ABSTRACT

This white paper describes the high availability features of SUSE Linux® Enterprise Server for SAP Applications when used with SAP HANA® System Replication and Lenovo System x servers.

Included are the best practices and configuration guidelines for SUSE Linux Enterprise Server for SAP Applications and Lenovo System x® servers which ensure the highest availability of SAP HANA database instances in a scale up OLAP/OLTP environment.

August 30, 2016

Table of contents

ABSTRACT ................................................................................................................................. 1
Table of contents .......................................................................................................................... 1
Executive Summary .................................................................................................................... 2
Introduction ................................................................................................................................ 2
  Purpose ..................................................................................................................................... 3
  Scope ....................................................................................................................................... 3
  Audience ................................................................................................................................. 3
Lenovo System x Server and SLES for SAP Applications ............................................................ 3
  What is this about? .................................................................................................................... 3
  Performance Optimized vs. Cost Optimized ........................................................................... 4
  What are the current requirements? ......................................................................................... 4
Preparations for HAE Installation and Configuration ................................................................. 5
  Network Configuration .......................................................................................................... 5
Network Description ................................................................................................................ 5
IP Addresses ............................................................................................................................. 5
Host Name ............................................................................................................................... 5
Net Mask ................................................................................................................................... 5
  Create SAP HANA DB User .................................................................................................... 6
  Backup the primary DB .......................................................................................................... 6
  Configuring SAP HANA System Replication ...................................................................... 6
  User access to SAP HANA system replication synchronization information .................... 7
Network configuration for system replication ......................................................................... 7
  SAP HANA parameter changes for system replication communication .............................. 8
Executive Summary

A majority of the SAP application install base is moving towards SAP HANA as a preferred underlying database for all the mission critical workloads ranging from OLAP\(^1\) (BW, data mart, etc) to OLTP\(^2\) (ERP, CRM, SRM, SCM, etc). This technology shift brings-in its own challenges like data availability, disaster recovery which should meet the ever evolving customer requirements.

With the highest level data protection requirements which needs near zero RTO and zero RPO, It always becomes critical to have a stable, integrated, well tested HA/DR solution which will ensure continuous serviceability, availability and stability to the IT infrastructure and the critical customer data.

Introduction

---

\(^1\) On-Line Analytical Processing
\(^2\) On-Line Transactional Processing
Purpose
To meet customers’ needs for SAP HANA high availability (HA), Lenovo offers two distinct delivery alternatives: IBM GPFS® HA with a quorum solution or SUSE Linux Enterprise Server (SLES) for SAP Applications, which includes the SUSE High Availability Extension (HAE) and SAP HANA system replication. IBM GPFS HA is part of the fully integrated, highly optimized Lenovo System x solution that offers the performance and robustness needed to handle the challenges of a combined transactional and analytical platform. The solution is fully certified by both Lenovo and SAP.

However, when SAP HANA system replication is used as the data replication mechanism for HA instead of IBM GPFS, Lenovo and SUSE provide a fully automated high availability solution that can be used for the Lenovo System solutions for SAP HANA servers using IBM GPFS or any other Lenovo System x server that has been approved by SAP on their SAP HANA hardware directory for use with the SAP HANA application. SLES for SAP Applications for SAP HANA is fully authorized by SAP SE to give you the flexibility to specify key components within your SAP HANA infrastructure.

Scope
This white paper serves describes the following:

- HAE installation and configuration details on Lenovo System x servers.
- Different types of solution offerings from SLES for SAP Applications for SAP HANA.
- Tested scenario with the Lenovo Systems solution for SAP HANA and SLES for SAP Applications.
- Key benefits of using SLES for SAP Applications with SAP HANA system replication.

Audience
This whitepaper is intended for Lenovo system integrators, systems administrators, partners, and customers, members of Lenovo lab services and professional services who will be involved in SAP HANA implementation on Lenovo System x servers.

Lenovo System x Server and SLES for SAP Applications
What is this about?
The basic idea behind SLES for SAP Applications using HAE for SAP HANA is that synchronizing the data to the second SAP HANA instance is not enough, as this only solves the problem of having the data shipped to a second instance. To increase the availability you need a cluster solution, which controls the takeover (automatically) of the second instance as well as providing the service address for the client access to the database.

The bandwidth required between the two sites depends by the change rates of the database. Therefore, you should start with a 10-Gbps Layer 2 connection for one scale-up system. If you configure a scale-out system, the bandwidth can be increased to the number of nodes x 10 Gbps.

If the change rate is not high, less bandwidth may be required. In this case, even a switched (Layer 3) connection is possible.

---

3 In this paper we discuss SAP HANA scale up architecture and high-availability feature of SLES for SAP Applications.
SAP HANA allows the installation of a second development or test SAP HANA DB instance within one physical server. When using a SAP HANA DB instance in system replication mode with a second non-productive SAP HANA DB instance on the secondary (fail over) node, this is defined as a cost optimized scenario for SAP HANA replication. When only a single SAP HANA DB instance is installed in system replication mode across the primary and secondary site, this is defined as a performance optimized scenario for SAP HANA system replication.

While this paper describes the performance optimized scenario, the cost optimized scenario is based upon this scenario with a few extra steps for the second SAP HANA DB instance. These steps are documented quite well on the SAP Community Network (SCN) in two parts:

- Part 1 ([http://scn.sap.com/docs/DOC-65899](http://scn.sap.com/docs/DOC-65899)) describes the concept, the setup of the SUSE Linux Enterprise Server for SAP Applications and the set-up and installation of SAP HANA
- Part 2 ([http://scn.sap.com/docs/DOC-68633](http://scn.sap.com/docs/DOC-68633)) describes the cluster configuration and SAP HANA integration into the cluster

### Performance Optimized vs. Cost Optimized

What are the current requirements?

- The current SLES for SAP Applications using SAP HANA system replication is limited to scale up configurations only
- The cluster must include a valid STONITH method
- Both nodes are in the same L2 network layer
- Name resolution between the cluster nodes and VIP must be available
- NTP configured between the cluster nodes
- SAP HANA DB is installed on both the nodes (Tip: SID and instance number in the both the systems must be the same)
- SAP HANA system replication enabled in the cluster nodes
  - From an ISL bandwidth requirement this is similar to sync and async system replication requirement
Ensure that the AUTOMATE_REGISTER feature of SLE HAE is set to ‘false’. The value of “true”, is needed only when you would like to register the failed node after a failover back in the replication. This is not recommended by SUSE to be set.

- Automated start of SAP HANA instances during system boot must be switched off (This can be done, when you install SAP HANA)
- This paper describes features provided by SAPHanaSR version 0.149. Newer versions may change some of these commands.
- SAP host agent is mandatory
- Running QA/DEV on standby node is supported by SUSE but, it is not the scope of this release of the Lenovo whitepaper
- Only one system replication is performed for the SAP HANA DB

Preparations for HAE Installation and Configuration

Network Configuration

Before you start the HAE cluster configuration you will need to gather the following networking details. For our paper, we have used the following assumptions.

Table 1: Network configuration requirements.

<table>
<thead>
<tr>
<th>Network Description</th>
<th>IP Addresses</th>
<th>Host Name</th>
<th>Net Mask</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAP HANA System Replication Node 1</td>
<td>192.168.2.1</td>
<td>server01-rep</td>
<td>255.255.255.0</td>
</tr>
<tr>
<td>SAP HANA System Replication Node 2</td>
<td>192.168.2.2</td>
<td>server02-rep</td>
<td>255.255.255.0</td>
</tr>
<tr>
<td>Productive Node 1</td>
<td>10.5.1.1</td>
<td>server01</td>
<td>255.255.255.0</td>
</tr>
<tr>
<td>Productive Node 2</td>
<td>10.5.1.2</td>
<td>server02</td>
<td>255.255.255.0</td>
</tr>
<tr>
<td>System Management 1</td>
<td>10.10.10.1</td>
<td>server01-imm</td>
<td>255.255.255.0</td>
</tr>
<tr>
<td>System Management 2</td>
<td>10.10.10.2</td>
<td>server01-imm</td>
<td>255.255.255.0</td>
</tr>
<tr>
<td>Multicast Address</td>
<td>217.15.14.1</td>
<td>n/a</td>
<td>n/a</td>
</tr>
</tbody>
</table>

In addition to the table above you should record the details below:

- **Virtual IP Address:** 10.5.1.3
- **User Key:** slehaloc
- **HAWK Port:** 7630

*Note:* If you are using a Lenovo System solution for SAP HANA server, do not alter the SAP HANA and IBM GPFS bonded IP addresses. These should remain as installed by your Lenovo Professional Services representative.

It is assumed in the paper that an SAP HANA database instance has been successfully installed and is up and running.
Create SAP HANA DB User

Create a new user and assign the system privilege “DATA_ADMIN” to the new user. We use here the user: slehasync. If system privilege “DATA_ADMIN” is too powerful for your security policies, you need to check for more granular rights in the SAP documentation.

Run the commands below to create the user key on both the cluster nodes as a root user.

```
server01:~ # PATH="/usr/sap/SLE/HDB00/exe"
server01:~ # hdbsql -u system -i 00 'CREATE USER slehasync PASSWORD Linuxlab'
server01:~ # hdbsql -u system -i 00 'GRANT DATA ADMIN TO slehasync'
server01:~ # hdbsql -u system -i 00 'ALTER USER slehasync DISABLE PASSWORD LIFETIME'
```

Note: While the database user needs to be created only on one side (the other will get the user by the system replication), the user keys for password free access needs to be created on both nodes as the user keys are stored in the file system and not in the database.

The name of the user key “slehaloc” is a fix name used in the resource agent. The port should be setup as 3nn15, where nn is the SAP instance number like “00”.

```
server01:~ # PATH="/usr/sap/SLE/HDB00/exe"
server01:~ # hdbuserstore SET slehaloc localhost:30015 slehasync Linuxlab
```

Verify the created setup as Linux user root. The option “-U key” tells hdbsql to use the credentials which are stored in the secure store. The table “dummy” is available for any database user; here each user has its own dummy table. So far the following test only shows that the user key has been defined correctly and we can login to the database.

```
server01:~ # hdbsql -U slehaloc "select * from dummy"
```

If the output is = DUMMY, then all seems good to proceed.

“X”

Backup the primary DB

```
server01:~ # hdbsql -u system -i 00 "BACKUP DATA USING FILE ('backup')"
```

You should use <SID>adm Username/Password here when prompted.

Configuring SAP HANA System Replication

As the Linux user sidadm, enable the system replication at the primary node. You need to define a site name (like “PRIMARY”). This site name must be unique for all SAP HANA databases which are connected via system replication. This requires that the other site have a different name, such as SECONDARY.

Configure SAP HANA system replication on the primary node:

```
server01:~ su - sleadm
server01:/usr/sap/SLE/HDB00--> hdbnsutil -sr_enable --name=PRIMARY
checking local nameserver:
  checking for active nameserver ...
nameserver is running, proceeding ...
  configuring ini files ...
successfully enabled system as primary site ...
done.
```

Check the replication status

```
server01:/usr/sap/SLE/HDB00--> hdbnsutil -sr_state
checking for active or inactive nameserver ...
System Replication State
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
mode: primary
site id: 1
```
Configuration steps on the Secondary Node

The SAP HANA database instance on the secondary site must be stopped before the instance could be registered for the system replication. You should use HDB Stop. After the database instance has stopped successfully, you should register the instance using hdbnsutil.

Run the following command to register the 2nd node in the replication:

```bash
server02:/usr/sap/SLE/HDB00~> sapcontrol -nr 00 -function StopSystem
server02:/usr/sap/SLE/HDB00~> hdbnsutil -sr_register --remoteHost=server01 --remoteInstance=00 --mode=sync --name=SECONDARY
adding site ...
checking for inactive nameserver ...
nameserver SUSE02:30001 not responding.
collecting information ...
updating local ini files ...
done.
```

Verify the status

```bash
server01:/usr/sap/SLE/HDB00> hdbnsutil -sr_state
checking for active or inactive nameserver ...
System Replication State
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
mode: primary
Site id: 1
site name: PRIMARY
Host Mappings:
~~~~~~~~~~~~~~
server01 -> [PRIMARY] server01
server01 -> [SECONDARY] server02
done.
```

User access to SAP HANA system replication synchronization information

Now please check if the database user has enough rights to access the synchronization information.

```bash
server01:~ # PATH="PATH:/usr/sap/SLE/HDB00/exe"
server01:~ # hdbsql -U slehaloc 'select distinct REPLICATION_STATUS from SYS.M_SERVICE_REPLICATION'
```

If you get a result like “error”, “unknown”, “syncing” or “active” than the user is able to access the synchronization info.

Start the Secondary Node

```bash
server02:/usr/sap/SLE/HDB00> sapcontrol -nr 00 -function StartSystem
11.10.2015 13:24:04
StartSystem
OK
```

Verify SAP HANA system replication status

```bash
server01:/usr/sap/SLE/HDB00> hdbsql -U slehaloc 'select distinct REPLICATION_STATUS from SYS.M_SERVICE_REPLICATION'
1 row selected (overall time 6130 usec; server time 6219 usec)
REPLICATION_STATUS
"ACTIVE"
```

Network configuration for system replication

If nothing is configured, SAP HANA uses the access network (IP set declared in table 1) to synchronize the systems. The solution in this document uses a separate network for system replication, so you should configure the network (Figure 2).
For more information, see:

- http://help.sap.com/saphelp_hanaplatform/helpdata/en/32/c22f81e8c14268a4e5de01cd033e8f/content.htm
- http://help.sap.com/saphelp_hanaplatform/helpdata/en/0c/6738ab85c64da1aed0fa91c25ed47c/content.htm

**SAP HANA parameter changes for system replication communication**

**Edit the primary node (server01) SAP HANA custom config settings as follows:**

```
server01:/usr/sap/SLE/HDB00> cd glo
server01:/usr/sap/SLE/SYS/global> cd hdb/custom/config
server01:/usr/sap/SLE/SYS/global/hdb/custom/config> cat global.ini

[persistence]
basepath_datavolumes = /hana/data/SLE
basepath_logvolumes = /hana/log/SLE

[system_replication]
mode = primary
actual_mode = primary
site_id = 1
site_name = PRIMARY
```

**Edit the secondary node (server02) SAP HANA custom config settings as follows:**

```
server02:/usr/sap/SLE/HDB00> cd glo
server02:/usr/sap/SLE/SYS/global> cd hdb/custom/config/
server02:/usr/sap/SLE/SYS/global/hdb/custom/config> cat global.ini

[persistence]
basepath_datavolumes = /hana/data/SLE
basepath_logvolumes = /hana/log/SLE

[system_replication]
```

Figure 2: SLES for SAP Applications network configuration used with system replication
Now we should modify the custom global.ini to use a separate network path for system replication.

**Primary Node (server01) global.ini**

```
server01:/usr/sap/SLE/SYS/global/hdb/custom/config> cat global.ini
[persistence]
basepath_datavolumes = /hana/data/SLE
basepath_logvolumes = /hana/log/SLE

[system_replication]
mode = primary
actual_mode = primary
site_id = 1
site_name = PRIMARY

[communication]
listeninterface=.internal

[internal_hostname_resolution]
192.168.1.1 = server01
192.168.2.1 = server02

/system_replication_hostname_resolution]
10.5.1.1 = server01
10.5.2.1 = server02
```

**Secondary Node (server02) global.ini**

```
server02:/usr/sap/SLE/SYS/global/hdb/custom/config> cat global.ini
[persistence]
basepath_datavolumes = /hana/data/SLE
basepath_logvolumes = /hana/log/SLE

[system_replication]
site_id = 2
mode = sync
actual_mode = sync
site_name = SECONDARY

[system_replication_site_masters]
1 = server01:30001

[communication]
listeninterface=.internal

[internal_hostname_resolution]
192.168.1.1 = server01
192.168.2.1 = server02

/system_replication_hostname_resolution]
10.5.1.1 = server01
10.5.2.1 = server02
```

Then you may restart the SAP HANA services.

**Note:** The `listeninterface=.global/.internal` parameter in the `[communication]` section is required for the communication between SAP HANA services (nameserver, indexserver, and so on) in a distributed system; it is not specific to SAP HANA system replication.
Setting up HAE

**HAE Cluster Software Installation**

The following HAE packages must be installed. These may be installed individually or with the appropriate zypper pattern.

```
server01:~ # zypper install --type pattern ha_sles
```

```
server02:~ # zypper install --type pattern ha_sles
```

Finally, you should install the SAP HANA system replication resource agents provided by SUSE on the SLES for SAP Applications update channel. After registering the product with SUSE you are able to download and install the SAP HANA system replication package directly with zypper.

```
server01:~ # zypper install SAPHanaSR
```

```
server02:~ # zypper install SAPHanaSR
```

Make sure that the each node has the same host and user names along with the same password files. Ensure that the host names, user names and passwords are correct on server01 and copy the appropriate files to server02 (or use the appropriate tools in YaST on each server to copy these values.

```
server02: / # scp server01:/etc/{hosts,group,shadow,passwd} /etc/
```

**HAE Cluster configuration**

**Set up the SAP Host Control Agent**

Before you configure on the cluster, make sure that the SAP Host Control (SAPHostCtrl) agent is installed and working properly on all nodes. You should use the latest available SAP host agent you can download from SAP Marketplace.

**Install the latest SAP host agent on all nodes.**

```
server01:/sapcd # rpm -Uvh saphostagentrpm_203-20005731.rpm
Preparing... ########################################### [100%]
1:SAPHostAgent ########################################### [100%] sapinit 0:off 1:off
2:off 3:on 4:off 5:on 6:off
```

```
server02:/sapcd # rpm -Uvh saphostagentrpm_203-20005731.rpm
```

**Restart the SAP host agent on all nodes.**

```
server01:/usr/sap/hostctrl/exe # service sapinit restart
saphostexec is already running (pid=18222). Stopping...Stopped
start hostcontrol using profile /usr/sap/hostctrl/exe/host_profile
Impromptu CCC initialization by 'rscpCInit'.
See SAP note 1266393.
Impromptu CCC initialization by 'rscpCInit'.
See SAP note 1266393.
```

```
server02:/usr/sap/hostctrl/exe # service sapinit restart
```
Test the host control functions.

```
server01:/usr/sap/hostctrl/exe # ./saphostctrl -function ListInstances
Inst Info : SLE - 00 - server01 - 781, patch 42, changelist 1444691
```

```
server02:/usr/sap/hostctrl/exe # ./saphostctrl -function ListInstances
Inst Info : SLE - 00 - server02 - 781, patch 42, changelist 1444691
```

initialize the HAE cluster

After setting up the SAP host controller, you may begin to initialize and configure HAE. This process is interactive from the command line. All inputs below are marked in **boldface** type.

```
server01:~ # sleha-init
WARNING: NTP is not configured to start at system boot.
Do you want to continue anyway? [y/N] y
Enabling sshd service
/root/.ssh/id_rsa already exists - overwrite? [y/N] N
Configuring csync2
Generating csync2 shared key (this may take a while)...done
Enabling csync2 service
Enabling xinetd service
csync2 checking files
Configure Corosync:
This will configure the cluster messaging layer. You will need to specify a network address over which to communicate (default is mgmt's network, but you can use the network address of any active interface), a multicast address and multicast port.
Network address to bind to (e.g.: 192.168.1.0) **192.168.1.0**
Multicast address (e.g.: 239.x.x.x) **217.15.14.1**
Multicast port [5405] **5405**
Configure SBD:
If you have shared storage, for example a SAN or iSCSI target, you can use it avoid split-brain scenarios by configuring SBD. This requires a 1 MB partition, accessible to all nodes in the cluster. The device path must be persistent and consistent across all nodes in the cluster, so /dev/disk/by-id/* devices are a good choice. Note that all data on the partition you specify here will be destroyed.
Do you wish to use SBD? [y/N] N
WARNING: Not configuring SBD - STONITH will be disabled.
Configuring hawk service
HA Web Konsole is now running, to see cluster status go to: https://192.168.10.17:7630/
Log in with username 'hacluster', password 'linux'
WARNING: You should change the hacluster password to something more secure!
Enabling openais service
Waiting for cluster........done
Loading initial configuration
Done (log saved to /var/log/sleha-bootstrap.log)
```

Note: The configuration discussed in this paper does not use a **shared block device (SBD)** for the STONITH connection. Instead, the Lenovo Integrated Management Console (IMM) interface can be configured to enable a STONITH setup similar to SBD. This will be set up at a later point.

Disable auto cluster start during start up.

```
server01:~ # service openais stop
server01:~ # chkconfig |grep openais
openais off
```

```
server02:~ # service openais stop
server02:~ # chkconfig |grep openais
openais off
```

Configure corosync.conf file for UDP

```
server01:~ # cat /etc/corosync/corosync.conf
compatibility: whitetank
```
Lenovo Solution White Paper  
SUSE HA with SAP HANA

```plaintext
aisexec {
  user: root
  group: root
}
service {
  ver: 0
  name: pacemaker
  use_mgmtd: yes
  use_logd: yes
}
totem {
  version: 2
  token: 5000
  token_retransmits_before_loss_const: 10
  join: 60
  consensus: 6000
  vsftype: none
  max_messages: 20
  clear_node_high_bit: new
  secauth: off
  interface {
    ringnumber: 0
    bindnetaddr: 192.168.1.0
    member {
      memberaddr: 192.168.1.1
    }
    member {
      memberaddr: 192.168.2.1
    }
    Mcastaddr: 217.15.14.1
    Mcastport: 5405
    Ttl: 1
  }
  transport: udpu
  threads: 4
}
logging {
  fileline: off
  to_stderr: no
  to_logfile: no
  to_syslog: yes
  syslog_facility: daemon
  debug: off
  timestamp: off
  logger_subsys {
    subsys: AMF
    debug: off
  }
}

Copy the corosync.conf file to the secondary node.

```
server01: # scp /etc/corosync/corosync.conf server02:/etc/corosync/
```

Adding the second node to the cluster

```
service02: # sleha-join
WARNING: NTP is not configured to start at system boot.
Do you want to continue anyway? [y/N] y

Join This Node to Cluster:
You will be asked for the IP address of an existing node, from which configuration will be copied. If you have not already configured passwordless ssh between nodes, you will be prompted for the root password of the existing node.

IP address or hostname of existing node (e.g.: 192.168.1.1) [] server01
Enabling sshd service
/root/.ssh/id_rsa already exists - overwrite? [y/N] N
Configuring csync2
Enabling csync2 service
Enabling xinetd service
Merging known_hosts
Probing for new partitions......done
Enabling hawk service
```
HA Web Konsole is now running, to see cluster status go to: https://192.168.10.17:7630/
Log in with username 'hacluster', password 'linux'
WARNING: You should change the hacluster password to something more secure!
Enabling opensais service
Waiting for cluster... done
Done (log saved to /var/log/sleha-bootstrap.log)

Restart cluster service on both the nodes.

server01:~ # service opensais stop
Stopping OpenAIS/corosync daemon (corosync): .done OK

server02:~ # service opensais stop
Stopping OpenAIS/corosync daemon (corosync): .done OK

server01:~ # service opensais start
Starting OpenAIS/Corosync daemon (corosync): starting... OK

server02:~ # service opensais start
Starting OpenAIS/Corosync daemon (corosync): starting... OK

Run resource manager to verify the cluster status.

server01:~ #crm_mon -1 -r
Last change: Sun Oct  1 10:04:10 2015 by root via crm_attribute on server01
Stack: classic opensais (with plugin)
Current DC: server02 - partition with quorum
Version: 1.1.11-3ca8c3b
2 Nodes configured, 2 expected votes
0 Resources configured
Online: [ server01 server02 ]
Full list of resources:

Configure the STONITH using IMM.

server01:~ #crm configure
primitive STONITH-Server01 stonith:external/ipmi op monitor interval="300s" timeout="60s" on-fail="restart" op start interval="0" timeout="60s" on-fail="restart" params hostname="server01" ipaddr="<IMM IP>" userid="USERID" passwd="PASSW0RD" interface="lanplus"
primitive STONITH-Server02 stonith:external/ipmi op monitor interval="300s" timeout="60s" on-fail="restart" op start interval="0" timeout="60s" on-fail="restart" params hostname="server02" ipaddr="<IMM IP>" userid="USERID" passwd="PASSW0RD" interface="lanplus"

Make the IPMI service (IMM) runs on both the cluster nodes

server01:~ # location LOC_STONITH_Server01 STONITH-Server01 -inf: server01
server02:~ # location LOC_STONITH_Server02 STONITH-Server02 -inf: server02

server01:~ #crm configure property no-quorum-policy="ignore"
server01:~ #crm configure property stonith-action="reboot"
server01:~ #crm configure property startup-fencing="false"
server01:~ #crm configure property stonith-timeout="30s"

Restart the cluster using service opensais stop and service opensais start commands on both the cluster nodes.

Run crm_mon to check the latest status of the cluster.

server01:~ #crm_mon -l -r
Configure base “bootstrap” settings in the cluster, crate the file and upload the configuration.

```
server01:~ # cat crm-base.txt
property $id="cib-bootstrap-options" \
    no-quorum-policy="ignore" \ 
    stonith-enabled="true" \ 
    stonith-action="reboot" \ 
    stonith-timeout="150s" \ 
    rsc_defaults $id="rsc-options" \ 
        resource-stickiness="1000" \ 
        migration-threshold="5000" \ 
        op_defaults $id="op-options" \ 
            timeout="600"
```

```
server01:~ # cat saphanatop.txt
primitive rsc_SAPHanaTopology_SLE_HDB00 ocf:suse:SAPHanaTopology \ 
    operations $id="rsc_sap2_SLE_HDB00-operations" \ 
    op start interval="0" timeout="600" \ 
    op stop interval="0" timeout="300" \ 
    params SID="SLE" InstanceNumber="00" 
    clone cln_SAPHanaTopology_SLE_HDB00 rsc_SAPHanaTopology_SLE_HDB00 \ 
    meta is-managed="true" clone-node-max="1" target-role="Started" \ 
    interleave="true"
```

```
server01:~ # cat saphana.txt
primitive rsc_SAPHana_SLE_HDB00 ocf:suse:SAPHana \ 
    operations $id="rsc_sap_SLE_HDB00-operations" \ 
    op start interval="0" timeout="3600" \ 
    op stop interval="0" timeout="3600" \ 
    op promote interval="0" timeout="3600" \ 
    op monitor interval="60" role="Master" timeout="1400" \ 
    op monitor interval="61" role="Slave" timeout="1400" \ 
    params SID="SLE" InstanceNumber="00" PREFER_SITE_TAKEOVER="true" \ 
    DUPLICATE_PRIMARY_TIMEOUT="7200" AUTOMATED_REGISTER="false" 
    ms msl_SAPHana_SLE_HDB00 rsc_SAPHana_SLE_HDB00 \ 
    meta is-managed="true" notify="true" clone-max="2" clone-node-max="1" \ 
    target-role="Started" interleave="true"
```

```
server01:~ # cat saphanatop.txt
```

```
server01:~ # crm configure load update crm-base.txt
```

```
server01:~ # crm configure load update saphanatop.txt
```

```
server01:~ # cat saphana.txt
```

```
server01:~ # crm configure load update saphana.txt
```
Virtual IP for application access.

```
server01:~ # cat crm-ip.txt
primitive rsc_ip_SLE_HDB00 ocf:heartbeat:IPaddr2 
meta target-role="Started" is-managed="true" 
operations $id="rsc_ip_SLE_HDB00-operations" 
  op monitor interval="10s" timeout="20s" 
  params ip="192.168.10.1"
server01:~ #crm configure load update crm-ip.txt
```

Two constraints are organizing the correct placement of the virtual IP address for the client
database access and the start order between the two resource agents SAPHana and
SAPHanaTopology.

```
server01:~ # cat crm-cs.txt
colocation col_saphana_ip_SLE_HDB00 2000: rsc_ip_SLE_HDB00:Started 
  msl_SAPHana_SLE_HDB00:Master
order ord_SAPHana_SLE_HDB00 2000: cln_SAPHanaTopology_SLE_HDB00 msl_SAPHana_SLE_HDB00
server01:~ #crm configure load update crm-cs.txt
```

Cluster status after resource configuration

```
server01:/ # crm_mon --r -1
Last change: Sun Oct 1 10:04:10 2015 by root via crm_attribute on server01

Stack: classic openais (with plugin)
Current DC: server02 - partition with quorum
Version: 1.1.11-3ca8c3b
2 Nodes configured, 2 expected votes
7 Resources configured
Online: [ server01 server02 ]
Full list of resources:

STONITH-Server01 (stonith:external/ipmi): Started server01
STONITH-Server02 (stonith:external/ipmi): Started server02
Clone Set: cln_SAPHanaTopology_SLE_HDB00 [rsc_SAPHanaTopology_SLE_HDB00]
  Started: [ server01 server02 ]
Master/Slave Set: msl_SAPHana_SLE_HDB00 [rsc_SAPHana_SLE_HDB00]
  Masters: [ server01 ]
  Slaves: [ server02 ]
rsc_ip_SLE_HDB00 (ocf::heartbeat:IPaddr2): Started server02
```
Appendix A

SAP HANA compute node details

- 2 × Lenovo System x3850 X6 servers with Intel® Xeon® E7- 8880 v2 CPU @ 3.20GHz and 512GB RAM
- Emulex OneConnect 10Gb NIC (be3) (rev 02)
- SUSE Linux Enterprise Server for SAP Applications 11 SP3
- IBM GPFS 4.1.0.7

SAP HANA DB and SUSE SR agent used for the tests

- SAP HANA 1.0 SPS08 (Revision 85)
- 2-Node scale-up configuration (1 worker + 1 stand-by)
- SAPHanaSR-0.149-0.8.1.noarch
- SAPHanaSR-doc-0.149-0.8.1.noarch
- Lenovo Integrated Management Module for STONITH technique

Resources

Lenovo documentation:
- SAP HANA System replication
- SAP HANA network configuration
- Network configuration for SAP HANA system replication
- SAP HANA failover explained

SUSE documentation:
- Automate your SAP HANA System Replication Failover using SLE HAE

SAP documentation:
- SAP HANA Administration Guide
- SAP HANA Server Installation Guide
- SAP HANA HA guidelines

Notices

Lenovo may not offer the products, services, or features discussed in this document in all countries. Consult our local Lenovo representative for information on the products and services currently available in your area. Any reference to a Lenovo product, program, or service is not intended to state or imply that only that Lenovo product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any Lenovo intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any other product, program, or service.

Lenovo may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:
Why Lenovo?

The Lenovo server portfolio includes data warehouse appliances that integrate database, server and storage into a single, easy-to-manage appliance that requires minimal setup and ongoing administration, and provides fast and consistent analytic performance. Lenovo also offers pre-built and pre-integrated workload-optimized data warehousing and analytics platforms and data warehouse software for operational intelligence. These offerings are enhanced with additional support for big data and new types of analytics workloads, including continuous and fast analysis of massive volumes of data-in-motion.

For more information

To learn more about the Lenovo System x and ThinkServer product lines, contact your Business Partner or visit lenovo.com/systems/servers