address `192.168.101.3`.
- Add following lines to `/etc/hosts/` on both node:

  ```
  192.168.100.2 node1
  192.168.100.3 node2
  ```

- Enable ssh communication from `node1` to `node2`.

Software Installation Following software components need to be installed for the test setup on the nodes:

- openldap2
- PaceMaker
- Resource Agents
- CoroSync
- ClusterGlue

You can find these on *High Availability Extensions* Addon.

Time Synchronization For MirrorMode Replication to work accurately it is important that the system clocks on the participating nodes are tightly in sync. So it is recommended to setup NTP to synchronize the clocks of both nodes.

2 OpenLDAP MirrorMode Configuration

Setup an OpenLDAP server on the both nodes with yast2 auth-server module with subtree `dc=example,dc=com`. No TLS is required. Set the LDAP-Administrator password to `secret`. Because the `slapd` service must be controlled by the cluster disable starting on boot:

```
systemctl disable slapd.service
```

2.1 Setup first node

The first node can be configured by executing following command:

```
ldapadd -x -D cn=Administrator,dc=example,dc=com -w secret -f node1.add.ldif
ldapadd    -Y external -H ldapi:/// -f node1.api.add.ldif
ldapmodify -Y external -H ldapi:/// -f node1.api.modify.ldif
```

2.2 Second Node

The second node can be configured by executing following command:

```
ldapadd    -Y external -H ldapi:/// -f node2.api.add.ldif
ldapmodify -Y external -H ldapi:/// -f node2.api.modify.ldif
```

3 Pacemaker/Corosync Setup

3.1 Basic Setup

Execute node1.ha.setup.sh on `node1` to basic setup of the cluster.

3.1.1 Starting and Verifying the Configuration

The cluster software can now be started to verify the configuration. On SUSE Linux Enterprise it is done by running this command on every node:

```
systemctl enable corosync.service
systemctl enable pacemaker.service
systemctl start corosync.service
systemctl start pacemaker.service
```

To check the cluster status now the `crm_mon` command is used. You have to wait some seconds until the cluster get ready. If all nodes are online the output should be similar to this:

```
Last updated: Sat Mar  1 18:24:56 2014
Last change: Sat Mar  1 18:21:38 2014 by hacluster via crmd on node2
Stack: corosync
Current DC: node2 (1084777475) - partition with quorum
Version: 1.1.10+git20140207.6290953-1.10-1.1.10+git20140207.6290953
2 Nodes configured
0 Resources configured


Online: [ node1 node2 ]
```

3.2 Resources

Execute node1.ha.recources.sh on `node1` to setup the resources on the cluster.
If everything went right the virtual IP address `192.168.101.200` has been set up on one of the node. You can verify it by the command: `ip addr`. The `slapd` should be running on both nodes in the cluster now. Use `crm_mon` to verify that. Its output should contain something like:

```
Last updated: Mon Mar  3 15:22:59 2014
Last change: Mon Mar  3 14:57:39 2014 by root via cibadmin on node2
Stack: corosync
Current DC: node1 (1084777474) - partition with quorum
Version: 1.1.10+git20140207.6290953-1.10-1.1.10+git20140207.6290953
2 Nodes configured
3 Resources configured


Online: [ node1 node2 ]

 Clone Set: ldap_clone [slapd_mirrormode]
     Started: [ node1 node2 ]
ldap-v_ip       (ocf::heartbeat:IPaddr2):       Started node1
```

3.3 Testing and Monitoring

The setup is now basically complete and ready for some testing. Monitoring the current status of the cluster can be done with the `crm_mon` tool. For testing if the failover of the virtual IP address correctly works when slapd crashes on the currently active node, you can for example kill the slapd process on that node. Pacemaker will first try to restart slapd. After slapd was killed for the third time, pacemaker should re-assign the virtual IP address to the second node. After that the `slapd_mirrormode` resource should be shown as `stopped` on the failed node.

```
Last updated: Mon Mar  3 15:27:04 2014
Last change: Mon Mar  3 14:57:39 2014 by root via cibadmin on node2
Stack: corosync
Current DC: node1 (1084777474) - partition with quorum
Version: 1.1.10+git20140207.6290953-1.10-1.1.10+git20140207.6290953
2 Nodes configured
3 Resources configured


Online: [ node1 node2 ]

 Clone Set: ldap_clone [slapd_mirrormode]
     Started: [ node2 ]
     Stopped: [ node1 ]
ldap-v_ip       (ocf::heartbeat:IPaddr2):       Started node2

Failed actions:
    slapd_mirrormode_monitor_10000 on node1 'not running' (7): call=22, status=complete, la
```