SAP

Best Practices

Best Practices

SAP HANA HA Cross-Zone with SLES HAE

SAP HANA HA Cross-Zone with SLES HAE

- Solution Overview
    - SAP HANA System Replication
    - HAE of SLES
    - Alibaba Cloud Infrastructure
    - Architecture Overview
    - ECS instance
    - VPC
    - Cloud Disk
    - OSS
    - Shared block storage
- Infrastructure Preparation
    - Infrastructure List
    - Creating VPC
    - Creating ECS Instances
    - Creating ENIs and binding to ECS instances
    - Creating NAT Gateway and configure SNAT entry
    - Creating STONITH device and Virtual IP Resource Agent
- Software Preparation
    - Software List
    - SLES for SAP HAE Installation
    - UUIDD Installation
    - SAP HANA Installation
    - SAP HANA Client Installation
    - SAP Host Agent Installation
- Configuring SAP HANA System Replication
- Configuring HAE for SAP HANA
    - Mechanism of SLES for SAP HAE
    - Configuration of Corosync

- Configuration of Pacemaker
- Verify the HA take over

- Reference

Version Control:

Version	Revision Date	Types Of Changes	Author	Approval	Effective Date
1.0			Tony Zhang	Li Bing	2018.3.7

Solution Overview

SAP HANA System Replication

SAP HANA provides a feature called System Replication, which replicates the data and the redo log of every committed transaction from the primary SAP HANA database to a secondary SAP HANA database with the same SID and instance number.

By leveraging the HANA System Replication feature, users are able to build their own SAP HANA High Availability solution. However, SAP HANA System Replication feature itself doesn’ t provide the automatic fail-over function, which means when the primary SAP HANA database fails, users have to manually switch the cluster over to the secondary SAP HANA database.

HAE of SLES

To build up a more user-friendly and robust HA solution, it is quite popular to combine the SUSE Linux HAE with SAP HANA System Replication together.

SUSE High Availability Extension (HAE) is a high availability solution based on Corosync (Messaging Layer, handling heartbeat & cluster membership) and Pacemaker (Cluster Resource Management Layer). With SUSE Linux Enterprise Server (SLES) for SAP, SUSE provides SAP specific Resource Agents (SAPHana, SAPHanaTopology etc.) used by Pacemaker to help users to buildup SAP HANA HA solution more effectively.

Alibaba Cloud Infrastructure

Alibaba Cloud is built on a global infrastructure. Alibaba Could infrastructure services locates in different Regions around the world. Alibaba Could Regions enable you to deploy your SAP applications in a location which is closer to your users, and to meet legal or other requirements. Each region contains multiple isolated locations called Zones. Each Zone is designed to be isolated from failures in other Zones, and provides inexpensive, high speed network connectivity to other Zones within the same Region.

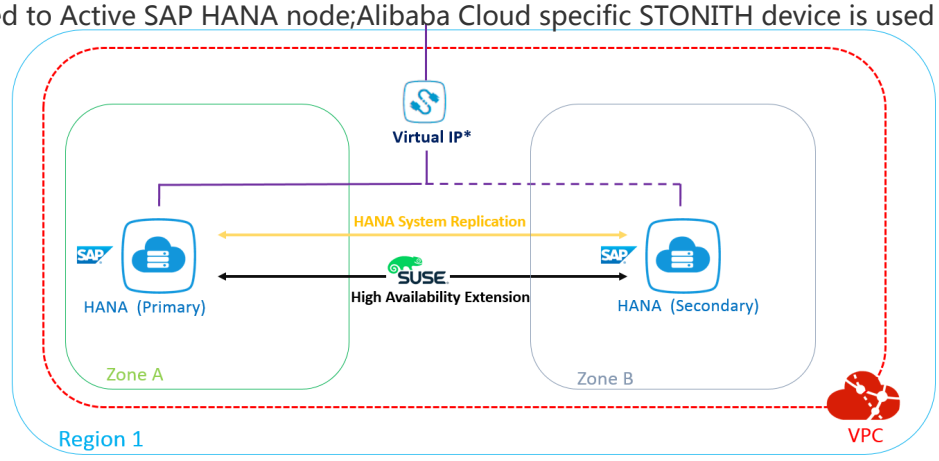
To leverage the cutting edge of Alibaba Cloud Infrastructure, it is more desirable to deploy your SAP High Availability solution cross Zones within the same Region. To facilitate the deployment of cross Zones HANA HA solution, Alibaba Cloud provides an Alibaba Cloud Specific STONITH device (which is calls Alibaba Cloud OpenAPI inside the ECS instance, to shut down a ECS server with specific

instance ID) and Resource Agent (to provide Service IP between 2 ECS instances locates in different Zones).

Architecture Overview

This document guides you on how to deploy a SAP HANA HA solution cross different Zones. Following is a brief architecture:

- HAE of SLES for SAP is used to setup the HA Cluster;
- SAP HANA System Replication is activated between the two HANA nodes;
- Two HANA nodes locates in different Zones of the same Region;
- Alibaba Cloud Specific Virtual IP Resource Agent is used to allow Moving IP automatically switched to Active SAP HANA node;Alibaba Cloud specific STONITH device is used for fencing;



* **Virtual IP** – Alibaba Cloud provides a specific Resource Agent to switch Service IP across different ECS instances locates in different zones;

ECS instance

Elastic Compute Service (ECS) is a type of computing service that features elastic processing capabilities. ECS has a simpler and more efficient management mode than physical servers. You can create instances, change the operating system, and add or release any number of ECS instances at any time to fit your business needs.An ECS instance is a virtual computing environment that includes CPU, memory, and other basic computing components. An instance is the core component of ECS and is the actual operating entity offered by Alibaba Cloud. Other resources, such as disks, images, and snapshots, can only be used in conjunction with an ECS instance.Before create SQL server instances you have to create ECS instances first using the ECS console, about the detail information, please refer to Create ECS instances.

VPC

Virtual Private Cloud (VPC) creates an isolated network environment for you SQL Server environment. You can select an IP address range, divide networks, and configure the routing list and gateway. The interflow of VPC intranet and between VPC and physical IDC machine rooms can be realized among regions or users. About how to create VPC please refer to the detail.

Cloud Disk

Ultra Cloud Disk: When you create ECS instance, Ultra Cloud Disk as the system disk provides a high-performance location for operating system and windows page file.
SSD Cloud Disk: When you create ECS instance we recommend you choose SSD cloud disk store the database files, tempdb,log file separately. Separate SSD cloud disks provide high performance and high reliability.

- High performance: A single SSD cloud disk provides a maximum of 20,000 random reading/writing IOPS and 300 MBps throughput of storage performance.
- IOPS=min{1200+30*disk_size, 20000}. The base is 1200 IOPS, and each GB provides 30 random IOPS up to a maximum of 20,000.
- Throughput=min{80+0.5*disk_size, 300} MBps. The base is 80 MBps, and each GB adds an additional 0.5 MBps up to a maximum of 300 MBps throughput performance.
- Reliability: SSD cloud disks use Alibaba Cloud’ s Apsara distributed storage technology, based on three distributed copies, which can guarantee 99.9999999% data reliability.

For how to create a cloud disk, please refer to create a cloud disk.

OSS

Alibaba Cloud Object Storage Service (OSS) is a network-based data access service. OSS enables you to store and retrieve unstructured data including text files, images, audios, and videos. We recommend you backup your SQL Server database into OSS. For how to use OSS please see Get started with Object Storage Service

Shared block storage

Shared Block Storage is designed for the high availability architecture of enterprise-class applications and provide shared access to block storage devices in a Share-everything architecture, such as the SQL Server always on with WSFC node architecture, which is common among government departments, enterprises, and financial customers, and the high availability server cluster architecture. For about shared block storage detail, please see Shared block storage FAQ

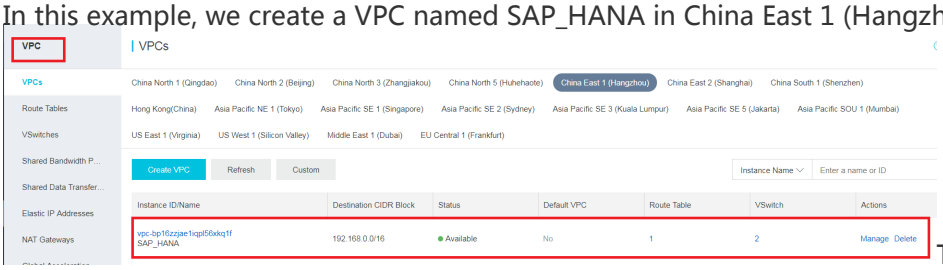
Infrastructure Preparation

Infrastructure List

- 1 VPC network;
- 2 ECS instances in different zones of the same VPC;
- 2 Elastic Network Interface (ENI), one for each ECS instance;
- Alibaba Cloud specific Virtual IP Resource Agent and STONITH device;
- NAT Gateway and SNAT entry;

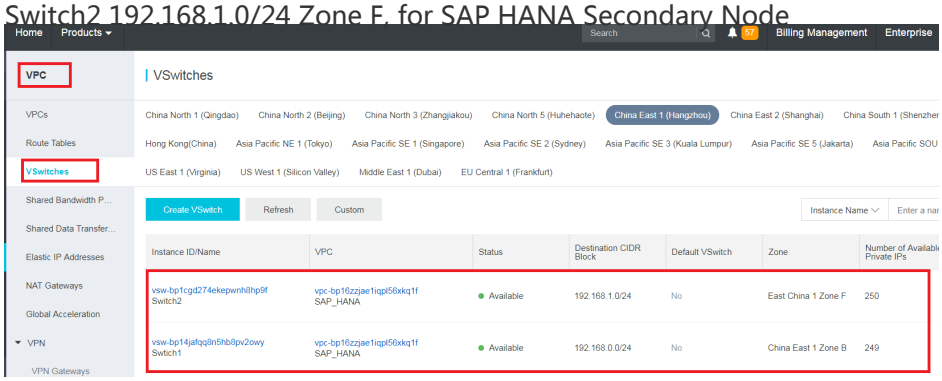
Creating VPC

First of all, a VPC should be created.
In this example, we create a VPC named SAP_HANA in China East 1 (Hangzhou) Region as follow:



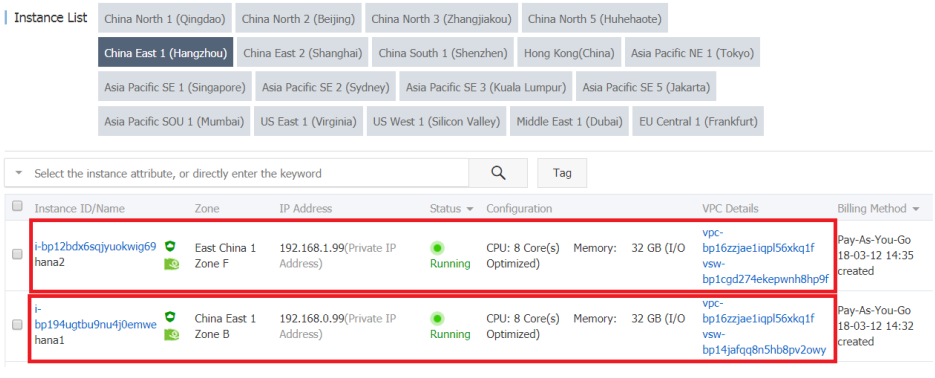
There should be at least 2 VSwitches(subnets) defined within the VPC network, each VSwitch bound to a different Zone.
In this example, we have following 2 VSwitches(subnets):

- Switch1 192.168.0.0/24 Zone B, for SAP HANA Primary Node
- Switch2 192.168.1.0/24 Zone F, for SAP HANA Secondary Node



Creating ECS Instances

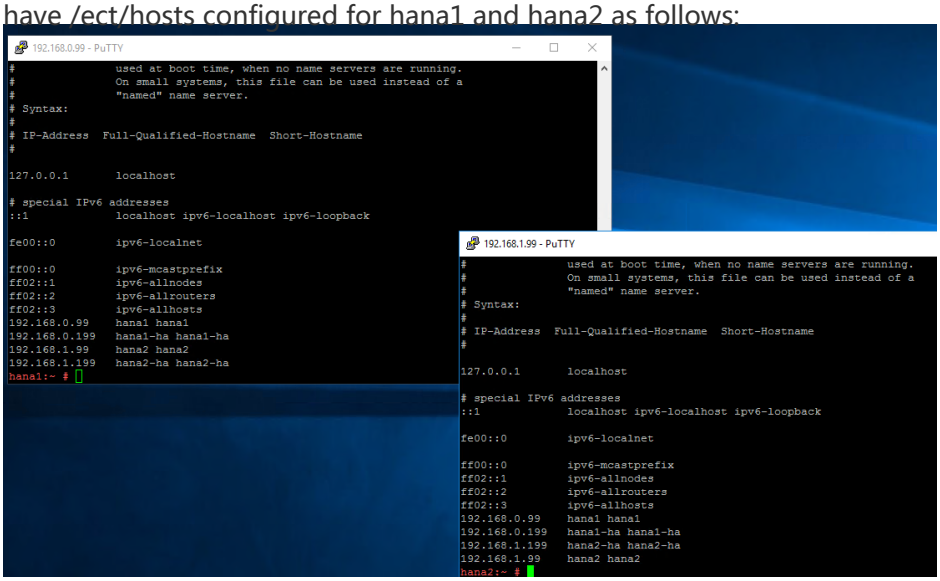
Two ECS instances are created in different Zones of the same VPC. Choose the SLES for SAP image from the Image Market place.
In this example, 2 ECS instances (hostname: hana1 and hana2) are created in China East 1 Region, Zone F and Zone B, within VPC: SAP_HANA, with SLES 12 SP2 for SAP image from the Image Market Place. Host hana1 is the primary SAP HANA database node, and hana2 is the secondary SAP HANA database node.



Creating ENIs and binding to ECS instances

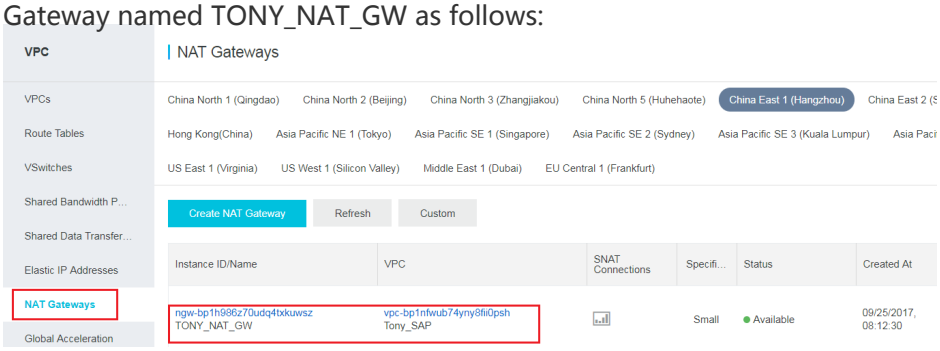
Create two ENIs, and attach one for each ECS instance, for HANA System Replication purpose. Configure the IP addresses of the ENIs to the subnet for HANA System Replication only.
In this example, the ENIs are attached to ECS instances hana1 and hana2, and IP addresses are configured as 192.168.0.199 and 192.168.1.199 within the same VSwitches of hana1 and hana2, and put in the VPC: SAP_HANA

Meanwhile, within the Guest OS, /etc/hosts should also be configured as well.In this example, we have /ect/hosts configured for hana1 and hana2 as follows:



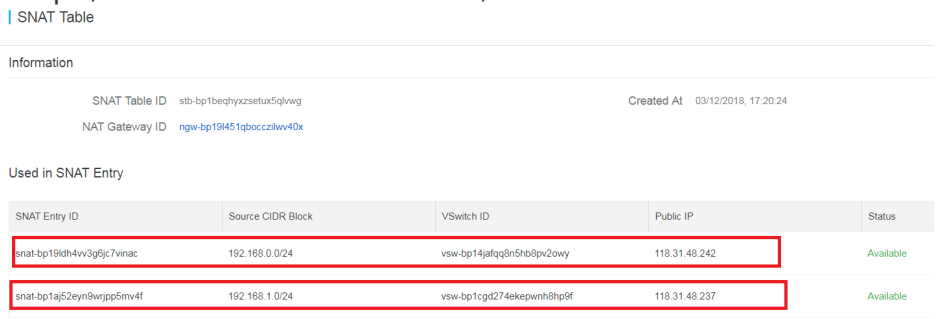
Creating NAT Gateway and configure SNAT entry

First of all, create a NAT Gateway attached to the given VPC; In our example, we create a NAT Gateway named TONY_NAT_GW as follows:



After creating NAT Gateway, you need to create corresponding SNAT entry to allow ECS instances within the VPC can access public address on Internet. (Caution: Alibaba Cloud specific STONITH device and Virtual IP Resource Agent, need to access Alibaba Cloud OpenAPI through a public domain);

In our example, we create to two SNAT entries, for ECS instances locates in different network range as follows:



Creating STONITH device and Virtual IP Resource Agent

Download software from with following command:

wget http://repository-iso.oss-cn-beijing.aliyuncs.com/ha/aliyun-ecs-pacemaker.tar.gz

```
hana1:/hana/tmp # ls
122.05  122.05.tgz  SAPHOSTAGENT  SAP_HANA_CLIENT  aliyun  corosync.conf
hana1:/hana/tmp # wget http://repository-iso.oss-cn-beijing.aliyuncs.com/ha/aliyun-ecs-pacemaker.tar.gz
--2018-03-05 16:12:13--  http://repository-iso.oss-cn-beijing.aliyuncs.com/ha/aliyun-ecs-pacemaker.tar.gz
Resolving repository-iso.oss-cn-beijing.aliyuncs.com (repository-iso.oss-cn-beijing.aliyuncs.com)... 59.110.190.40
Connecting to repository-iso.oss-cn-beijing.aliyuncs.com (repository-iso.oss-cn-beijing.aliyuncs.com)|59.110.190.40|:80
.. connected.
HTTP request sent, awaiting response... 200 OK
Length: 4125 (4.0K) [application/x-gzip]
Saving to: 'aliyun-ecs-pacemaker.tar.gz'

100%[=======================================================================>] 4,125       --.-K/s   in 0s

2018-03-05 16:12:13 (503 MB/s) - 'aliyun-ecs-pacemaker.tar.gz' saved [4125/4125]

hana1:/hana/tmp #
```

Extract the package and install the software

tar –xvf aliyun-ecs-pacemaker.tar.gz

./install

```
hana1:/hana/tmp # ls -l
total 3102992
drwxrwxr-x 8 root root       4096 Mar 30  2017 122.05
-rw-r--r-- 1 root root 3177440026 Dec 24 18:14 122.05.tgz
drwxr-xr-x 2 root root       4096 Feb 28 12:14 SAPHOSTAGENT
drwxrwxrwx 4 root root        131 Feb 28 12:14 SAP_HANA_CLIENT
drwxr-xr-x 3 root root         69 Mar  5 15:31 aliyun
-rw-r--r-- 1 root root       4125 Mar  2 18:24 aliyun-ecs-pacemaker.tar.gz
-rw-r--r-- 1 root root       3957 Mar  1 10:29 corosync.conf
hana1:/hana/tmp # tar -xvf aliyun-ecs-pacemaker.tar.gz
aliyun-ecs-pacemaker/ecs-pacemaker/
aliyun-ecs-pacemaker/ecs-pacemaker/fence_aliyun.py
aliyun-ecs-pacemaker/ecs-pacemaker/vpc-move-ip
aliyun-ecs-pacemaker/install.sh
hana1:/hana/tmp # ./install
```

Install Alibaba Cloud OpenAPI SDK

pip install aliyun-python-sdk-ecs aliyun-python-sdk-vpc aliyuncli

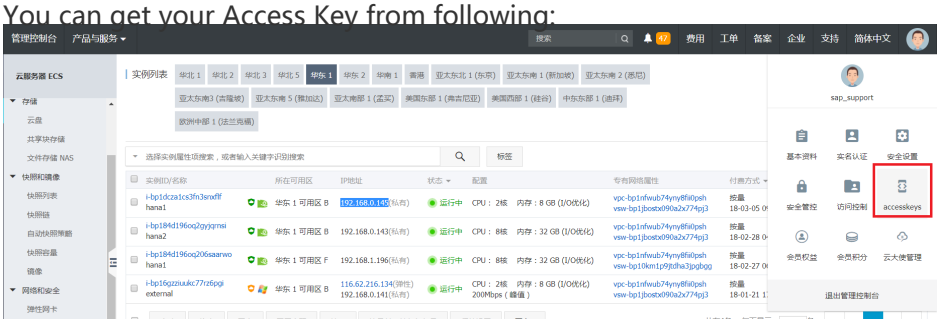
```
hana1:~ # pip install aliyun-python-sdk-ecs aliyun-python-sdk-vpc aliyuncli
Collecting aliyun-python-sdk-ecs
  Downloading http://mirrors.aliyun.com/pypi/packages/02/f6/f63a3ba3357613caade16cf4fa69178eaacccd9cf2fc
6692368024499083/aliyun-python-sdk-ecs-4.6.4.tar.gz (41kB)
    100% |                                | 51kB 31.4MB/s
Collecting aliyun-python-sdk-vpc
  Downloading http://mirrors.aliyun.com/pypi/packages/39/eb/ec999243088e6d59710f9e634ca4b43c8ff895f256aa
5e3a5ca9b2303982/aliyun-python-sdk-vpc-3.0.2.tar.gz
Collecting aliyuncli
  Downloading http://mirrors.aliyun.com/pypi/packages/f1/6c/ecb0313299cc8f886b75d4a117c6ead329ec89562fb1
b82e1c6f083d3c5a/aliyuncli-2.1.9-py2-none-any.whl (69kB)
    100% |                                | 71kB 41.4MB/s
Collecting aliyun-python-sdk-core>=2.0.2 (from aliyun-python-sdk-ecs)
  Downloading http://mirrors.aliyun.com/pypi/packages/fa/28/045cf5cc1e80cc482c3bd1979287e62db73b3c42ef0e
429f2d0d8dd2bfbe/aliyun-python-sdk-core-2.8.1.tar.gz
Collecting colorama<=0.3.3,>=0.2.5 (from aliyuncli)
  Downloading http://mirrors.aliyun.com/pypi/packages/24/84/29ce4167d1f5c4a320aaad91e1178e5a1baf9cfe1c63
```

Configure Alibaba Cloud OpenAPI SDK and Client

aliyuncli configure

```
hana1:~ #
hana1:~ # aliyuncli configure
Aliyun Access Key ID [None]: LTAIPYT3XsfmFfNs
Aliyun Access Key Secret [None]: wjM285NLWOS8dx32bewKwPIjTipHUf
Default Region Id [None]: cn-hangzhou
Default output format [None]: table
hana1:~ #
```

You can get your Access Key from following:



1. Add additional route entry, with following command:route add -host 100.100.100.200 dev eth0 gw IP-Addr-of-your-eth0-gateway

IP-Addr-of-your-eth0-gateway should be replace by real IP address of gateway of your eth0.

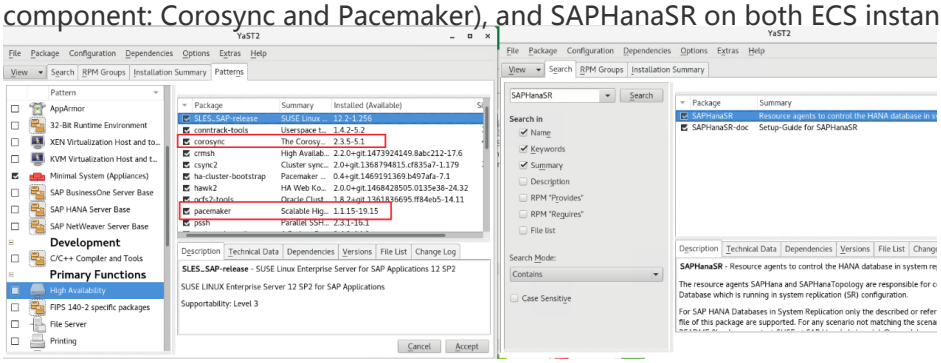
Software Preparation

Software List

- SLES 12 SP2 for SAP
- HANA Installation Media
- HANA Client Installation Media
- SAP Host Agent Installation Media

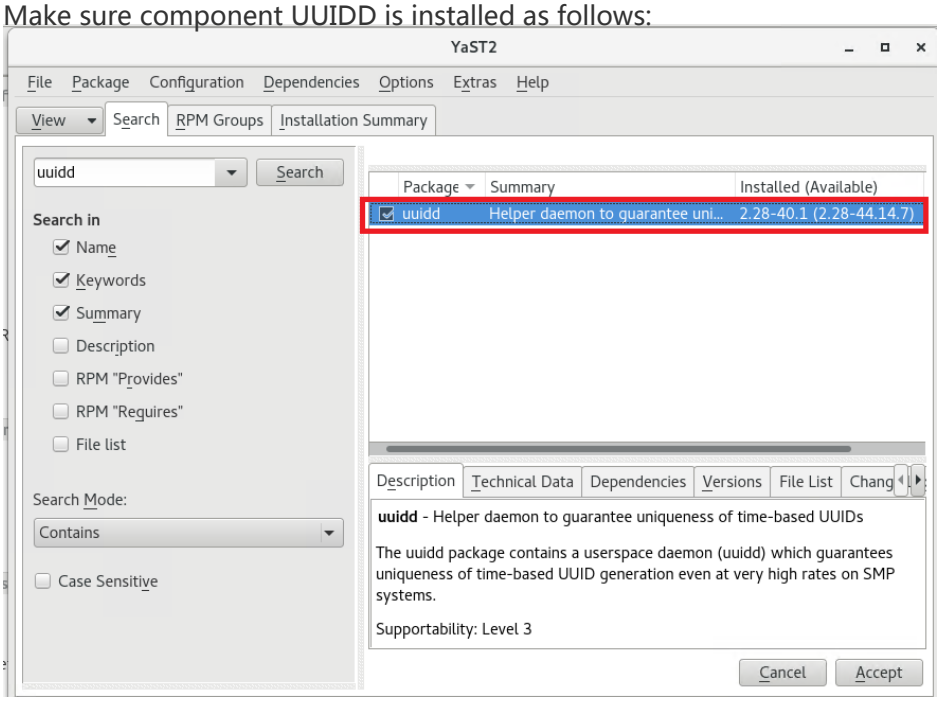
SLES for SAP HAE Installation

Both ECS instances are created with the SLES 12 SP2 for SAP image. Both ECS instances should install the HAE component, as well as package SAPHanaSR.In this example, we install HAE (major software component: Corosync and Pacemaker), and SAPHanaSR on both ECS instances as follows:



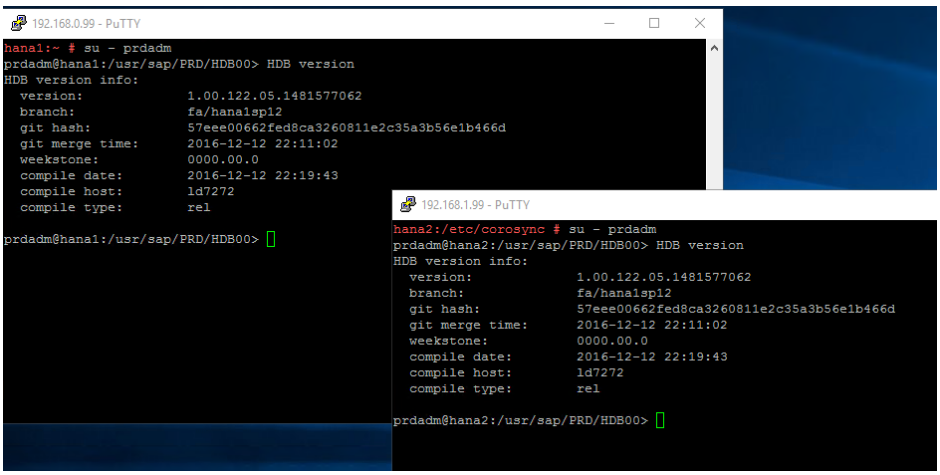
UUIDD Installation

Make sure component UUIDD is installed as follows:



SAP HANA Installation

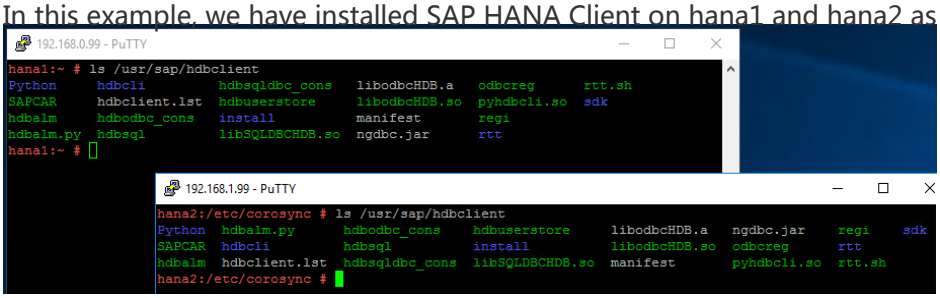
Install SAP HANA software on both ECS instances, and make sure the SAP HANA SID and Instance Number are the same (requirement by SAP HANA System Replication).In this example, both node are installed with SAP HANA (Rev. 1.00.122.05), and SID: **HDB**, Instance Number: **00**.



SAP HANA Client Installation

Install SAP HANA Client on both ECS instances. Default Installation path is as follows: /usr/sap/hdbclient

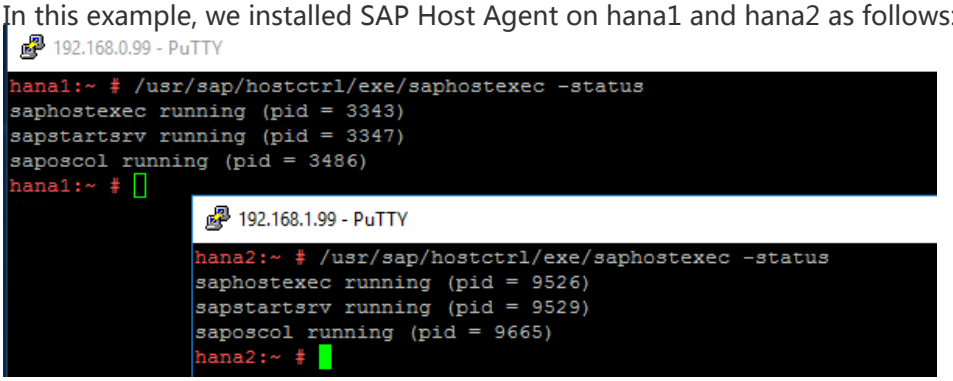
In this example, we have installed SAP HANA Client on hana1 and hana2 as follows:



SAP Host Agent Installation

Install SAP Host Agent on both ECS instances.

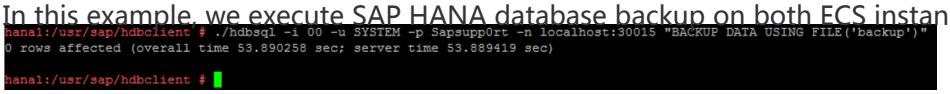
In this example, we installed SAP Host Agent on hana1 and hana2 as follows:



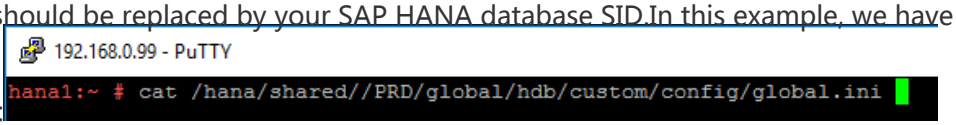
Configuring SAP HANA System Replication

Backup database on both ECS instances for the first time;

Execute following command as root, in the path /usr/sap/hdbclient;
./hdbsql -i InstanceNumber -u SYSTEM -p Password -n localhost:30015 "BACKUP DATA USING FILE('backup')"
InstanceNumber should be replaced by your SAP HANA Instance Number;
Password should be replaced by your SAP HANA SYSTEM user' s password;
In this example, we execute SAP HANA database backup on both ECS instances as follows:



Configuration on SAP HANA Primary Node:
a) Logon SAP HANA primary node as root, and switch to user **[sid]adm**, to stop SAP HANA database using HDB Stop;
b) Switch back to user root, and configure following file:
/hana/shared/SID/global/hdb/custom/config/global.ini
SID should be replaced by your SAP HANA database SID.In this example, we have following path:

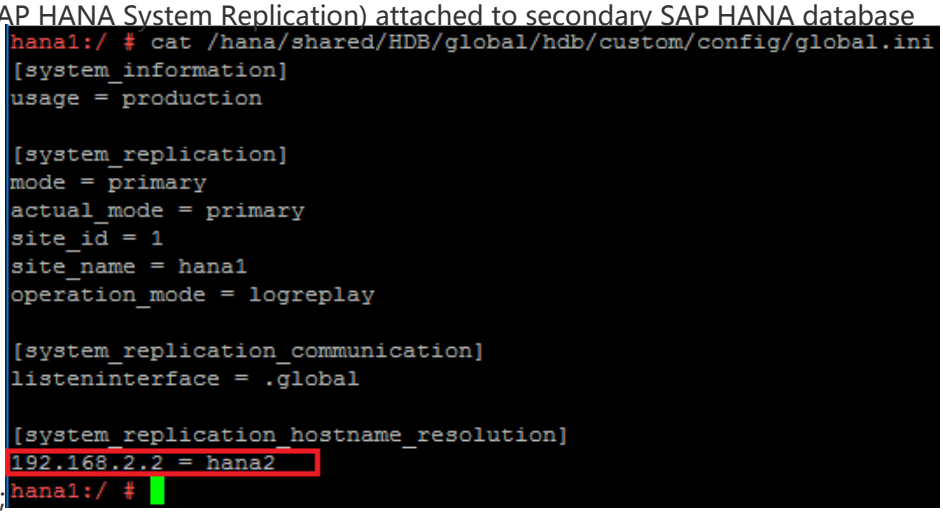


Add following content:

```
[system_replication_communication]
listeninterface = .global
[system_replication_hostname_resolution]
```

IP Address of ENI of Secondary HANA node = **Hostname of Secondary HANA node**

IP Address of ENI of Secondary HANA node should be address of the ENI (for SAP System Replication) attached to the Secondary SAP HANA node;
Hostname of Secondary HANA node should be hostname of the Secondary SAP HANA node (if you are not sure about your ECS instance hostname, use command hostname);
In this example, we have following configuration:
192.168.2.2 of ENI (for SAP HANA System Replication) attached to secondary SAP HANA database node (hostname: hana2);



Configuring SAP HANA System Replication

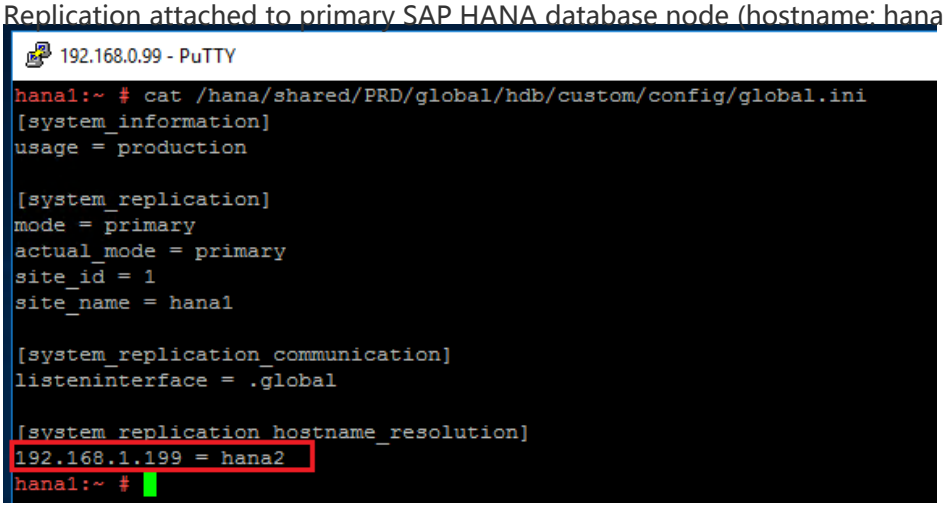
1. Configuration on SAP HANA Secondary Node; (Similar to step 2)
   a) Stop SAP HANA database as user [sid]adm;
   b) Edit following file as root:
   /hana/shared/SID/global/hdb/custom/config/global.ini

Add following content:

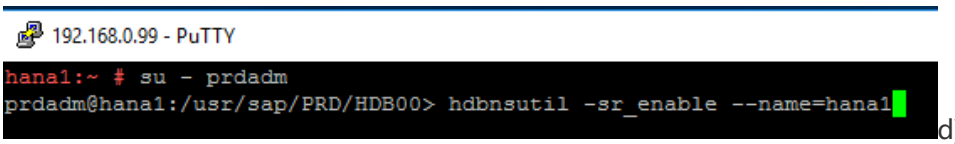
```
[system_replication_communication]
listeninterface = .global
[system_replication_hostname_resolution]
```

IP Address of ENIs of Primary HANA node = **Hostname of Primary HANA node**

IP Address of ENI of Primary HANA node should be address of the ENI (for SAP System Replication) attached to the Primary SAP HANA node;
Hostname of Primary HANA node should be hostname of the Primary SAP HANA node (if you are not sure about your ECS instance hostname, use command hostname);
In this example, we have following configuration: 192.168.2.1 of ENI for SAP HANA System Replication attached to primary SAP HANA database node (hostname: hana1);

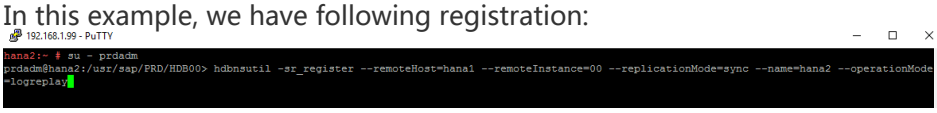


Enable SAP HANA System Replication on SAP HANA on primary node
a) Log onto the primary HANA node with **[sid]adm**;
b) Start SAP HANA database using HDB start;
c) Enable System Replication by executing following command:
hdbnsutil -sr_enable --name= [primary HANA node hostname]
primary HANA node hostname should be replaced by your primary HANA node's hostname.
In this example, we have following setup:

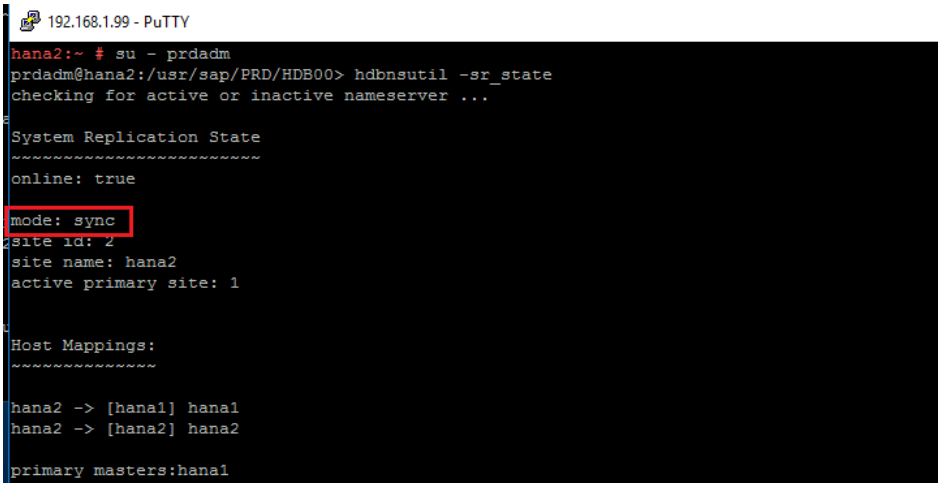


d) Verify the System Replication Status on the primary node by following command:
hdbnsutil -sr_state
In this example, we have following status on primary HANA node hana1:

Register the Secondary HANA node to the Primary HANA node
a) Log onto the secondary HANA node as **[sid]adm**;
b) Stop SAP HANA database by executing: HDB stop;
c) Register the Secondary HANA node to the Primary HANA node by running following command:
hdbnsutil -sr_register --remoteHost=[hostname of primary Node] --remoteInstance=[instance number of primary node] --replicationMode=sync --name=[hostname of the secondary node] --operationMode=logreplay
In this example, we have following registration:

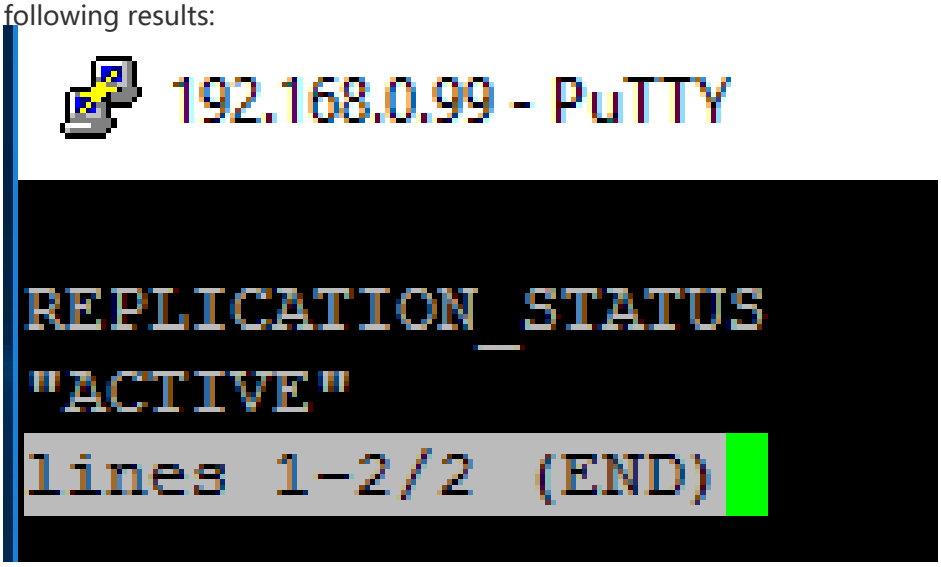


d) Start SAP HANA database on secondary HANA node by executing: HDB start;
e) Verify the System Replication Status on the secondary node by following command:
hdbnsutil -sr_state
In this example, we have following status on secondary HANA node hana2:



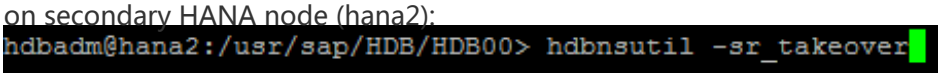
Verify SAP HANA System Replication Status by following command:
./hdbsql -i [InstanceNumber] -u SYSTEM -p [Password] 'select distinct REPLICATION_STATUS from SYS.M_SERVICE_REPLICATION'
InstanceNumber should be replaced by the Instance Number of your SAP HANA instance;
Password should be replaced by password of your SAP HANA system account SYSTEM;
In this example, we execute the following command:

And get following results:



Notes:

1. For HANA 1.0, SAP HANA System Replication only supports Active-Passive mode, which means, the secondary node is not accessible from outside such as HANA Studio.
2. When the primary HANA node fails, you have to manually switch to Secondary node, by executing command: hdbnsutil -sr_takeover;In this example, we run following commands on secondary HANA node (hana2):



3. When we fix the problem of the primary HANA node, want the primary node rejoin the SAP HANA System Replication Cluster, we need to register the primary HANA node as the secondary;In this example, we run following commands on primary HANA node (hana1):

```
hana1:/usr/sap/hdbclient # su - hdbadm
hdbadm@hana1:/usr/sap/HDB/HDB00> hdbnsutil -sr_register --remoteHost=hana2 --remoteInstance=00 --replicationMode=sync --name=hana1 --operationMode=l
ogreplay
```

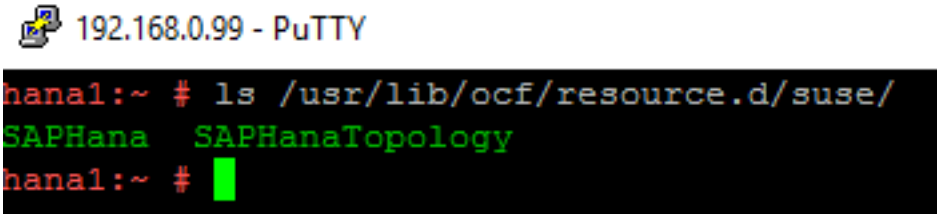
Configuring HAE for SAP HANA

Mechanism of SLES for SAP HAE

SLES for SAP HAE uses Corosync as the cluster infrastructure for messaging and membership purpose, and uses Pacemaker for Cluster Resource Management.

Pacemaker manages all Resource Agents (For Pacemaker, anything that can be scripted can be clustered as Resource; and A Resource Agent is an external program that abstracts the service it provides and present a consistent view to the cluster); For more details of Pacemaker, please kindly refer to: http://clusterlabs.org/doc/ .

For SAP HANA High Availability Solution, SLES for SAP HAE provides two SAP HANA specific Resource Agents (SAPHanaTopology and SAPHana) for managing the SAP HANA HA cluster.You can find the two RA from path: /usr/lib/ocf/resource.d/suseIn our example, we have following:



SAPHanaTopology is a resource agent (RA) that analyzes the SAP HANA topology and “sends” all findings via the node status attributes to all nodes in the cluster. These attributes are taken by the SAPHana RA to control the SAP Hana Databases. In addition it starts and monitors the local SAP Host Agent. (For more information, please kindly use command: man ocf_suse_SAPHanaTopology)

SAPHana is a resource agent for SAP HANA databases. It manages a SAP HANA database with system replication in an OCF master/slave configuration. System replication will help to replicate the database data from one computer to another computer in order to compensate for database failures. With this mode of operation, internal SAP HANA high-availability (HA) mechanisms and the resource agent must work together.
The SAPHana resource agent (RA) performs the actual check of the SAP HANA database instances and is configured as a master/slave resource. Managing the two SAP HANA instances means that the resource agent controls the start/stop of the instances. In addition the resource agent is able to monitor the SAP HANA databases on landscape host configuration level. For this monitoring the resource agent relies on interfaces provided by SAP.
A third task of the resource agent is to also check the synchronization status of the two SAP HANA databases. If the synchronization is not “SOK”, than the cluster avoids to failover to the secondary

side, if the primary fails. This is to improve the data consistency. (For more information, please kindly use command: man ocf_suse_SAPHana)

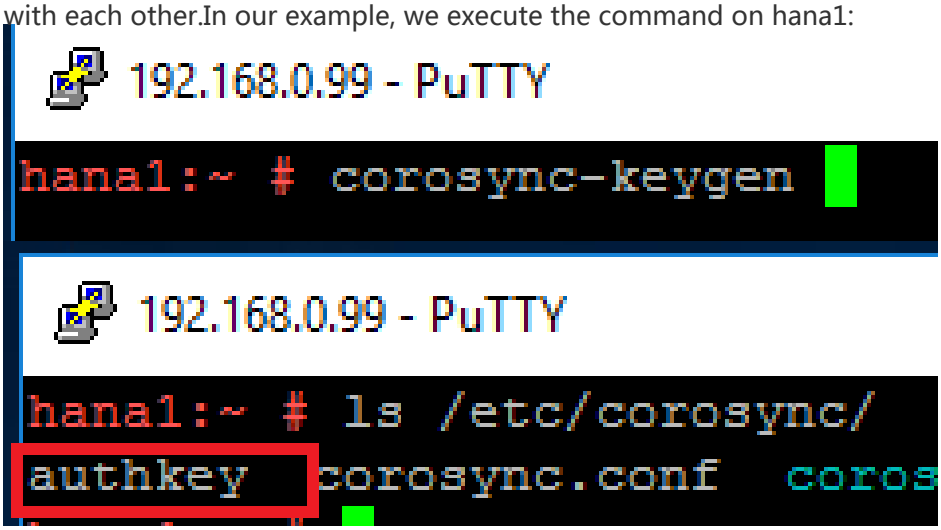
SAP HANA HA Solution is built on top of leveraging RA SAPHanaTopology and SAPHana in Pacemaker.

STONITH: fence_aliyun
For a HA solution, a fencing device is a must. Alibaba Cloud provides its own STONITH device, which allows the servers in the HA cluster to shut down the other which is not responsible. The STONITH device leverage Alibaba Cloud OpenAPI underneath the ECS instance, which is similar to a physical reset / shutdown on a on-premise environment.

Configuration of Corosync

It is desirable that, you add more redundancy for messaging (Heartbeat) by using separate ENIs attached to the ECS instances with separate network range.On Alibaba Cloud, it is strongly suggested that, only using Unicast for the transport setting in Corosync.Follow the following steps to configure Corosync:

Create Key for communication by executing command as root on primary HANA node: corosync-keygenAnd the generated key will be located in the file: /etc/corosync/authkeyThe key will be used by Corosync on different nodes to communicate with each other.In our example, we execute the command on hana1:

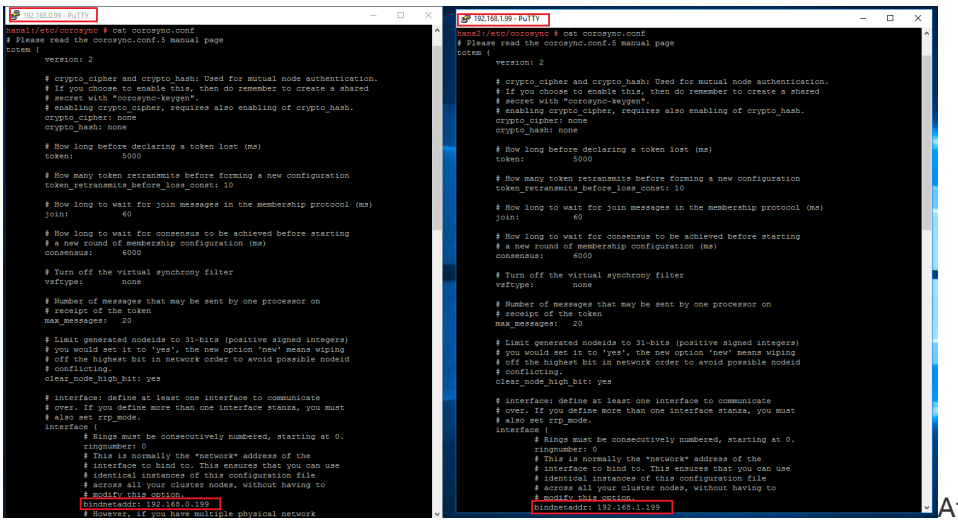


Configure /etc/corosync/corosync.conf with following content as root on primary HANA node:

```
totem {
version: 2
token: 5000
token_retransmits_before_loss_const: 6
crypto_cipher: none
crypto_hash: none
```

```
clear_node_high_bit: yes
interface {
ringnumber: 0
bindnetaddr: **IP-address-for-heart-beating-for-the-current-server**
mcastport: 5405
ttl: 1
}
# On Alibaba Cloud, transport should be set to udpu, means: unicast
transport: udpu
}
logging {
fileline: off
to_logfile: yes
to_syslog: no
logfile: /var/log/cluster/corosync.log
debug: off
timestamp: on
logger_subsys {
subsys: QUORUM
debug: off
}
}
nodelist {
node {
ring0_addr: **ip-node-1**
nodeid: 1
}
node {
ring0_addr: **ip-node-2**
nodeid: 2
}
}
quorum {
# Enable and configure quorum subsystem (default: off)
# see also corosync.conf.5 and votequorum.5
provider: corosync_votequorum
expected_votes: 2
two_nodes: 1
}
```

IP-address-for-heart-beating-for-the-current-server should be replaced by the IP address of the current server, used for messaging (heartbeat) or HANA System Replication. In our example, we use IP address of ENI of the current node; Caution: this value will be different on primary and secondary node.
nodelist directive is used to list all nodes in the cluster.
ip-node-1 and **ip-node-2** should be replaced by the IP addresses of the ENIs attached to ECS instances for Heartbeat Purpose or HANA System Replication Purpose.
Following is an example:



After completing edit of /etc/corosync/corosync.conf on primary HANA node, copy the /etc/corosync/authkey and /etc/corosync/corosync.conf to /etc/corosync on the secondary HANA node with following command:

scp /etc/corosync/authkey root@**hostnameOfSecondaryNode**:/etc/corosync

scp /etc/corosync/corosync.conf root@**hostnameOfSecondaryNode**:/etc/corosync

In our example, we execute following command:

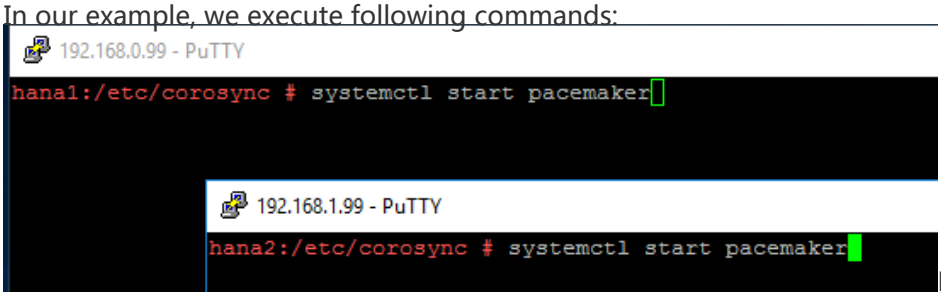
```
hana1:/ # scp /etc/corosync/authkey root@hana2:/etc/corosync
hana1:/ # scp /etc/corosync/corosync.conf root@hana2:/etc/corosync
```

After copy the corosync.conf to the secondary node, please kindly configure the bindnetaddr as above to the local heart beating IP address.

Verify Corosync Configuration

After configuration, we start the cluster for the first time by executing following command on both nodes: systemctl start pacemaker;

In our example, we execute following commands:



Execute command: crm_mon –r

In this example, you get following results after executing crm_mon –r;

```
Stack: corosync
Current DC: hana2 (version 1.1.15-19.15-e174ec8) - partition with quorum
Last updated: Thu Mar  1 10:34:47 2018
Last change: Thu Mar  1 10:34:45 2018 by hacluster via crmd on hana2

2 nodes configured
0 resources configured

Online: [ hana1 hana2 ]

Full list of resources:
```

Configuration of Pacemaker

For SAP HANA HA solution, we need to configure 7 Resource Agents and corresponding constraints in Pacemaker.

Cluster bootstrap and more;Add configuration of bootstrap and default setting of resource and operations to the cluster; Save following scripts in a file: crm-bs.txt

```
property $id='cib-bootstrap-options' \
expected-quorum-votes="2" \
no-quorum-policy="ignore" \
stonith-enable="true" \
stonith-action="off" \
stonith-timeout="150s"
rsc_defaults $id="rsc-options" \
resource-stickness="1000" \
migration-threshold="5000"
op_defaults $id="op-options" \
timeout="600"
```

Execute command to add setting to the cluster: crm configure load update crm-bs.txt
In our example, we have following setup:

192.168.0.99 - PuTTY

```
hana1:/hana/tmp/HAscripts # cat crm-bs.txt
property $id='cib-bootstrap-options' \
              expected-quorum-votes="2" \
              no-quorum-policy="ignore" \
              stonith-enable="true" \
              stonith-action="off" \
              stonith-timeout="150s"
rsc_defaults $id="rsc-options" \
              resource-stickness="1000"     \
              migration-threshold="5000"
op_defaults $id="op-options" \
              timeout="600"
hana1:/hana/tmp/HAscripts # crm configure load update crm-bs.txt
```

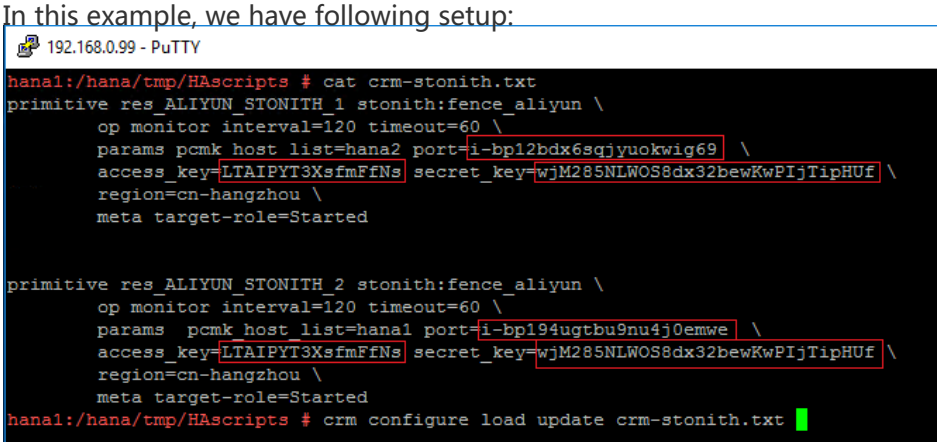
Resource Agents
a) Fencing Device – stonith:fence_aliyun

This part defines STONITH devices in the cluster;
Save following scripts in a file: crm-stonith.txt

```
primitive res_ALIYUN_STONITH_1 stonith:fence_aliyun \
op monitor interval=120 timeout=60 \
params pcmk_host_list=<secondary node hostname> port=<secondary node instance id> \
access_key=<access key> secret_key=<secret key> \
region=<region> \
meta target-role=Started
primitive res_ALIYUN_STONITH_2 stonith:fence_aliyun \
op monitor interval=120 timeout=60 \
params pcmk_host_list=<primary node hostname> port=<primary node instance id> \
access_key=<access key> secret_key=<secret key> \
region=<region> \
meta target-role=Started
```

[secondary node hostname] / **[primary node hostname]** should be replaced by the real hostname of your secondary node;
[secondary node instance id] / **[secondary node instance id]** should be replaced by the real instance-id of your secondary node; you can get this from the console;
[access key] should be replaced with real access key;
[secret key] should be replaced with real secret key;
[region] should be replaced with real region name where the node locates;

Execute command to add the resource to the cluster: crm configure load update crm-stonith.txt
In this example, we have following setup:



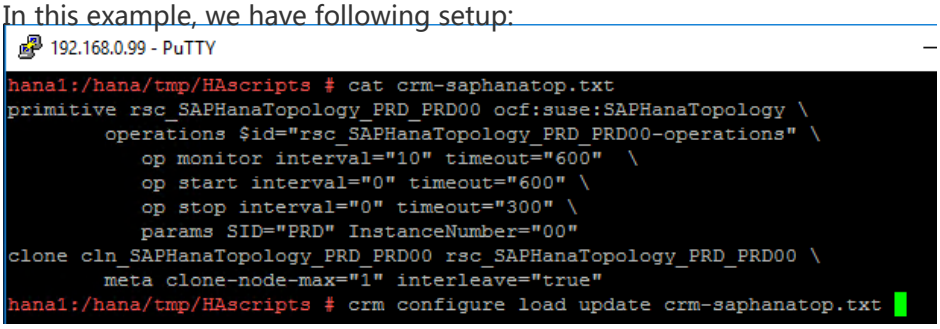
b) SAPHanaTopology
This part defines a SAPHanaTopology RA, and a clone of SAPHanaTopology on both nodes in the cluster. Save following scripts in a file: crm-saphanatop.txt

```
primitive rsc_SAPHanaTopology_HDB_HDB00 ocf:suse:SAPHanaTopology \
operations $id="rsc_SAPHanaTopology_HDB_HDB00-operations" \
op monitor interval="10" timeout="600" \
op start interval="0" timeout="600" \
op stop interval="0" timeout="300" \
params SID="HDB" InstanceNumber="00"
```

```
clone cln_SAPHanaTopology_HDB_HDB00 rsc_SAPHanaTopology_HDB_HDB00 \
meta clone-node-max="1" interleave="true"
```

HDB should be replaced by the real SAP HANA SID;
00 should be replaced by the real SAP HANA Instance Number;
Execute command to add resources to the cluster:
crm configure load update crm-saphanatop.txt

In this example, we have following setup:

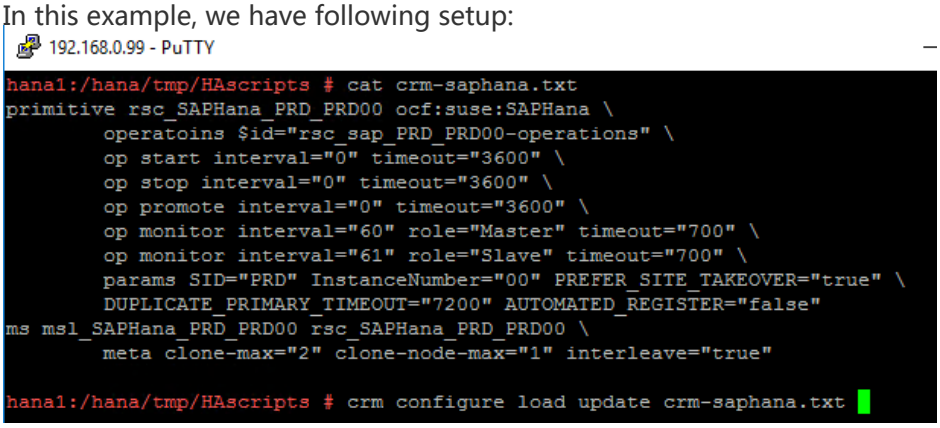


c) SAPHanaThis part defines a SAPHana RA, and a Multi-state resource of SAPHana on both nodes in the cluster. Save following scripts in a file: crm-saphana.txt

```
primitive rsc_SAPHana_HDB_HDB00 ocf:suse:SAPHana \
operatoins $id="rsc_sap_HDB_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="700" \
op monitor interval="61" role="Slave" timeout="700" \
params SID="HDB" InstanceNumber="00" PREFER_SITE_TAKEOVER="true" \
DUPLICATE_PRIMARY_TIMEOUT="7200" AUTOMATED_REGISTER="false"
ms msl_SAPHana_HDB_HDB00 rsc_SAPHana_HDB_HDB00 \
meta clone-max="2" clone-node-max="1" interleave="true"
```

HDB should be replaced by the real SAP HANA SID;
00 should be replaced by the real SAP HANA Instance Number;
Execute command to add resources to the cluster:
crm configure load update crm-saphana.txt

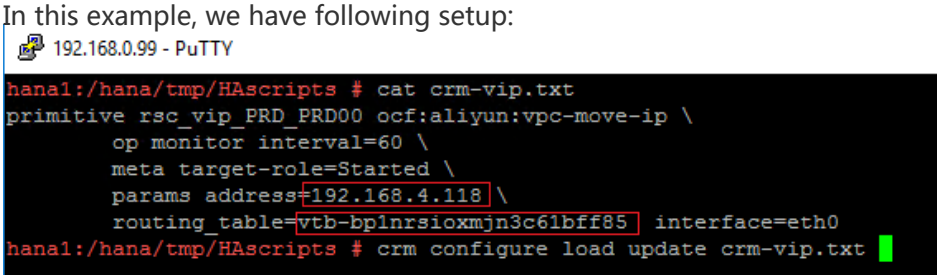
In this example, we have following setup:



d) Virtual IPThis part defines a Virtual IP RA in the cluster. Save following scripts in a file: crm-vip.txt.

```
primitive res_ALIYUN_IP ocf:aliyun:vpc-move-ip \
op monitor interval=60 \
meta target-role=Started \
params address=<virtual_IPv4_address> routing_table=<route_table_ID> interface=eth0
```

[virtual_IP4_address] should be replaced by the real IP address you prefer toprovide service;
[route_table_ID] should be replaced by the route table ID of your VPC;
Execute command to add the resource to the cluster:
crm configure load update crm-vip.txt
In this example, we have following setup:

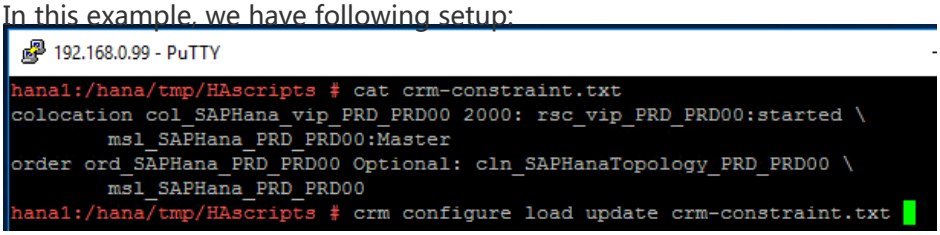


1. ConstraintsTwo constraints are organizing the correct placement of the virtual IP address for the client database access and the start order between the two resource agents SAPHana and SAPHanaTopology. Save following scripts in a file: crm-constraint.txt

```
colocation col_SAPHana_vip_HDB_HDB00 2000: rsc_vip_HDB_HDB00:started \
msl_SAPHana_HDB_HDB00:Master
order ord_SAPHana_HDB_HDB00 Optional: cln_SAPHanaTopology_HDB_HDB00 \
msl_SAPHana_HDB_HDB00
```

HDB should be replaced by the real SAP HANA SID;
00 should be replaced by the real SAP HANA Instance Number;

Execute command to add the resource to the cluster:
crm configure load update crm-constraint.txt
In this example, we have following setup:



Start HANA HA Cluster
Execute command: systemctl start uuidd
Execute command: systemctl start pacemaker

Monitor the HANA HA Cluster

Execute command: systemctl status pacemaker

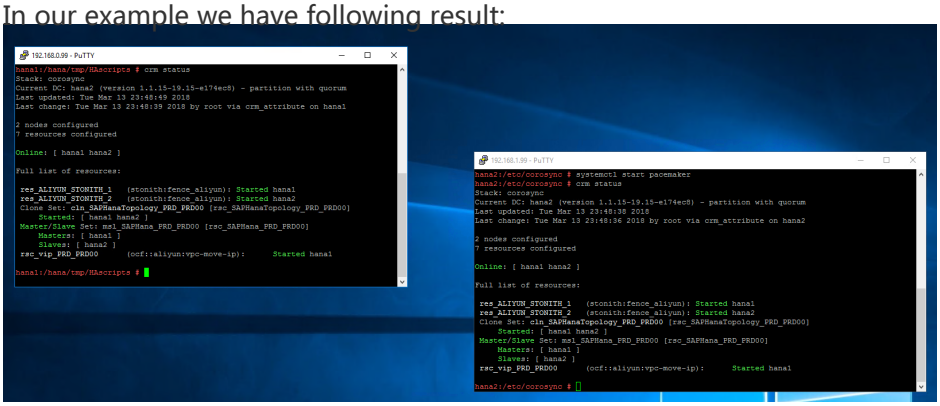
In our example, we have following output:

```
hana1:~ # systemctl status pacemaker
● pacemaker.service - Pacemaker High Availability Cluster Manager
   Loaded: loaded (/usr/lib/systemd/system/pacemaker.service; disabled; vendor preset: disabled)
   Active: active (running) since Wed 2018-01-03 22:39:46 CST; 28min ago
     Docs: man:pacemakerd
           http://clusterlabs.org/doc/en-US/Pacemaker/1.1-pcs/html/Pacemaker_Explained/index.html
 Main PID: 21337 (pacemakerd)
    Tasks: 11 (limit: 512)
   CGroup: /system.slice/pacemaker.service
           ├─21337 /usr/sbin/pacemakerd -f
           ├─21339 /usr/lib64/pacemaker/cib
           ├─21340 /usr/lib64/pacemaker/stonithd
           ├─21341 /usr/lib64/pacemaker/lrmd
           ├─21342 /usr/lib64/pacemaker/attrd
           ├─21343 /usr/lib64/pacemaker/pengine
           ├─21344 /usr/lib64/pacemaker/crmd
           ├─27938 /bin/bash /usr/lib/ocf/resource.d/suse/SAPHanaTopology monitor
           ├─28096 /bin/bash /usr/lib/ocf/resource.d/suse/SAPHanaTopology monitor
           ├─28097 /bin/bash /usr/lib/ocf/resource.d/suse/SAPHanaTopology monitor
           └─28098 timeout 60 su - hdbadm -c true; /usr/sap/HDB/HDB00/HDBSettings.sh hdbnsutil -sr_stateConfiguration --sapcontrol=1

Jan 03 23:07:50 hana1 su[27415]: (to hdbadm) root on none
Jan 03 23:07:50 hana1 su[27415]: pam_unix(su-l:session): session opened for user hdbadm by (uid=0)
Jan 03 23:07:51 hana1 su[27537]: (to hdbadm) root on none
Jan 03 23:07:51 hana1 su[27537]: pam_unix(su-l:session): session opened for user hdbadm by (uid=0)
Jan 03 23:07:53 hana1 su[27713]: (to hdbadm) root on none
Jan 03 23:07:53 hana1 su[27713]: pam_unix(su-l:session): session opened for user hdbadm by (uid=0)
Jan 03 23:08:04 hana1 su[27969]: (to hdbadm) root on none
Jan 03 23:08:04 hana1 su[27969]: pam_unix(su-l:session): session opened for user hdbadm by (uid=0)
Jan 03 23:08:04 hana1 su[28099]: (to hdbadm) root on none
Jan 03 23:08:04 hana1 su[28099]: pam_unix(su-l:session): session opened for user hdbadm by (uid=0)
hana1:~ #
```

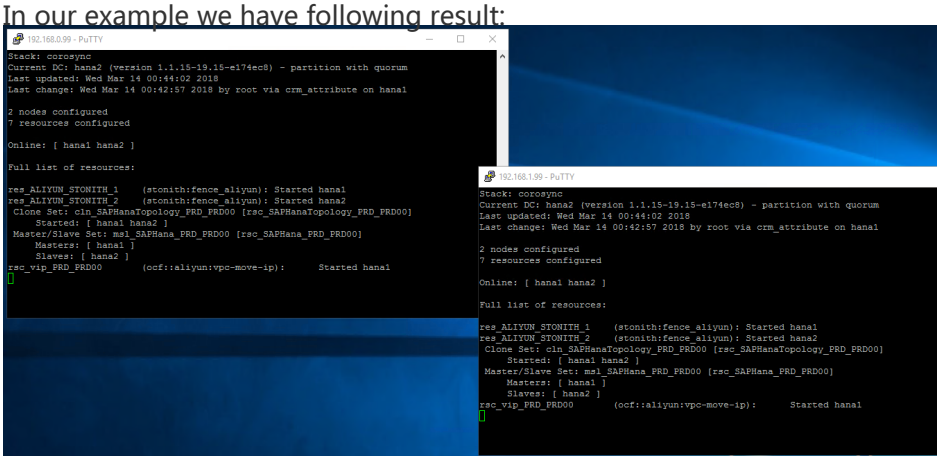
Execute command: crm status

In our example we have following result:



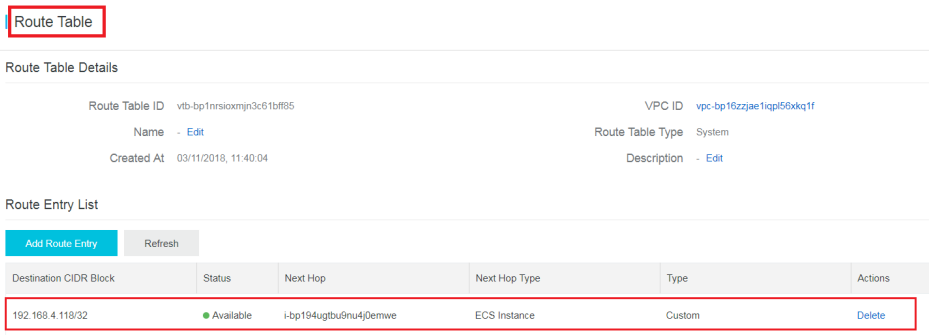
Execute command: crm_mon –r

In our example we have following result:



Meanwhile, please kindly check, if a new entry **[virtual_IP4_address]** is added into the route table of VPC.

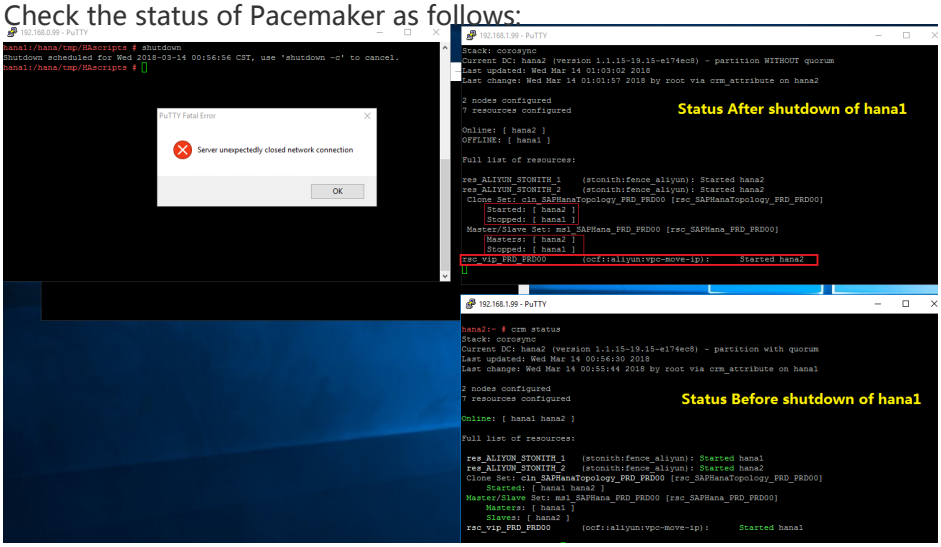
In our example, we have following:



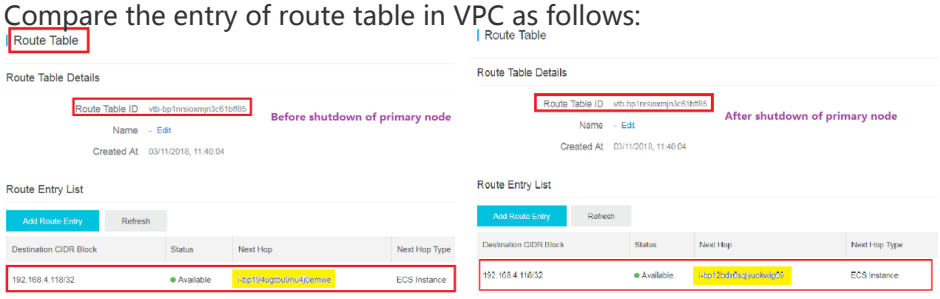
Verify the HA take over

1. Shutdown the primary node as follows:

   Check the status of Pacemaker as follows:



   Compare the entry of route table in VPC as follows:



Reference

- Pacemaker 1.1 Configuration Explained
- SAP HANA SR Performance Optimized Scenario – SUSE Linux Enterprise Server for SAP Applications 12 SP1

Microsoft SQL Server on Alibaba Cloud

Microsoft SQL Server on Alibaba Cloud

- Getting Started
- Prerequisites
    - Alibaba Cloud account and RAM
    - SQL Server Images and Version
    - ECS instance
    - VPC
    - Cloud Disk
    - OSS
    - Shared block storage
- Installation
    - Create SQL Server instance
    - Windows Settings
- High Availability
- Backup
- Performance
    - Separate Cloud Disk
    - Table Compression
    - Enable Buffer Pool Extension(BPE)
    - Max degree of parallelism setting
- Monitor
    - Monitoring Capability
    - Alarm Capability
- Management Studio
    - Remote SQL Server Management Studio
    - Default SQL Server Management Studio

Getting Started

Make use of Alibaba Cloud reliable and flexible cloud computing infrastructure and platform services can help you to run Microsoft SQL Server more stably and smoothly. Microsoft SQL server on Alibaba cloud Elastic Compute Service(ECS) just like when you installed on-premises, you are responsible for installation, administering the database, including backups and recovery, patching the operating system and the database, tuning of the operating system and database parameters, managing

security, and configuring high availability or replication. It gives you complete control over every setting. To learn what is ECS, please refer to the detail .This article provides you with the best practice for Microsoft SQL Server instance running on ECS instance. Please be aware this article used **SQL server 2016 Enterprise Edition** as the example, so not every option is appropriate for every version. It also presents a simplistic procedure to guide you for running your Microsoft SQL server. If you want to choose Alibaba RDS for SQL server, please refer to Quick Start for SQL Server.

Prerequisites

Alibaba Cloud account and RAM

You must have registered to an Alibaba Cloud account. We recommend you to enable RAM to manage your account. Resource Access Management (RAM) is a cloud service that helps you manage user identities and control resources access. Using RAM, you can create and manage user accounts, and control the operation permissions that these user accounts possess for resources under your account, for example, employees, systems, and applications. For the detail information please refer to Quick Start and RAM best practices.

SQL Server Images and Version

Alibaba cloud support Bring Your Own License (BYOL) and Images.SQL Server Enterprise, Standard, and Express Editions are licensed for production use. For Enterprise and Standard Editions, contact your software vendor for the installation media. You can find purchasing information and a directory of Microsoft partners on Microsoft official purchasing website.Free editions you can find at Microsoft official website: SQL Server Downloads.

ECS instance

Elastic Compute Service (ECS) is a type of computing service that features elastic processing capabilities. ECS has a simpler and more efficient management mode than physical servers. You can create instances, change the operating system, and add or release any number of ECS instances at any time to fit your business needs.An ECS instance is a virtual computing environment that includes CPU, memory, and other basic computing components. An instance is the core component of ECS and is the actual operating entity offered by Alibaba Cloud. Other resources, such as disks, images, and snapshots, can only be used in conjunction with an ECS instance.Before create SQL server instances you have to create ECS instances first using the ECS console, about the detail information, please refer to Create ECS instances.

VPC

Virtual Private Cloud (VPC) creates an isolated network environment for you SQL Server environment. You can select an IP address range, divide networks, and configure the routing list and gateway. The interflow of VPC intranet and between VPC and physical IDC machine rooms can be realized among

regions or users. About how to create VPC please refer to the detail.

Cloud Disk

Ultra Cloud Disk: When you create ECS instance, Ultra Cloud Disk as the system disk provides a high-performance location for operating system and windows page file.
SSD Cloud Disk: When you create ECS instance we recommend you choose SSD cloud disk store the database files, tempdb,log file separately. Separate SSD cloud disks provide high performance and high reliability.

- High performance: A single SSD cloud disk provides a maximum of 20,000 random reading/writing IOPS and 300 MBps throughput of storage performance.
- IOPS=min{1200+30*disk_size, 20000}. The base is 1200 IOPS, and each GB provides 30 random IOPS up to a maximum of 20,000.
- Throughput=min{80+0.5*disk_size, 300} MBps. The base is 80 MBps, and each GB adds an additional 0.5 MBps up to a maximum of 300 MBps throughput performance.
- Reliability: SSD cloud disks use Alibaba Cloud’ s Apsara distributed storage technology, based on three distributed copies, which can guarantee 99.9999999% data reliability.

For how to create a cloud disk, please refer to create a cloud disk.

OSS

Alibaba Cloud Object Storage Service (OSS) is a network-based data access service. OSS enables you to store and retrieve unstructured data including text files, images, audios, and videos. We recommend you backup your SQL Server database into OSS. For how to use OSS please see Get started with Object Storage Service

Shared block storage

Shared Block Storage is designed for the high availability architecture of enterprise-class applications and provide shared access to block storage devices in a Share-everything architecture, such as the SQL Server always on with WSFC node architecture, which is common among government departments, enterprises, and financial customers, and the high availability server cluster architecture. For about shared block storage detail, please see Shared block storage FAQ

Installation

This section provides general information about how to create a SQL server instance on Elastic Compute Service (ECS). The tutorial includes the following tasks：

Create SQL Server instance

We recommend you to close the windows update setting before you create your SQL Server instance as below:

1.Click “run” button and input gpedit.msc
2.You will open the “Local Computer Policy”:

- Select “Administrative template”
- Select ” Windows Components”
- Select ” Windows Update”

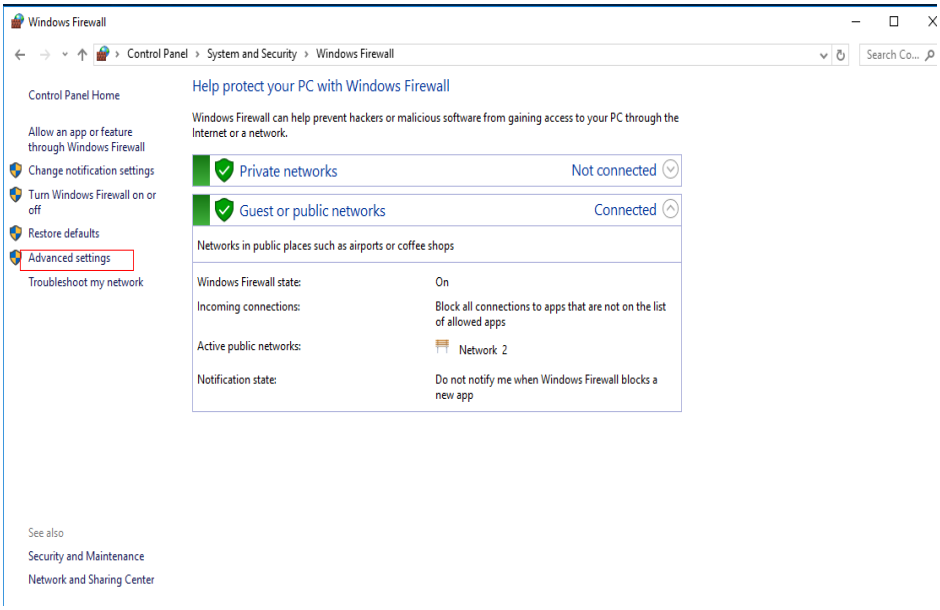
3.Change the “Configure Automatic Updates” option todisablestatus.

After the SQL server instance creating you can decide whether need to enable the update setting. It is the same method with on premise to create your SQL Server instance on ECS instance. You can choose installation wizard, command-line or using a configuration file. For how to create or install please refer to Microsoft website.

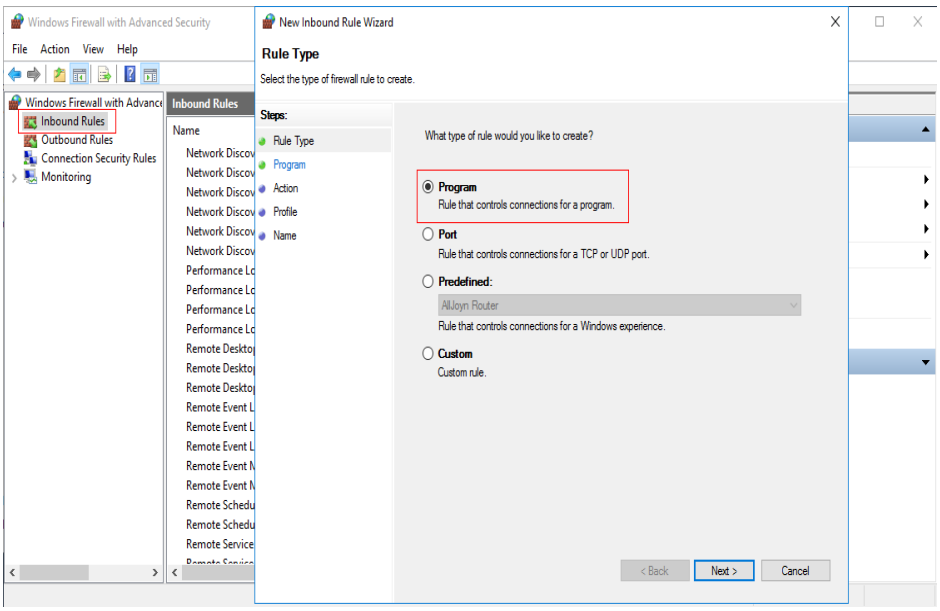
Windows Settings

This section provides you the settings about how to configure windows settings to better optimize your SQL Server instance. We recommend you to setting the windows server firewall rule to specify the IP addresses for your client computer. It is very important security policy when you create your SQL Server instance to be able to connect to the database from other client machines. Configure the firewall to allow incoming traffic:

- Open your windows firewall with advanced security.

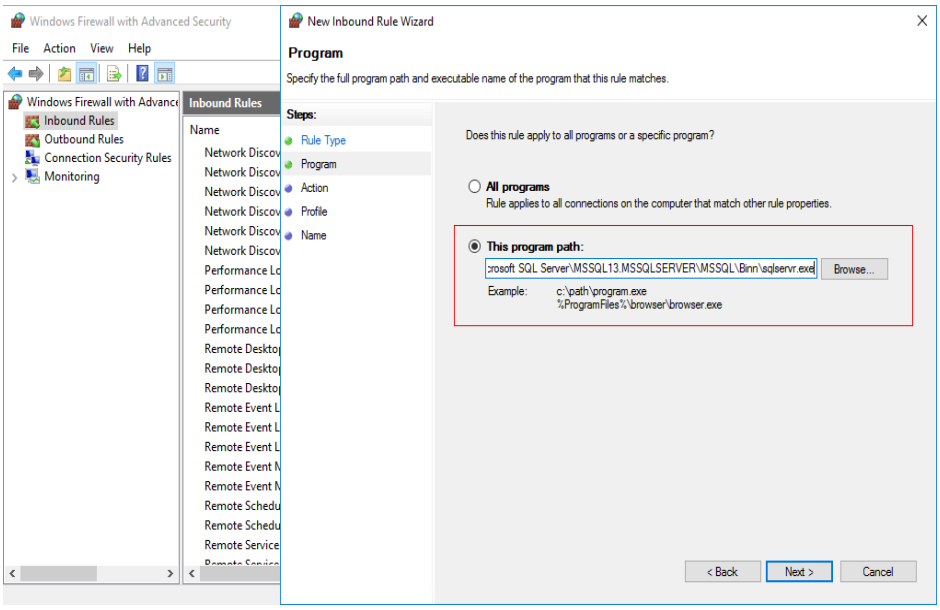


- Click right button to create a new inbound rule.

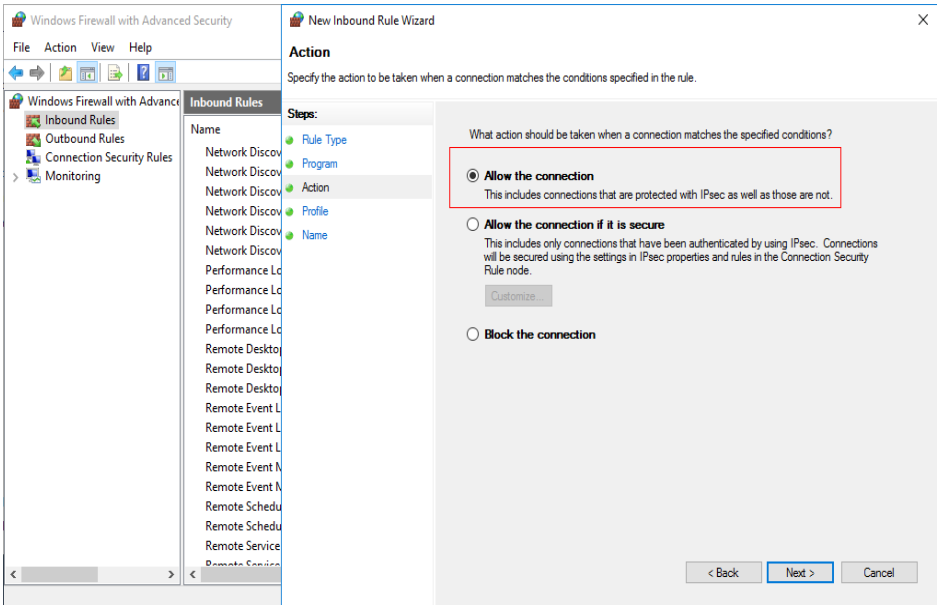


- Select your program path.

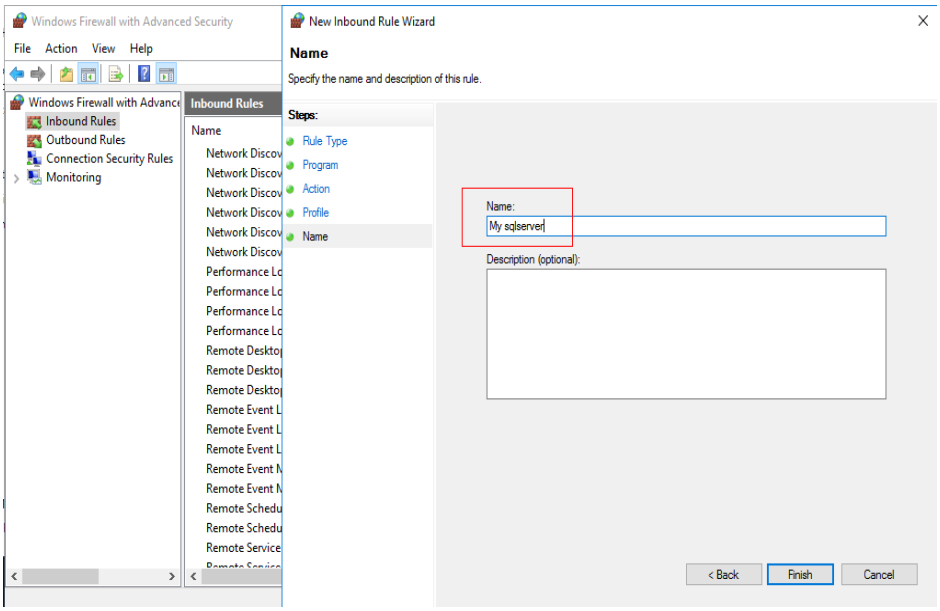
Input:%ProgramFiles%\Microsoft SQL Server\MSSQL13.MSSQLSERVER\MSSQL\Binn\sqlservr.exe



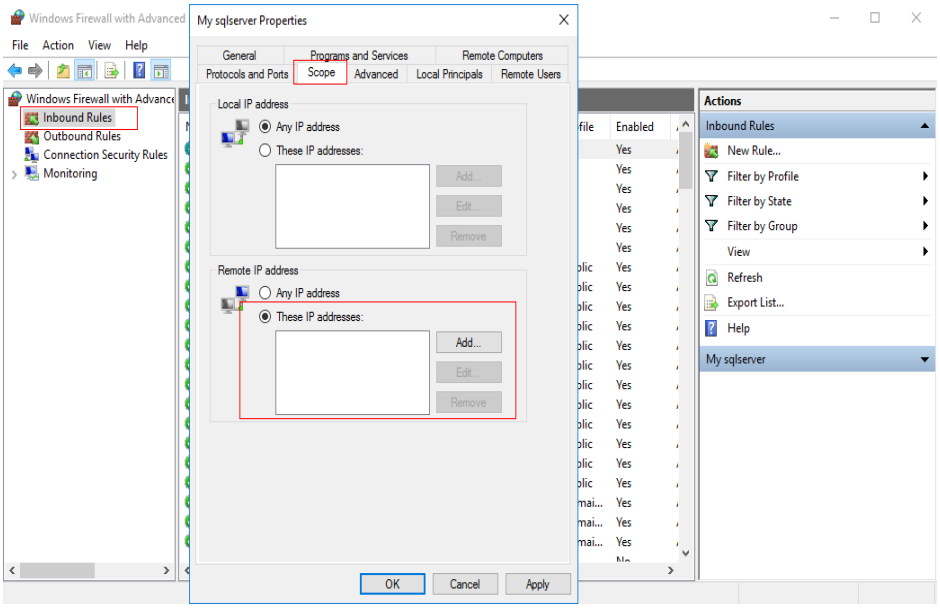
- Allow the connection.



- Name your rule,for example:mysqlserver.



- Set your remote policy.



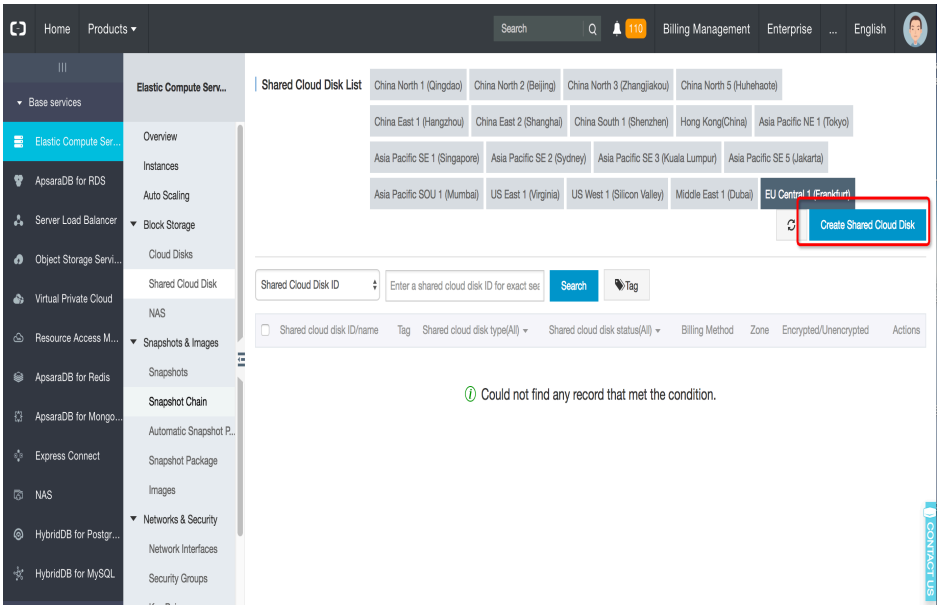
The system default network settings are usually sufficient. Alibaba cloud offers you the high capacity and performance network. When you create ECS instances you can choose the network bandwidth from 1M up to 100M,about the bandwidth, Please refer to ECS Bandwidth FAQs.Windows requires anti-virus software to be installed. Install enterprise level anti-virus software and enable virus library updating and real-time protection, however, if the antivirus software is not configured correctly, it can negatively impact your database performance. Microsoft provides advice about how to choose antivirus software.

High Availability

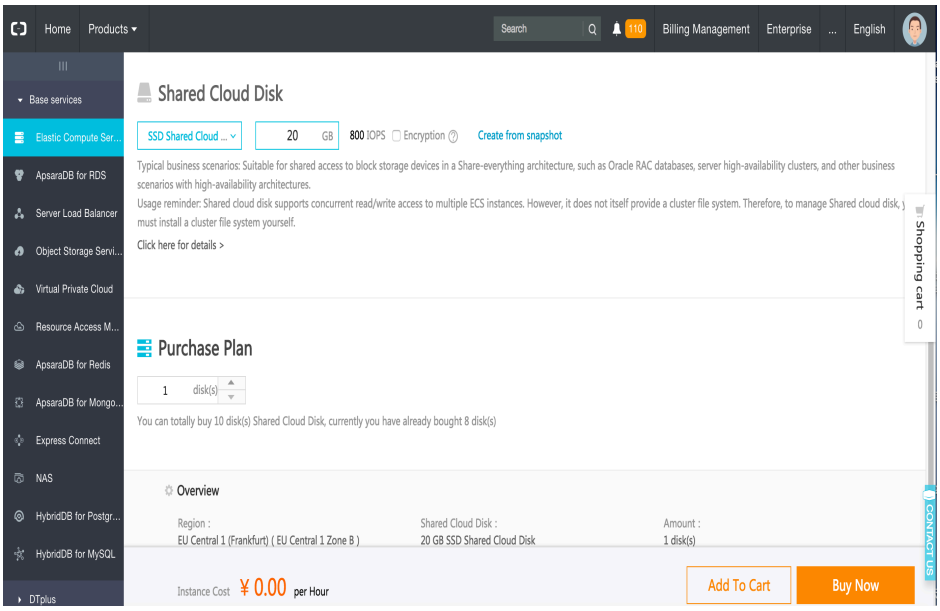
We recommend you to use Windows Server Failover Clustering and SQL Server AlwaysOn Availability Groups as your SQL Server high availability solution on ECS instances.
The Always On feature must be enabled for the server instance ‘sqlserver’ before you can create an availability group on this instance. To enable Always On:

- Open the SQL Server Configuration Manager.
- Select SQL Server Services.
- Right-click the SQL Server instance name.
- Select Properties,and use the Always On High Availability tab of the SQL Server Properties dialog. (ObjectExplorer).

You should create 3 ECS instances, one is for the DC(domain controller)and DNS, the other two are the cluster nodes. You need to create a shared cloud disk as the shared block storage as below:



and then choose the region and size, please pay attention you can only buy 20GB at least as the shared block storage.



For how to plan, configure your WSFC and Always on group please see Microsoft official website.

Backup

The SQL Server backup and restore component provides an essential safeguard for protecting critical data stored in your SQL Server databases. We strongly recommend place the databases, logs, backups on separate SSD Cloud disk. Placing the data and backups on SSD Cloud disk also enhances the I/O performance for both writing backups and the production use of the database. A backup and restore strategy contains a backup portion and a restore portion. Designing an effective backup and restore strategy requires careful planning, implementation, and testing. There is no difference

between doing SQL Server database backup and restore from Alibaba cloud and on premise version. Backing Up and Restoring How-to Topics (SQL Server Management Studio)and Backing Up and Restoring How-to Topics (Transact-SQL)

provides best practice for how to implement a solid backup and maintenance action.
Use the Cloud SSD disk to store your backups and then copy them into OSS bucket or you can use the windows task scheduler copy them as the regular task.

Performance

This section provides you with how to tuning your SQL server instance performance on ECS instances. Running SQL Server on ECS environment we recommend you continue using the same database performance tuning options that are applicable to SQL Server in on-premises server environment SQL Server Enterprise Edition has a long list of added capabilities over Standard Edition. If you are migrating an existing license to ECS, there are some performance options that you might consider implementing.

Separate Cloud Disk

We recommend you to place the databases, logs, backups on separate SSD Cloud disk during you create your SQL Server instances

Table Compression

Generally, data compression reduces the space occupied by the data. It can help improve performance of I/O intensive workloads because the data is stored in fewer pages and queries need to read fewer pages from disk. Data compression can be performed for a table, clustered index, non-clustered index. We recommend you to enable table and index compression. It might seem counter-intuitive that compressing tables could make your system perform faster, but in most cases, that’ s what happens. The tradeoff is using a small amount of CPU cycles to compress the data and eliminate the extra disk IO required to read and write the bigger blocks. Generally, the less disk IO your system uses, the better its performance will be. Instructions for estimating and enabling table and index compression please refer to Micosoft website

Enable buffer pool extension (BPE)

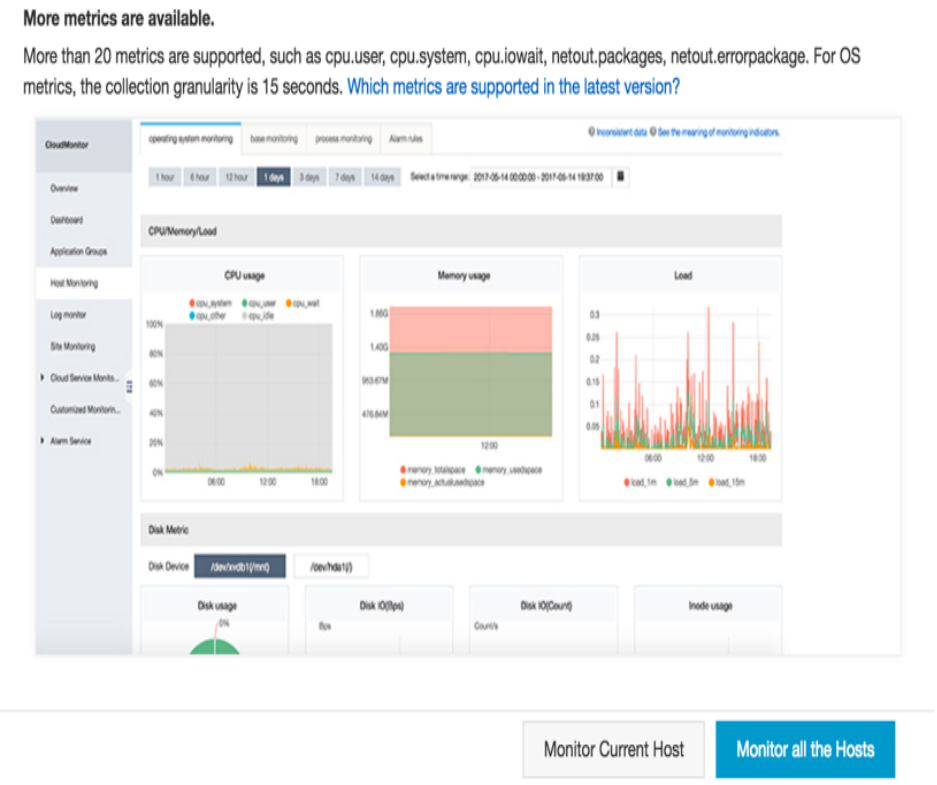
We recommend you to use the buffer pool extension to speed data access. The buffer pool extension feature enables you to push clean pages to the SSD Cloud disk, instead of dropping them. This works along the same lines as virtual memory, which is to say by swapping, and gives you access to the clean pages on the SSD Cloud disk, which is faster than you would get by going to the regular disk to fetch the data. This technique is not nearly as fast as having enough memory, but it can give you a modest increase in throughput when your available memory is low. For how to enable BPE and the technology detail please refer to Microsoft website

Max degree of parallelism setting

We recommend you to configure the max degree of parallelism option to 8. When your SQL Server instance runs on ECS that has more than one processor, it detects the best degree of parallelism, that is, the number of processors employed to run a single statement, for each parallel plan execution. You can use the max degree of parallelism option to limit the number of processors to use in parallel plan execution.
This value is set using sp_configure system procedure or you use SQL Server Management Studio. The default value is 0 which means there is no upper limit and SQL Server can use all available processors. If you set Max Degree of Parallelism to 1 then all queries will execute serially. This setting is ignored on servers with a single processor. Occasionally you might find that parallelism actually hinders performance of some queries. In this case the cost of initializing and synchronizing parallel plans might exceed the benefit of running portions of the query on multiple threads. If you feel that serial execution of a particular query can provide better performance you can override this setting using MAXDOP option within an individual query. For how to configure please refer to Microsoft website

Monitor

We recommend you to use CloudMonitor to monitor your ECS instances. Make sure that your ECS monitoring agents are functional to collect metric data. Otherwise, you must install the agent manually. For more information, see How to install CloudMonitor agent.



Monitoring capability

CloudMonitor allows more than 30 metrics covering CPU, memory, disk, and network to meet the basic monitoring and O&M requirements of the servers. Click here to view the full list of metrics the switch.

Alarm capability

CloudMonitor provides alarm service for all metrics, allowing you to set alarm rules for individual servers, application groups, and all the other resources. You can use the alarm service as per your business requirements. CloudMonitor provides Host monitoring metrics to set alarm rules for individual servers, application groups, and all the other resources. You can use the alarm services as per your business requirements. You can use the alarm service directly in the host monitoring list, or use it in your application group once you add servers to the group. You can add the alarm rules directly in the host monitoring list, or use it in your application group once you add servers to the group. For how to create an alarm service, please see here

Management Studio

You can use SQL Server Management Studio to perform most administrative tasks. This section provides you with how to manage your SQL server instance on ECS instances.

Remote SQL Server Management Studio

Microsoft offered SQL Server Management Studio to configure SQL Server databases. You can download and installed it on your desktop, connect to database remotely.

Default SQL Server Management Studio

You also can use the default SQL Server Management Studio which running on the instance itself. With this method you should connect to your SQL Server instance through RDP.SQL Server 2012 and SQL Server 2014 both include the SQL Server Management Studio by default. For SQL Server 2016, you must download the SQL Server Management Studio from the Microsoft website and install it on the instance.By default, SQL Server uses Windows Authentication mode to control remote access to SQL Server itself. If you need to use SQL Server Authentication mode, change the authentication mode.