redhat.

SAP Training @ HPE Walldorf

SAP HANA System Replication with Pacemaker

Markus Koch

Objective

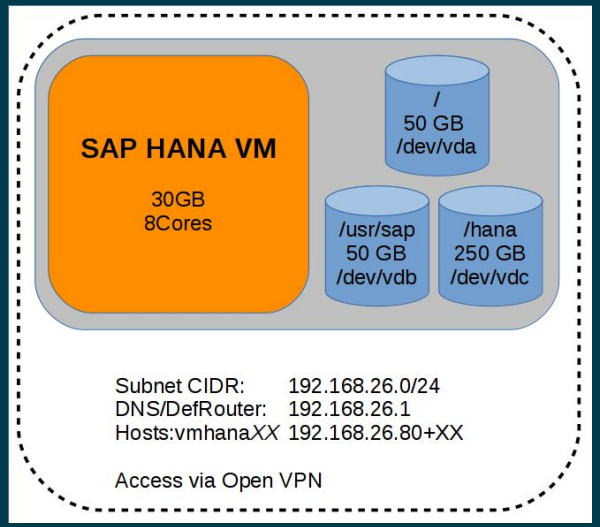
In the first part of this course you have learned how to install a HANA database on RedHat Enterprise Linux and how to find documentation to keep up to date for newer versions

This part guides you through the setup of a pacemaker cluster for HANA Scale-Up scenarios

In the end you will have working cluster setup

WORKSHOP ENVIRONMENT

RHEL 7.5 Instance hosted on RHV in HPE Performance Center in BBN



- Instance size: 16 cores, 48GB
- 10GB & 120GB (thin provisioned)

- SAP HANA pre provisioned (depending on lab)
- SSH available - cloud-user/redhat01
- Use “sudo” to become root
- Service addresses used for cluster:
  - hpi-vhana50 to hpi-vhana80

Pacemaker Cluster Workflow

The following steps need to be performed

- “Triple” check DNS and time setup is working
- Setup HANA system replication and check it is working
- Install the pacemaker cluster packages
- Configure Cluster incl. Fencing
- Check fencing is working
- Include resource groups for HANA into cluster
- Check again

Overview of test environment

SAP HANA Configuration

SID: L90 Instance Number: 10

Host	Site Name	Role
hpi-vmXX (node1)	DC1	master
hpi-vmYY (node2)	DC2	slave

Please note: For this class we share the production and heartbeat network. This is not recommended for production environments. You can use 172.25.25.XX/24 for heartbeat.

HANA System Replication

Setting up HANA System Replication

Supported scenarios

- SAP HANA Scale-Up System Replication only
- Engineering Support for HANA Scale-Out SR
- No support for HANA SR installations configured with a standby node
- Partial Support for cost-optimized scenario, because customized setup required.
  See also HOW TO SET UP SAPHanaSR IN THE COST OPTIMIZED SCENARIO
- "Multitier System Replication"/"replication chains" are possible. Please note: tertiary site can not be managed by the cluster
- SAP HANA System Replication "Multiple components One Database (MCOD)" or “Multiple Database Containers (MDC)” (http://scn.sap.com/docs/DOC-59893)
- MCOS is only supported if all databases running on the hosts are replicated and the replication is always to the same secondary node.

Setting up HANA System Replication

Supported scenarios

Using Full Sync Replication is possible, but due to the way Full Sync Replication works some functionality of the cluster will be restricted.

- for example the automatic start of the HANA instances on both nodes when the cluster is started will not work and the HANA instance on the secondary node will have to be started manually for the cluster to be able to resume operation if Full Sync Replication is enabled; to determine if Full Sync Replication is active the command "`hdbcons -e hdbindexserver "replication info"`" can be used (check for "`[system_replication] enable_full_sync`" and `"ReplicationFullSync"`)

Setting up HANA System Replication

Technical Requirements

- Two-node clusters only
- Both SAP HANA instances must have the same SAP Identifier (SID) and Instance-Number (HXE and 90)
- All nodes must be in the same network segment (layer 2)
- Technical users and groups such as "<SID>adm" must be identically defined on all cluster nodes
- Name resolution of the cluster nodes and the virtual IP address can be done locally on all cluster nodes
- Time on all cluster nodes must be in sync (using NTP or some other time synchronization method)

Note: If the cluster nodes are installed in different data centers or data center areas, the environment must match both the requirements defined by SAP for HANA System Replication (see chapter "4.2 Distance between data centers" in the SAP "How to Perform System Replication for SAP HANA" guide) and also the RHEL HA add-on stretch cluster requirements (see Support for Red Hat Enterprise Linux High Availability Cluster Stretch and Multi-Site Architectures), specifically the network latencies between the nodes and the recommended maximum distance

Setting up HANA System Replication

Configuring HANA for System Replication - Check log_mode

Done by ansible role saphana-hsr

- For the System replication to work the SAP HANA "log_mode" variable must be set to "normal". You can verify this after installation with the following command: (must be done on all nodes)

```
# su - hxeadm
[hxeadm]#  hdbsql -u system -i 90 "select value from "SYS"."M_INIFILE_CONTENTS" \
-d SystemDB  where
key='log_mode'"
Password:  System123
VALUE
"normal"
1 row selected (overall time 191.885 msec; server time 188.704 msec)
```

Setting up HANA System Replication

Configuring HANA for System Replication - Check Autostart

Done by ansible role saphana-hsr

- Please make sure that SAP HANA is not configured to automatically start after the system boot, since startup and shutdown of SAP HANA will be controlled by the cluster (must be done on all nodes):

```
# grep Autostart /usr/sap/HXE/SYS/profile/*
/usr/sap/HXE/SYS/profile/HMC_HDB90_hpi-vmXX:Autostart = 0
```

Setting up HANA System Replication

Configuring HANA for System Replication - Reduce amount of log backups

DO NOT USE IN PRODUCTION

- We are on a test environment with limited disk space, so we want only most necessary log entries, so we will set "log_backup_timeout_s" in the [persistence] section of the global.ini to 0. See SAP Note 1642148 - FAQ: SAP HANA SB Backup & Recovery for more options.
  (must be done on all nodes):

```
hxeadm# hdbsql -i 90 -u system -p System123 "alter system alter configuration
('global.ini', 'SYSTEM') SET ('persistence', 'log_backup_timeout_s') = 0"
```

- Check the current active value

```
[hxeadm]# hdbsql -i 90 -u system -p System123 "select value from
"SYS"."M_INIFILE_CONTENTS" where key='log_backup_timeout_s'
"VALUE
"0"
1 row selected (overall time 168.393 msec; server time 157.142 msec)
```

Setting up HANA System Replication

Configuring HANA for System Replication - Prepare Node 01 (Master)

Done by ansible role saphana-hsr

- Backup the DB (SAP HANA System replication will not work until a backup has been performed)

```
[hxeadm]# hdbsql -i 90 -u system -p System123 "BACKUP DATA USING FILE ('/tmp/foo')"
0 rows affected (overall time 35.458355 sec; server time 35.426372 sec)
```

- Setup the replication

```
[hxeadm]# hdbnsutil -sr_enable --name=DC1
checking for active nameserver ...
nameserver is active, proceeding ...
successfully enabled system as system replication source site
done.
```

Setting up HANA System Replication

Done by ansible role saphana-hsr

Configuring HANA for System Replication - Prepare Node 01 (Master)

- Check the replication configuration

```
[hxeadm]# hdbnsutil -sr_state
checking for active or inactive nameserver ...
System Replication State
~~~~~~~~~~~~~~~~~~~~~~~~
mode: primary
site id: 1
site name: DC1
Host Mappings:
~~~~~~~~~~~~~~
```

Setting up HANA System Replication

Done by ansible role saphana-hsr

Configuring HANA for System Replication - Prepare Node 02 (Secondary)

- Register the secondary site:

```
[hxeadm]# HDB stop
[hxeadm]# hdbnsutil -sr_register --remoteHost=node1 --remoteInstance=90
--replicationMode=syncmem --name=DC2
adding site ...
checking for inactive nameserver ...
nameserver node2:30001 not responding.
collecting information ...
updating local

 ini files ...
done.
[hxeadm]# HDB start
```

Setting up HANA System Replication

Configuring HANA for System Replication - Prepare Node 02 (Secondary)

Done by ansible role saphana-hsr

- Check the replication configuration

```
[hxeadm]#  hdbnsutil -sr_state
checking for active or inactive nameserver ...
[...]


Host Mappings:
~~~~~~~~~~~~~~

hpi-vmYY -> [DC1] hpi-vmXX
hpi-vmYY -> [DC2] hpi-vmYY

primary masters:hpi-vmXX

done.
```

Setting up HANA System Replication

Check Replication status (run on primary Node)

Done by ansible role saphana-hsr

```
[hxeadm]# python /usr/sap/HXE/HDB90/exe/python_support/systemReplicationStatus.py

| Host         | Port  | Service Name | Volume ID | Site ID | Site Name | Secondary    | Secondary | Secondary | Secondary | Secondary     | Replication | Replication | Replication    |
|              |       |              |           |         |           | Host         | Port      | Site ID   | Site Name | Active Status | Mode        | Status      | Status Details |
| ------------ | ----- | ------------ | --------- | ------- | --------- | ------------ | --------- | --------- | --------- | ------------- | ----------- | ----------- | -------------- |
| mkoch-hana02 | 39007 | xsengine     |         2 |       2 | DC2       | mkoch-hana90 |     39007 |         1 | DC1       | YES           | SYNCMEM     | ACTIVE      |                |
| mkoch-hana02 | 39090 | nameserver   |         1 |       2 | DC2       | mkoch-hana90 |     39090 |         1 | DC1       | YES           | SYNCMEM     | ACTIVE      |                |
| mkoch-hana02 | 39003 | indexserver  |         3 |       2 | DC2       | mkoch-hana90 |     30103 |         1 | DC1       | YES           | SYNCMEM     | ACTIVE      |                |

status system replication site "1": ACTIVE
overall system replication status: ACTIVE

Local System Replication State
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

mode: PRIMARY
site id: 2
site name: DC2
```

Setting up HANA System Replication

Final Notes

Replication will be active now, but note that **after every takeover you will have two primary systems**. This makes sense since system replication can also be used to duplicate a system. If you do not want this, you will have to declare one system to be the secondary ("register" or "re-setup replication") after a takeover.

When replication is active the command *hdbnsutil -sr_takeover* must be used to switch the role of the secondary node to primary. Using *hdbsql* -sr_enable after the System Replication has already been established will cause a complete reinitialization of the replication which can potentially cause data loss.

Note: when replication is active you will (as of SAP HANA SPS08) not be able to read from, write to or connect to the secondary system using tools like *hdbsql or HANA Studio*.

Testing HANA System Replication

Perform Manual Takeover
(and perform the same steps vice versa)

- On Node 2

```
[hxeadm]#  hdbnsutil -sr_takeover
```

- On Node 1

```
[hxeadm]# HDB stop
[hxeadm]# hdbnsutil -sr_register --remoteHost=hpi-vmYY --remoteInstance=90
--replicationMode=syncmem --name=DC1
[hxeadm]# HDB start
```

- Check replication state with python scripts on both nodes

Pacemaker Cluster Installation

Setup Pacemaker Cluster

Cluster Installation

Bonus Lab:
Do an ansible role or playbook for the next steps

- Check your repositories

```
# yum repolist
Geladene Plugins: package_upload, product-id, search-disabled-repos, subscription-manager
Repo-ID                                          Repo-Name:                                                                    Status
!rhel-7-server-eus-rpms/x86_64                   Red Hat Enterprise Linux 7 Server - Extended Update Support (RPMs)            11.405
!rhel-7-server-eus-satellite-tools-6.2-rpms/x86_64 Red Hat Satellite Tools 6.2 (for RHEL 7 Server - EUS) (RPMs)                69
!rhel-ha-for-rhel-7-server-eus-rpms/x86_64       Red Hat Enterprise Linux High Availability (for RHEL 7 Server) - Extended Update Support (RP   225
!rhel-sap-hana-for-rhel-7-server-eus-rpms/x86_64 RHEL for SAP HANA (for RHEL 7 Server) Extended Update Support (RPMs)          14
```

- Install Pacemaker on both cluster nodes

```
node01# yum install pcs fence-agents-all resource-agents-sap-hana
node02# yum install pcs fence-agents-all resource-agents-sap-hana
```

Pacemaker Cluster
Hana Preparation

Setup Pacemaker Cluster

Create monitoring account in SAP HANA

Only required for HANA 1. The agent uses the API in HANA 2

- Create user rhelhasync in HANA on the **primary node**

```
[hxeadm]# hdbsql -i 90 -u system -p System123 "create user rhelhasync password P4ssw0rd"
[hxeadm]# hdbsql -i 90 -u system -p System123 "grant CATALOG READ to rhelhasync"
[hxeadm]# hdbsql -i 90 -u system -p System123 "grant MONITOR ADMIN to rhelhasync"
[hxeadm]# hdbsql -i 90 -u system -p System123 "ALTER USER rhelhasync DISABLE PASSWORD
LIFETIME"
```

- Verify that the user has been created correctly

```
[hxeadm]# hdbsql -i 90 -u system -p System123 "select * from sys.users where
USER_NAME=’RHELHASYNC’"
```

Setup Pacemaker Cluster

Create monitoring account in SAP HANA

Only required for HANA 1. The agent uses the API in HANA 2

- As root add a userkey “SAPHANA<SID>SR” for this user to the local HANA userstore for the root user on **each node**

```
# /usr/sap/HXE/HDB90/exe/hdbuserstore SET SAPHANAHMCSR localhost:39015 rhelhasync P4ssw0rd
```

- Verify that the userkey has been created correctly. Run `hdbuserstore list` on **each node**

```
# /usr/sap/HXE/HDB90/exe/hdbuserstore list
DATA FILE       : /root/.hdb/mkoch-hana/SSFS_HDB.DAT
KEY FILE        : /root/.hdb/mkoch-hana01/SSFS_HDB.KEY

KEY SAPHANAHXESR
  ENV : localhost:39015
  USER: rhelhasync
```

Pacemaker Cluster
Basic Cluster Config

Setup Pacemaker Cluster

Initialize the cluster

- Make sure both cluster nodes are configured in /etc/hosts

```
# All entrys need to have the following syntax:
<ip>   <FQDN> <short hostname>
```

- Run the following command **on each cluster node** to configure the cluster infrastructure and start the cluster

```
# echo password | passwd hacluster --stdin
# systemctl enable pcsd
# systemctl start pcsd
```

Setup Pacemaker Cluster

Initialize the cluster

- Now authenticate cluster communication (on one node)

```
# pcs cluster auth hpi-vmXX hpi-vmYY
Username: hacluster
Password: password
hpi-vmXX: Authorized
hpi-vmYY: Authorized
```

- Run the following command **on each cluster node** to configure the cluster infrastructure and start the cluster

```
# systemctl start pacemaker
# systemctl enable pacemaker
```

Setup Pacemaker Cluster

Initialize the cluster

Now the cluster is configured, so that the following must be run one only one host

- Set some basic cluster parameters

```
# pcs resource defaults default-resource-stickness=1000
# pcs resource defaults default-migration-threshold=5000
# pcs resource op defaults timeout=600s
```

Pacemaker Cluster Fencing

Detailed RHV Fencing is described here: https://access.redhat.com/articles/3335601

Setup Pacemaker Cluster

Configure fencing (STONITH)

- List the nodes available on rhev for fencing:

```
# fence_rhevm -a rhvm.epc.ext.hpe.com -l fence@internal -p FenceRH -z -o list \
   --shell-timeout=60 --ssl-insecure
[...]
hpi-vmXX,
hpi-vm1XX,
[...]
```

- Find your nodes and remember the names (refer to as vm-name)
- You can check the status of each node with

```
# fence_rhevm -a rhvm.epc.ext.hpe.com -l fence@internal -p FenceRH -z -o status -n <vm-name>\
    --shell-timeout=60 --ssl-insecure
Status: On
```

Setup Pacemaker Cluster

Configure fencing (STONITH)

- Create fencing for your two devices:

```
# pcs property set default-action-timeout=120s
# pcs stonith create rhevfence01 fence_rhevm \
  pcmk_host_map="hpi-vmXX:hpi-vmXX" ipaddr=rhvm.epc.ext.hpe.com ssl=1 \
  ssl_insecure=1 login=fence@internal passwd=FenceRH shell_timeout=60

# pcs stonith create rhevfence02 fence_rhevm delay=30 \
  pcmk_host_map="hpi-vmYY:hpi-vmYY" ipaddr=rhvm.epc.ext.hpe.com ssl=1 \
  ssl_insecure=1 login=fence@internal passwd=FenceRH shell_timeout=60
```

- Check fencing configuration

```
# pcs stonith show --full
```

Setup Pacemaker Cluster

Configure fencing (STONITH)

Before you continue with setup, make sure your fencing is working. It's a good idea to shut down the HANA database before testing fencing.

```
# pcs stonith fence <hostname>
```

The fenced host should reboot and rejoin the cluster. Check with

```
# pcs status
[...]
Online: [ hpi-vmXX hpi-vmYY ]

Full list of resources:

 rhevfence02    (stonith:fence_rhevm):    Started hpi-vmXX
 rhevfence01    (stonith:fence_rhevm):    Started hpi-vmYY
[...]
```

Pacemaker Cluster Resource Configuration

Setup Pacemaker Cluster

Create resource for virtual IP

- Start your HANA databases and check synchronisation status
- Get your ip address for the Virtual IP [hpi-vhana50-hpi-vhana80]
- It's 192.168.25.50+[1-20] (DNS: hpi-vhana50 to hpi-vhana80)
- Create the resource:

```
# pcs resource create rsc_ip_SAPHana_HXE_HDB90 IPaddr2 ip=192.168.25.XXX
```

Setup Pacemaker Cluster

Create the SAPHanaTopology resource

- Inspect the SAP Hana cluster agent with

```
# pcs resource describe SAPHana
```

- Create SAPHanaTopology resource

```
# pcs resource create rsc_SAPHanaTopology_HXE_HDB90 SAPHanaTopology \
   SID=HXE InstanceNumber=01 \
   op start timeout=600 \
   op stop timeout=300  \
   op monitor interval=10 timeout=600
```

Setup Pacemaker Cluster

Create resources

- Create clone resource for SAPHana Topology

```
# pcs resource clone rsc_SAPHanaTopology_HXE_HDB90 meta is-managed=true  clone-max=2 \
   clone-node-max=1 interleave=true
```

- Create SAPHana resource

```
# pcs resource create rsc_SAPHana_HXE_HDB90 SAPHana \
       SID=HXE InstanceNumber=01 \
       PREFER_SITE_TAKEOVER=true \
       DUPLICATE_PRIMARY_TIMEOUT=7200 \
       AUTOMATED_REGISTER=false \
       op start timeout=3600 \
       op stop timeout=3600 \
       op promote timeout=3600 \
       op demote timeout=3600 \
       op monitor interval=59 role="Master" timeout=700 \
       op monitor interval=61 role="Slave" timeout=700
```

Setup Pacemaker Cluster

Create resources

- Create Master/Slave for SAP Hana

```
# pcs resource master msl_rsc_SAPHana_HXE_HDB90 rsc_SAPHana_HXE_HDB01 \
      meta is-managed=true notify=true clone-max=2 clone-node-max=1 interleave=true
```

- Setup Constraints

```
# pcs constraint colocation \
      add rsc_ip_SAPHana_HXE_HDB90 with master msl_rsc_SAPHana_HXE_HDB90 2000
# pcs constraint order \
      rsc_SAPHanaTopology_HXE_HDB90-clone then msl_rsc_SAPHana_HXE_HDB01 symmetrical=false
```

Setup Pacemaker Cluster

Cluster in good state

```
# pcs status
Cluster name: hanasr
Last updated: Sat Nov 12 02:15:01 2016       Last change: Sat Nov 12 02:14:08 2016 by root via crm_attribute on mkoch-hana01
Stack: corosync
Current DC: mkoch-hana02 (version 1.1.13-10.el7_2.4-44eb2dd) - partition with quorum
2 nodes and 7 resources configured

Online: [ mkoch-hana01 mkoch-hana02 ]

Full list of resources:

 rhevfence02    (stonith:fence_rhevm):    Started mkoch-hana02
 rhevfence01    (stonith:fence_rhevm):    Started mkoch-hana01
 rsc_ip_SAPHana_HXE_HDB01    (ocf::heartbeat:IPaddr2):    Started mkoch-hana01
 Clone Set: rsc_SAPHanaTopology_HXE_HDB90-clone [rsc_SAPHanaTopology_HXE_HDB90]
           Started: [ mkoch-hana90 mkoch-hana02 ]
 Master/Slave Set: msl_rsc_SAPHana_HXE_HDB90 [rsc_SAPHana_HXE_HDB90]
           Masters: [ mkoch-hana01 ]
           Slaves: [ mkoch-hana02 ]

PCSD Status:
  mkoch-hana01: Online
  mkoch-hana02: Online

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

Setup Pacemaker Cluster

Move SAP HANA SR primary to secondary (perform takeover) using cluster commands

- make sure that the cluster is in a clean state, and that there are no location constrains that have been created by previous manual takeovers using the cluster

```
# pcs resource cleanup <name_of_resource>
# pcs resource clear <name_of_resource>
```

- If the cluster is in a clean state (pcs status)

```
# pcs resource move msl_rsc_SAPHana_HXE_HDB90 [destination host]
```

- After the failover you have to clear the Master/Slave resource

```
# pcs resource clear msl_rsc_SAPHana_HXE_HDB90
```

Pacemaker Cluster Useful Commands

Setup Pacemaker Cluster

Useful commands

```
# pcs status                                                    # Show cluster status
# pcs config                                                    # Show cluster config
# pcs cluster [start|stop] --all                                # Start/stop cluster
# pcs resource [enable|disable] <resourcename>                  # Start/stop resource
# pcs resource failcount show <resourcename>                    # View failcount
# pcs resource failcount reset <resourcename>                   # Delete failcount
# pcs resource cleanup <resourcename>                           # Cleanup resource status
# pcs resource show <resourcename>                              # List resource attributes
# pcs resource update <resourcename> SAPHanaFilter="all"  # Change resource parameter
# pcs resource [manage|unmanage] <resourcename>                 # Switch resource to (un)managed mode
```

Comprehensive Command Overview: https://access.redhat.com/articles/1351733

Useful Documentation

- SAP Note 2235581 - SAP HANA: Supported Operating Systems
- SAP Note 2063657 - HANA System Replication takeover decision guideline
- SAP Note 2303243 - SAP HANA Multitier System Replication – supported replication modes between sites
- SAP Note 2315257 - Parallel execution of "hdbnsutil -sr_state" on NFS3 can cause data corruption
- SAP Note 2340501 - Prohibit execution of hdbnsutil, systemReplicationStatus.py and landscapeHostConfiguration.py as root user
- SAP HANA Administration Guide - High Availability for SAP HANA
- How to Perform System Replication for SAP HANA
- Support Policies for RHEL High Availability Clusters - Management of SAP HANA in a Cluster
- SAP Netweaver in pacemaker cluster
- SAP HANA system replication in pacemaker cluster
- HANA System Replication - Take-over process
- Red Hat Enterprise Linux Cluster, High Availability Knowledge Base Index
- What is the proper way to simulate a network failure on a RHEL Cluster?