aws

SAP HANA Guides

SAP HANA on AWS

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

SAP HANA on AWS: SAP HANA Guides

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

Table of Contents

Home **1**
AWS Backint Agent for SAP HANA **2**
How AWS Backint Agent for SAP HANA works 3
Billing 3
Supported operating systems 3
Supported databases 4
Supported Regions 4
Amazon S3 4
Prerequisites 4
Install and configure 8
Backup and restore 42
Uninstall 46
AWS Backup 47
Prerequisites 47
Install and configure 47
Backup and restore 49
Verify signature 49
Troubleshoot 52
Agent logs 52
Installation 53
Backup and recovery 54
Backup deletion 60
Version history 61
EBS snapshots for SAP HANA **66**
Considerations 66
How to automate the creation of EBS snapshots for SAP HANA 68
Restoring SAP HANA from EBS snapshots 69
Step 1: Prepare for a restore 70
Step 2: Attach or replace the restored EBS volumes 71
Step 3: Recover SAP HANA database 72
Step 4: Resume standard operations 73
Migrating SAP HANA to AWS **74**
..... 74
Migration frameworks 74

6 Rs ... 75
AWS CAF ... 76
Planning ... 77
Understanding on-premises resource utilization ... 77
Reviewing AWS automation tools for SAP ... 77
Prerequisites ... 77
SAP HANA sizing ... 78
Memory requirements for rehosting ... 78
Memory requirements for replatforming ... 79
Instance sizing for SAP HANA ... 80
Network planning and sizing ... 80
SAP HANA scale-up and scale-out ... 81
Migration tools and methodologies ... 82
AWS Launch Wizard ... 83
AWS Migration Hub Orchestrator ... 83
Amazon S3 Transfer Acceleration ... 83
AMIs ... 84
AWS Snowball ... 84
Amazon S3 Transfer Acceleration ... 84
SAP HANA HSR with initialization via backup and restore ... 85
Migration using DMO with System Move ... 85
SAP HANA classical migration ... 86
SAP Software SUM DMO ... 86
DMO Move to SAP S/4HANA on AWS (single step) – DMOVE2S4 ... 87
Backup/restore tools ... 88
Migration scenarios ... 89
Migrating *anyDB* to SAP HANA on AWS ... 91
Migrating SAP HANA to AWS ... 93
Option 1: AWS Migration Hub Orchestrator ... 93
Option 2: Backup and restore ... 94
Option 3: Classical migration ... 95
Option 4: HSR ... 96
Option 5: HSR (with initialization via backup and restore) ... 97
Option 6: On-premises to AWS Cloud ... 98
Migrating SAP HANA to a High Memory instance ... 99
Option 1: Resizing an instance with host or dedicated tenancy ... 100

Option 2: Migrating from an instance with default tenancy 100
Option 3: Migrating from Amazon EC2 High Memory metal instance with Virtualized High Memory Host Tenancy 109
Security 111
Document history 112
SAP HANA Environment Setup 113
Prerequisites 114
Specialized knowledge 114
Technical requirements 114
Plan the deployment 115
Compute 115
Operating system 115
AMI 116
Storage 116
Network 117
Configure operating system 117
SLES 12/15 117
RHEL 7/8/9 121
Configure storage 125
Storage architecture 125
Configure storage (EBS) 129
Configure storage (FSx for ONTAP) 139
Configure storage (EFS) 169
Configure ENA Express 170
Prerequisites 170
Configure operating system 171
ENA Express settings 172
Check performance 172
Post deployment steps 173
SAP HANA on AWS Operations Guide 174
About this guide 174
Introduction 174
Administration 175
Starting and stopping instances 175
Tagging SAP resources 176
Monitoring 178

Automation ... 178
Patching ... 179
Restoring backups and snapshots ... 193
Automated patching ... 195
SAP references ... 196
Architecture ... 196
Prerequisites ... 114
SSM automation document ... 198
AWS services ... 200
Preparations to run the document ... 205
Troubleshoot ... 205
Version reporting ... 206
Storage configuration ... 207
gp2 and gp3 ... 208
io1, io2, and io2 Block Express ... 235
Root and binaries ... 261
Backup options ... 267
Networking ... 268
EBS-optimized instances ... 268
Elastic network interfaces ... 269
Security groups ... 270
Network configuration for HSR ... 271
Configuration for logical network separation ... 272
SAP support access ... 273
Support channel setup with SAProuter on AWS ... 273
Support channel setup with SAProuter on premises ... 275
Security ... 276
OS hardening ... 277
Disabling HANA services ... 277
API call logging ... 277
Notifications on access ... 277
Architecture patterns for SAP HANA on AWS ... 277
SAP HANA System Replication ... 278
Secondary SAP HANA instance ... 278
Overview of patterns ... 278
Single Region patterns ... 280

Multi-Region patterns 284
High availability and disaster recovery 291
Amazon EC2 recovery options 291
SAP HANA service auto-restart 293
SAP HANA backup/restore 293
AWS Backint Agent for SAP HANA 293
Amazon EBS snapshots 296
Cluster solutions 297
Pacemaker cluster 297
AWS Launch Wizard for SAP 298
AWS Application Migration Service and AWS Elastic Disaster Recovery 300
SAP HANA system replication 300
Testing 317
Troubleshoot 323
Appendix: Configuring Linux to recognize Ethernet devices for multiple network interfaces .. 324
Document history 326
SAP HANA Data Tiering on AWS Overview 328
Overview 328
Prerequisites 328
Specialized Knowledge 328
Technical Requirements 328
SAP Data Tiering 328
Warm Data Tiering Options 330
SAP HANA native storage extension 331
SAP HANA Dynamic Tiering 331
SAP HANA Extension Node 333
Data Aging 334
Cold Data Tiering Options 335
DLM with SAP HANA Spark Controller 336
Cold Tier Options for SAP BW 337
SAP BW Near Line Storage (NLS) with SAP IQ 337
SAP BW NLS with Hadoop 338
SAP BW/4HANA DTO with Data Hub 339
Cold Tier Options for SAP S/4HANA or Suite on HANA 340
SAP ILM with SAP IQ 340
SAP Archiving 341

Document Revisions 343
SAP on AWS High Availability with Overlay IP Address Routing 344
Overview 344
Prerequisites 344
Specialized Knowledge 344
SAP on AWS High Availability Setup 345
Overlay IP Routing using AWS Transit Gateway 346
Architecture 346
Configuration Steps for AWS Transit Gateway 347
Overlay IP Routing with Network Load Balancer 352
Architecture 352
Configuration Steps for Network Load Balancer 353
Additional Implementation Notes 358
Document Revisions 359
SAP HANA on AWS: High Availability Configuration Guide for SLES and RHEL 360
Automated deployment of SAP HANA on AWS with high availability 360
Manual deployment of SAP HANA on AWS with high availability clusters 360
AWS infrastructure, operating system setup and HANA installation 363
Configuring the SAP HANA HA/DR provider hook 366
Cluster configuration prerequisites 369
HA cluster configuration on SLES 374
HA cluster configuration on RHEL 412
High availability cluster and shared Amazon VPC 448
Document history 453

SAP HANA Guides

This section covers the following guides.

- AWS Backint Agent
- Migrating SAP HANA to AWS
- SAP HANA environment setup on AWS
- SAP HANA on AWS operations guide
- SAP HANA Data Tiering on AWS
- SAP on AWS High Availability with Overlay IP Address Routing
- SAP HANA on AWS: High Availability Configuration Guide for SLES and RHEL
- SAP HANA on AWS with Amazon FSx for NetApp ONTAP

Additional SAP on AWS documentation

- General SAP guides
- SAP NetWeaver on AWS
- Databases for SAP applications on AWS
- AWS Launch Wizard for SAP
- AWS Systems Manager for SAP
- AWS SDK for SAP ABAP
- SAP BusinessObjects on AWS
- AWS Migration Hub Orchestrator

AWS Backint Agent for SAP HANA

AWS Backint Agent for SAP HANA (AWS Backint agent) is an SAP-certified backup and restore application for SAP HANA workloads running on Amazon EC2 instances in the cloud. AWS Backint agent runs as a standalone application that integrates with your existing workflows to back up your SAP HANA database to Amazon S3 and AWS Backup. AWS Backint agent restores SAP HANA workloads using SAP HANA Cockpit, SAP HANA Studio, and SQL commands. AWS Backint agent supports full, incremental, and differential backup of SAP HANA databases. Additionally, you can back up log files and catalogs to Amazon S3 and AWS Backup.

AWS Backint agent runs on an SAP HANA database server, where backups and catalogs are transferred from the SAP HANA database to AWS Backint agent. Based on the configurations in your agent file, AWS Backint agent stores your files in Amazon S3 or AWS Backup. To restore your SAP HANA database server, SAP HANA reads the stored catalog files using AWS Backint agent. It then initiates a request to restore the required files.

If you want to deploy an SAP HANA database application with AWS Backint agent, you can use AWS Launch Wizard for SAP, a service that guides you through the sizing, configuration, and deployment of SAP applications on AWS, and follows AWS cloud application best practices.

Topics

- How AWS Backint Agent for SAP HANA works
- Billing
- Supported operating systems
- Supported databases
- Supported Regions
- Backup and restore SAP HANA workloads to Amazon S3
- AWS Backup
- Verify the signature of the AWS Backint agent and installer for SAP HANA
- Troubleshoot AWS Backint Agent for SAP HANA
- Version history

How AWS Backint Agent for SAP HANA works

You can deploy the AWS Backint agent to your SAP HANA instances from the AWS Systems Manager (SSM) console. From the AWS SSM console, an AWS SSM document is executed on the instances to install the agent. You provide the configuration information in the document as parameters. You can also download and manually install and configure the agent. When the agent is installed, you can back up your SAP HANA database to Amazon S3 or AWS Backup.

AWS Backint agent increases scalability through parallel processing of backup and restore processes, providing maximum throughput and reducing backup Recovery Time Objective (RTO) during recovery.

To use AWS Backup with AWS Backint agent, see the following documentation.

- AWS Backup for AWS Backint agent
- AWS Systems Manager for SAP
- AWS Backup

Billing

AWS Backint agent is a free service. You pay for only the underlying AWS services that you use, for example Amazon S3 or AWS Backup. See the following references for more information.

- Amazon S3 pricing
- AWS Backup pricing

Supported operating systems

AWS Backint agent is supported on the following operating systems:

- SUSE Linux Enterprise Server
- SUSE Linux Enterprise Server for SAP
- Red Hat Enterprise Linux for SAP

Supported databases

AWS Backint agent supports the following databases:

- SAP HANA 1.0 SP12 (single node and multi node)
- SAP HANA 2.0 and later (single node and multi node)

Supported Regions

AWS Backint agent is available in all commercial Regions, as well as in China (Beijing), China (Ningxia), and GovCloud.

AWS Backint agent with storage on AWS Backup is available in all commercial Regions.

Backup and restore SAP HANA workloads to Amazon S3

This section provides information about setting up and using AWS Backint agent to backup and restore your SAP HANA workloads to Amazon S3.

Topics

- Prerequisites
- Install and configure AWS Backint Agent for SAP HANA
- Backup and restore your SAP HANA system with the AWS Backint Agent for SAP HANA
- Uninstall

Prerequisites

After your SAP HANA system is successfully running on an Amazon EC2 instance, verify the following prerequisites to install AWS Backint agent using the Amazon EC2 Systems Manager document or using AWS Backint installer.

Topics

- AWS Identity and Access Management
- AWS Systems Manager Agent (SSM Agent)
- Amazon S3 bucket

- AWS CLI

AWS Identity and Access Management

1. To access the AWS resources required to install AWS Backint agent with AWS Systems Manager, you must attach the `AmazonSSMManagedInstanceCore` managed policy to your IAM role.

   > **Note**
   >
   > If you choose to install the AWS Backint agent using the AWS Backint installer, you can skip this step.

2. To allow your Amazon EC2 instance to access your target Amazon S3 bucket, you must create or update an inline IAM policy with the following permissions and attach it to your EC2 service role. Replace the resource names, such as the S3 bucket name, to match your resource name. You must provide the AWS Region and Amazon S3 bucket owner account ID along with the Amazon S3 bucket name.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor1",
            "Effect": "Allow",
            "Action": [
                "s3:GetBucketPolicyStatus",
                "s3:GetBucketLocation",
                "s3:ListBucket",
                "s3:GetBucketAcl",
                "s3:GetBucketPolicy"
            ],
            "Resource": [
                "arn:aws:s3:::<Bucket Name>/*",
                "arn:aws:s3:::<Bucket Name>"
            ]
        },
        {
            "Sid": "VisualEditor2",
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt",
```

```
                "kms:GenerateDataKey"
            ],
            "Resource": "<KMS Arn>"
        },
          {
              "Sid": "VisualEditor0",
              "Effect": "Allow",
              "Action": [
                  "s3:PutObjectTagging",
                  "s3:PutObject",
                  "s3:GetObject",
                  "s3:DeleteObject"
              ],
              "Resource": "arn:aws:s3:::<bucket name>/<folder name>/*"
          }
    ]
}
```

> **Note**
>
> If you want to allow cross-account backup and restore, you must add your account details under a principal element in your policy. For more information about principal policies, see AWS JSON Policy Elements: Principal in the *AWS Identity and Access Management User Guide*. In addition, you must ensure that the S3 bucket policies allow your account to perform the actions specified in the IAM policy example above. For more information, see the example for Bucket owner granting cross-account bucket permissions in the *Amazon S3 Developer Guide*.

For more information about managed and inline policies, see the IAM User Guide.

AWS Systems Manager Agent (SSM Agent)

To install the AWS Backint agent with the AWS Systems Manager Agent (SSM Agent) document, you must install the AWS Systems Manager Agent (SSM Agent) version 2.3.274.0 or later, and your instance must be a managed instance that is configured for AWS Systems Manager. If you want to install AWS Backint agent using AWS Backint installer, you can skip this step. For more information about managed instances, see AWS Systems Manager Managed Instances. To update the SSM Agent, see Update SSM Agent by using Run Command.

> **Note**
>
> The SSM Agent will not work if you do not attach the `AmazonSSMManagedInstanceCore` policy to your EC2 instance role.

Amazon S3 bucket

When you install the AWS Backint agent, you must provide the name of the S3 bucket where you want to store your SAP HANA backups. Only Amazon S3 buckets created after May 2019 are compatible with AWS Backint agent. If you do not own a bucket created after May 2019, create a new S3 bucket in your target Region. Additionally, ensure that the Amazon S3 bucket where you want to store your backups doesn't have public access enabled. If the S3 bucket has public access enabled, backups will fail.

AWS Backint agent supports backing up to Amazon S3 with VPC endpoints. Amazon S3 gateway endpoint can improve performance, and help potentially avoid timeouts. It increases security while reducing cost. For more information, see VPC Endpoints.

S3 storage classes —AWS Backint agent supports backing up your SAP HANA database to an Amazon S3 bucket with the S3 Standard, S3 Standard-IA, S3 One Zone-IA, and S3 Intelligent-Tiering storage classes. S3 Reduced Redundancy, Deep Archive, and Glacier storage classes are not supported by AWS Backint agent. By default, the S3 Standard storage class is used to store your backups. You can change the storage class to use for backups by modifying the AWS Backint agent configuration file. Alternatively, you can change your backup files to one of the supported storage classes through S3 LifeCycle configuration or directly using APIs. To learn more about Amazon S3 storage classes, see Amazon S3 Storage Classes in the *Amazon S3 Developer Guide*.

> **Note**
>
> S3 Intelligent-Tiering storage class enables movement of objects between four access tiers. It can also move objects to the archival tiers. However, **AWS Backint agent for SAP HANA does not support backup and recovery from archival tiers.** To recover or delete objects from the archival tiers, you must first restore the archived S3 objects before initiating a recovery or deletion with the AWS Backint agent.

Encryption— AWS Backint agent supports encrypting your SAP HANA backup files while storing them in Amazon S3, using server-side encryption with AWS KMS (KMS). You can encrypt your backups with a `aws-managed-key` called `aws/s3` or you can use your own custom symmetrical AWS KMS key stored in KMS. To encrypt your backup files with keys stored in KMS (AWS-managed or custom), you must provide the KMS ARN during the install, or update the AWS Backint agent configuration file at a later time. To learn more about encrypting your S3 objects using AWS KMS, see How Amazon S3 uses AWS KMS in the *AWS Key Management Service Developer Guide*. Alternatively, you can enable default encryption for your Amazon S3 bucket using keys managed by Amazon S3. To learn more about enabling default encryption for your bucket, see How do I enable default encryption for an Amazon S3 bucket? in the *Amazon S3 Console User Guide*.

Object locking— You can store objects using a *write-once-read-many* (WORM) model with S3 Object Lock. Use S3 Object Lock if you want to prevent your SAP HANA backup files from being accidentally deleted or overwritten for a specific time period or indefinitely. If S3 Object Lock is enabled, you can't delete your SAP HANA backups stored in Amazon S3 using SAP HANA Cockpit, SAP HANA Studio, or SQL commands until the retention period expires. To learn about S3 Object Lock, see Locking objects using S3 Object Lock in the *Amazon S3 Developer Guide*.

Object tagging — By default, AWS Backint agent adds a tag called `AWSBackintAgentVersion` when it stores your SAP HANA backup files in your S3 bucket. This tag helps to identify the AWS Backint version and the SAP HANA version used when backing up your SAP HANA database. You can list the value of the tags from S3 console or using APIs. To disable default tagging, modify the AWS Backint agent configuration file.

AWS CLI

AWS Backint agent installation leverages the AWS CLI to validate S3 bucket properties. To install or update to the AWS CLI, see Install or update to the latest version of the AWS CLI.

Install and configure AWS Backint Agent for SAP HANA

This section provides information to help you install the AWS Backint agent using an AWS Systems Manager document or AWS Backint installer. It also provides information to help you configure the agent, view logs, and get the current agent version.

Topics

- Install AWS Backint agent using the AWS Systems Manager document
- Install AWS Backint agent using AWS Backint installer — interactive mode

- Install AWS Backint agent using AWS Backint installer — silent mode
- Use a proxy address with AWS Backint agent
- Backint-related SAP HANA parameters
- Modify AWS Backint agent configuration parameters
- Configure SAP HANA to use a different Amazon S3 bucket and folder for data and log backup
- Configure SAP HANA to use a different Amazon S3 bucket and folder for catalog backup
- Configure AWS Backint agent to use shorter Amazon S3 paths
- View AWS Backint agent logs
- Get the currently installed AWS Backint agent version
- Update to the newest version or install a previous version of AWS Backint agent
- Performance tuning
- Subscribe to AWS Backint agent notifications

Install AWS Backint agent using the AWS Systems Manager document

Use the following steps to install the AWS Backint agent using the AWS SSM document.

> **Important**
>
> Disable any existing backup processes (including scheduled log backups) before continuing with the installation. If you don't disable existing backup processes before running the SSM document, you can corrupt an in-progress backup, which can impact your ability to recover your database.

1. From the AWS Management Console, choose **Systems Manager** under **Management & Governance**, or enter **`Systems Manager`** in the **Find Services** search bar.
2. From the Systems Manager console, choose **Documents** under **Shared Resources** in the left navigation pane.
3. On the Documents page, select the **Owned by Amazon** tab. You should see a document named **AWSSAP-InstallBackint**.
4. Select the **AWSSAP-InstallBackint** document and choose **Run command**.
5. Under the Command parameters, enter the following

a. **Bucket Name**. Enter the name of the Amazon S3 bucket where you want to store your SAP HANA backup files.

b. **Bucket Folder**. Optionally, enter the name of the folder within your Amazon S3 bucket where you want to store your SAP HANA backup files.

c. **System ID**. Enter your SAP HANA System ID, for example **HDB**.

d. **Bucket Region**. Enter the AWS Region of the Amazon S3 bucket where you want to store your **SAP HANA backup files**. AWS Backint agent supports cross-Region and cross-account backups. You must provide the AWS Region and Amazon S3 bucket owner account ID along with the Amazon S3 bucket name for the agent to perform successfully.

e. **Bucket Owner Account ID**. Enter the account ID of the Amazon S3 bucket where you want to store your SAP HANA backup files.

f. **Kms Key**. Enter the ARN of AWS KMS that AWS Backint agent can use to encrypt the backup files stored in your Amazon S3 bucket.

g. **Installation Directory**. Enter the path of the directory location where you want to install the AWS Backint agent. Avoid using `/tmp` as the install path.

h. **Agent Version**. Enter the version number of the agent that you want to install. If you do not enter a version number, the latest published version of the agent is installed.

> **Note**
>
> 1.0 versions are unavailable in the GovCloud Regions.

i. **Modify Global ini file**. Choose how you want to modify the `global.ini` file. The `global.ini` file of the SAP HANA SYSTEM DB must be updated to complete the setup.

   i. “modify” — SSM will update the `global.ini` file directly.

   ii. “sql” — SSM will create a file called `modify_global_ini.sql` with SQL statements that you can run in your target SAP HANA system to set the required parameters. You can find the `modify_global_ini.sql` file in the *<installation directory>*`/aws-backint-agent/` folder.

   iii. “none” — No action will be taken by SSM to modify the `global.ini` file. You must manually update it to complete the setup.

j. **Ignore Bucket Checks**. Select **yes** to ignore sanity checks of the S3 bucket. S3 Bucket sanity checks verify the following:

- the bucket exists in your account
- the bucket Region is correct
- the bucket is public

k. **Debug Mode**. Select **yes** to activate debug mode.

l. **Important! Ensure No Backup In Process**. Choose **Yes** to confirm that you have disabled existing backups and are ready to proceed with the installation. **The SSM document will fail if you choose "No"**.

6. Under **Targets**, select the method for your target instance to use to install the AWS Backint agent, and then choose the instance on which to install it. If you are not able to find your instance in the list, verify that you have followed all of the steps in the prerequisites.

7. Under **Other parameters**, leave the field empty and choose **Run**.

> **Important**
>
> If you do not have the latest version of the SSM Agent installed (2.3.274.0 or later), **Run Command** will fail to execute.

8. When the agent is successfully installed, you will see the **Success** status under the **Command ID**.

9. To verify the installation, log in to your instance and view the `/<install directory>/aws-backint-agent` directory. You should see the following files in the directory: the AWS Backint agent binary, `THIRD_PARTY_LICENSES.txt` file, which contains licenses of libraries used by the agent, the launcher script, the YAML configuration file, and the optional `modify_global_ini.sql` file. In addition, a source file (`aws-backint-agent.tar.gz`) of AWS Backint agent is stored in the package directory. You can verify the signature of this file to ensure that the downloaded source file is original and unmodified. See the Verifying the signature of AWS Backint agent and installer for SAP HANA section in this document for details.

   The SSM document creates symbolic links (symlinks) in the SAP HANA global directory for the Backint configuration. Verify that the symlink for `hdbbackint` exists in the `/usr/sap/<SID>/SYS/global/hdb/opt` directory and the symlink for `aws-backint-agent-config.yaml` exists in the `/usr/sap/<SID>/SYS/global/hdb/opt/hdbconfig` directory.

Install AWS Backint agent using AWS Backint installer — interactive mode

Another way to install the AWS Backint agent is with the AWS Backint installer. You can download the AWS Backint installer from an Amazon S3 bucket.

The name of the S3 bucket is `s3://awssap-backint-agent/`

> **Note**
>
> For AWS GovCloud (US-East), the name of the S3 bucket is `s3://awssap-backint-agent-us-gov-east-1`.
> For AWS GovCloud (US-West), the name of the S3 bucket is `s3://awssap-backint-agent-us-gov-west-1`.

The latest installer can always be found at `s3://awssap-backint-agent/binary/latest/install-aws-backint-agent`

> **Note**
>
> For AWS GovCloud (US-East), the latest installer can always be found at `s3://awssap-backint-agent-us-gov-east-1/binary/latest/install-aws-backint-agent`.
> For AWS GovCloud (US-West), the latest installer can always be found at `s3://awssap-backint-agent-us-gov-west-1/binary/latest/install-aws-backint-agent`.

Follow these steps to install AWS Backint agent using the AWS Backint installer from an SSH session on your SAP HANA instance.

> **Important**
>
> Disable any existing backup processes (including scheduled log backups) before continuing with the installation. If you don't disable existing backup processes before running the AWS Backint agent installer, you can corrupt an in-progress backup, which can impact your ability to recover your database.

1. Navigate to `/tmp` (or another temporary directory where you downloaded the installer).

```
cd /tmp
```

2. Run one of the following commands to download the installer.

```
sudo aws s3 cp s3://awssap-backint-agent/binary/latest/install-aws-backint-agent /
tmp/ --region us-east-1
```

or

```
sudo wget https://s3.amazonaws.com/awssap-backint-agent/binary/latest/install-aws-
backint-agent -O /tmp/install-aws-backint-agent
```

> **Note**
>
> If you encounter permission issues while downloading the AWS Backint installer using the AWS CLI, check your IAM policy and ensure that your policies allow for downloading objects from the `awssap-backint-agent` bucket. See the Identity and Access Management section of this documentation for details.

3. (Optional) For AWS GovCloud (US-East) and AWS GovCloud (US-West), run one of the following commands to download the installer.

```
sudo aws s3 cp s3://awssap-backint-agent-us-gov-east-1/binary/latest/install-aws-
backint-agent /tmp/ --region us-gov-east-1
```

```
sudo aws s3 cp s3://awssap-backint-agent-us-gov-west-1/binary/latest/install-aws-
backint-agent /tmp/ --region us-gov-west-1
```

or

```
sudo wget https://awssap-backint-agent-us-gov-east-1.s3.us-gov-
east-1.amazonaws.com/binary/latest/install-aws-backint-agent -O /tmp/install-aws-
backint-agent
```

```
sudo wget https://awssap-backint-agent-us-gov-west-1.s3.us-gov-
west-1.amazonaws.com/binary/latest/install-aws-backint-agent -O /tmp/install-aws-
backint-agent
```

4. Run the installer with the `-h` flag to find all of the available options.

```
sudo python install-aws-backint-agent -h
```

5. Run the following command to execute the installer.

```
sudo python install-aws-backint-agent
```

> **Note**
>
> Run the installer with the `-l` flag if you want the installer to get the AWS Backint agent binary file from your own file system or Amazon S3 bucket. Specify the location of the `aws-backint-agent.tar.gz` file.
>
> ```
> sudo python install-aws-backint-agent -l /tmp/backint/aws-backint-agent.tar.gz
> ```
>
> ```
> sudo python install-aws-backint-agent -l s3://AWSDOC-EXAMPLE-BUCKET/aws-backint-agent.tar.gz
> ```
>
> ```
> sudo python install-aws-backint-agent -l https://AWSDOC-EXAMPLE-BUCKET.s3.amazonaws.com/aws-backint-agent.tar.gz
> ```

6. Enter information for the following parameters.

   a. **Installation directory** — Enter the path of the directory location where you want to install the AWS Backint agent. The default value for the installation directory is `/hana/shared/`.

   b. **Amazon S3 bucket owner** — Enter the account ID of the Amazon S3 bucket owner of the bucket where you want to store your SAP HANA backup files.

   c. **Amazon S3 bucket Region** — Enter the AWS Region of the Amazon S3 bucket where you want to store your SAP HANA backup files.

   d. **Amazon S3 bucket name** — Enter the name of the Amazon S3 bucket where you want to store your SAP HANA backup files.

   e. **Folder in the S3 bucket** — Enter the name of the folder in the Amazon S3 bucket where you want to store your SAP HANA backup files. This parameter is optional.

f. **Amazon S3 SSE KMS ARN** — Enter the ARN of the AWS KMS that AWS Backint agent can use to encrypt the backup files stored in your Amazon S3 bucket.

> **Note**
>
> If you leave this field empty, AWS Backint installer will prompt you to confirm that you don't want to encrypt your backup files with encryption keys stored in AWS KMS. If you do not confirm that you do not want to encrypt with the kms-key, the installer will abort. We strongly recommend that you encrypt your data.

g. **SAP HANA system ID** — Enter your SAP HANA System ID, for example HDB.

h. **HANA opt dir** — Confirm the location of the SAP HANA opt directory.

i. **Modify global.ini [modify/sql/[none]]** — Choose how you want to modify the `global.ini` file. The `global.ini` file of the SAP HANA SYSTEM must be updated to complete the setup.

   i. "modify" — AWS Backint installer will update the `global.ini` file directly.

   ii. "sql" — AWS Backint installer will create a file called `modify_global_ini.sql` with SQL statements that you can run in your target SAP HANA system to set the required parameters. You can find the `modify_global_ini.sql` file in the *<installation directory>*`/aws-backint-agent/` folder.

   iii. "none" — No action will be taken by AWS Backint installer to modify the `global.ini` file. You must manually update them to complete the setup.

j. **HANA SYSTEM db global.ini file** — Confirm the location of `global.ini` file.

k. **Verify signature of the agent binary `.tar` file** —

- Choose `y` to verify the signature of the AWS Backint agent source file. If you choose `y`, enter the Amazon S3 bucket location of the signature file of the agent binary `.tar` file, for example, `https://s3.amazonaws.com/awssap-backint-agent/binary/latest/aws-backint-agent.sig`. Or, provide a local file that is stored on the instance. If you proceed without making a selection, the default location listed within brackets ([]) is used.
- Choose `n` if you do not want to verify the signature of the AWS Backint agent source file.

l. **Save responses for future usage?** — You can save your information for the AWS Backint installer to a file. You can then use it later to run the installer in silent mode, if needed.

m. **Do you want to proceed with the installation?** — Confirm that you have disabled the existing backups and are ready to proceed with the installation.

7. To verify the installation, log in to your instance and view the */<install directory>/aws-backint-agent* directory. You should see the following files in the directory: the AWS Backint agent binary, the `THIRD_PARTY_LICENSES.txt` file, which contains licenses of libraries used by the agent, the launcher script, the YAML configuration file, and the optional `modify_global_ini.sql` file. In addition, a source file (`aws-backint-agent.tar.gz`) of AWS Backint agent is stored in the package directory. You can verify the signature of this file to ensure that the downloaded source file is original and unmodified. See the Verifying the signature of AWS Backint agent and installer for SAP HANA section in this document for details.

   In addition, the AWS Backint installer creates symbolic links (symlinks) in the SAP HANA global directory for the Backint configuration. Verify that the symlink for `hdbbackint` exists in the */usr/sap/<SID>/SYS/global/hdb/opt* directory, and that the symlink for `aws-backint-agent-config.yaml` exists in the */usr/sap/<SID>/SYS/global/hdb/opt/hdbconfig* directory.

   > **Note**
   >
   > If your installation fails due to validation errors and you want to ignore the validation and proceed with the installation, you can execute the installer with the `-n` flag to ignore the validation steps. You can also use the `-d` flag to run the installer in debug mode to generate detailed installation logs for troubleshooting.

Install AWS Backint agent using AWS Backint installer — silent mode

You can install the AWS Backint agent with the AWS Backint installer in a silent mode. Choose this option if you want the installation process to be automated without manual intervention.

To run the AWS Backint installer in silent mode, create a response file with all of the required installation parameters. Follow the steps in the section on installing using the interactive mode to download AWS Backint installer and create a response file. You don't have to confirm to continue with the AWS Backint agent installation in interactive mode. AWS Backint installer will create a response file called `aws-backint-agent-install-YYYYMMDDHHMMSS.rsp`.

When you have a response file, you can modify it with a vim editor and adjust the parameters as needed.

The following is an example response file.

```
[DEFAULT]
s3_bucket_name = awsdoc-example-bucket
s3_bucket_owner_account_id = 111122223333
modify_global_ini = sql
s3_bucket_region = us-east-1
s3_sse_kms_arn = arn:aws:kms:us-east-1:111122223333:key/1abcd9b9-
ab12-1a2a-1abc-12345abc12a3
s3_bucket_folder = myfolder
hana_sid = TST
installation_directory = /hana/shared/
```

If you want to generate the response file programmatically instead of using AWS Backint installer in interactive mode, you can use the `-g` flag to generate a new response file. The following is an example of how to generate a response file using AWS Backint installer.

```
sudo python install-aws-backint-agent -g "s3_bucket_owner_account_id =
 111122223333,s3_bucket_name = awsdoc-example-bucket,s3_bucket_region = us-
east-1,hana_sid = TST,s3_sse_kms_arn = arn:aws:kms:us-east-1:111122223333:key/1abcd9b9-
ab12-1a2a-1abc-12345abc12a3,s3_bucket_folder = myfolder,installation_directory = /hana/
shared/,modify_global_ini = sql" -f myresponse.rsp
```

After the response file is created, use the following steps to run AWS Backint installer in silent mode.

> **Important**
>
> Disable any existing backup processes (including scheduled log backups) before continuing with the installation. If you don't disable existing backup processes before running the AWS Backint agent installer, you can corrupt an in-progress backup, which can impact your ability to recover your database.

Run the following command to execute the installer using the generated response file.

```
sudo python install-aws-backint-agent -m silent -f backint-agent-install-
YYYYMMDDHHMMSS.rsp -a yes
```

If you want to choose the location from which to install the agent, run the command with the `-l` flag and specify the location.

```
sudo python install-aws-backint-agent -f aws-backint-agent-install-YYYYMMDDHHMMSS.rsp -
m silent -a yes -d -l /tmp/backint/aws-backint-agent.tar.gz
```

> **Note**
>
> You must confirm that you have disabled the existing backups and are ready to proceed with the installation in silent mode by passing an acknowledgement flag (`-a yes`). If you don't pass the acknowledgement flag, AWS Backint installer will fail to execute.

Use a proxy address with AWS Backint agent

If you use a proxy address in your SAP HANA environment when you install the agent, you must use the following shell script to install the agent to ensure that the correct proxy settings are used by the AWS Backint agent installer.

```
#!/bin/bash
export https_proxy=<PROXY_ADDRESS>:<PROXY_PORT>
export HTTP_PROXY=<PROXY_ADDRESS>:<PROXY_PORT>
export no_proxy=169.254.169.254
export NO_PROXY=169.254.169.254
sudo python install-aws-backint-agent
```

If you use a proxy address in your SAP HANA environment, you must update the `aws-backint-agent-launcher.sh` file, which is located in the AWS Backint agent installation directory (for example, `/hana/shared/aws-backint-agent/`). You must perform the following update to ensure that the correct proxy settings are used by AWS Backint agent during backup and restore operations.

Add `http_proxy`, `HTTP_PROXY`, `no_proxy`, and `NO_PROXY` variables to the `aws-backint-agent-launcher.sh` script. It is important to exclude the `196.254.169.254` address with the `no_proxy` variable. If you do not exclude this address, instance metadata service calls made

by AWS Backint agent will fail and cause errors during backup and restore operations. For more information about instance metadata and user data, see Instance metadata and user data in the *Amazon EC2 User Guide for Linux Instances*.

```
#!/bin/bash
export https_proxy=<PROXY_ADDRESS>:<PROXY_PORT>
export HTTP_PROXY=<PROXY_ADDRESS>:<PROXY_PORT>
export no_proxy=169.254.169.254
export NO_PROXY=169.254.169.254
/hana/shared/aws-backint-agent/aws-backint-agent "$@"
```

Backint-related SAP HANA parameters

To enable SAP HANA backups using AWS Backint agent, you must set the following SAP HANA parameters. If you chose the “modify” option for the `global.ini` file update, the SSM document or AWS Backint installer adds or updates the following backup related SAP HANA parameters in `global.ini` for the system database. If you chose “sql”, you can run the SQL statements specified in the `modify_global_ini.sql` file to update these parameters. For more details about these parameters, see Backup Configuration Parameters in the *SAP HANA Administration Guide*.

```
[backup]
catalog_backup_parameter_file = /usr/sap/<SID>/SYS/global/hdb/opt/hdbconfig/aws-
backint-agent-config.yaml
data_backup_parameter_file = /usr/sap/<SID>/SYS/global/hdb/opt/hdbconfig/aws-backint-
agent-config.yaml
log_backup_parameter_file = /usr/sap/<SID>/SYS/global/hdb/opt/hdbconfig/aws-backint-
agent-config.yaml
catalog_backup_using_backint = true
log_backup_using_backint = true
parallel_data_backup_backint_channels = 8
data_backup_buffer_size = 4096
max_recovery_backint_channels = 1
[communication]
tcp_backlog = 2048
[persistence]
enable_auto_log_backup = yes
verify_signature = yes
input_signature_filepath = https://s3.amazonaws.com/awssap-backint-agent/binary/latest/
aws-backint-agent.sig
```

Note

Changing the `tcp_backlog` parameter requires a restart of SAP HANA to take effect. `max_recovery_backint_channels` determines the number of log files restored/recovered in parallel during the recovery process. When multistreamed backups are recovered, SAP HANA always uses the same number of channels that were used during the backup. For more information, see Multistreaming Data Backups with Third-Party Backup Tools in the SAP documentation.

Modify AWS Backint agent configuration parameters

The AWS Backint agent configuration parameters are maintained in a YAML file in the `/<installation directory>/aws-backint-agent/` directory. The name of the configuration file is `aws-backint-agent-config.yaml`. The following tables summarize the configuration parameters added as part of the AWS Backint agent installation process, and additional parameters that you can add or change.

Parameters added to the `aws-backint-agent-config.yaml` during initial setup

Name of the parameter	Description	Default value
`S3BucketName`	Name of the Amazon S3 bucket where you want to store your SAP HANA backup files. For example, `awsdoc-example-bucket` .	N/A
`S3BucketAwsRegion`	AWS Region of your Amazon S3 bucket. For example, `us-east-1` .	N/A
`S3BucketFolder`	Name of the folder in the Amazon S3 bucket where you want to store your SAP HANA backup files. For example, `my-folder` .	Empty

Name of the parameter	Description	Default value
`S3BucketOwnerAccountID`	12-digit account ID of the Amazon S3 bucket owner. For example, `111122223333` .	N/A
`LogFile`	Location of the AWS Backint agent log file.	`/hana/shared/aws-backint-agent/aws-backint-agent.log`
`S3SseKmsArn`	ARN of the kms-key that AWS Backint agent can use to encrypt the backup files stored in Amazon S3. For example, `arn:aws:kms:us-east-1:111122223333:key/5bfbc9b9-ab12-ab12-a123-11111xxx22xx` .	Empty
`S3SseEnabled`	Specifies whether KMS encryption is enabled.	Set to `false` if the `S3SseKmsArn` parameter is empty. Otherwise, set to `true`.

Parameters that can be added to the `aws-backint-agent-config.yaml` file to update default values

Name of the parameter	Description	Default value	Supported since
`BackupObjectTags`	Enables support for additional S3 object tags. `EnableTagging` must be set to	N/A	Version 1.03

Name of the parameter	Description	Default value	Supported since
	`true` in order to use `BackupObjectTags` . Allowed values: must be a valid JSON string that uses the following syntax: `-BackupObjectTags: "[{Key=string,Value=string},{Key=string,Value=string},...]` For applicable tag restrictions, see Tag restrictions in the *Amazon EC2 User Guide*.		

Name of the parameter	Description	Default value	Supported since
`EnableTagging`	Enables or disables default object tagging for backups files stored in S3. Tagging helps to identify the AWS Backint version and SAP HANA version used during the backup. Allowed values: `true` or `false`.	`true`	Version 1.03
`LogLevel`	Specifies the logging level for agent logs. Allowed values: `info` or `debug`.	`info`	Version 1.0
`LogRotationFrequency`	Specifies the `aws-backint-agent.log` file rotation frequency. Allowed values: `minute`, `hour`, `day`, or `never`.	`never`	Version 1.03

Name of the parameter	Description	Default value	Supported since
`S3StorageClass`	Specifies the S3 storage class type that AWS Backint agent can use while storing your backup files. Allowed values: `STANDARD`, `STANDARD_IA` , `ONEZONE_IA` , or `INTELLIGENT_TIERING` .	`STANDARD`	Version 1.0 (Intelligent-Tiering since version 1.05)
`UploadConcurrency`	Specifies the number of Amazon S3 threads that can work in parallel during backup. Allowed values: 1 to `200`.	`100`	Version 1.0
`UploadChannelSize`	Specifies the number of files that can be uploaded in parallel to the S3 bucket during the backups. Allowed values: 1 to `32`.	`10`	Version 1.0

Name of the parameter	Description	Default value	Supported since
`MaximumConcurrentFilesForRestore`	Specifies the number of files that can be downloaded in parallel from S3 during the restore. Allowed values: 1 to 32.	5	Version 1.0
`S3ShortenBackupDestinationEnabled`	Specifies whether to use a shorter Amazon S3 path. Allowed values: `true` or `false`.	`false`	Version 1.05
`DownloadConcurrency`	Specifies the number of Amazon S3 threads that can work in parallel during restore Allowed values: 1 to 200.	100	Version 1.0

Configure SAP HANA to use a different Amazon S3 bucket and folder for data and log backup

AWS Backint agent uses the same parameters by default for the data and log backups. It stores the data and log backups in the same Amazon S3 bucket and folder.

```
data_backup_parameter_file = /usr/sap/<SID>/SYS/global/hdb/opt/hdbconfig/aws-backint-agent-config.yaml
log_backup_parameter_file = /usr/sap/<SID>/SYS/global/hdb/opt/hdbconfig/aws-backint-agent-config.yaml
```

To use a different Amazon S3 bucket and folder for the data and log backups, follow these steps.

1. **Check the SAP HANA backup parameters**

   Locate the `data_backup_parameter_file` and `log_backup_parameter_file` parameters. The default value of these parameters should be `/<installation directory>/aws-backint-agent/aws-backint-agent-config.yaml`. If you do not see this default value, check the configuration file to confirm that it is displaying the same Amazon S3 location.

2. **Retain access to the logs backup stored in the previous Amazon S3 location**

   If this is a new setup or you do not want to retain the previous logs backup, skip this step and continue with Step 3.

   Move the previous logs backup with source type `volume` to the new Amazon S3 location for logs backup only. You can confirm the source type by running the following SQL command.

   ```
   select SOURCE_TYPE_NAME, DESTINATION_PATH from M_BACKUP_CATALOG_FILES
   ```

   The backup catalog is assigned a name in the following format: `log_backup_0_0_0_0.<BackupID>`. This type of backup is managed by a different SAP HANA parameter, has a source type `catalog`, and should remain in the data backup location. This file contains the backup catalog file that stores the history of all backups. Only the log backups with source type `volume` should be moved to the new Amazon S3 location. To change the Amazon S3 location for catalog backup, see the section called “Configure SAP HANA to use a different Amazon S3 bucket and folder for catalog backup”.

   The following table provides an example of a SYSTEM DB folder structure:

Backup folder	Descriptions
COMPLETE_DATA_BACKUP_databackup_0_1/	Nameserver data backup with the source type "topology"
COMPLETE_DATA_BACKUP_databackup_1_1/	Nameserver data backup with the source type "volume"
log_backup_0_0_0_0/	Log file with source type "catalog"

Backup folder	Descriptions
log_backup_1_0_*<backup ID>*_*<backup ID>*	Log file with source type "volume"

The following table provides an example of a TENANT DB folder structure:

Backup folder	Descriptions
COMPLETE_DATA_BACKUP_databackup_0_1/	Indexserver data backup with the source type "topology"
COMPLETE_DATA_BACKUP_databackup_2_1/	Indexserver data backup with the source type "volume"
COMPLETE_DATA_BACKUP_databackup_3_1/	Xsengine data backup with the source type "volume"
log_backup_0_0_0_0/	Log file with source type "catalog"
log_backup_2_0_*<backup ID>*_*<backup ID>*	Log file with source type "volume"
log_backup_3_0_*<backup ID>*_*<backup ID>*	Log file with source type "volume"

Note

Before doing steps a and b, ensure that there is no backup process running.

a. **Change the location of the logs backup for SYSTEM DB**

Run the following commands to move the volume type of SYSTEM DB logs. In the example, we use the same Amazon S3 bucket, but create another folder for the logs backup.

```
# Create the folder structure
aws s3api put-object --bucket <S3 bucket> --key <S3 folder for logs>/<SID>/usr/
sap/<SID>/SYS/global/hdb/backint/SYSTEMDB/ --region us-east-1

# Execute a Dry Run to check
aws s3 cp s3://<S3 bucket>/<S3 folder>/<SID>/usr/sap/<SID>/SYS/global/hdb/
backint/SYSTEMDB/ s3://<S3 bucket>/<S3 folder for logs>/<SID>/usr/sap/<SID>/
SYS/global/hdb/backint/SYSTEMDB/ --exclude "*" --include "log_backup_1_0*" --
recursive --dryrun --region us-east-1

# Run the command to move the logs to the new S3 location
aws s3 cp s3://<S3 bucket>/<S3 folder>/<SID>/usr/sap/<SID>/SYS/global/hdb/
backint/SYSTEMDB/ s3://<S3 bucket>/<S3 folder for logs>/<SID>/usr/sap/<SID>/
SYS/global/hdb/backint/SYSTEMDB/ --exclude "*" --include "log_backup_1_0*" --
recursive --region us-east-1

# Check the output of the S3 location for logs
aws s3 ls s3://<S3 bucket>/<S3 folder for logs>/<SID>/usr/sap/<SID>/SYS/global/
hdb/backint/SYSTEMDB/ --region us-east-1
```

b. **Change the location of the logs backup for TENANT DB**

Run the following commands to move the volume type TENANT DB logs. In the example, we use the same Amazon S3 bucket, and create another folder for the logs backup. You need to repeat this step for every TENANT DB.

```
#Create the folder structure
aws s3api put-object --bucket <S3 bucket> --key <S3 folder for logs>/<SID>/usr/
sap/<SID>/SYS/global/hdb/backint/DB_<SID>/ --region us-east-1

#Execute a Dry Run
aws s3 cp s3://<S3 bucket>/<S3 folder>/<SID>/usr/sap/<SID>/SYS/global/hdb/
backint/DB_<SID>/ s3://<S3 bucket>/<S3 bucket for logs>/<SID>/usr/sap/<SID>/
SYS/global/hdb/backint/DB_<SID>/ --exclude "" --include "log_backup_2_0" --
include "log_backup_3_0" --recursive --dryrun --region us-east-1

#Run the command to move the logs to the new S3 location
aws s3 cp s3://<S3 bucket>/<S3 folder>/<SID>/usr/sap/<SID>/SYS/global/hdb/
backint/DB_<SID>/ s3://<S3 bucket>/<S3 bucket for logs>/<SID>/usr/sap/<SID>/
SYS/global/hdb/backint/DB_<SID>/ --exclude "" --include "log_backup_2_0" --
include "log_backup_3_0" --recursive --region us-east-1
```

```
#Check the output of the S3 location for logs
aws s3 ls s3://<S3 bucket>/<S3 bucket for logs>/<SID>/usr/sap/<SID>/SYS/global/
hdb/backint/DB_<SID>/ --region us-east-1
```

3. **Create the `aws-backint-agent-config-logs.yaml` parameter file**

   a. Make a copy of the existing AWS Backint agent configuration for logs backup.

   ```
   cp /hana/shared/aws-backint-agent/aws-backint-agent-config.yaml  \
   /hana/shared/aws-backint-agent/aws-backint-agent-config-logs.yaml
   ```

   b. Modify the `S3BucketName`, `S3BucketFolder`, and `LogFile` parameters in `aws-backint-agent-config-logs.yaml`, using your preferred editor.

   ```
   S3BucketName: "<Amazon S3 bucket for SAP HANA logs>"
   S3BucketFolder: "<Amazon S3 folder for SAP HANA logs>"
   LogFile: "/hana/shared/aws-backint-agent/aws-backint-agent-logs.log"
   ```

   c. Create a `hdbbackint` soft link from `/usr/sap/<SID>/SYS/global/hdb/opt/hdbconfig/` to `/hana/shared/aws-backint-agent/`.

   ```
   ln -s /hana/shared/aws-backint-agent/aws-backint-agent-config-logs.yaml  \
   /usr/sap/<SID>/SYS/global/hdb/opt/hdbconfig/aws-backint-agent-config-logs.yaml
   ```

4. **Run reconfiguration for the update**

   Run `hdbnsutil -reconfig` for the update to take effect.

5. **Validate to ensure that all steps have been processed correctly**

   a. Run a point-in-time recovery to a previous state, to ensure that you can access the previous log files in the new Amazon S3 location.

   b. Verify that new logs are uploaded to the new S3 location.

6. **Delete previous backups**

   After a successful validation, we recommend waiting for at least a week before deleting the previous logs.

   When you're ready, delete the previous logs with the following commands.

   ```
   # Delete previous backups in SYSTEMDB
   ```

```
aws s3 rm s3://<S3 bucket>/<S3 folder>/<SID>/usr/sap/<SID>/SYS/global/hdb/backint/
SYSTEMDB/ --exclude "" --include "log_backup_1_0" --recursive --dryrun --region us-
east-1
aws s3 rm s3://<S3 bucket>/<S3 folder>/<SID>/usr/sap/<SID>/SYS/global/hdb/backint/
SYSTEMDB/ --exclude "" --include "log_backup_1_0" --recursive --region us-east-1

# Delete previous backups in the TENANT database (Repeat for each tenant)
aws s3 rm s3://<S3 bucket>/<S3 folder>/<SID>/usr/sap/<SID>/SYS/global/hdb/
backint/DB_<SID>/ --exclude "" --include --include "log_backup_2_0" --include
 "log_backup_3_0" --recursive --dryrun --region us-east-1
aws s3 rm s3://<S3 bucket>/<S3 folder>/<SID>/usr/sap/<SID>/SYS/global/hdb/backint/
DB_<SID>/ --exclude "" --include "log_backup_2_0" --include "log_backup_3_0" --
recursive --region us-east-1
```

Configure SAP HANA to use a different Amazon S3 bucket and folder for catalog backup

AWS Backint agent uses the same parameters by default for the data, log, and catalog backups. It stores all of the backups in the same Amazon S3 bucket and folder.

```
data_backup_parameter_file = /usr/sap/<SID>/SYS/global/hdb/opt/hdbconfig/aws-backint-
agent-config.yaml
log_backup_parameter_file = /usr/sap/<SID>/SYS/global/hdb/opt/hdbconfig/aws-backint-
agent-config.yaml
catalog_backup_parameter_file = /usr/sap/<SID>/SYS/global/hdb/opt/hdbconfig/aws-
backint-agent-config.yaml
```

To use a different Amazon S3 bucket and folder for catalog backup, follow these steps.

1. **Check the SAP HANA backup parameters**

   Locate the `data_backup_parameter_file`, `log_backup_parameter_file`, and `catalog_backup_parameter_file` parameters. The default value of these parameters should be */<installation directory>*`/aws-backint-agent/aws-backint-agent-config.yaml`. If you do not see this default value, check the configuration file to confirm that it is displaying the same Amazon S3 location.
2. **Retain access to the logs backup stored in the previous Amazon S3 location**

   If this is a new setup or you do not want to retain the previous catalog backup, skip this step and continue with Step 3.

Move the previous catalog backup with source type `catalog` to the new Amazon S3 location for catalog backup only. You can confirm the source type by running the following SQL command.

```
select SOURCE_TYPE_NAME, DESTINATION_PATH from M_BACKUP_CATALOG_FILES
```

The backup catalog is assigned a name in the following format: `log_backup_0_0_0_0.<BackupID>`. This type of backup has a source type `catalog`. This file contains the backup catalog file that stores the history of all backups. Only the catalog backups with source type `catalog` should be moved to the new Amazon S3 location. To change the Amazon S3 location for log backup, see the section called “Configure SAP HANA to use a different Amazon S3 bucket and folder for data and log backup”.

The following table provides an example of a SYSTEM DB folder structure:

Backup folder	Descriptions
COMPLETE_DATA_BACKUP_databackup_0_1/	Nameserver data backup with the source type "topology"
COMPLETE_DATA_BACKUP_databackup_1_1/	Nameserver data backup with the source type "volume"
log_backup_0_0_0_0/	Log file with source type "catalog"
log_backup_1_0_*<backup ID>*_*<backup ID>*	Log file with source type "volume"

The following table is an example of a TENANT DB folder structure:

Backup folder	Descriptions
COMPLETE_DATA_BACKUP_databackup_0_1/	Indexserver data backup with the source type "topology"

Backup folder	Descriptions
COMPLETE_DATA_BACKUP_databackup_2_1/	Indexserver data backup with the source type "volume"
COMPLETE_DATA_BACKUP_databackup_3_1/	Xsengine data backup with the source type "volume"
log_backup_0_0_0_0/	Log file with source type "catalog"
log_backup_2_0_*<backup ID>*_*<backup ID>*	Log file with source type "volume"
log_backup_3_0_*<backup ID>*_*<backup ID>*	Log file with source type "volume"

Note

Before doing steps a and b, ensure that there is no backup process running.

a. **Change the location of the catalog backup for SYSTEM DB**

Run the following commands to move the `catalog` type of SYSTEM DB logs. In the example, we use the same Amazon S3 bucket, but create another folder for catalog backup.

```
# Create the folder structure
aws s3api put-object --bucket <S3 bucket> --key <S3 folder for catalog>/<SID>/
usr/sap/<SID>/SYS/global/hdb/backint/SYSTEMDB/ --region us-east-1

# Execute a Dry Run to check
aws s3 cp s3://<S3 bucket>/<S3 folder>/<SID>/usr/sap/<SID>/SYS/global/hdb/
backint/SYSTEMDB/ s3://<S3 bucket>/<S3 folder for catalog>/<SID>/usr/sap/<SID>/
SYS/global/hdb/backint/SYSTEMDB/ --exclude "*" --include "log_backup_0_0_0_0*"
 --recursive --dryrun --region us-east-1

# Run the command to move the logs to the new S3 location
```

```
aws s3 cp s3://<S3 bucket>/<S3 folder>/<SID>/usr/sap/<SID>/SYS/global/hdb/
backint/SYSTEMDB/ s3://<S3 bucket>/<S3 folder for catalog>/<SID>/usr/sap/<SID>/
SYS/global/hdb/backint/SYSTEMDB/ --exclude "*" --include "log_backup_0_0_0_0*"
 --recursive --region us-east-1

# Check the output of the S3 location for logs
aws s3 ls s3://<S3 bucket>/<S3 folder for catalog>/<SID>/usr/sap/<SID>/SYS/
global/hdb/backint/SYSTEMDB/ --region us-east-1
```

b. **Change the location of the catalog backup for TENANT DB**

Run the following commands to move the `catalog` type tenant database logs. In the example, we use the same Amazon S3 bucket, and create another folder for catalog backup. You need to repeat this step for every TENANT DB.

```
#Create the folder structure
aws s3api put-object --bucket <S3 bucket> --key <S3 folder for catalog>/<SID>/
usr/sap/<SID>/SYS/global/hdb/backint/DB_<SID>/ --region us-east-1

#Execute a Dry Run
aws s3 cp s3://<S3 bucket>/<S3 folder>/<SID>/usr/sap/<SID>/SYS/global/hdb/
backint/DB_<SID>/ s3://<S3 bucket>/<S3 bucket for catalog>/<SID>/usr/sap/<SID>/
SYS/global/hdb/backint/DB_<SID>/ --exclude "" --include "log_backup_0_0_0_0*"
 --recursive --dryrun --region us-east-1

#Run the command to move the catalog to the new S3 location
aws s3 cp s3://<S3 bucket>/<S3 folder>/<SID>/usr/sap/<SID>/SYS/global/hdb/
backint/DB_<SID>/ s3://<S3 bucket>/<S3 bucket for catalog>/<SID>/usr/sap/<SID>/
SYS/global/hdb/backint/DB_<SID>/ --exclude "" --include "log_backup_0_0_0_0*"
 --recursive --region us-east-1

#Check the output of the S3 location for catalog
```

3. **Create the `aws-backint-agent-config-catalog.yaml` parameter file**

a. Make a copy of the existing AWS Backint agent configuration for catalog backup.

```
cp /hana/shared/aws-backint-agent/aws-backint-agent-config.yaml  \
/hana/shared/aws-backint-agent/aws-backint-agent-config-catalog.yaml
```

b. Modify the `S3BucketName`, `S3BucketFolder`, and `LogFile` parameters in `aws-backint-agent-config-catalog.yaml`, using your preferred editor.

```
S3BucketName: "<Amazon S3 bucket for SAP HANA catalog>"
S3BucketFolder: "<Amazon S3 folder for SAP HANA catalog>"
LogFile: "/hana/shared/aws-backint-agent/aws-backint-agent-catalog.log"
```

c. Create a hdbbackint soft link from /usr/sap/*<SID>*/SYS/global/hdb/opt/hdbconfig/ to /hana/shared/aws-backint-agent/.

```
ln -s /hana/shared/aws-backint-agent/aws-backint-agent-config-catalog.yaml  \
/usr/sap/<SID>/SYS/global/hdb/opt/hdbconfig/aws-backint-agent-config-
catalog.yaml
```

4. **Run reconfiguration for the update**

   Run hdbnsutil -reconfig for the update to take effect.

5. **Validate to ensure that all steps have been processed correctly**

   a. Run a point-in-time recovery to a previous state to ensure that you can access the previous log files in the new Amazon S3 location.

   b. Verify that new logs are uploaded to the new S3 location.

6. **Delete previous backups**

   After a successful validation, we recommend waiting for at least a week before deleting the previous catalog.

   When you're ready, delete the previous logs with the following commands.

```
# Delete previous backups in SYSTEMDB
aws s3 rm s3://<S3 bucket>/<S3 folder>/<SID>/usr/sap/<SID>/SYS/global/hdb/backint/
SYSTEMDB/ --exclude "" --include "log_backup_0_0_0_0" --recursive --dryrun --
region us-east-1
aws s3 rm s3://<S3 bucket>/<S3 folder>/<SID>/usr/sap/<SID>/SYS/global/hdb/backint/
SYSTEMDB/ --exclude "" --include "log_backup_0_0_0_0" --recursive --region us-
east-1

# Delete previous backups in the TENANT database (Repeat for each tenant)
aws s3 rm s3://<S3 bucket>/<S3 folder>/<SID>/usr/sap/<SID>/SYS/global/hdb/backint/
DB_<SID>/ --exclude "" --include --include "log_backup_0_0_0_0" --recursive --
dryrun --region us-east-1
```

```
aws s3 rm s3://<S3 bucket>/<S3 folder>/<SID>/usr/sap/<SID>/SYS/global/hdb/backint/
DB_<SID>/ --exclude "" --include "log_backup_0_0_0_0" --recursive --region us-
east-1
```

Configure AWS Backint agent to use shorter Amazon S3 paths

AWS Backint agent uses the SAP HANA operating system path as the default location for backups, but you can configure it to use a shorter path.

Default path	s3://<Amazon-s3-bucket>/<Amazon-s3-folder>/*<SID>*/usr/sap/*<SID>*/SYS/global/hdb/backint/
New path	s3://<Amazon-s3-bucket>/<Amazon-s3-folder>/*<SID>*/

To use a shorter path, complete the following steps.

1. **Check the SAP HANA backup parameters**

   Locate the `data_backup_parameter_file`, `log_backup_parameter_file`, and `catalog_backup_parameter_file` parameters. If you are using the same parameter for data, log, and catalog backups, you only need to make this change in the `aws-backint-agent-config.yaml` file. If you are using different files, these changes need to be made in both files.
2. **Retain access to backups that are stored in the previous Amazon S3 location**

   If this is a new setup or you do not want to retain the previous catalog backup, skip this step and continue with Step 3.

   Ensure that there is no backup process running, then run the following command to move all of the previous backups to the new Amazon S3 location. This step assumes that you are using the same configuration parameter for both data and log. The example below uses the same S3 bucket, but you can use a new bucket.

   ```
   # Execute a Dry Run to check
   aws s3 cp s3://<S3 bucket>/<S3 folder>/<SID>/usr/sap/<SID>/SYS/global/hdb/backint/
    s3://<S3 bucket>/<S3 folder>/<SID>/ --recursive --dryrun --region us-east-1

   # Run the command to move the backups to new S3 location
   ```

```
aws s3 cp s3://<S3 bucket>/<S3 folder>/<SID>/usr/sap/<SID>/SYS/global/hdb/backint/
 s3://<S3 bucket>/<S3 folder>/<SID>/ --recursive --region us-east-1

# Check the output of both S3 location
aws s3 ls s3://<S3 bucket>/<S3 folder>/<SID>/ --region us-east-1
```

3. Modify `aws-backint-agent-config.yaml`.

```
vi /hana/shared/aws-backint-agent/aws-backint-agent-config.yaml
```

Add the `S3ShortenDestinationBackupEnabled` parameter in `aws-backint-agent-config.yaml`, using your preferred editor.

```
S3ShortenBackupDestinationEnabled: "true"
```

4. **Validate to ensure that all steps have been processed correctly**

   a. Run a point-in-time recovery to a previous state to ensure that you can access the previous log files in the new Amazon S3 location.

   b. Verify that new logs are uploaded to the new S3 location.

5. **Delete previous backups**

   After a successful validation, we recommend waiting for at least a week before deleting the previous catalog.

   When you're ready, delete the previous logs with the following commands.

```
# Execute a Dry Run to make sure
aws s3 rm s3://<S3 bucket>/<S3 folder>/<SID>/usr --recursive --dryrun --region us-
east-1

# Run the command to delete it in the previous S3 location
aws s3 rm s3://<S3 bucket>/<S3 folder>/<SID>/usr --recursive --region us-east-1

# Check the output of both S3 location
aws s3 ls s3://<S3 bucket>/<S3 folder>/<SID>/ --region us-east-1
```

View AWS Backint agent logs

When the AWS Backint agent is called by SAP HANA for backup and restore related operations, the logs are written as `aws-backint-agent.log` to the *<installation directory>*`/aws-backint-agent/` folder. If you want to change the location of AWS Backint agent logs, you can update the parameter `LogFile` in the `aws-backint-agent-config.yaml` file.

Get the currently installed AWS Backint agent version

To display the backint version and the current AWS Backint agent version that it supports, run the `hdbbackint` command with the `–v` parameter from the install directory as the `<SID>adm` user as shown in the following example.

```
/usr/sap/<SID>/SYS/global/hdb/opt/hdbbackint -v
```

For instance, running the preceding command on a system with `SID` as `HDB` returns the AWS Backint agent version as 1.05 as displayed in the image below.

```
hdbadm@hanabackint:/usr/sap/HDB/HDB00> /usr/sap/HDB/SYS/global/hdb/opt/hdbbackint -v
INFO[0000] Starting execution.
"backint 1.04" "AWS Backint Agent 1.05"
```


```
/usr/sap/<SID>/SYS/global/hdb/opt/hdbbackint -v
```


Update to the newest version or install a previous version of AWS Backint agent

Prerequisite

You must complete the following tasks before updating the agent.

- Disable scheduled data backups as these can fail during the version update.
- Stop log backups from SAP HANA Cockpit, SAP HANA Studio or through SQL command.

Update using installation method

The latest and previous versions of the installer can be found at the following S3 bucket locations.

- *Latest version* – `s3://awssap-backint-agent/binary/latest/install-aws-backint-agent`

AWS GovCloud (US-East) latest version – `s3://awssap-backint-agent-us-gov-east-1/binary/latest/install-aws-backint-agent`

AWS GovCloud (US-West) latest version – `s3://awssap-backint-agent-us-gov-west-1/binary/latest/install-aws-backint-agent`

- *Previous version* – `s3://awssap-backint-agent/binary/`**`agent-version`**`/install-aws-backint-agent`

  AWS GovCloud (US-East) previous version – `s3://awssap-backint-agent-us-gov-east-1/binary/`**`agent-version`**`/install-aws-backint-agent`

  AWS GovCloud (US-West) previous version – `s3://awssap-backint-agent-us-gov-west-1/binary/`**`agent-version`**`/install-aws-backint-agent`

Update using agent binary

1. Based on your AWS Region, download the agent binary tar file into a temporary location from the relevant Amazon S3 location.

   a.
   ```
   cd /tmp
   ```

   b.
   ```
   mkdir agent_download && cd agent_download
   ```

   c.
   ```
   aws s3 cp s3://awssap-backint-agent/binary/<agent-version>/aws-backint-agent.tar.gz aws-backint-agent.tar.gz --region us-east-1
   ```

   The latest and previous versions of the installer can be found at the following S3 bucket locations.

   - *Latest version* – `s3://awssap-backint-agent/binary/latest/aws-backint-agent.tar.gz`

     AWS GovCloud (US-East) latest version – `s3://awssap-backint-agent-us-gov-east-1/binary/latest/aws-backint-agent.tar.gz`

     AWS GovCloud (US-West) latest version – `s3://awssap-backint-agent-us-gov-west-1/binary/latest/aws-backint-agent.tar.gz`

- *Previous version* – `s3://awssap-backint-agent/binary/`**`agent-version`**`/aws-backint-agent.tar.gz`

  AWS GovCloud (US-East) previous version – `s3://awssap-backint-agent-us-gov-east-1/binary/`**`agent-version`**`/aws-backint-agent.tar.gz`

  AWS GovCloud (US-West) previous version – `s3://awssap-backint-agent-us-gov-west-1/binary/`**`agent-version`**`/aws-backint-agent.tar.gz`

2. Extract the binary using the following command.

```
tar -xf aws-backint-agent.tar.gz
```

3. Disable scheduled data and log backups, if not already disabled as prerequisite.
4. Backup existing agent binary using the following command. This is to ensure that you have a backup if you need to revert the agent version.

```
cp <INSTALLATION_DIR>/aws-backint-agent/aws-backint-agent <INSTALLATION_DIR>/aws-backint-agent/aws-backint-agent.<mmddyy>
```

5. Copy the newly extracted agent binary using the following command.

```
cp aws-backint-agent <INSTALLATION_DIR>/aws-backint-agent/aws-backint-agent
```

6. Change the ownership and mode with the following commands.

   a.
   ```
   cd <INSTALLATION_DIR>/aws-backint-agent
   ```

   b.
   ```
   chmod 770 aws-backint-agent
   ```

   c.
   ```
   chown <sid>adm:sapsys aws-backint-agent
   ```

7. Once the installation or update is complete, you can re-enable scheduled data backups and log backups.

See the section called “Modify AWS Backint agent configuration parameters” for the configuration parameters of the agent that are applicable according to the version.

For installing the agent with AWS Systems Manager, leave the **Agent version** blank or input a specific agent version, and follow the steps in the section called "Install AWS Backint agent using the AWS Systems Manager document".

Performance tuning

AWS Backint agent is installed with default values that optimize the performance of backup and restore operations. If you want to further optimize the performance of your backup and restore operations, you can adjust the `UploadChannelSize` and `MaximumConcurrentFilesForRestore` parameters. Ensure that you are using the right instance type and storage configurations to get the best performance. AWS Backint agent is constrained by the resources available in the instance.

The `UploadChannelSize` parameter is used to determine how many files can be uploaded in parallel to the S3 bucket during backups. The default value for this parameter is `10` and it provides optimal performance in most cases.

The `UploadConcurrency` parameter is used to determine how many S3 threads can work in parallel during backups. The default value for this parameter is `100` and it provides optimal performance in most cases.

The `MaximumConcurrentFilesForRestore` parameter is used to determine how many files can be downloaded in parallel from S3 during a restore operation. The default value for this parameter is 5, which provides the optimal performance for most use cases.

If you want to adjust these parameters, you can add them to the `aws-backint-agent-config.yaml` file and adjust the values (up to the allowed maximum). We strongly recommend that you test both the backup and recovery operations after the change to ensure there is no unintended impact to your backup and restore operations, as well as to other standard operations.

For non-production servers and SAP HANA instances smaller than 512 GB, you can lower the Amazon S3 upload and restore parameters to avoid maxing out the data volume Amazon EBS throughput. You can assign lower parameter values for non-production instances.

`UploadConcurrency`	10
`UploadChannelSize`	5

You can test the speed of your backup and Amazon EBS usage before increasing the parameter values for optimal backup time and disk usage. For more information, see Storage configuration for SAP HANA.

Subscribe to AWS Backint agent notifications

Amazon Simple Notification Service (Amazon SNS) can notify you when new versions of AWS Backint agent or AWS Backint installer are released. The following procedure shows how to subscribe to these notifications.

To subscribe to AWS Backint agent notifications

1. Open the Amazon SNS console at https://console.aws.amazon.com/sns/v3/home.
2. From the Region selector in the navigation bar, choose **US East (N. Virginia)**, if it is not selected already. You must select this Region because the SNS notifications for AWS Backint agent that you are subscribing to are generated from this Region only.
3. In the navigation pane, choose **Subscriptions**.
4. Choose **Create subscription**.
5. For **Create subscription**, do the following:

    a. For **Topic ARN**, use the following Amazon Resource Name (ARN):

    ```
    arn:aws:sns:us-east-1:464188257626:AWS-Backint-Agent-Update
    ```

    For AWS GovCloud (US-East) and AWS GovCloud (US-West), use `arn:aws-us-gov:sns:us-gov-east-1:516607370456:AWS-Backint-Agent-Update`

    b. For **Protocol**, choose **Email** or **SMS**.

    c. For **Endpoint**, enter an email address that you can use to receive the notifications. If you choose **SMS**, enter an area code and number.

    d. Choose **Create subscription**.
6. If you chose **Email**, you'll receive an email asking you to confirm your subscription. Open the email and follow the directions to complete your subscription.

    Whenever a new version of AWS Backint agent or AWS Backint installer is released, we send notifications to subscribers. If you no longer want to receive these notifications, use the following procedure to unsubscribe.

To unsubscribe from AWS Backint agent notifications

1. Open the Amazon SNS console.
2. In the navigation pane, choose **Subscriptions**.
3. Select the subscription and then choose **Actions**, **Delete subscriptions**. When prompted for confirmation, choose **Delete**.

Backup and restore your SAP HANA system with the AWS Backint Agent for SAP HANA

When the AWS Backint agent is installed and configured on your Amazon EC2 instance, you can initiate backup and recovery using SQL statements, SAP HANA Cockpit, or SAP HANA Studio.

Topics

- Backup and recovery using SQL statements
- Backup and recovery using SAP HANA Cockpit or SAP HANA Studio
- Get backup and recovery status
- Find your backup in an Amazon S3 bucket
- Schedule and manage backups
- Backup retention

Backup and recovery using SQL statements

The following are a limited number of examples of SQL statements that you can use to perform backup and recovery. We recommend that you always refer to the SAP, SAP HANA Administration, or SQL Reference guides to find the syntax of all of the other options for your specific SAP HANA version. For more details, see Backup and Recovery Statements in the *SAP HANA SQL Reference Guide*.

The following example shows the syntax to initiate a full data backup of the system database.

```
BACKUP DATA USING BACKINT ('/usr/sap/<SID>/SYS/global/hdb/backint/SYSTEMDB/
<MY_PREFIX>')
```

The following example shows the syntax to initiate a full data backup of the tenant database.

```
BACKUP DATA FOR <TENANT DB ID> USING BACKINT ('/usr/sap/<SID>/SYS/global/hdb/backint/
DB_<TENANT DB ID>/<MY_PREFIX >')
```

The following example shows the syntax to initiate a differential data backup of the tenant database.

```
BACKUP DATA DIFFERENTIAL FOR <TENANT DB ID> USING BACKINT ('/usr/sap/<SID>/SYS/global/
hdb/backint/DB_<TENANT DB ID>/<MY_PREFIX >')
```

The following example shows the syntax to initiate an incremental data backup of the tenant database.

```
BACKUP DATA INCREMENTAL FOR <TENANT DB ID> USING BACKINT ('/usr/sap/<SID>/SYS/global/
hdb/backint/DB_<TENANT DB ID>/<MY_PREFIX >')
```

The following example shows the syntax to recover your tenant database to a particular point in time.

```
RECOVER DATABASE FOR <TENANT DB ID> UNTIL TIMESTAMP 'YYYY-MM-DD HH:MM:SS' USING DATA
 PATH ('/usr/sap/<SID>/SYS/global/hdb/backint/DB_<TENANT DB ID>/') USING LOG PATH ('/
usr/sap/<SID>/SYS/global/hdb/backint/DB_<TENANT DB ID>') USING BACKUP_ID 1234567890123
 CHECK ACCESS USING BACKINT
```

The following example shows the syntax to recover your tenant database with a specific data backup using catalogs stored in S3.

```
RECOVER DATA FOR <TENANT DB ID> USING BACKUP_ID 1234567890123 USING CATALOG BACKINT
 USING DATA PATH ('/usr/sap/<SID>/SYS/global/hdb/backint/DB_<TENANT DB ID>/') CLEAR LOG
```

The following example shows the syntax to recover your tenant database with a specific data backup without using a catalog.

```
RECOVER DATA FOR <TENANT DB ID>  USING BACKINT ('/usr/sap/<SID>/SYS/global/hdb/backint/
DB_<TENANT DB ID>/<MY_PREFIX >') CLEAR LOG
```

With AWS Backint agent, you can perform system copies by restoring a backup of the source database into the target database. To perform system copies using AWS Backint agent, verify the following requirements.

1. You must have AWS Backint agent configured in both the source and target systems.
2. Check the compatibility of the SAP HANA software version of the source and target systems.
3. The AWS Backint agent in your target system should be able to access the Amazon S3 bucket where the backups of the source system are stored. If you use a different Amazon S3 bucket for backups in the source and target systems, you have to adjust the configuration parameters of the AWS Backint agent in the target system to temporarily point to the Amazon S3 bucket where the backups are stored in the source system.
4. If you are performing a system copy across two different AWS accounts, ensure that you have the appropriate IAM permissions and Amazon S3 bucket policies in place. See the Identity and Access Management section in this document for details.

The following is the syntax to restore a specific backup of the source tenant database into your target tenant database.

```
RECOVER DATA FOR <TARGET TENANT DB ID>  USING SOURCE '<SOURCE TENANT DB ID>@<SOURCE
 SYSTEM ID>' USING BACKUP_ID 1234567890123 USING CATALOG BACKINT USING DATA PATH ('/
usr/sap/<SOURCE SYSTEM ID>/SYS/global/hdb/backint/DB_<SOURCE TENANT DB ID>/') CLEAR LOG
```

The following is an example of a SQL statement to restore a specific backup of the source tenant database, called SRC, in the source system QAS into a target tenant database called TGT.

```
RECOVER DATA FOR TGT USING SOURCE 'SRC@QAS' USING BACKUP_ID 1234567890123 USING CATALOG
 BACKINT USING DATA PATH ('/usr/sap/QAS/SYS/global/hdb/backint/DB_SRC/')  CLEAR LOG
```

The following is an example of a SQL statement to perform a point-in-time recovery of a source tenant database, called SRC, in a source system QAS into a target tenant database called TGT.

```
RECOVER DATABASE FOR TGT UNTIL TIMESTAMP '2020-01-31 01:00:00' CLEAR LOG USING SOURCE
 'SRC@QAS' USING CATALOG BACKINT USING LOG PATH ('/usr/sap/QAS/SYS/global/hdb/backint/
DB_SRC') USING DATA PATH ('/usr/sap/QAS/SYS/global/hdb/backint/DB_SRC/') USING
 BACKUP_ID 1234567890123 CHECK ACCESS USING BACKINT
```

Backup and recovery using SAP HANA Cockpit or SAP HANA Studio

In addition to using SQL statements, you can initiate the backup and recovery process from SAP HANA Cockpit or SAP HANA Studio. For more information, see Backup and Recovery and Reference: Backup Console (SAP HANA Studio) in the SAP documentation. Ensure that you are using the latest version of SAP HANA Cockpit or SAP HANA studio to get all of the latest features from SAP.

Get backup and recovery status

Use your current backup and restore methods to confirm the status of a backup and restore request, and to verify whether the AWS Backint agent is working correctly. For example, if you are using SAP HANA Studio to monitor the progress of a running backup, you can do the same for any backup requests triggered by the AWS Backint agent. For failure scenarios, you can review the AWS Backint agent logs or the SAP HANA backup logs for errors, and take action or reach out to AWS Support for assistance.

Find your backup in an Amazon S3 bucket

You can verify the backup files in your Amazon S3 bucket from the Amazon S3 console or by using APIs. AWS Backint agent stores your backup files using a designated folder structure within your Amazon S3 bucket. During backup and restore, SAP HANA uses this folder structure to stream data into a pipe that Backint agents can read and write. AWS Backint agent maintains this same folder structure in the Amazon S3 bucket. We recommend that you do not change this structure after you back up your files. Changing the folder structure can cause issues during the restore operation and impact your recoverability.

For system and tenant databases, you can find your data, log, and catalog backups in the following locations. Your data backups will include an additional prefix that you used during the backup.

```
<awsdoc-example-bucket>/<optional-my-folder>/<SID>/usr/sap/<SID>/SYS/global/hdb/
backint/SYSTEMDB/
```

```
<awsdoc-example-bucket>/<optional-my-folder>/<SID>/usr/sap/<SID>/SYS/global/hdb/
backint/DB_<Tenant ID>/
```

Schedule and manage backups

You can use SAP HANA Cockpit to schedule periodic backups of your target SAP HANA database, including log backups. Ensure that you choose Backint as the backup type when scheduling your backup. For more details, see Schedule Backups in the *SAP HANA Administration with SAP HANA Cockpit Guide*.

Backup retention

Beginning with SAP HANA 2 SPS 03, you can use SAP HANA Cockpit to set the retention policies for your SAP HANA database backups. Based on your retention policies, SAP HANA Cockpit can

automatically trigger jobs to delete old backups from catalogs, as well as the physical backups. This process also automatically deletes backup files stored in your Amazon S3 buckets. For more information, see “Retention Policy” under Backup Configuration Settings in the *SAP HANA Administration with SAP HANA Cockpit Guide*.

Uninstall

Follow along these steps to uninstall the agent installed with Amazon S3.

1. Disable scheduled data and log backups if you are still using the agent for backups.
2. Remove the following symbolic links from the SAP HANA opt directory `/usr/sap/<SID>/SYS/global/hdb/opt`.
   a. SAP HANA link – `<HANA Opt directory>/hdbbackint`
   b. Config YAML link – `<HANA Opt directory>/hdbconfig/aws-backint-agent-config.yaml`
3. Remove or rename the agent installation directory.
4. Modify or remove the agent configuration parameters in the `global.ini` file.

   Reset the following parameters that are modified during agent installation, to default.
   a. `catalog_backup_parameter_file`
   b. `data_backup_parameter_file`
   c. `log_backup_parameter_file`
   d. `catalog_backup_using_backint` – Set to false
   e. `log_backup_using_backint` – Set to false
5. Reconfigure the changes as database administrative user for them to take effect.

```
hdbnsutil -reconfig
```

Your backups on Amazon S3 remain intact even after uninstalling AWS Backint agent from your Amazon EC2 instances. If you do not need the backups on Amazon S3, you can delete them from your S3 bucket.

AWS Backup

This section provides information about setting up and using AWS Backint agent to backup and restore your SAP HANA databases to AWS Backup.

Topics

- Prerequisites
- Install and configure AWS Backint Agent for SAP HANA
- Backup and restore your SAP HANA system with the AWS Backint Agent for SAP HANA

Prerequisites

The following prerequisites must be completed before to use AWS Backint Agent for SAP HANA to backup and restore SAP HANA databases to AWS Backup. For more information, see Get started with AWS Systems Manager for SAP and Register your SAP HANA databases with AWS Systems Manager for SAP .

- Set up required permissions for Amazon EC2 instance running SAP HANA database
- Set up required permissions for Amazon EC2 instance for backup and restore of SAP HANA database
- Register SAP HANA database credentials in AWS Secrets Manager
- Verify AWS Systems Manager Agent (SSM Agent) is running
- Verify parameters before registering your SAP HANA database
- Register your SAP HANA databases with AWS Systems Manager for SAP

Install and configure AWS Backint Agent for SAP HANA

Topics

- AWS Systems Manager Agent (SSM Agent)
- Systems Manager document
- Switch to AWS Backup from Amazon S3

AWS Systems Manager Agent (SSM Agent)

To install the AWS Backint agent with the AWS Systems Manager Agent (SSM Agent) document, you must install the AWS Systems Manager Agent (SSM Agent) version 2.3.274.0 or later, and your instance must be a managed instance that is configured for AWS Systems Manager. For more information about managed instances, see AWS Systems Manager Managed Instances. To update the SSM Agent, see Update SSM Agent by using Run Command.

> **Note**
>
> The SSM Agent will not work if you do not attach the `AmazonSSMManagedInstanceCore` policy to your Amazon EC2 instance role.

Systems Manager document

Ensure that your installed SSM Agent is running, and then follow these steps.

1. Go to https://console.aws.amazon.com/systems-manager/ > Shared Resources > Documents.
2. Search for the **AWSSAP-InstallBackintForAWSBackup** document.
3. Select **Run command**.
4. Specify the following parameters in **Command parameters**.
   - System ID – Enter a system ID for your SAP HANA database. For instance, `HDB`.
   - Installation Directory Confirmation – yes
   - Modify Global Ini File – modify
   - Confirm Log Backup Post Install – yes
   - Ensure No Backup In Process – yes

   You can retain the other parameters without any manual changes.
5. Under **Target selection**, choose **Choose instances manually**, and search for the Amazon EC2 instance on which your SAP HANA database is running.

   Alternatively, you can select an instance with the `SSMForSAPManaged: True` tag.
6. **Run** the **AWSSAP-InstallBackintForAWSBackup** SSM document.

The Run command takes a few minutes to complete. You can refresh the page to check the status. On successful completion, the *Overall status* and *Detailed status* display Success.

Switch to AWS Backup from Amazon S3

You can switch your storage media to be AWS Backup if you have AWS Backint agent setup with Amazon S3. Before you do that, ensure the following:

- Scheduled data backups are disabled – these can fail during switch-over.
- Scheduled backup from SAP HANA Cockpit, SAP HANA Studio or through SQL to stop log backups to Amazon S3 are disabled – these are re-enabled with AWS Backup.

To make the switch from Amazon S3 to AWS Backup, you must reinstall AWS Backint agent with Systems Manager document. The **AWSSAP-InstallBackintForAWSBackup** document replaces existing AWS Backint agent with a newer version that supports AWS Backup. For more details, see the preceding section the section called “Systems Manager document”.

Once the switch-over is complete, setup AWS Systems Manager for SAP for an automated backup solution. For more information, see Get started with AWS Systems Manager for SAP.

You can now create a backup plan or perform on-demand backups. For more information, see Backup Operations in the AWS Backup console.

Backup and restore your SAP HANA system with the AWS Backint Agent for SAP HANA

For details about backup and restore of your SAP HANA databases on AWS Backup, see SAP HANA databases on Amazon EC2 instances backup.

Verify the signature of the AWS Backint agent and installer for SAP HANA

The source file of AWS Backint agent (`aws-backint-agent.tar.gz`) and AWS Backint installer (`install-aws-backint-agent`) supports signature verification. You can use a public key to verify that the downloaded source file and AWS Backint installer are original and unmodified. You can find the AWS Backint installer in your `/tmp` directory or any other location where you have

downloaded the installer. You can find the source file (`aws-backint-agent.tar.gz`) of AWS Backint agent under `<installation directory>/aws-backint-agent/package/`.

Automatic signature verification

To enable automatic signature verification during agent installation, see the parameter descriptions at Install AWS Backint agent using AWS Backint installer — interactive mode (Step 6k).

To verify the AWS Backint agent package on a Linux server

1. Download the public key.

```
shell$ wget https://s3.amazonaws.com/awssap-backint-agent/binary/public-key/aws-
backint-agent.gpg
```

2. (Optional) For AWS GovCloud (US-East) or AWS GovCloud (US-West), download one of the following keys.

```
shell$ wget https://awssap-backint-agent-us-gov-east-1.s3.us-gov-
east-1.amazonaws.com/binary/public-key/aws-backint-agent.gpg
```

```
shell$ wget https://awssap-backint-agent-us-gov-west-1.s3.us-gov-
west-1.amazonaws.com/binary/public-key/aws-backint-agent.gpg
```

3. Import the public key into your keyring.

```
shell$ gpg --import aws-backint-agent.gpg
gpg: key 1E65925B: public key "AWS Backint Agent" imported
gpg: Total number processed: 1
gpg: imported: 1 (RSA: 1)
```

   Make a note of the key value, as you will need it in the next step. In the preceding example, the key value is 1E65925B.

4. Verify the fingerprint by running the following command.

```
shell$ gpg --fingerprint 1E65925B
pub 2048R/1E65925B 2020-03-18
Key fingerprint = BD35 7A5F 1AE9 38A0 213A 82A8 80D8 5C5E 1E65 925B
uid [ unknown] AWS Backint Agent
```

The fingerprint should be equal to the following:

```
BD35 7A5F 1AE9 38A0 213A 82A8 80D8 5C5E 1E65 925B
```

If the fingerprint string doesn't match, don't install the agent. Contact Amazon Web Services.

After you have verified the fingerprint, you can use it to verify the signature of the AWS Backint agent binary.

5. Download the signature files for the source file and the installer.

```
shell$ wget https://s3.amazonaws.com/awssap-backint-agent/binary/latest/aws-
backint-agent.sig

shell$ wget https://s3.amazonaws.com/awssap-backint-agent/binary/latest/install-
aws-backint-agent.sig
```

6. (Optional) For AWS GovCloud (US-East) and AWS GovCloud (US-West), download the signature files from one of the following locations.

```
shell$ wget https://awssap-backint-agent-us-gov-east-1.s3.us-gov-
east-1.amazonaws.com/binary/latest/aws-backint-agent.sig

shell$ wget https://awssap-backint-agent-us-gov-east-1.s3-us-gov-
east-1.amazonaws.com/binary/latest/install-aws-backint-agent.sig
```

```
shell$ wget https://awssap-backint-agent-us-gov-west-1.s3.us-gov-
west-1.amazonaws.com/binary/latest/aws-backint-agent.sig

shell$ wget https://awssap-backint-agent-us-gov-west-1.s3-us-gov-
west-1.amazonaws.com/binary/latest/install-aws-backint-agent.sig
```

7. To verify the signature, run `gpg --verify` against the `aws-backint-agent.tar.gz` source file and `install-aws-backint-agent` installer.

```
shell$ gpg --verify aws-backint-agent.sig aws-backint-agent.tar.gz
gpg: Signature made Fri 08 May 2020 12:24:48 AM UTC using RSA key ID 1E65925B
gpg: Good signature from "AWS Backint Agent" [unknown]
gpg: WARNING: This key is not certified with a trusted signature!
gpg: There is no indication that the signature belongs to the owner.
Primary key fingerprint: BD35 7A5F 1AE9 38A0 213A  82A8 80D8 5C5E 1E65 925B
```

```
shell$ gpg --verify install-aws-backint-agent.sig install-aws-backint-agent
gpg: Signature made Fri 08 May 2020 12:15:40 AM UTC using RSA key ID 1E65925B
 gpg: Good signature from "AWS Backint Agent" [unknown]
 gpg: WARNING: This key is not certified with a trusted signature!
 gpg: There is no indication that the signature belongs to the owner.
 Primary key fingerprint: BD35 7A5F 1AE9 38A0 213A  82A8 80D8 5C5E 1E65 925B
```

If the output includes the phrase BAD `signature`, check whether you performed the procedure correctly. If you continue to get this response, contact Amazon Web Services and avoid using the downloaded files.

> **Note**
>
> A key is trusted only if you or someone you trust has signed it. If you receive a warning about trust, this doesn't mean that the signature is invalid. Instead, it means that you have not verified the public key.

Troubleshoot AWS Backint Agent for SAP HANA

The following documentation can help you troubleshoot problems that you might have with your AWS Backint Agent for SAP HANA installation or backups.

Topics

- Agent logs
- Installation
- Backup and recovery
- Backup deletion

Agent logs

To find logs to help you troubleshoot errors and failures, check the following locations.

Agent logs

```
{INSTALLATION DIRECTORY}/aws-backint-agent/aws-backint-agent.log
```

System db backup/recovery logs

```
/usr/sap/<SID>/HDB<Instance No>/<hostname>/trace/backup.log
/usr/sap/<SID>/HDB<Instance No>/<hostname>/trace/backint.log
```

Tenant db backup/recovery logs

```
/usr/sap/<SID>/HDB<Instance No>/<hostname>/trace/DB_<TENANT>/backup.log
/usr/sap/<SID>/HDB<Instance No>/<hostname>/trace/DB_<TENANT>/backint.log
```

Installation

Problem: Error returned when installing AWS Backint agent.

Error returned:

```
SyntaxError: Non-UTF-8 code starting with '\xf3' in file install-aws-backint-agent on
 line 1, but no encoding declared; see http://python.org/dev/peps/pep-0263/ for details
```

- **Root Cause**: Only Python version 3 is installed on the user environment.
- **Resolution**: Run the following commands to install Python version 2 and create a symbolic link to `usr/bin/python`.

  ```
  yum install -y python2
  ```

  ```
  ln -s /usr/bin/python2.7 /usr/bin/python
  ```

Problem: Unable to view the instance listed for installation with the SSM document.

- **Root Causes**:
  1. The SSM Agent is not installed on the instance.
  2. If the SSM Agent is installed, either the instance is not running or the SSM Agent on the instance is not running.
  3. The SSM Agent installed on the instance is a version older than 2.3.274.0.
- **Resolution**: Follow the steps listed at Practice Installing or Updating SSM Agent on an Instance. You can verify whether the SSM Agent is running with the following command.

```
sudo systemctl status amazon-ssm-agent
```

Problem: The following error is returned when you use the SSM installation document.

```
failed to download manifest - failed to retrieve package document
description: InvalidDocument: Document with name AWSBackintAgent with
version x does not exist.
```

- **Root Cause**: An unsupported version of AWS Backint agent was entered.
- **Resolution**: See the version history for AWS Backint agent. For more information, see the section called “Version history”.

Backup and recovery

Problem: `AccessDenied` appears in agent logs.

- **Root Causes**:
  1. The IAM role for the EC2 instance does not have the correct permissions to access the S3 bucket.
  2. The agent configuration file does not have the `S3BucketOwnerAccountID` in double quotes. The `S3BucketOwnerAccountID` is the 12-digit AWS Account ID.
  3. The S3 bucket is not owned by the provided account for `S3BucketOwnerAccountID`.
  4. The S3 bucket provided for the `S3BucketOwnerAccountID` was created before May 2019.
- **Resolution**: Verify the prerequisite steps for installing the AWS Backint agent.

Problem: Backup or recovery failed due to S3 connectivity

- **Root Cause**: The IAM role attached to the instance does not have the correct permissions to access the S3 bucket.
- **Resolution**: Verify the prerequisite steps for installing the AWS Backint agent.

Problem: Agent logs display `Backint cannot execute hdbbackint` or `No such file or directory`.

- **Root Causes**:
  1. If you are installing the agent manually, the creation of a symlink for the agent executable did not succeed.
  2. If you are using the SSM agent, step 2 of the agent failed while creating symlinks. You can verify this by viewing the RunCommand implementation details.
- **Resolution**: Verify that you have correctly followed the installation steps in this document.

Problem: The following error is displayed when initiating a backup from the SAP HANA console:

```
Could not start backup for system <SID> DBC: [447]: backup could not be
completed: [110091] Invalid path selection for data backup using backint: /
usr/sap/<SID>/SYS/global/hdb/backint/COMPLETE_DATA_BACKUP must start with /
usr/sap/<SID>/SYS/global/hdb/backint/DB_<TENANT>
```

- **Root Cause**: When adding your SAP HANA system to SAP HANA Studio, you chose the single container mode instead of the multiple container mode.
- **Resolution**: Add the SAP HANA system to SAP HANA Studio and select multiple container mode, and then try to initiate your backup again. For more details, see Invalid path selection for data backup using backint (portal access required).

Problem: Your backup fails and the following error appears in `aws-backint-agent.log`:

```
Error creating uploadId: AuthorizationHeaderMalformed: The authorization
header is malformed; the region '<region id>' is wrong; expecting '<region
id>'
```

- **Root Cause**: You specified an incorrect Region ID for the `AwsRegion` parameter in the `aws-backint-agent-config.yaml` configuration file.
- **Resolution**: Specify the AWS Region of your Amazon S3 bucket and initiate the backup again. You can find the Region in which your Amazon S3 bucket is created from the Amazon S3 console.

Problem: Any AWS Backint agent operation fails with one of the following errors, which appear in the `aws-backint-agent.log`:

`“Error creating upload id for bucket:<mys3bucket>"`

or

`"NoCredentialProviders: no valid providers in chain.`

- **Potential Root Cause**: No IAM role is attached to your Amazon EC2 instance.
- **Resolution**: AWS Backint agent requires an attached IAM role to your EC2 instance to access AWS resources for backup and restore operations. Attach an IAM role to your EC2 instance and attempt the operation again. For more information, see the prerequisites for installing AWS Backint agent.
- **Potential Root Cause**: Use of proxy for HANA instance on which agent is run causes agent failure.
- **Resolution**: When using a proxy for the HANA instance on which the agent is run, do not use a proxy for the instance metadata call, otherwise the call hangs. Instance metadata information can not be obtained via proxy, so it must be excluded. Update the launcher script at `{INSTALLATION DIRECTORY}/aws-backint-agent-launcher.sh` to designate `169.254.169.254` as a `no_proxy` host.

  ```
  # cat aws-backint-agent-launcher.sh
  #!/bin/bash
  export https_proxy=<PROXY_ADDRESS>:<PROXY_PORT>
  export HTTP_PROXY=<PROXY_ADDRESS>:<PROXY_PORT>
  export no_proxy=169.254.169.254
  export NO_PROXY=169.254.169.254
  /hana/shared/aws-backint-agent/aws-backint-agent "$@"
  ```

  For more information about using a proxy address in your SAP HANA environment, see Use a proxy address with AWS Backint agent.

Problem: When you initiate a backup or restore, you get the following error in SAP HANA Studio or SAP HANA Cockpit:

`backup could not be completed, Backint cannot execute /usr/sap/<SID>/SYS/global/hdb/opt/hdbbackint, Permission denied (13)`

- **Root Cause**: The AWS Backint agent binary or launcher script doesn't have the execute permission at the operating system level.
- **Resolution**: Set the execute permission for AWS Backint agent binary `aws-backint-agent` and for the launcher script `aws-backint-agent-launcher.sh` in the installation directory (for example, `/hana/shared/aws-backint-agent/`).

Problem: My backup is running too slowly and is taking a longer time to complete.

- **Root Cause**: The performance of backup and restore depends on many factors, such as the type of EC2 instance used, the EBS volumes, and the number of SAP HANA channels. If your database size is less than 128 GB, SAP HANA defaults to a single channel, or your SAP HANA parameter `parallel_data_backup_backint_channels` is set to 1.
- **Resolution**: The speed of your database backup depends on how much storage throughput is available to your SAP HANA data volumes (/hana/data). Total storage throughput available for SAP HANA data volumes depends on your Amazon EBS storage type and the number of volumes used for striping. For best performance, follow the storage configuration best practices. You can switch your Amazon EBS volumes associated with SAP HANA data filesystem to `io1`, `io2` or gp3 volume type. Additionally, if your database size is greater than 128 GB, you can improve your backup performance by adjusting the number of parallel backup channels. Increase the value of `parallel_data_backup_backint_channels` and try to initiate your backup again. We recommend that you take the resource contention with normal system operation performance into consideration when you try to tune the performance of your backup.

Problem: My backup and restore fails with one of the following errors:

1. `Backint exited with exit code 1 instead of 0. console output: Crashed during fetch and conversion read/write tcp 10.0.2.83:56192->52.216.88.123:443: use of closed network connection`
2. `Backint exited with exit code 1 instead of 0. console output: Crashed during fetch and conversion caused by: read tcp 10.0.2.83:54890->52.216.130.243:443: read: connection reset by peer`

- **Root Cause**: The connection between AWS Backint agent and S3 fails due to high throughput.
- **Resolution**: Use the following steps to troubleshoot this issue.

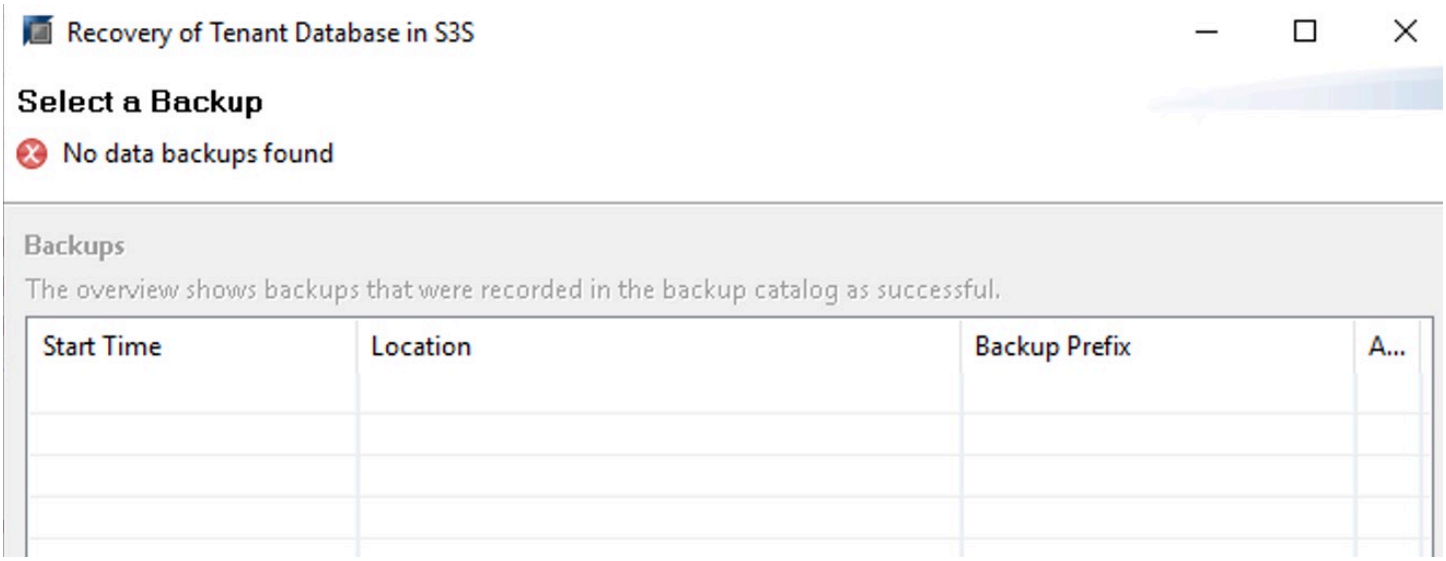
1. Update AWS Backint agent version to 2.0.4.768 or higher. These versions have improved resiliency to S3 connection timeouts.
   - Once the agent is updated, ensure that SAP HANA picks up the latest version of the agent. Run the following command to verify the version of the agent.

     ```
     /usr/sap/<SID>/SYS/global/hdb/opt/hdbbackint -v
     ```

     For more information, see Get the currently installed AWS Backint agent version.
2. Use these steps if the issue persists – lower the following backup and restore parameters.
   - **Backup**
     - `UploadConcurrency`
     - `UploadChannelSize`
   - **Restore**
     - `MaximumConcurrentFilesForRestore`
     - `DownloadConcurrency`

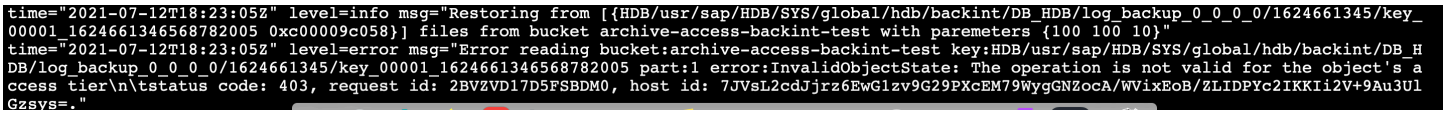
   These values reduce concurrency and parallelism used by AWS Backint agent to achieve high performance during backup and restore. See Modify AWS Backint agent configuration parameters for the default values of the preceding parameters.
3. Review network setup and configuration.
4. Perform trace route to see if Amazon S3 traffic goes through firewall package scanners or any other software that could significantly increase network latency.

Problem: When you set the S3ShortenBackupDestinationEnabled = 'true' parameter in the `aws-backint-agent-config.yaml`, a 'No data backups found' error is displayed when processing a database recovery.



- **Root Cause**: AWS Backint agent searches for the logs and data backups only in the Amazon S3 path that's provided in the configuration file. Because the `S3ShortenBackupDestinationEnabled` parameter changes the Amazon S3 folder, it cannot find the backup.
- **Resolution**: You can either change the `S3ShortenBackupDestinationEnabled` parameter to **`false`** and run the restore, or you can move the previous backups and the SAP HANA backup catalog to the new S3 location. For more details, see the section called "Configure AWS Backint agent to use shorter Amazon S3 paths".

Problem: When processing a database recovery, a 'No data backups found' error is displayed and the agent log shows, 'The operation is not valid for the objects' access tier'.



- **Root Cause**: With the **S3StorageClass = 'INTELLIGENT_TIERING'** parameter set in the `aws-backint-agent-config.yaml`, the objects have moved to archival storage tiers. AWS Backint agent does not support recovery from archival tiers.

- **Resolution**: You must first restore the archived S3 objects to move them in the access tier. This can take from a few minutes to 12 hours, depending on the archival tier and restore option that is selected. After the S3 restore is complete, you can initiate recovery for the HANA database.

Problem: Backup request initiated by IAM doesn't have access to your Amazon S3 bucket.

Error returned:

```
Error Fetching Bucket: Access Denied
```

- **Root Cause**: Credentials for internal tasks are configured in the `./aws` folder which is picked by default instead of the configured IAM role for initiating a backup request.
- **Resolution**: When you initialize a new service client without providing any credential arguments, the SDK uses the default credential provider chain to find AWS credentials. The SDK uses the first provider in the chain that returns credentials without an error. The default provider chain looks for credentials in the following order:
  1. Environment variables
  2. Shared credentials file
  3. If your application uses an Amazon ECS task definition or RunTask API operation, IAM role for tasks
  4. If your application is running on an Amazon EC2 instance, IAM role for Amazon EC2

  For more information, see Configuring the AWS SDK for Go.

Backup deletion

Problem: You deleted your SAP HANA backup from the SAP HANA backup console (SAP HANA Studio or SAP HANA Cockpit) but the deleted backup files still appear in the Amazon S3 folder.

- **Root Cause**: AWS Backint agent couldn’t delete the associated backup files from the Amazon S3 bucket due to a permission issue.
- **Resolution**: AWS Backint agent requires `s3:DeleteObject` permission to delete the backup files from your target Amazon S3 bucket when you delete the backup from the SAP HANA backup console. Ensure that the IAM profile attached to your EC2 instance has `s3:DeleteObject` permission. For backups that are already deleted from SAP HANA, you can manually delete the associated files from the Amazon S3 bucket. We recommend that you take

additional precaution before manually deleting any backup files. Manually deleting the wrong backup file could impact your ability to recover your SAP HANA system in the future.

Version history

The following table summarizes the changes for each release of AWS Backint agent.

Version	Details	Release date
2.0.4.797	Manual installer update	January 10, 2024
2.0.4.779	**Agent** • Support for `ap-southeast-4` (Melbourne), `eu-central-2` (Zurich), and `eu-south-2` (Spain) Regions.	January 2, 2024
2.0.4.768	**Agent** • Improved resiliency for Amazon S3 connection timeouts.	October 06, 2023
2.0.3.755	**Agent** • Improved `AWSSAP-InstallBackint` and `AWSSAP-InstallBackintForAWSBackup` input validation. • Removed boto3 requirement for installation of the agent.	September 13, 2023
2.0.2.732	**Agent** • Set `config file` permissions to `sapsys` group. • Support for `ap-south-2` (Hyderabad) and `me-central-1` (UAE) Regions.	July 14, 2023
2.0.1.671	**Agent**	April 17, 2023

Version	Details	Release date
	• Added support for AWS Backup as storage media. • Support for `backint protocol` 1.5. • Support for Python 3.	
1.05.4	**Agent** • Support for `ap-southeast-3` (Jakarta) Region.	February 22, 2023
1.05.4	**Agent** • Improve `AWSSAP-InstallBackint SSM` document AWS KMS input validation.	February 25, 2022
1.05.3	**Agent** • Improve `AWSSAP-InstallBackint SSM` document input validation.	February 04, 2022
1.05.2	**Agent** • Bug fix: robust error handling for Amazon S3 connection failures.	December 08, 2021

Version	Details	Release date
1.05	**Agent** • Support for Intelligent-Tiering S3 storage class. • Support for shortening S3 paths. • Support for separate log, data, and catalog backup S3 paths. **SSM & manual installer** • Support for `python3` non-compiled version of the installer. • Support for installation through Ansible configura tions. • Bug fix: Removal of ASCII characters. **Manual installer** • Bug fix: Agent binary signature verification in silent mode.	August 30, 2021
1.04	**Agent** • Support for bucket owner full control access to backup objects for cross-account backups. • Bug fix: Parallel restore configuration issue. **Manual installer** • Support for Amazon EC2 Instance Metadata Service (IMDS) v2.	May 28, 2021

Version	Details	Release date
1.03	**Agent** • Support for `ap-northeast-3` (Osaka-Local) Region. • Support for rotating agent log files. • Support for additional S3 object tags. • Improvements to parallel restore using efficient parallelism. **SSM installer** • Bug Fix: SSM document to locate `python2` library for installation. **Manual installer** • Bug Fix: Support for isolated instances to make Regional S3 calls. • Support for automatic agent signature verification.	March 31, 2021
1.02.1	**Agent** • Bug Fix: kms-key formatting issue.	December 4, 2020
1.02	**Agent** • Bug Fix: Backup failure at high throughput due to failed connection with S3.	November 19, 2020

Version	Details	Release date
1.01	**Agent** • Support for GovCloud Regions. • Support for specifying number of S3 threads that can run in parallel using UploadConcurrency parameter in configuration file. **Manual installer** • Removed -o flag. • Added -l flag, which allows you to specify the location of the agent .tar file. **SSM installer** • Added support for specifying agent installation version. • Added feature to ignore S3 bucket validations. • Bug Fix: Occasional installation failure when AWS CLI installation is selected.	July 17, 2020
1.0	Initial release.	May 18, 2020

Amazon EBS snapshots for SAP HANA

Amazon EBS snapshots provide point-in-time backups of your EBS volumes to Amazon S3. EBS snapshots are stored incrementally, which means that only the blocks that have changed after your last snapshot are saved, and you are billed only for the changed blocks.

> **Note**
>
> EBS snapshots are created at the block level. Creating EBS snapshots does not utilize storage throughput from the underlying EBS volume, or network bandwidth or CPU resources from the Amazon EC2 instance.

If you have Amazon EC2 instances running SAP HANA, you can automate the creation and retention of application-consistent EBS snapshots of the EBS volumes attached to those instances. You can then use these EBS snapshots to restore the entire SAP HANA database to the point-in-time when the EBS snapshot creation was initiated. EBS snapshots are created in a specific AWS Region, and they can be used to restore EBS volumes in any Availability Zone in that Region. EBS snapshots can also be copied to secondary Regions or AWS accounts for Disaster Recovery (DR) purposes.

The time it takes to restore an EBS volume from an EBS snapshot depends on several factors that can impact the initialization of volume. To reduce the time needed to restore a volume from a snapshot, we recommend that you enable the snapshot for Amazon EBS fast snapshot restore. Fast snapshot restore enables you to create a volume from a snapshot that is fully initialized at creation. When you automate application-consistent EBS snapshots for SAP HANA with Amazon Data Lifecycle Manager, you can configure your policy to automatically enable those snapshots for fast snapshot restore. For more information, see Considerations.

We recommend that you use AWS Backint Agent for SAP HANA as your primary backup mechanism, and to create application-consistent EBS snapshots for SAP HANA to supplement your DR strategy, by maintaining copies in other Regions and/or accounts.

Considerations

Limitations

Only the following configurations are supported.

- SAP HANA 2.0 SPS 05 and later with multi-tenant configuration.
- Single SAP HANA databases. Multiple SAP HANA Systems on One Host (MCOS) is not supported.
- SAP HANA scale-out systems are not supported.

Supported Regions

You can automate the creation and retention of application-consistent snapshots of your SAP HANA workloads using Amazon Data Lifecycle Manager in all AWS Regions where AWS Systems Manager for SAP is available.

Fast snapshot restore

- Ensure that the EBS snapshot you use for the restore has fast snapshot restore in the `enabled` state for the required Availability Zone.
- It takes 60 minutes per TiB to enable a snapshot for fast snapshot restore after the snapshot reaches the `COMPLETED` state.
- Ensure that the snapshot has enough volume creation credits to restore volumes with the full performance benefit of fast snapshot restore.
- Ensure that you have sufficient fast snapshot restore quota in your account and Region to meet your recovery needs. The total quota required depends on several factors, including the number of volumes supporting the SAP HANA database, the snapshot creation frequency, and the snapshot retention period.
- You can use Amazon CloudWatch metrics and Amazon EventBridge events to monitor the fast snapshot restore state for a snapshot.

Other considerations

- We recommend that you do not use the SSM document for SAP HANA without Amazon Data Lifecycle Manager. Doing this will result in EBS snapshots that are not managed by Amazon Data Lifecycle Manager.
- It is your responsibility to ensure that the SAP HANA database is prepared to create snapshots and that it has at least 2 percent available memory and CPU resources. Otherwise, Amazon Data Lifecycle Manager will not initiate the instructions to freeze I/O and to create the application-consistent EBS snapshots.
- The time required to complete snapshot creation depends on several factors, including the amount of data that has changed since the last snapshot of the EBS volume.

- The time it takes to restore a SAP HANA database from EBS snapshots will be impacted by the initialization of EBS volumes. You can use fast snapshot restore to ensure that the EBS volumes created from the EBS snapshot are fully-initialized at creation and instantly deliver all of their provisioned performance.
- If you choose to not use fast snapshot restore, you can manually initialize the EBS volume after creation. However, this can take several minutes or up to several hours, depending on your EC2 instance bandwidth, the IOPS provisioned for the volume, and the size of the volume.
- You can verify that application-consistent EBS snapshots of your SAP HANA workloads were successfully created by reviewing the snapshot tags, the emitted Amazon CloudWatch metrics, and the emitted Amazon EventBridge events. For more information see Identifying snapshots created with pre and post scripts and Monitoring pre and post script execution.

How to automate the creation of EBS snapshots for SAP HANA

In a running database, to be application-consistent, EBS snapshots must be aligned with an internal database snapshot. For more information, see Create a Data Snapshot in the SAP documentation.

To create application-consistent snapshots, Amazon Data Lifecycle Manager performs the following steps with pre and post scripts:

1. In the pre script, operating system checks are performed, the I/O is paused, and the SAP HANA SQL command is run to create a consistent internal database snapshot.
2. Amazon Data Lifecycle Manager initiates EBS snapshot creation for the volumes attached to the targeted instance.
3. In the post script, the SAP HANA SQL command is run to mark the internal snapshot as either completed or failed.

Amazon Data Lifecycle Manager also provides monitoring capabilities and manages the retention of the EBS snapshots after creation.

To automate the creation of application-consistent EBS snapshots for SAP HANA using Amazon Data Lifecycle Manager, you need the following:

- An Amazon Data Lifecycle Manager policy that is enabled for pre and post scripts for SAP HANA and that uses an IAM role with the permissions required to manage application-consistent

snapshots. We recommend that you also configure the policy to automatically enable the EBS snapshots for fast snapshot restore. For more information, see Considerations.

- AWS Systems Manager Agent (SSM Agent) installed and running on the target instances with the SAP HANA workloads that you want to back up.
- Access to the Systems Manager document for SAP HANA, `AWSSystemsManagerSAP-CreateDLMSnapshotForSAPHANA`, which is available in all AWS Regions where AWS Systems Manager for SAP is available.
- (Recommended) A resource tagging strategy that includes tagging your Amazon EBS volumes in a way that enables you to map them to your specific SAP HANA workloads.

For more information about setting up your target instances, the Amazon Data Lifecycle Manager policy , and your SAP HANA environment for automated application-consistent snapshots, see Automating application-consistent snapshots with pre and post scripts.

Restoring SAP HANA from EBS snapshots

A successful restore strategy is dependent on many factors, including:

- The failure scenario or event that led to the restore
- The recovery point to which you need to restore
- The date and time of your last successful backup

All of these factors can impact the restore and recovery approach. It is recommended that a comprehensive disaster recovery (DR) strategy is developed, tested, and well documented to ensure that the recovery process and recovery times are well understood.

The following steps do not cover detailed instructions on log recovery, which is likely to involve a secondary backup mechanism, such as AWS Backint for SAP HANA. Ensure that when you select the volumes to recover, you consider the potential impact on your recovery point.

Topics

- Step 1: Prepare for a restore
- Step 2: Attach or replace the restored EBS volumes
- Step 3: Recover SAP HANA database
- Step 4: Resume standard operations

Step 1: Prepare for a restore

1. Get information about the EBS volume to restore. This information can help you identify which volumes need to be restored. For example, it can help you to identify the data volumes and the log volumes for your workload. Use this information to tag your volumes or save the information in a location that will not be impacted by the loss of the instance.

   Run the following commands and note the volume's ID, serial number, UUID, mount point information, fstab configuration, and attachment information.

   ```
   $ lsblk -o +LABEL,UUID,SERIAL | sed 's/vol/vol-/g'
   ```

   ```
   $ cat /etc/fstab | column -t
   ```

   ```
   $ aws ec2 describe-volumes \
     --filters Name=attachment.instance-id,Values=instance_id \
     --query 'Volumes[*].
   [VolumeId,Size,Attachments[0].Device,Attachments[0].InstanceId,Attachments[0].State]'
    \
     --output table
   ```

2. Review the fast snapshot restore state for the EBS snapshots. EBS volumes that are created from snapshots with fast snapshot restore instantly deliver all of their provisioned performance. This eliminates the latency of I/O operations on a block when it is accessed for the first time. This is important for SAP HANA restores as tables are read from disk on startup so that they can be loaded into memory. Before you create the EBS volume, ensure that fast snapshot restore is in the `enabled` state for the snapshot in the Availability Zone in which you want to create the volume. You will also need sufficient volume creation credits. For more information, see Considerations.

3. Identify the backup and backup catalog. If possible, identify the time stamp and backup ID of the backup you plan to restore to. Ensure that the backup catalog is available in a location that will be available after the EBS volume is restored.

4. Stop SAP HANA and any backup schedules. If you are restoring in place, ensure a clean SAP HANA and operating system state for the recovery.

Run the following command to stop SAP HANA and any connected SAP applications using sapcontrol or other SAP tools, and to remove remaining processes and clean shared memory segments.

```
$ cleanipc hana_sys_no remove
```

(Optional) Do the following to prevent SAP HANA from trying to start up before the restore is complete. This can be helpful if you need to restart the operating system before the restore is complete.

```
$ cd /usr/sap/hana_sid/SYS/profile
$ vi hana_start_profile
```

Then, check and change the value of `autostart` to `0`.

```
#autostart=1 # Previous value changed for restore.
autostart=0
```

5. (Optional) Temporarily disable or modify Amazon Data Lifecycle Manager policies or schedules to exclude the EC2 instance where you are performing the restore. This prevents interference with the restore and ensures that the required snapshots are retained for the duration of the restore process.

Step 2: Attach or replace the restored EBS volumes

The following steps are dependent on whether you are restoring to the EC2 instance where the backups were created or to a new EC2 instance. If you are replacing EBS volumes on the same instance, then you must detach all previous volumes.

1. Identify the mount points to be restored from the EBS snapshot and, if applicable, unmount the filesystems associated with the old volumes from the EC2 Instance. For example, run the following command as the root user.

```
$ umount /hana/data
```

If you are concerned that the state of the volumes might impact the ability to reboot the instance, you can comment out the entries in `/etc/fstab`.

2. Follow the prescriptive guidance on Restoring an EBS volume from an EBS snapshot to create volumes from the snapshots that match the backup time and the volumes that need to be restored, and then attach the volumes to the instance using the mapping information that you noted in Step 1.

   If you are using striped Logical Volume Manager (LVM) volumes, take additional care to ensure that all required volumes in the volume group are recovered from the same point.

3. Scan or refresh connected volumes. Run the following command as the root user.

   ```
   $ pvscan --cache -aay
   ```

   If you are using LVM, run the following command.

   ```
   $ vgchange --refresh
   ```

4. Remount the volumes and ensure that `/etc/fstab` reflects the required filesystems. For example, run the following command.

   ```
   $ mount /hana/data
   ```

   Review the operating system logs for any errors.

Step 3: Recover SAP HANA database

After the EBS volumes have been restored, follow the instructions in the SAP documentation to recover the SAP HANA system database and all tenants. Ensure that the backup catalog and any required logs for roll forward are available. This can include access to AWS Backing for SAP HANA and/or local filesystems.

As both the system and tenant databases generally share the same filesystems, all databases need to be recovered.

1. Recover the SAP HANA system database. For more information, see Recover SAP HANA From a Data Snapshot in the SAP documentation.
2. Recover all SAP HANA tenant databases. For more information, see Recover all Tenant Databases From a Data Snapshot in the SAP documentation.

Step 4: Resume standard operations

If you previously disabled the Amazon Data Lifecycle Manager policy when you started the restore process, then you should now re-enable the policy so that it will continue to automate the creation of application-consistent EBS snapshots for all targeted EC2 instances.

You might also consider changing the `autostart` back to 1 so that SAP HANA restarts automatically after a system reboot.

Migrating SAP HANA to AWS: Patterns for AWS Migrations

Last updated: April, 2023

This guide describes the most common scenarios, use cases, and options for migrating SAP HANA systems from on-premises or other cloud platforms to the Amazon Web Services Cloud.

This guide is intended for SAP architects, SAP engineers, IT architects, and IT administrators who want to learn about the methodologies for migrating SAP HANA systems to AWS, or who want to have a better understanding of migration approaches to AWS in general.

This guide does not replace AWS and SAP documentation and is not intended to be a step-by-step, detailed migration guide. Some of the migration scenarios may involve additional technology, expertise, and process changes, as discussed later in this guide.

> **Note**
>
> To access the SAP notes and Knowledge Base articles (KBA) referenced in this guide, you must have an SAP ONE Support Launchpad user account. For more information, see the SAP Support website.

About this Guide

This guide is part of a content series that provides detailed information about hosting, configuring, and using SAP technologies in the AWS Cloud. For the other guides in the series, ranging from overviews to advanced topics, see https://aws.amazon.com/sap/docs/.

Migration Frameworks

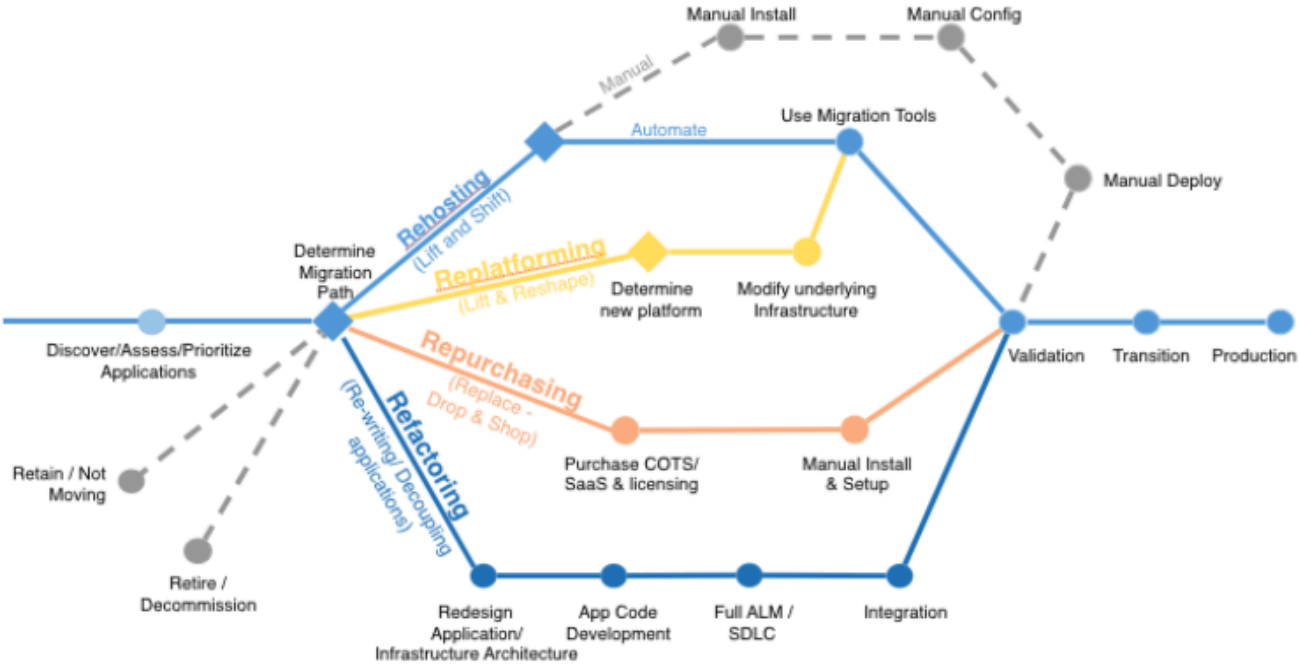
Although this guide focuses on SAP HANA migrations to AWS, it is important to understand AWS migrations in a broader context. To help our customers conceptualize and understand AWS migrations in general, we have developed two major guidelines: 6 Rs and CAF.

6 Rs Framework

The 6 Rs migration strategy helps you understand and prioritize portfolio and application discovery, planning, change management, and the technical processes involved in migrating your applications to AWS. The 6 Rs represent six strategies listed in the following table that help you plan for your application migrations.

"R" migration strategy	Methodology
Rehosting	The application is migrated as is to AWS. This is also called a “lift-and-shift” approach.
Replatforming	The application is changed or transformed in some aspect as part of its migration to AWS.
Repurchasing	You move to a different application or solution on the cloud.
Refactoring / Re-archit**ecting**	The application is redesigned (for example, it’s converted from a monolithic architecture to microservices) as part of the migration to AWS.
Retiring	The application is retired during migration to AWS.
Retaining	The application isn’t migrated.

The decision tree diagram helps you visualize the end-to-end process, starting from application discovery and moving through each 6 R strategy.



The two strategies that are specifically applicable for SAP HANA migrations to AWS are rehosting and replatforming. Rehosting is applicable when you want to move your SAP HANA system as is to AWS. This type of migration involves minimal change and can be seen as a natural fit for customers who are already running some sort of SAP HANA system. Replatforming is applicable when you want to migrate from an *anyDB* source database (such as IBM DB2, Oracle Database, or SQL Server) to an SAP HANA database.

AWS CAF Framework

The second guideline is the AWS Cloud Adoption Framework (CAF). The AWS CAF breaks down the complex process of planning a move to the cloud into manageable pieces called *perspectives*. Perspectives represent essential areas of focus that span people, processes, and technology. Capabilities within each perspective identify the areas of your organization that require attention. From this information, you can build an action plan organized into prescriptive work streams that support a successful cloud journey. Both the CAF and 6 Rs frameworks help you understand and plan the broader context of an AWS migration and what it means to you and your company.

Planning

Before you start migrating your SAP environment to AWS, there are some prerequisites that we recommend you go over, to ensure minimal interruptions or delays. For details, see the SAP on AWS overview. The following sections discuss additional considerations for planning your migration.

Understanding On-Premises Resource Utilization

If you are planning to rehost your on-premises SAP HANA environment on AWS, AWS Application Discovery Service can help you understand the utilization of resources as well as hardware configuration, performance data, and network connections in your on-premises SAP HANA environment. You can use this information to ensure that appropriate communication ports are enabled between SAP HANA and other systems in the security groups or virtual private clouds (VPCs) on AWS.

Application Discovery Service can be deployed in an agentless mode (for VMware environments) or with an agent-based mode (all VMs and physical servers). We recommend that you run Application Discovery Service for a few weeks to get a complete, initial assessment of how your on-premises environment is utilized, before you migrate to AWS.

Reviewing AWS Automation Tools for SAP

It is a good idea to review AWS automation tools and services that can help you migrate your SAP environment to AWS. For example, AWS Launch Wizard for SAP helps you deploy workloads, such as SAP HANA and SAP NetWeaver application servers. For details, see the Migration Tools and Methodologies section later in this guide.

Prerequisites

SAP HANA system migration requires a moderate to high-level knowledge of the source and target IT technologies and environments. We recommend that you familiarize yourself with the following information:

AWS Cloud architecture and migration:

- AWS Well-Architected Framework
- An Overview of the AWS Cloud Adoption Framework
- Architecting for the Cloud: Best Practices
- Migrating Your Existing Applications to the AWS Cloud

AWS services:

- Amazon Virtual Private Cloud (Amazon VPC)
- Amazon Elastic Compute Cloud (Amazon EC2)
- Amazon Elastic Block Store (Amazon EBS)
- Amazon Simple Storage Service (Amazon S3)

SAP on AWS:

- SAP on AWS Implementation and Operations Guide
- AWS Launch Wizard for SAP
- SAP HANA Environment Setup on AWS
- SAP on Amazon Web Services High Availability Guide

SAP HANA Sizing

The size of the SAP HANA system required on the AWS Cloud depends on the migration scenario. As mentioned earlier, migrating SAP HANA to AWS involves two possible scenarios: rehosting or replatforming.

Memory Requirements for Rehosting

Because rehosting implies that you are already running SAP HANA, you can determine the size of the SAP HANA system you need on the AWS Cloud from the peak memory utilization of your existing SAP HANA system. You may have oversized your on-premises SAP HANA environment (for example, to support future growth), so measuring peak memory utilization is a better approach than measuring allocated memory. When you have determined the base memory requirement, you should choose the smallest SAP-certified EC2 instance that provides more memory than your base requirement.

There are three ways to determine peak memory utilization of your existing SAP HANA system:

- SAP HANA Studio: The overview tab of the SAP HANA Studio administration view provides a memory utilization summary.
- SAP EarlyWatch alerts: This is a free, automated service from SAP that helps you monitor major administrative areas of your SAP system. See the SAP portal for details.

- SQL statements: SAP provides SQL statements that you can use to determine peak memory utilization. For details, see SAP KBA 1999997 – FAQ: SAP HANA Memory and SAP Note 1969700 – SQL statement collection for SAP HANA.

> **Tip**
>
> We recommend determining peak memory utilization for a timeframe during which your system utilization is likely to be high (for example, during year-end processing or a major sales event).

Memory Requirements for Replatforming

The replatforming scenario involves two possibilities:

- You are already running SAP HANA but you want to change your operating system—for example, from Red Hat Enterprise Linux (RHEL) to SUSE Linux Enterprise Server (SLES) or the other way around—when you migrate to the AWS Cloud, or you are migrating from an IBM POWER system to the x86 platform. In this case, you should size SAP HANA as described for the rehosting scenario.
- You are migrating from *anyDB* to SAP HANA. There are multiple ways you can estimate your memory requirements:
  - SAP standard reports for estimation: This is the best possible approach and is based on standard sizing reports provided by SAP. For examples, see the following SAP Notes:
    - 1736976 – Sizing Report for BW on HANA
    - 1637145 – SAP BW on HANA: Sizing SAP In-Memory Database
    - 1872170 - Business Suite on HANA and S/4HANA sizing report
    - 1736976 – Sizing Report for BW on HANA
  - SQL statements: SAP provides scripts that you can run in your existing environment to get high-level SAP HANA sizing estimates. These scripts run SQL statements against your existing database to estimate SAP HANA memory requirements. For more information, see SAP Note 1793345 - Sizing for SAP Suite on SAP HANA .
  - Rule of thumb: See the PDF attached to SAP Note 1793345 - SAP HANA Sizing for SAP Suite on SAP HANA for instructions on estimating SAP HANA memory requirements manually. Note that this will be a very rough and generic estimate.

You should also consider the following SAP notes and Knowledge Base articles for SAP HANA sizing considerations:

- 2388483 – How-To: Data Management for Technical Tables
- 1855041 – Sizing Recommendation for Master Node in BW-on-HANA
- 1702409 – HANA DB: Optimal number of scale out nodes for BW on HANA

Instance Sizing for SAP HANA

AWS offers SAP-certified systems that are configured to meet the specific SAP HANA performance requirements. For more information, see SAP Note 1943937 – Hardware Configuration Check Tool - Central Note, and Amazon EC2 instances for SAP on AWS. After you have determined your SAP HANA sizing, you can map your requirements to the EC2 instance family sizes. That is, you map the maximum amount of memory required for each of your SAP HANA instances to the maximum amount of memory available for your desired EC2 instance type. You should also consider appropriate storage volume types and sizes to ensure optimal performance of the SAP HANA database. For best practices and recommendations for volume types and file system layout, see the AWS Launch Wizard for SAP.

> **Note**
>
> Only production SAP HANA systems need to run on certified configurations that meet SAP HANA key performance indicators (KPIs). SAP provides more flexibility when running SAP HANA non-production systems. For more information, see SAP HANA TDI – FAQ and OSS Note 2271345 on the SAP website.

Network Planning and Sizing

You will need to consider network planning and sizing for the amount of data you will be transferring to AWS. Data transfer time depends on network bandwidth available to AWS and influences total downtime. Higher bandwidth helps with faster data transfer and helps reduce overall migration time. For non-production systems where downtime isn't critical, you can use a smaller network pipe to reduce costs. Alternatively, to transfer extremely large data, you can use services like AWS Snowball for a physical (non-network) transport of data to AWS. We'll discuss AWS Snowball more extensively later in this guide.

As a guideline, you can use this formula to help estimate how long your network data transfer might take:

(Total bytes to be transferred / Transfer rate per second) = Total transfer time in seconds

For example, for a 1 TB SAP HANA appliance, the total bytes to be transferred is usually 50% of the memory, which would be 512 GB. The transfer rate per second is your network transfer rate—if you had a 1 Gb AWS Direct Connect connection to AWS, you could transfer up to 125 MB per second, and your total data transfer time would be:

512 GB / 125 MB per second = 4,096 seconds (or 1.1 hours)

After you determine the amount of data you need to transfer and how much time you have available to transfer the files, you can determine the AWS connectivity options that best fit your cost, speed, and connectivity requirements.

SAP HANA Scale-up and Scale-out

AWS provides several types of EC2 instances for SAP HANA workloads. This gives you options for your SAP HANA scale-up and scale-out deployments. In a scale-up scenario, you utilize the compute, memory, network, and I/O capacity of a single EC2 instance. If you require more capacity, you can resize your instances to a different EC2 instance type. For example, if you're using an R4 instance type and it becomes too small for your workload, you can change it to an R5, X1, or X1e instance type. The limitation is the maximum capacity of a single EC2 instance. In AWS, scale-up enables you to start with the smallest EC2 instance type that meets your requirements and grow as needed. If your requirements change or new requirements surface, you can easily scale up to meet the changing requirements.

In a scale-out scenario, you add capacity to your SAP HANA system by adding new EC2 instances to the SAP HANA cluster. For example, once you reach the maximum memory capacity of a single EC2 instance, you can scale out your SAP HANA cluster and add more instances. AWS has certified SAP HANA scale-out clusters that support up to 100 TiB of memory. Please note that the minimum number of recommended nodes in an SAP HANA scale-out cluster can be as low as two nodes; for more information, see SAP Note 1702409 - HANA DB: Optimal number of scale out nodes for BW on HANA. It's likely that your sizing estimates will reveal the need to plan for a scale-out configuration before you start your SAP HANA migration. AWS gives you the ability to easily deploy SAP HANA scale-out configurations when you use the AWS Launch Wizard for SAP.

The following table illustrates example scale-up and scale-out sizing.

Scenario	Source configuration	Target configuration
Scale-up	r5.8xlarge	r5.16xlarge
Scale-up	r5.16xlarge	x2idn.16xlarge
Scale-up	x2idn.32xlarge	x2iedn.32xlarge
Scale-out	3 nodes of x2idn.16xlarge	4 nodes of x2idn.16xlarge
Scale-out	x2idn.32xlarge	3 nodes of x2idn.16xlarge

When you finalize your SAP sizing and SAP HANA deployment models, you can plan your migration strategy.

In addition to SAP HANA sizing, you may also need to size your SAP application tier. To find the SAP Application Performance Standard (SAPS) ratings of SAP-certified EC2 instances, see SAP Standard Application Benchmarks and the SAP on AWS support note on the SAP website (SAP login required).

Migration Tools and Methodologies

This section provides an introduction to the tools and methodologies available to you for your SAP system migration.

Topics

- AWS Launch Wizard for SAP
- AWS Migration Hub Orchestrator
- Amazon EC2 Instance Resize
- AMIs
- AWS Snowball
- Amazon S3 Transfer Acceleration
- SAP HANA HSR with Initialization via Backup and Restore
- Migration Using DMO with System Move
- SAP HANA Classical Migration
- SAP Software SUM DMO

- DMO Move to SAP S/4HANA on AWS (single step) – DMOVE2S4
- Backup/Restore Tools

AWS Launch Wizard for SAP

AWS Launch Wizard for SAP is a service that guides you through the sizing, configuration, and deployment of SAP applications on AWS, and follows AWS cloud application best practices.

AWS Launch Wizard reduces the time it takes to deploy SAP applications on AWS. You input your application requirements, including SAP HANA settings, SAP landscape settings, and deployment details on the service console, and Launch Wizard identifies the appropriate AWS resources to deploy and run your SAP application. Launch Wizard provides an estimated cost of deployment, which allows you to modify your resources and instantly view the updated cost. When you finalize your settings, Launch Wizard provisions and configures the selected resources. It then optionally installs an SAP HANA database and supported SAP applications using customer-provided software.

After you deploy an SAP application, you can access it from the Amazon EC2 console. You can manage your SAP applications with AWS SSM.

For more information, see AWS Launch Wizard for SAP.

AWS Migration Hub Orchestrator

AWS Migration Hub Orchestrator simplifies and automates the migration of servers and enterprise applications to AWS. It provides a single location to run and track your migrations.

You can migrate SAP HANA scale-up and scale-out systems to AWS cloud with AWS Migration Hub Orchestrator. Migration Hub Orchestrator offers templates to create a migration workflow that can be customized to fit your unique migration requirements. For more information, see AWS Migration Hub Orchestrator?

You can access AWS Migration Hub Orchestrator from https://console.aws.amazon.com/migrationhub/orchestrator/ or from the AWS Command Line Interface.

Amazon EC2 Instance Resize

Amazon EC2 provides you with the ability to easily change your instance type in minutes, from the Amazon EC2 console, the AWS Command Line Interface (AWS CLI), or the Amazon EC2 API. You can start with an instance type that meets your current needs and size your instance up or down, when your requirements change. When you change your EC2 instance type, all instance metadata,

including the IP address, instance ID, and hostname, remains the same. This enables you to migrate your SAP HANA to a new instance type seamlessly, without incurring a longer downtime. For details, see the Changing the Instance Type in the Amazon EC2 documentation.

AMIs

You can use an Amazon Machine Image (AMI) to launch any EC2 instance. You can create an AMI of an EC2 instance that hosts SAP HANA, including the attached EBS volumes, through the Amazon EC2 console, the AWS CLI, or the Amazon EC2 API. You can then use the AMI to launch a new EC2 instance with SAP HANA in any Availability Zone within the AWS Region where the AMI was created. You can also copy your AMI to another AWS Region and use it to launch a new instance. You can use this feature to move your SAP HANA instance to another Availability Zone or AWS Region, or to change the tenancy type of your EC2 instance. For example, you can create an AMI of your EC2 instance with default tenancy and use it to launch a new EC2 instance with host or dedicated tenancy and vice versa. For details, see the Amazon Machine Images (AMIs) in the Amazon EC2 documentation.

AWS Snowball

With AWS Snowball, you can copy large amounts of data from your on-premises environment to AWS, when it's not practical or possible to copy the data over the network. AWS Snowball is a storage appliance that is shipped to your data center. You plug it into your local network to copy large volumes of data at high speed. When your data has been copied to the appliance, you can ship it back to AWS, and your data will be copied to Amazon S3 based on the desired target storage destination that you specify. AWS Snowball is very useful when you're planning very large, multi-TB SAP system migrations. For more information, see *When should I consider using Snowball instead of the Internet* in the AWS Snowball FAQ.

Amazon S3 Transfer Acceleration

Amazon S3 Transfer Acceleration provides a faster way to copy data from your on-premises environment to AWS by copying data first to Amazon CloudFront edge locations that are closest to the source, and then using an optimized network path to copy data to Amazon S3. There is a network charge associated with this type of transfer. You can run an AWS-provided test tool to compare the speed of Amazon S3 Transfer Acceleration to standard Amazon S3 data transfer. For SAP workloads, you can copy backups or DB logs at regular intervals over Amazon S3 Transfer Acceleration to reduce the transfer time, if your regular network connection is slow—for example,

if your SAP environment is hosted in a location that doesn't have very strong internet connectivity. For more information, see the Amazon S3 documentation.

SAP HANA HSR with Initialization via Backup and Restore

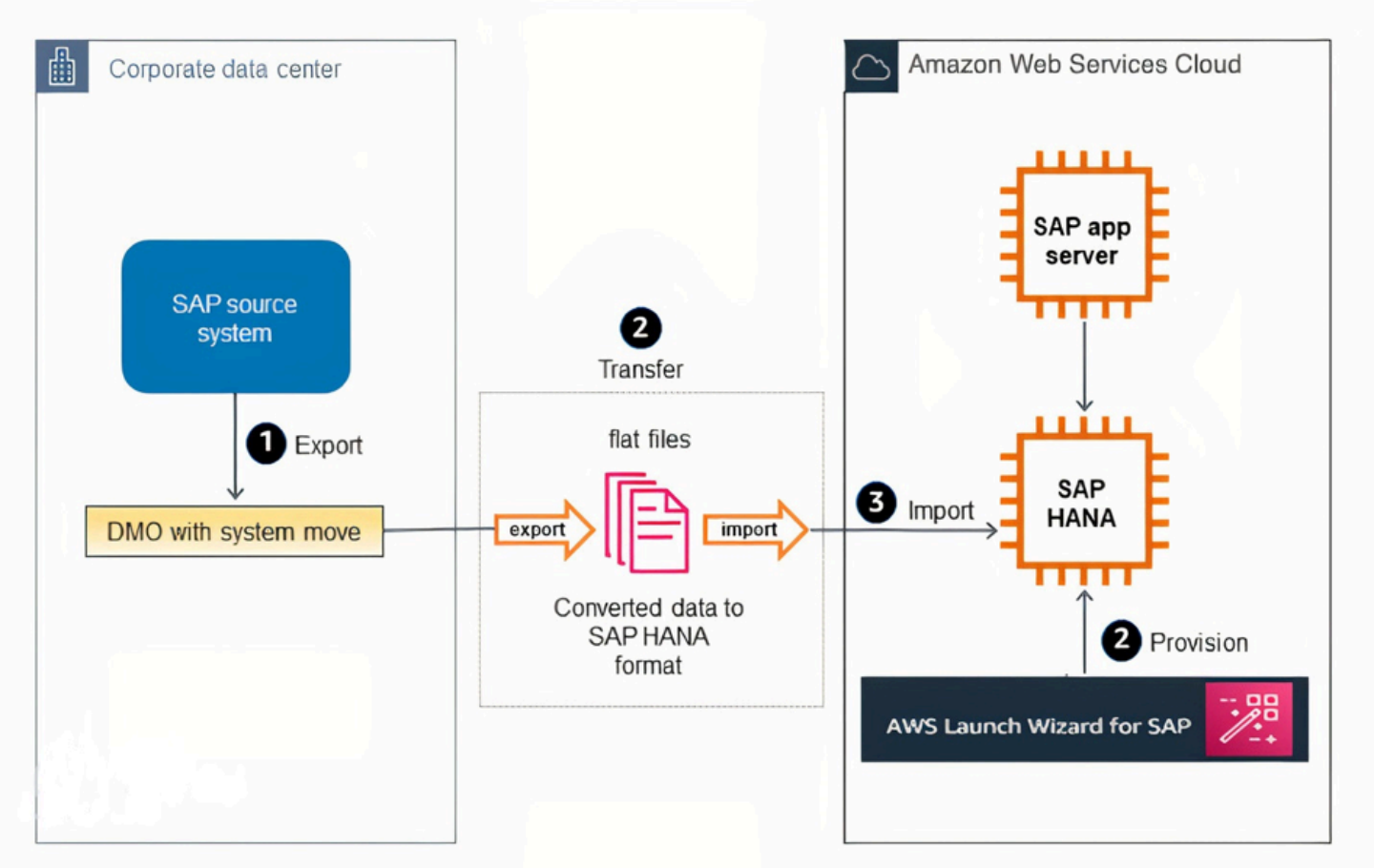
SAP supports the option of initializing the HSR target system with a backup and restore process. Using backup and restore can be useful if the network connection between your source SAP HANA system and the target system does not have enough bandwidth to replicate the data in a timely manner. Additionally, you may not want the data replication to consume part of your network traffic bandwidth. For details, see SAP Note 1999880 – FAQ: SAP HANA System Replication.

Migration Using DMO with System Move

SAP has enhanced the database migration option (DMO) of their Software Update Manager (SUM) tool to accelerate the testing of SAP application migrations. DMO with System Move enables you to migrate your SAP system from your on-premises environment to AWS by using a DMO tool and a special export and import process. You can use AWS services such as Amazon S3, Amazon EFS (over AWS Direct Connect), Storage Gateway file interface, and AWS Snowball to transfer your SAP export files to AWS.

You can then use the AWS Launch Wizard for SAP to rapidly provision SAP HANA instances and build your SAP application servers on AWS, when you are ready to trigger the import process of the DMO tool.

The SUM DMO tool can convert data from *anyDB* to SAP HANA or SAP ASE, with OS migrations, release/enhancement pack upgrades, and Unicode conversions occurring at the same time. Results are written to flat files, which are transferred to the target SAP HANA system on AWS. The second phase of DMO with System Move imports the flat files and builds the migrated SAP application with the extracted data, code, and configuration. Here's a conceptual flow of the major steps involved:



SAP HANA Classical Migration

SAP offers the SAP HANA classical migration option for migrating from other database systems to SAP HANA. This option uses the SAP heterogeneous system copy process and tools. To copy the exported files, you can use the options described in the Backup/Restore Tools section later in this guide. For details on the classical migration approach, see the classical migration overview on the SAP website.

SAP Software SUM DMO

SAP offers the standard SUM DMO approach as a one-step migration option from other database systems to HANA. This option uses the SAP DMO process and tool to automate multiple required migration steps. This is a preferred option if you are already running SAP on *anyDB* on AWS, as it will improve your migration times to SAP HANA, since there is no need for data export/import at a file system level. For details, see the DMO of SUM overview on the SAP website.

DMO Move to SAP S/4HANA on AWS (single step) – DMOVE2S4

Using SAP Database Migration Option (DMO) feature DMOVE2S4, you can migrate SAP ECC on SAP HANA or any other database, such as Oracle, SQL or others hosted on-premises to AWS Cloud. It combines migration with conversion. With this option, you can convert SAP ERP on SAP HANA to SAP S/4HANA while migrating the system to SAP S/4HANA cloud, private edition on AWS Cloud.

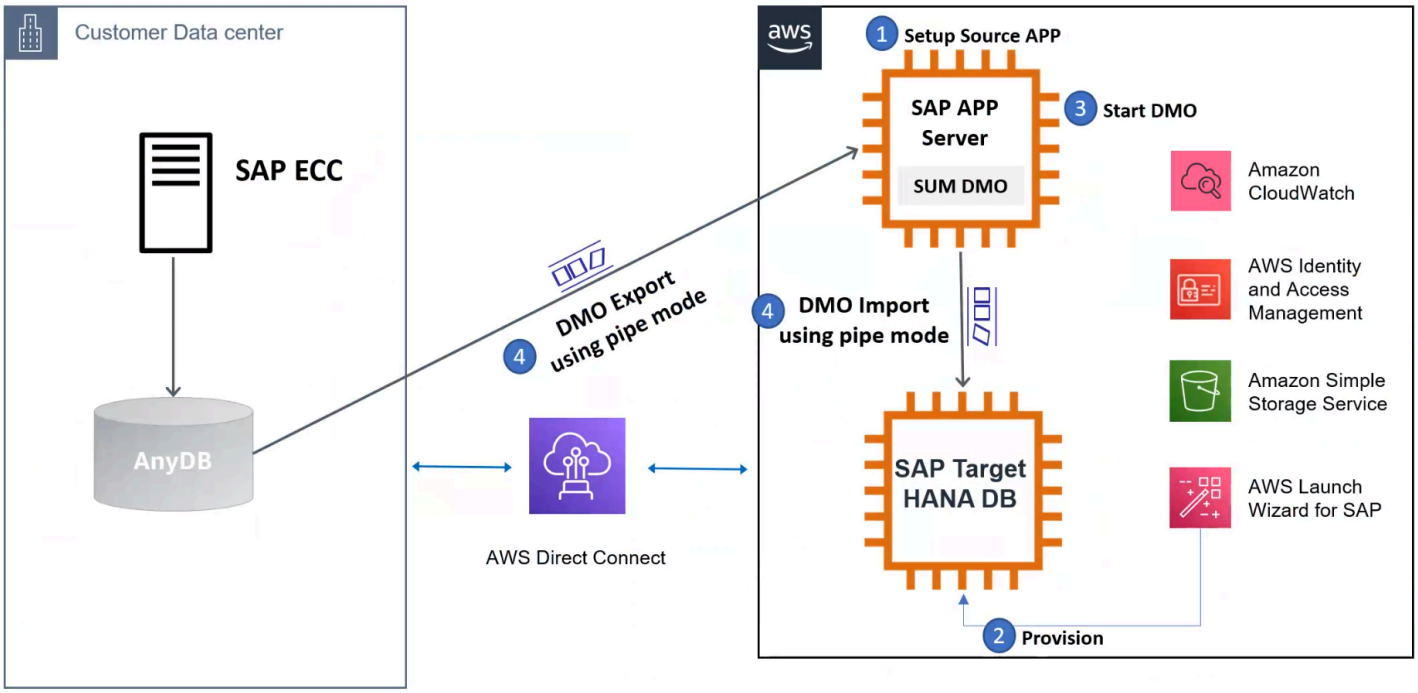
The migration is performed over network, using memory pipe option. It eliminates the requirement to import/export data over file system. You can further accelerate the migration by using one or both of the following options.

- Use AWS Direct Connect to enable secure and fast data transfer over the network. For more information, see What is AWS Direct Connect?
- Use an Amazon EC2 instance with a high number of vCPUs as the SAP application server running the import. This increases the parallel processing rate of the database load and reduces the time taken for migration and conversion.

One of the biggest advantages of a heterogeneous migration is that you can use DMO features, downtime optimized techniques, such as downtime optimized DMO (doDMO) or downtime optimized conversion (DOC), that are not available when using traditional DMO with system move option.

For more information, see the following SAP resources.

- SAP Note 3296427 - Database Migration Option (DMO) of SUM 2.0 SP17 (requires SAP portal access)
- SAP Documentation - DMO Move to SAP S/4HANA (on Hyperscaler)

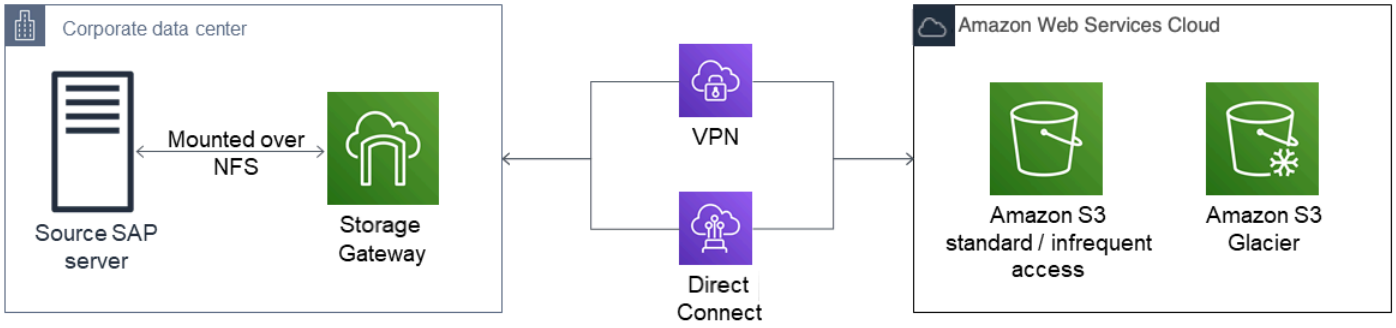


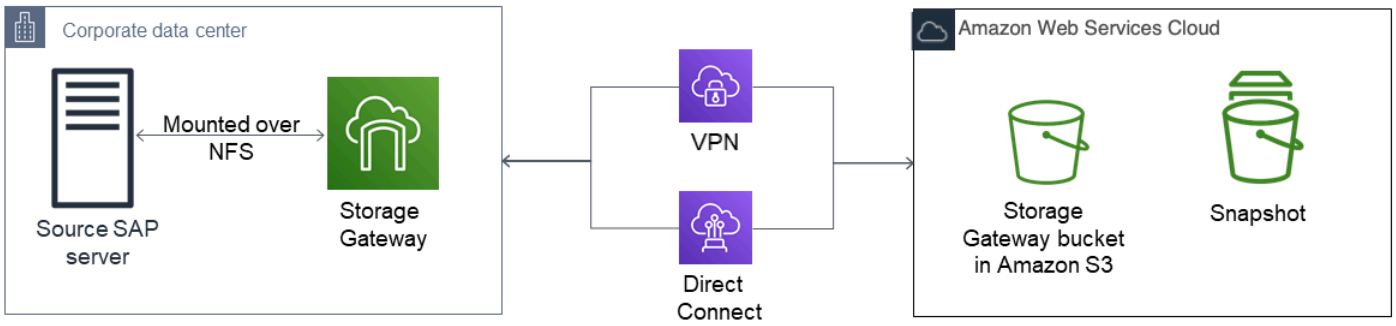
One-Step Migration SAP on AnyDB to SAP on HANA DB
One-Step Conversion from SAP ERP to SAP S/4HANA Using DMOVE2S4 to AWS

Backup/Restore Tools

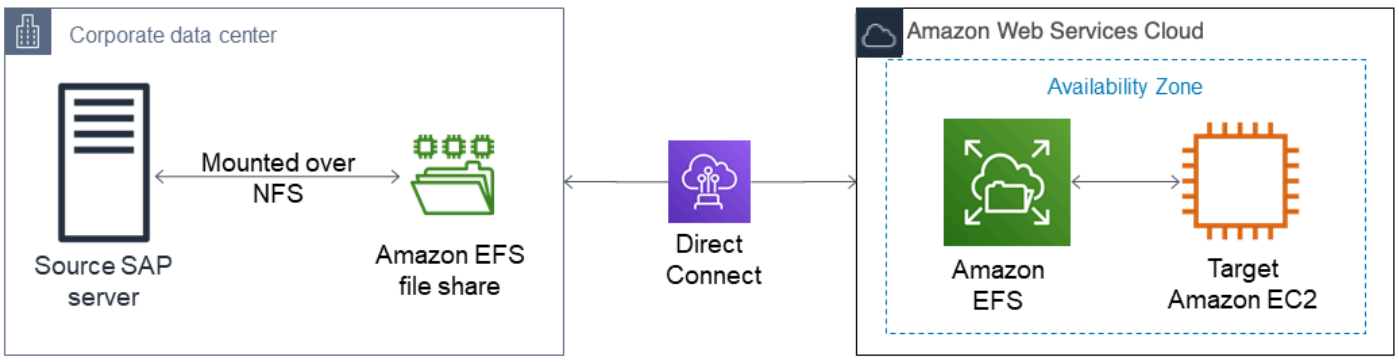
Backup and restore options are tried-and-true mechanisms for saving data on a source system and restoring it to another destination. AWS has various storage options available to help facilitate data transfer to AWS. Some of those are explained in this section. We recommend that you discuss which option would work best for your specific workload with your systems integrator (SI) partner or with an AWS solutions architect.

- Storage Gateway: This is a virtual appliance installed in your on-premises data center that helps you replicate files, block storage, or tape libraries by integrating with AWS storage services such as Amazon S3 and by using standard protocols like Network File system (NFS) or Internet Small Computer System Interface (iSCSI). Storage Gateway offers file-based, volume-based, and tape-based storage solutions. For SAP systems, we will focus on file replication using a file gateway and block storage replication using a volume gateway. For scenarios where multiple backups or logs need to be continuously copied to AWS, you can copy these files to the locally mounted storage and they will be replicated to AWS.





- Amazon EFS file transfer: AWS provides options to copy data from an on-premises environment to AWS by using Amazon Elastic File System (Amazon EFS). Amazon EFS is a fully managed service, and you pay only for the storage that you use. You can mount an Amazon EFS file share on your on-premises server, as long as you have AWS Direct Connect set up between your corporate data center and AWS.



Migration Scenarios

The following table lists the migration scenarios that we will cover in detail in this guide. The tools and methodologies listed in the table were discussed in the previous section.

Migration scenario	Source database	Target database	Migration tool or methodology
Migration of *anyDB* from other platforms to AWS*	*anyDB* (any non-SAP HANA database such as IBM DB2, Oracle Database, or SQL Server)	SAP HANA	[✓] SAP HANA classical migration [✓] SAP DMO with System Move [✓] SAP DMO move to SAP S/4 HANA (on hyperscaler) DMOVE2S4
Migration of SAP HANA from other platforms to AWS*	SAP HANA (scale-up and scale-out considerations apply here as well)	SAP HANA	[✓] AWS Migration Hub Orchestrator [✓] SAP HANA backup and restore [✓] SAP HANA classical migration (considered a homogeneous system copy in this scenario) ** [✓] SAP HANA HSR [✓] SAP HANA HSR with initialization via backup and restore [✓] SAP DMO with system move [✓] SAP DMO move to SAP S/4 HANA

Migration scenario	Source database	Target database	Migration tool or methodology
			(on hyperscaler) DMOVE2S4
Migration of SAP HANA from an existing EC2 instance to an EC2 High Memory instance	SAP HANA	SAP HANA	[✓] Instance resize [✓] Amazon Machine Image (AMI) [✓] SAP HANA HSR [✓] AWS Migration Hub Orchestrator [✓] SAP HANA backup and restore

* Other platforms include on-premises infrastructures and other cloud infrastructures outside of AWS.

** See SAP Note 1844468 – Homogeneous system copy on SAP HANA.

Migrating *AnyDB* to SAP HANA on AWS

Migrating from *anyDB* to HANA typically involves changes to the database platform and sometimes includes operating system changes. However, migration might also involve additional technical changes and impacts, such as the following:

- SAP ABAP code changes. For example, you might have custom code that has database or operating system dependencies, such as database hints coded for the *anyDB* platform. You might also need to change custom ABAP code so it performs optimally on SAP HANA. See SAP's recommendations and guidance for these SAP HANA-specific optimizations. For details and guidance, see Get started with the ABAP custom code migration process and SAP Notes 1885926 – ABAP SQL monitor and 1912445 – ABAP custom code migration for SAP HANA on the SAP website.

- Operating system-specific dependencies such as custom file shares and scripts that would need to be re-created or moved to a different solution.
- Operating system tunings (for example, kernel parameters) that would need to be accounted for. Note that the AWS Launch Wizard for SAP incorporates best practices from operating system partners like SUSE and Red Hat for SAP HANA.
- Technology expertise such as Linux administration and support, if your organization doesn't already have experience with Linux.

SAP provides tools and methodologies such as classical migration and SUM DMO to help its customers with the migration process for this scenario. (For more information, see the section Migration Tools and Methodologies.) AWS customers can use the SAP SUM DMO tool to migrate their database to SAP HANA on AWS. Some considerations for the SAP SUM DMO method are network bandwidth, amount of data to be transferred, and the amount of time available for the data to be transferred.

You can use the SUM DMO memory pipe option – DMO Move to SAP S/4HANA (DMOVE2S4) to accelerate your move to SAP S/4HANA. In a single step, you can migrate the database over network while converting to SAP S/4HANA. For more information, see SAP Documentation – DMO Move to SAP S/4HANA (on Hyperscaler). Use AWS Direct Connect to connect your on-premises environment with AWS Cloud for secure and fast data transfer over the network. When using DMOVE2S4, you must take the following into consideration.

- low latency – less than 20 ms
- high bandwidth – more than 400 Mbps

For more information, see the following SAP resources.

- SAP Note 3296427 - Database Migration Option (DMO) of SUM 2.0 SP17 (requires SAP portal access)
- SAP Blog - Two Major News with SUM 2.0 SP 17

Implementing SAP HANA on AWS enables quick provisioning of scale-up and scale-out SAP HANA configurations and enables you to have your SAP HANA system available in minutes. In addition to fast provisioning, AWS lets you quickly scale up by changing your EC2 instance type. With this capability, you can react to changing requirements promptly and focus less on getting your sizing

absolutely perfect. This means that you can spend less time sizing (that is, you can move through your project's planning and sizing phase faster) knowing that you can scale up later, if needed.

Migrating SAP HANA from Other Platforms to AWS

This scenario is more straightforward than migrating from *anyDB*, because you're already using SAP HANA. For this migration, you need to map your existing SAP HANA systems and sizing that are on a different platform to SAP HANA solutions on AWS.

EC2 instance memory capabilities give you the option to consolidate multiple SAP HANA databases on a single EC2 instance (scale-up) or multiple EC2 instances (scale-out). SAP calls these options HANA and ABAP One Server, Multiple Components in One Database (MCOD), Multiple Components in One System (MCOS), and Multitenant Database Containers (MDC). It is beyond the scope of this guide to recommend specific consolidation combinations; for possible combinations, see SAP Note 1661202 – Support for multiple applications on SAP HANA.

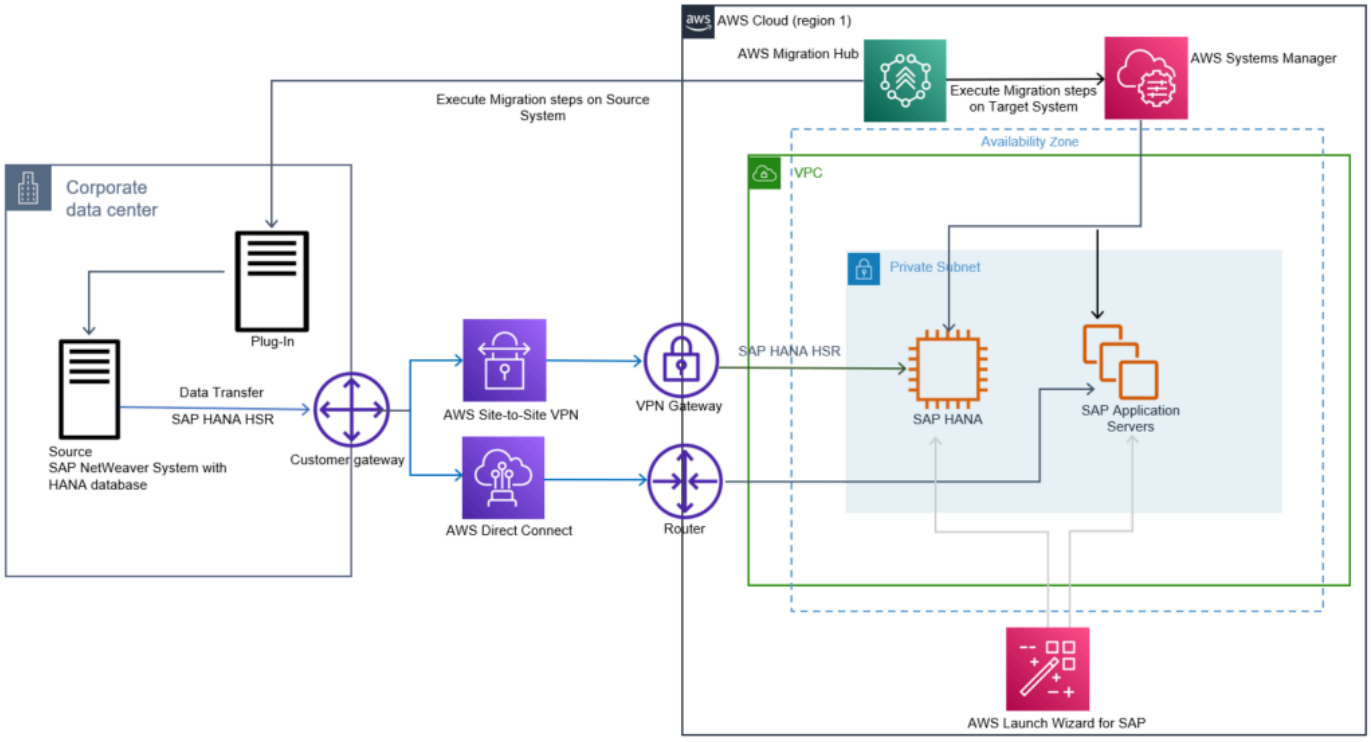
This migration scenario involves provisioning your SAP HANA system on AWS, backing up your source database, transferring your data to AWS, and installing your SAP application servers. If you are resizing your HANA environment from scale-up to scale-out, please follow the process highlighted in SAP Note 2130603. If you are resizing your HANA environment from scale-out to scale-up, refer to SAP Note 2093572. Depending on your specific scenario, you can use standard backup and restore, SAP HANA classical migration, SAP HANA HSR, AWS Server Migration Service (AWS SMS), or third-party continuous data protection (CDP) tools; see the following sections for details on each option.

Options

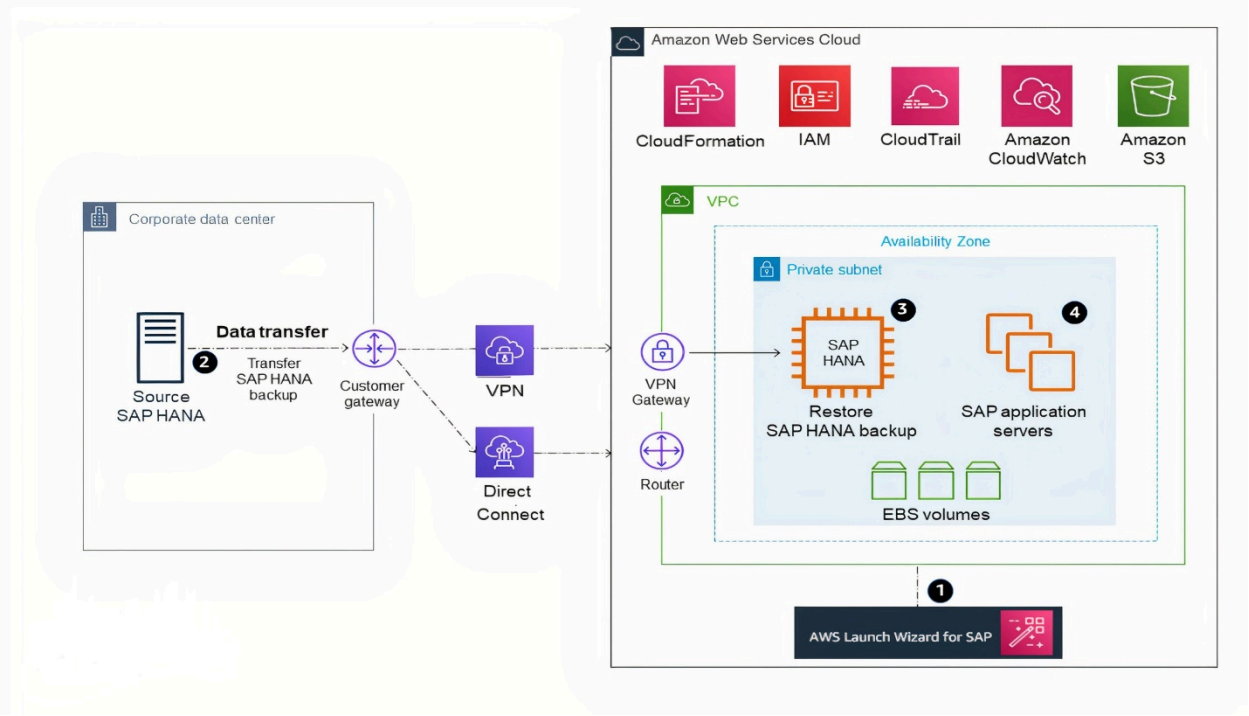
- Option 1: AWS Migration Hub Orchestrator
- Option 2: SAP HANA Backup and Restore
- Option 3: SAP HANA Classical Migration
- Option 4: SAP HANA HSR
- Option 5: SAP HANA HSR (with Initialization via Backup and Restore)
- Option 6: SAP HANA (on-premises) to SAP HANA (AWS Cloud)

Option 1: AWS Migration Hub Orchestrator

For details on how to migrate SAP HANA systems to AWS using AWS Migration Hub Orchestrator, see Migrate SAP NetWeaver based applications and SAP HANA databases to AWS.



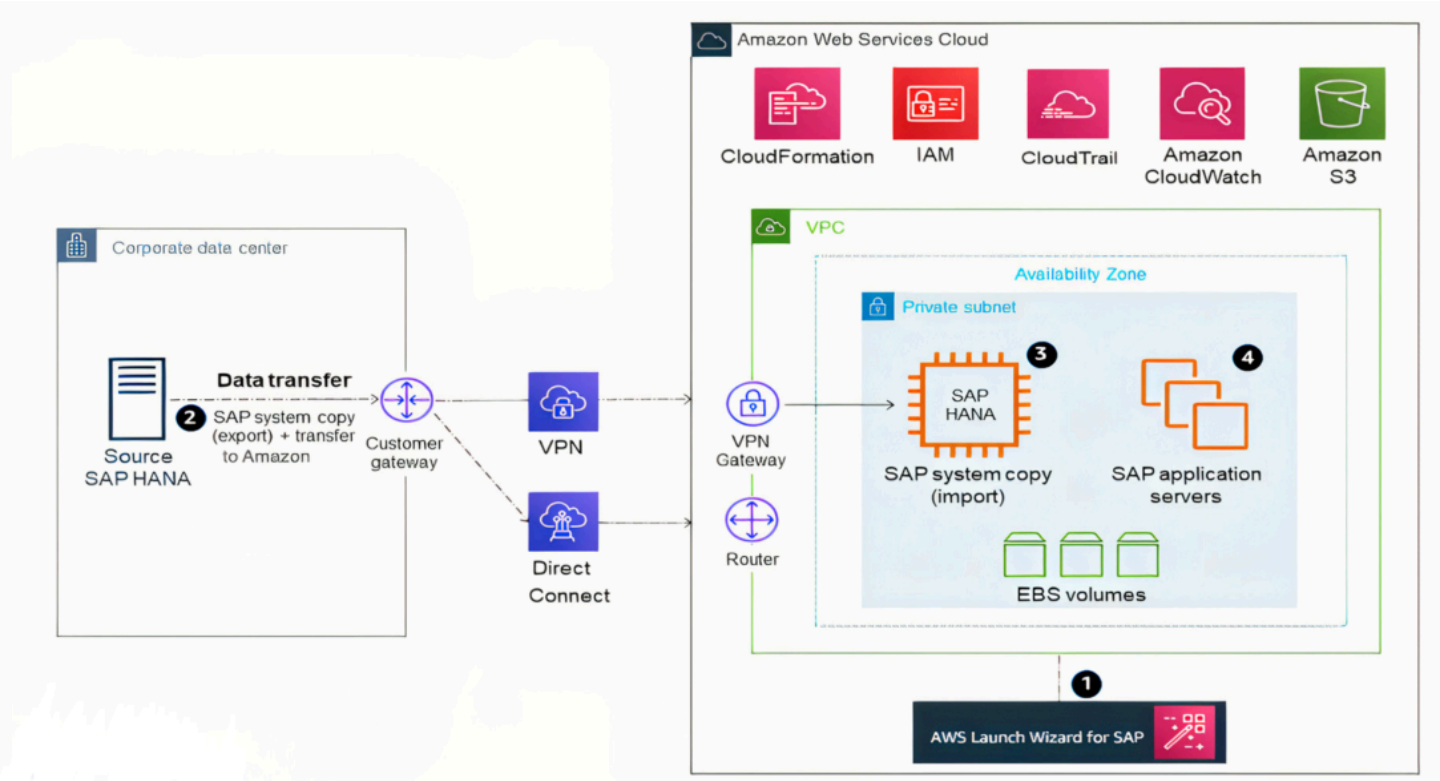
Option 2: SAP HANA Backup and Restore



1. Provision your SAP HANA system and landscape on AWS. (The AWS Launch Wizard for SAP can help expedite and automate this process for you.)

2. Transfer (**sftp** or **rsync**) a full SAP HANA backup, making sure to transfer any necessary SAP HANA logs for point-in-time recovery, from your source system to your target EC2 instance on AWS. A general tip here is to compress your files and split your files into smaller chunks to parallelize the transfer. If your transfer destination is Amazon S3, using the **aws s3 cp** command will automatically parallelize the file upload for you. For other options for transferring your data to AWS, see the AWS services listed previously in the Backup/Restore Tools section.
3. Recover your SAP HANA database.
4. Install your SAP application servers. (Skip this step if you used the AWS Launch Wizard for SAP in step 1.)
5. Depending on your application architecture, you might need to reconnect your applications to the newly migrated SAP HANA system.

Option 3: SAP HANA Classical Migration

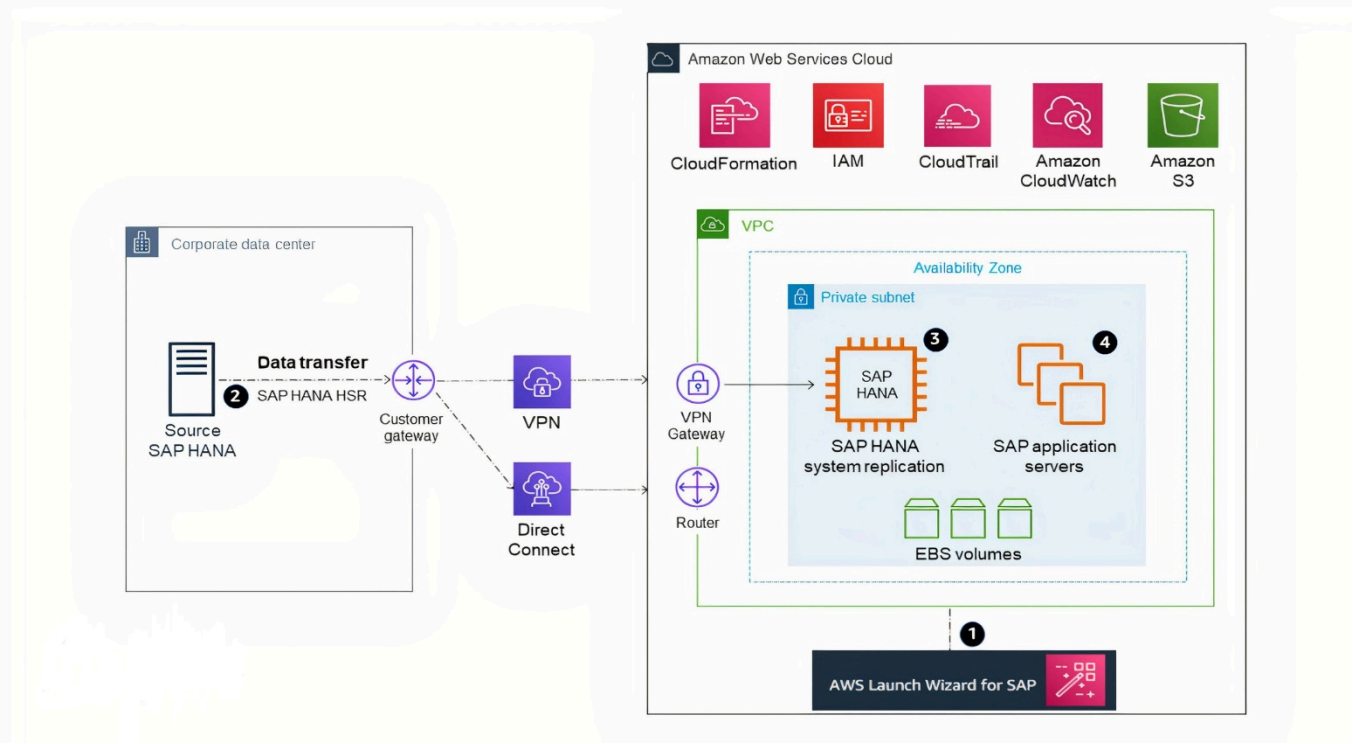


1. Provision your SAP HANA system and landscape on AWS. (The AWS Launch Wizard for SAP can help expedite and automate this process for you.)
2. Perform an SAP homogeneous system copy to export your source SAP HANA database. You may also choose to use a database backup as the export; see SAP Note 1844468 –

Homogeneous system copy on SAP HANA. When export is complete, transfer your data into AWS.

3. Continue the SAP system copy process on your SAP HANA system on AWS to import the data you exported in step 2.
4. Install your SAP application servers. (Skip this step if you used the AWS Launch Wizard for SAP in step 1.)
5. Depending on your application architecture, you might need to reconnect your applications to the newly migrated SAP HANA system.

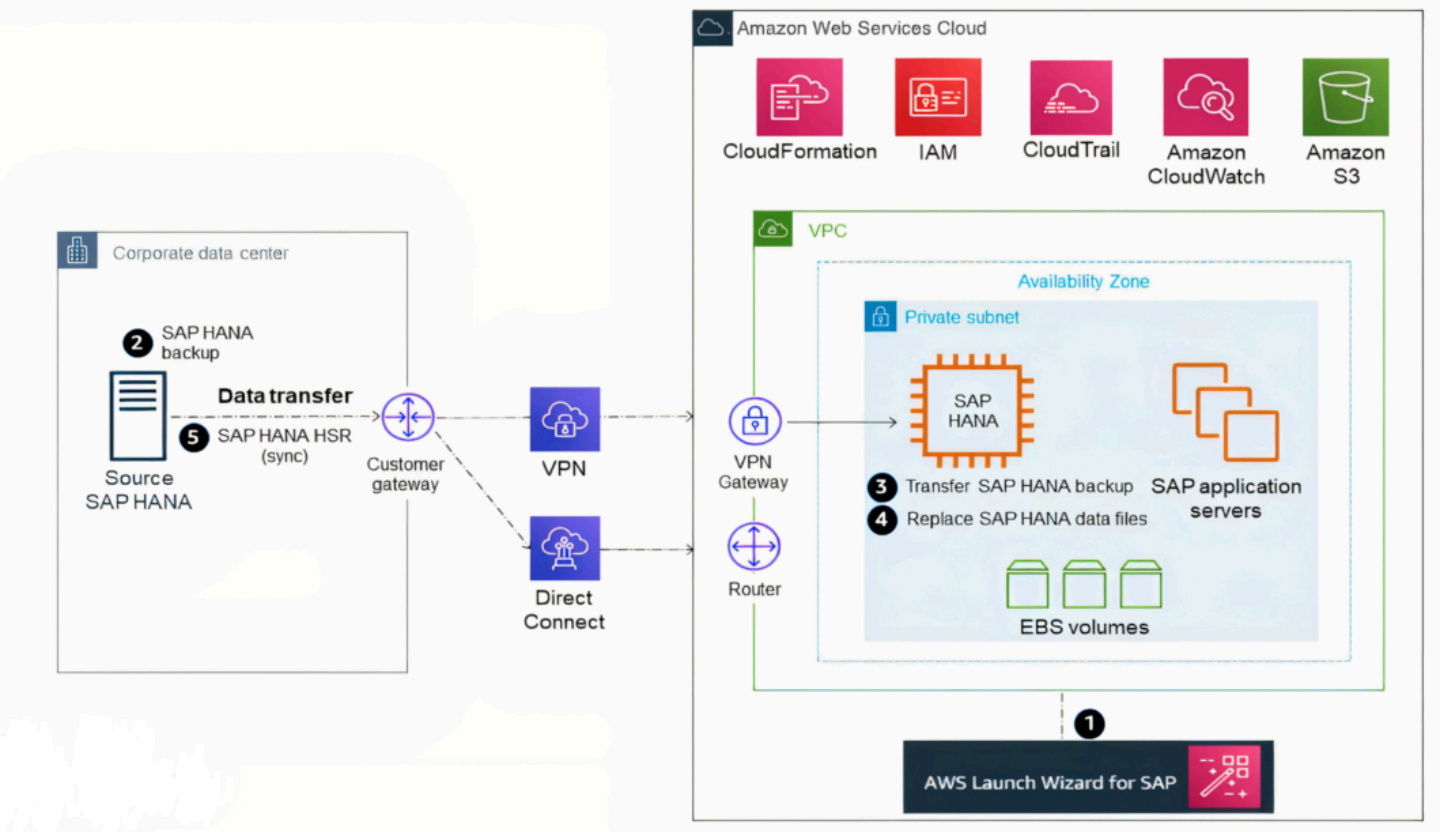
Option 4: SAP HANA HSR



1. Provision your SAP HANA system and landscape on AWS. (AWS Launch Wizard for SAP can help expedite and automate this process for you.) To save costs, you might choose to stand up a smaller EC2 instance type.
2. Establish asynchronous SAP HANA system replication from your source database to your standby SAP HANA database on AWS.
3. Perform an SAP HANA takeover on your standby database.
4. Install your SAP application servers. (Skip this step if you used AWS Launch Wizard for SAP in step 1.)

5. Depending on your application architecture, you might need to reconnect your applications to the newly migrated SAP HANA system.

Option 5: SAP HANA HSR (with Initialization via Backup and Restore)



1. Provision your SAP HANA system and landscape on AWS. (AWS Launch Wizard for SAP can help expedite and automate this process for you.) To save costs, you might choose to stand up a smaller EC2 instance type.
2. Stop the source SAP HANA database and obtain a copy of the data files (this is essentially a cold backup). After the files have been saved, you may start up your SAP HANA database again.
3. Transfer the SAP HANA data files to AWS, to the SAP HANA server you provisioned in step 1. (For example, you can store the data files in the /backup directory or in Amazon S3 during the transfer process.)
4. Stop the SAP HANA database on the target system in AWS. Replace the SAP HANA data files (on the target server) with the SAP HANA data files you transferred in step 3.

5. Start the SAP HANA system on the target system and establish asynchronous SAP HANA system replication from your source system to your target SAP HANA system in AWS.
6. Perform an SAP HANA takeover on your standby database.
7. Install your SAP application servers. (Skip this step if you used AWS Launch Wizard for SAP in step 1.)
8. Depending on your application architecture, you might need to reconnect your applications to the newly migrated SAP HANA system.

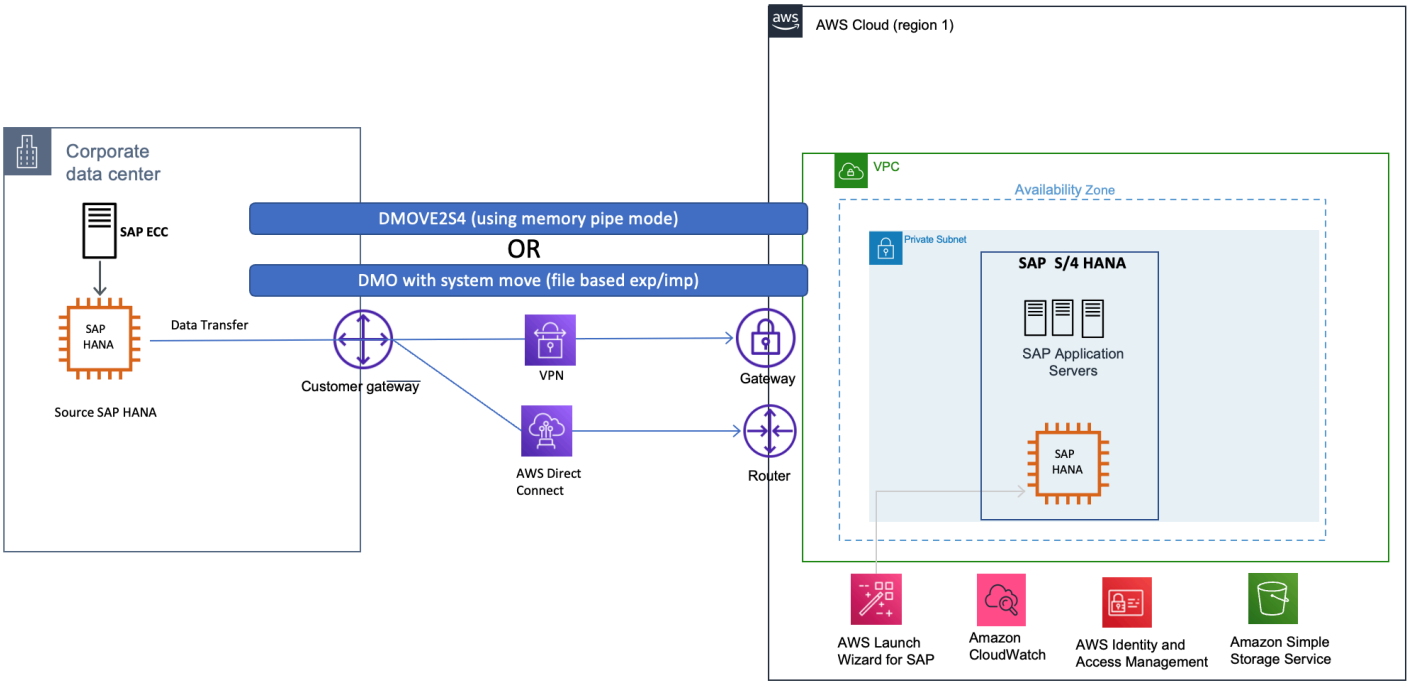
Option 6: SAP HANA (on-premises) to SAP HANA (AWS Cloud)

SAP added a new feature to the DMO option with SUM 2.0 SP-17 – DMO with system move, enabling SAP HANA to SAP HANA migration. It combines migration with conversion. With this option, you can convert SAP ERP on SAP HANA to SAP S/4HANA while migrating the system to SAP S/4HANA cloud, private edition on AWS Cloud.

DMOVE2S4 enables homogeneous migration (SAP HANA to SAP HANA). For this scenario, downtime optimized techniques, such as downtime optimized DMO (doDMO) or downtime optimized conversion (DOC) are currently not supported.

For more information, see SAP Blog - Two Major News with SUM 2.0 SP 17.

The following image displays this option.



Migrating SAP HANA on AWS to an EC2 High Memory Instance

EC2 High Memory instances are built on AWS Nitro System with up to 24TB of memory in a single instance to deliver scalable and elastic infrastructure capabilities for large in-memory databases, such as SAP HANA.

For SAP HANA workloads, EC2 High Memory instances support SUSE Linux Enterprise Server for SAP Applications (SLES for SAP) and Red Hat Enterprise Linux for SAP Solutions (RHEL for SAP) operating systems. The following table provides the minimum supported operating system version for SAP HANA workloads.

Instance Type	Supported operating system version
u-6tb1.metal, u-9tb1.metal, and u-12tb1.metal	SLES for SAP 12 SP3 and above; RHEL for SAP 7.4 and above
u-18tb1.metal and u-24tb1.metal	SLES for SAP 12 SP4 and above; RHEL for SAP 8.1 and above
u-3tb1.56xlarge	SLES for SAP 12 SP3 and above; RHEL for SAP 7.4 and above
u-6tb1.56xlarge	SLES for SAP 12 SP3 and above; RHEL for SAP 7.4, RHEL for SAP 7.7 and above
u-6tb1.112xlarge, u-9tb1.112xlarge, u-12tb1.112xlarge, u-18tb1.112xlarge, and u-24tb1.112xlarge	SLES for SAP 12 SP4 and above; RHEL for SAP 8.1 and above

See the SAP HANA hardware directory for a list of supported operating systems for your instance type.

Important

If you are using `u-*tb1.112xlarge` instance types with one of the following operating system version, verify that your system has the minimum required kernel version in order to use all available vCPUs.

- SLES for SAP 12 SP4 – 4.12.14-95.68
- SLES for SAP 12 SP5 – 4.12.14-122.60
- SLES for SAP 15 – 4.12.14-150.66
- SLES for SAP 15 SP1 – 4.12.14-197.83
- SLES for SAP 15 SP2 – 5.3.18-24.52
- RHEL for SAP 8.1 - 4.18.0-147.44.1.el8_1
- RHEL for SAP 8.2 - 4.18.0-193.47.1.el8_2

Note

`u-*tb1.metal` instances can be launched only as Amazon EC2 Dedicated Hosts with host tenancy. `u-6tb1.56xlarge` and `u-*tb1.112xlarge` instances can be launched with default, dedicated or host tenancy.
Before you start your migration, if you plan to use `u-*tb1.metal` instances, make sure that an `u-*tb1.metal` instance is allocated to your target account, Availability Zone, and AWS Region. If you plan to use `u-6tb1.56xlarge` or `u-*tb1.112xlarge`, ensure your account limit for resource "On-Demand High Memory instances" or "U*TB1 Dedicated Hosts" (required only if you intend to use it as dedicated host) is set appropriately. If needed, submit a request from AWS console to increase your account limit. For more information, see Amazon EC2 service quotas and On-Demand Instance limits in the AWS documentation.

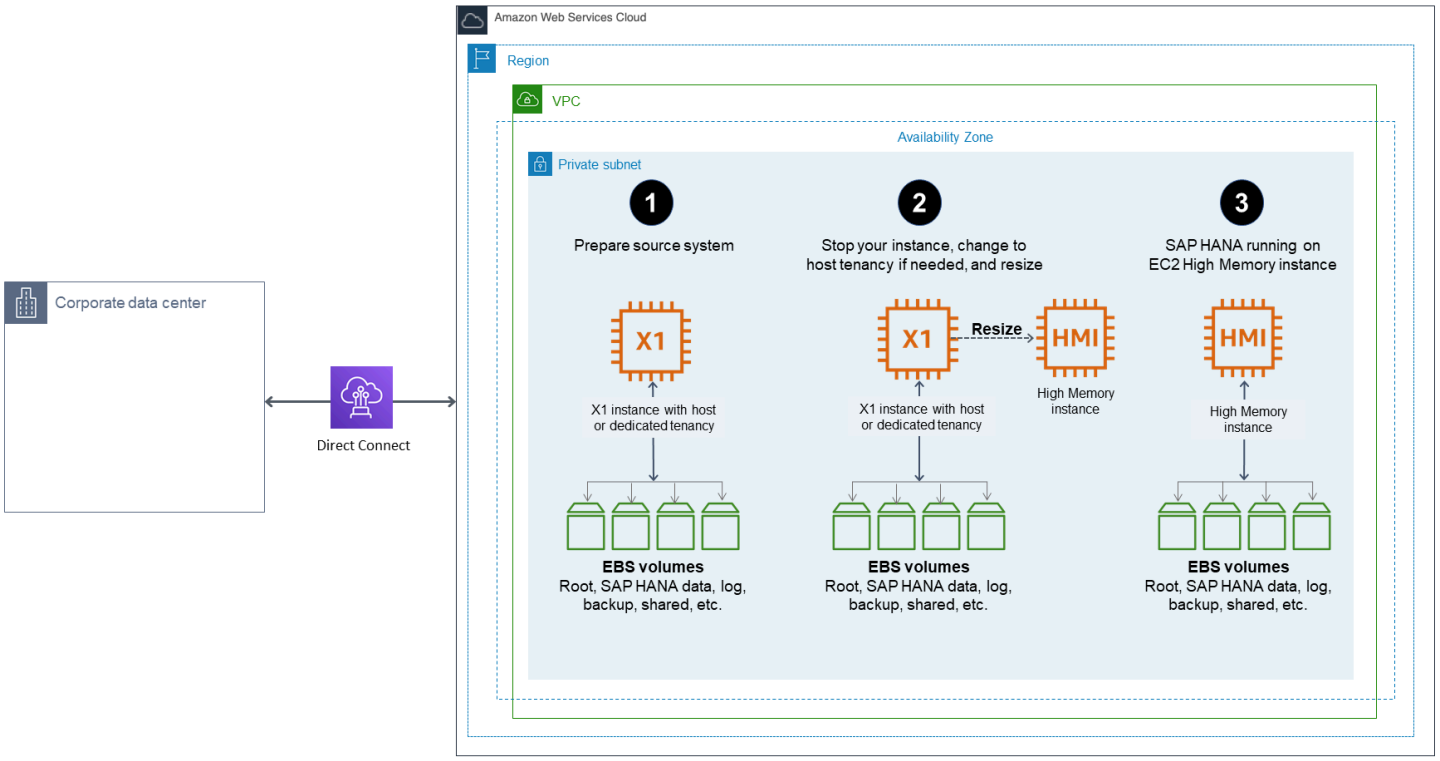
You have several options for migrating your existing SAP HANA workload on AWS to an EC2 High Memory instance, as discussed in the following sections.

In the following sections, we show X1 instance as the source instance type for migration. These procedures are applicable for any other source instance types as well.

Option 1: Resizing an Existing EC2 Instance with Host or Dedicated Tenancy

If your existing EC2 instance is running with host or dedicated tenancy, you can follow the steps in this section to migrate it to `u-*tb1.metal` EC2 High Memory instance. With this option, all your

instance properties, including IP addresses, hostnames, and EBS volumes, remain the same after migration.



1. Verify that your source system is running on a supported operating system version. If not, you might have to upgrade your operating system before resizing to an EC2 High Memory instance.
2. EC2 High Memory instances are based on the Nitro system. On Nitro-based instances, EBS volumes are presented as NVMe block devices. If your source system has any mount point entries in `/etc/fstab` with reference to block devices such as `/dev/xvd<x>`, you need to create a label for these devices and mount them by label before migrating to EC2 High Memory instances. Otherwise, you will face issues when you start SAP HANA on an EC2 High Memory instance.
3. Verify that you don't exceed the maximum supported EBS volumes to your instance. An `-*tb1.metalu` EC2 High Memory instance currently supports up to 19 EBS volumes. `u-6tb1.56xlarge` and `u-*tb1.112xlarge` instances supports up to 27 EBS volumes. For details, see Instance Type Limits in the AWS documentation.
4. When you are ready to migrate, make sure that you have a good backup of your source system. You can use AWS Backint Agent for SAP HANA to easily backup your SAP HANA database to Amazon S3. For details, see AWS Backint Agent for SAP HANA in the AWS documentation.
5. Stop the source instance in the Amazon EC2 console or by using the AWS CLI.

6. If your source EC2 instance is running with dedicated tenancy, modify the instance placement to host tenancy. For instructions, see Modifying instance Tenancy and Affinity in the AWS documentation. Skip this step if your instance is running with host tenancy.
7. Modify the instance placement of your existing instance to your target EC2 High Memory Dedicated Host through the Amazon EC2 console or the AWS CLI. For details, see modify-instance-placement in the AWS documentation.
8. Change your instance type to the desired EC2 High Memory instance type (for example, `u-12tb1.metal` or `u-12tb1.112xlarge`) through the AWS CLI or AWS Console.

   > **Note**
   >
   > You can change the instance type to `u-*tb1.metal` only through the AWS CLI or Amazon EC2 API.

9. Start your instance in the Amazon EC2 console or by using the AWS CLI.
10. When you increase the memory of your SAP HANA system, you might need to adjust the storage size of SAP HANA data, log, shared, and backup volumes as well to accommodate data growth and to get improved performance. For details, see SAP HANA on AWS Operations Guide.
11. Start your SAP HANA database and perform your validation.
12. Complete any SAP HANA-specific post-migration activities.
13. Complete any AWS-specific post-migration activities, such as setting up Amazon CloudWatch, AWS Config, and AWS CloudTrail.
14. Configure your SAP HANA system for high availability on the EC2 High Memory instance with SAP HANA HSR and clustering software, and test it.
15. Complete post-migration tasks to ensure that you are not charged.

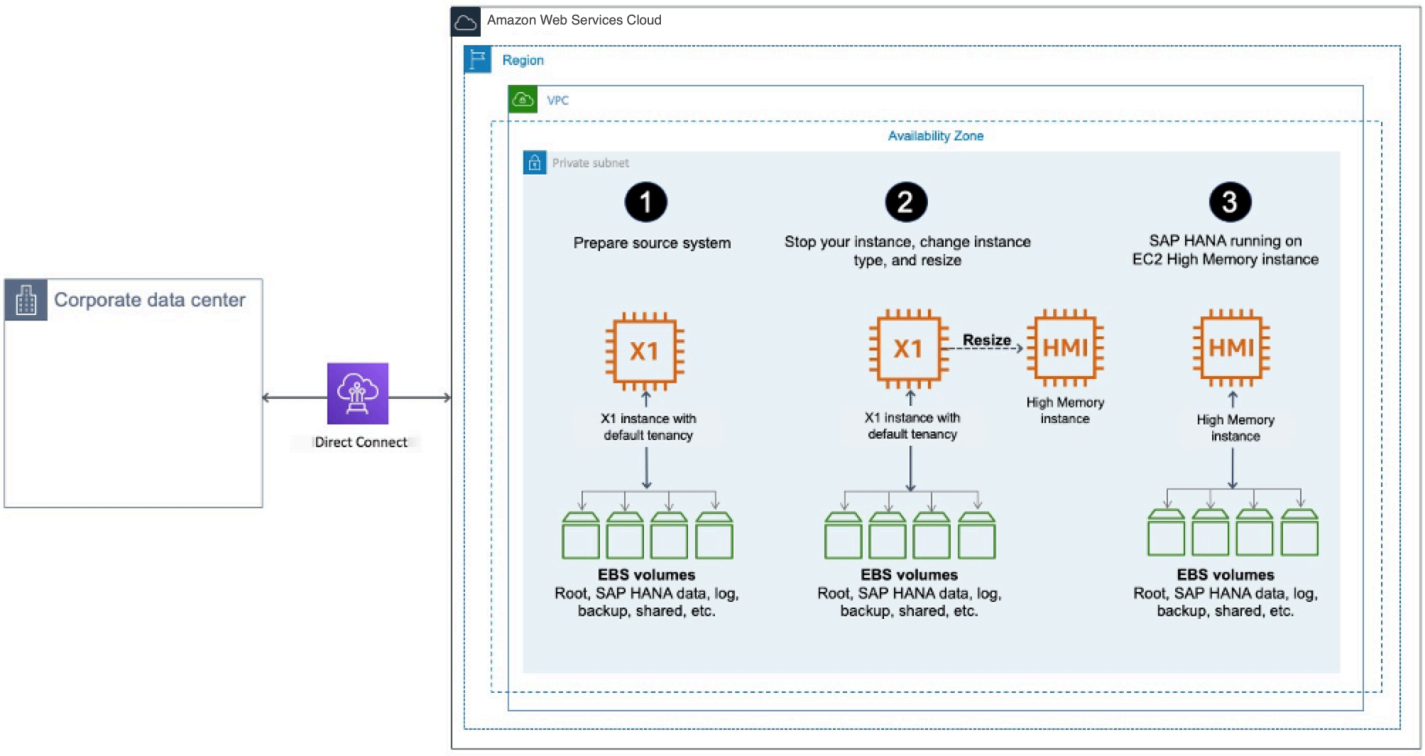
    - Review and confirm if you need to cancel reservations once migration is complete.
    - Review and confirm if you need to release Amazon EC2 Dedicated Hosts through the console. Once a reservation is cancelled, on-demand charging begins for the dedicated hosts until they are released from the console.

Option 2: Migrating from an Existing EC2 Instance with Default Tenancy

If your existing EC2 instance is running with default tenancy, you have multiple options to migrate it to an EC2 High Memory instance: If you plan to use `u-6tb1.56xlarge` or `u-*tb1.112xlarge` instance types, you can simply stop your instance and resize it to desired target instance size. Additionally, if you plan to use `u-*tb1.metal` instances, you can use an Amazon Machine Image (AMI) to launch your `u-*tb1.metal` EC2 High Memory instance with host tenancy, or you can set up a new SAP HANA on EC2 High Memory instance and then copy the data over from your source system.

Option 2(a): Resizing an existing EC2 instance

In this option, if you are using `u-6tb1.56xlarge` or `u-*tb1.112xlarge` instance types, you can simply resize your instance through AWS Management Console or AWS CLI.



1. Verify that your source system is running on a supported operating system version. If it isn't, you might have to upgrade your operating system before resizing to an EC2 High Memory instance.

2. EC2 High Memory instances are based on the Nitro system. On Nitro-based instances, EBS volumes are presented as NVMe block devices. If your source system has any mount point entries in `/etc/fstab` with reference to block devices such as `/dev/xvd<x>`, you need to create a label for these devices and mount them by label before migrating to EC2 High Memory instances. Otherwise, you will face issues during instance launch.
3. EC2 High Memory instances are based on the Nitro system. On Nitro-based instances, EBS volumes are presented as NVMe block devices. If your source system has any mount point entries in `/etc/fstab` with reference to block devices such as `/dev/xvd<x>`, you need to create a label for these devices and mount them by label before migrating to EC2 High Memory instances. Otherwise, you will face issues when you start SAP HANA on an EC2 High Memory instance.
4. When you are ready to migrate, verify that you have a good backup of your source system.
5. Stop the source instance in the Amazon EC2 console or by using the AWS CLI.
6. Change the instance type to target EC2 High Memory instance size such as `u-6tb1.56xlarge` or `u-*tb1.112xlarge`
7. When you increase the memory of your SAP HANA system, you might need to adjust the storage size of SAP HANA data, log, shared, and backup volumes as well to accommodate data growth and to get improved performance. For details, see the SAP HANA on AWS Operations Guide.
8. Start your SAP HANA database and perform your validation.

   Note

   If necessary, complete any SAP HANA-specific post-migration activities.

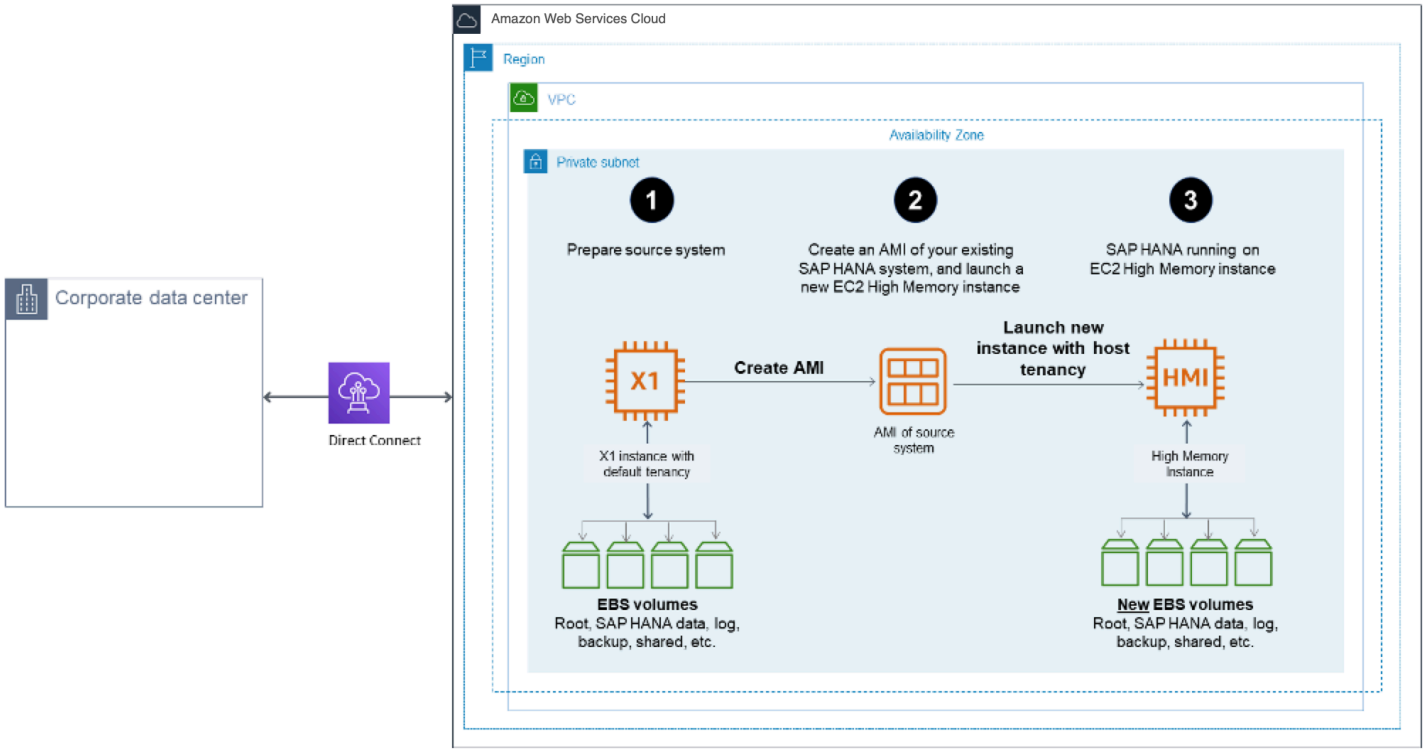
9. Check the connectivity between your SAP application servers and the new SAP HANA instance.
10. If necessary, complete any AWS-specific post-migration activities, such as setting up Amazon CloudWatch, AWS Config, and AWS CloudTrail.
11. Configure your SAP HANA system for high availability on the EC2 High Memory instance with SAP HANA HSR and clustering software, and test it.
12. Complete post-migration tasks to ensure that you are not charged.

    - Review and confirm if you need to cancel reservations once migration is complete.

- Review and confirm if you need to release Amazon EC2 Dedicated Hosts through the console. Once a reservation is cancelled, on-demand charging begins for the dedicated hosts until they are released from the console.

Option 2(b): Migrating Using an AMI

In this option, you launch a new EC2 High Memory instance based on the AMI that you created from your source system for the migration.



1. Verify that your source system is running on a supported operating system version. If it isn't, you might have to upgrade your operating system before resizing to an EC2 High Memory instance.
2. EC2 High Memory instances are based on the Nitro system. On Nitro-based instances, EBS volumes are presented as NVMe block devices. If your source system has any mount point entries in `/etc/fstab` with reference to block devices such as `/dev/xvd<x>`, you need to create a label for these devices and mount them by label before migrating to EC2 High Memory instances. Otherwise, you will face issues when you start SAP HANA on an EC2 High Memory instance.
3. When you are ready to migrate, verify that you have a good backup of your source system.
4. Stop the source instance in the Amazon EC2 console or by using the AWS CLI.

5. Create an AMI of your source instance. For details, see Creating an Amazon EBS-Backed Linux AMI in the AWS documentation.

   > **Tip**
   >
   > Creating an AMI for the first time with the attached EBS volumes could take a long time, depending on your data size. To expedite this process, we recommend that you take snapshots of EBS volumes attached to the instance ahead of time.

6. Launch a new EC2 High Memory instance with host tenancy for `u-*tb1.metal` instances. For `u-6tb1.56xlarge` and `u-*tb1.112xlarge`, you can launch a new EC2 High Memory instance with default, dedicated or host tenancy.
7. The new instance will have a new IP address. Update all references to the IP address of the source system, including the /etc/hosts file for the operating system and DNS entries, to reflect the new IP address. The hostname and storage layout will remain the same as on the source system.
8. When you increase the memory of your SAP HANA system, you might need to adjust the storage size of SAP HANA data, log, shared, and backup volumes as well to accommodate data growth and to get improved performance. For details, see the SAP HANA on AWS Operations Guide.
9. Start your SAP HANA database and perform your validation.

   > **Note**
   >
   > You might notice that SAP HANA is slow when loading data into memory for the first time after you create your instance with an AMI. This is expected behavior when EBS volumes associated with SAP HANA data are created from a snapshot. You will not experience the slowness after the initial hydration.

10. Complete any SAP HANA-specific post-migration activities.
11. Check the connectivity between your SAP application servers and the new SAP HANA instance.
12. Complete any AWS-specific post-migration activities, such as setting up Amazon CloudWatch, AWS Config, and AWS CloudTrail.
13. Configure your SAP HANA system for high availability on the EC2 High Memory instance with SAP HANA HSR and clustering software, and test it.
14. Complete post-migration tasks to ensure that you are not charged.

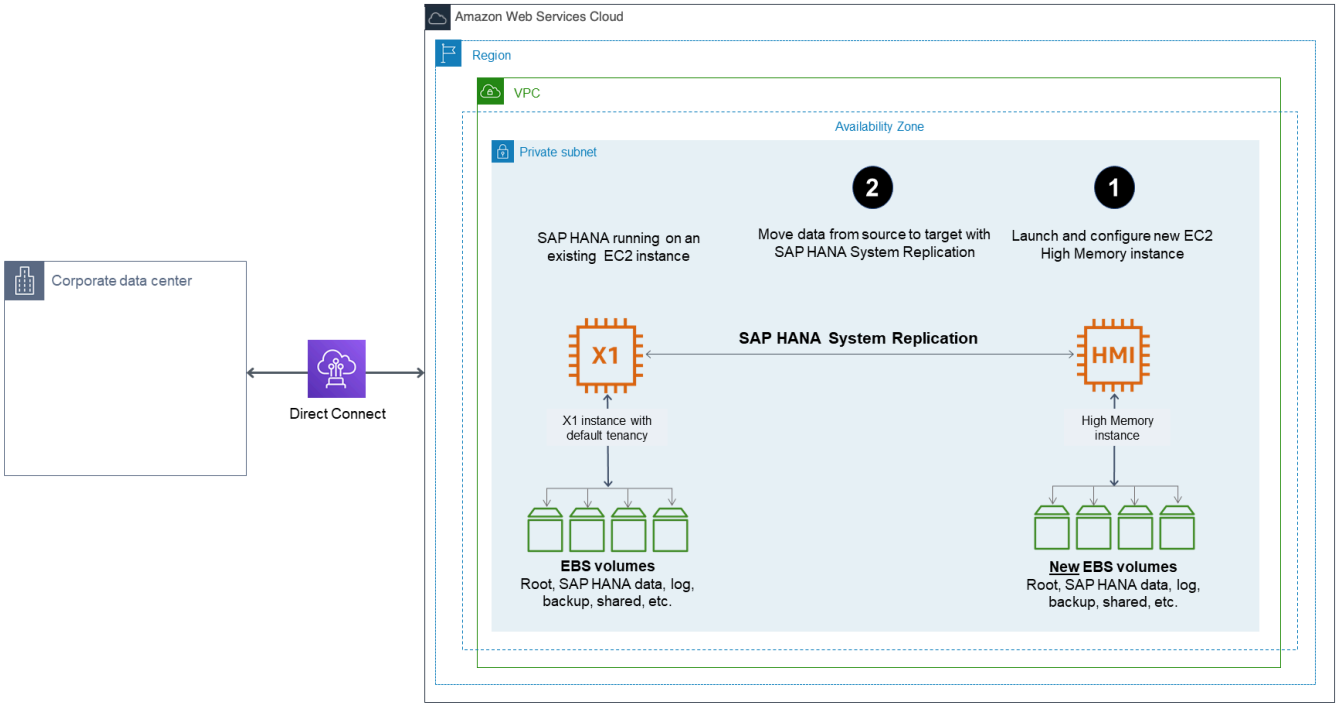
- Review and confirm if you need to cancel reservations once migration is complete.
- Review and confirm if you need to release Amazon EC2 Dedicated Hosts through the console. Once a reservation is cancelled, on-demand charging begins for the dedicated hosts until they are released from the console.

Option 2(c): Migrating Using SAP HANA HSR or SAP HANA backup and restore

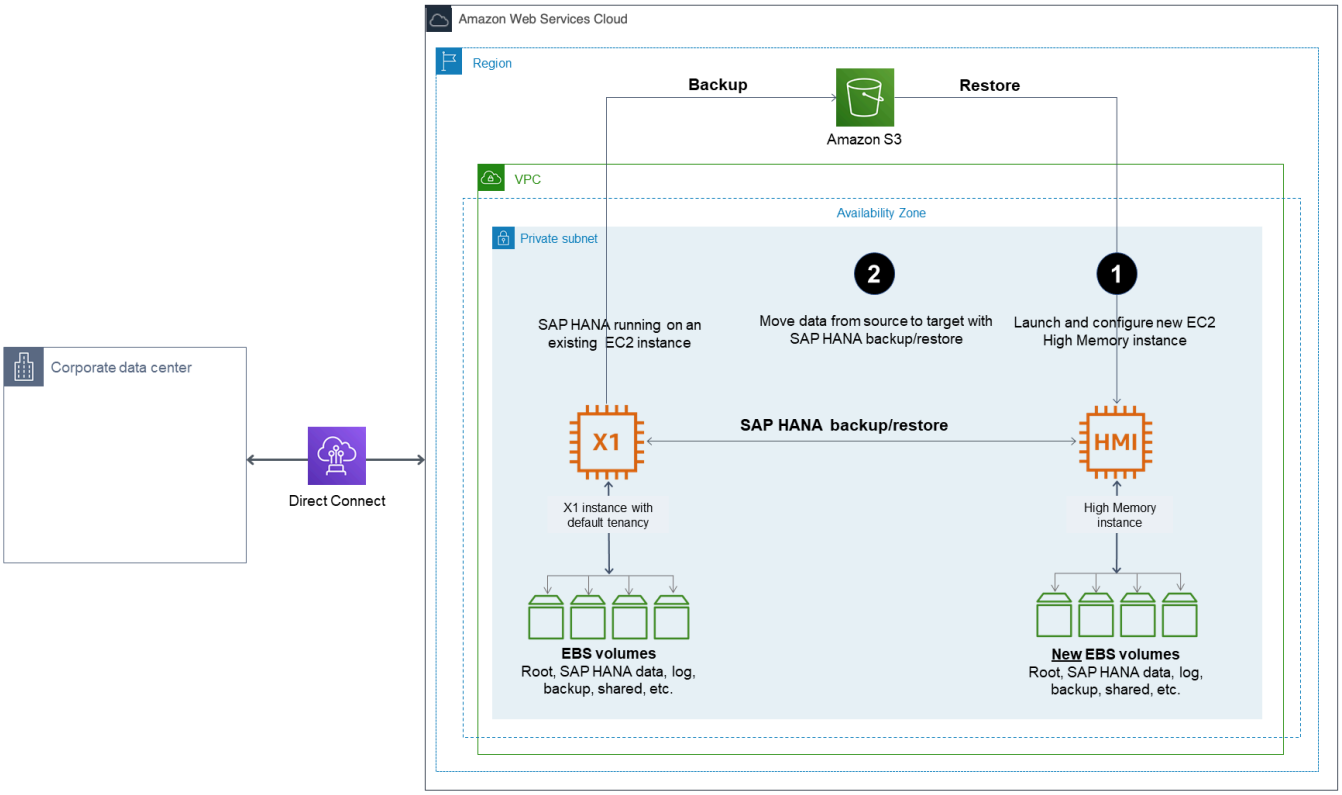
In this option, you launch a new EC2 High Memory instance, install and configure SAP HANA on the instance, and then copy the data over from your source system to complete the migration.

1. Launch a new SAP HANA EC2 High Memory instance with host tenancy for `u-*tb1.metal` instances. For `u-6tb1.56xlarge` and `u-*tb1.112xlarge`, you can launch your instance with default, dedicated or host tenancy. You can use the AWS Launch Wizard for SAP to set up your instance automatically, or follow the SAP HANA Environment Setup on AWS guide to set up your instance manually. Make sure that you are using an operating system that supports EC2 High Memory instances.
2. Complete any AWS-specific post-migration activities, such as setting up Amazon CloudWatch, AWS Config, and AWS CloudTrail, ahead of time.
3. Migrate the data from your existing SAP HANA instance by using SAP HANA HSR or SAP HANA backup and restore tools.

    - If you plan to use SAP HANA HSR for data migration, configure HSR to move data from your source system to your target system. For details, see the SAP HANA Administration Guide from SAP.



- If you plan to use the SAP HANA backup and restore feature to migrate your data, back up your source SAP HANA system. When backup is complete, move the backup data to your target system and perform a restore in your target system. If you back up your source SAP HANA system directly to Amazon S3 using AWS Backint Agent for SAP HANA, you can directly restore it in the target system from Amazon S3. For details, see the AWS Backint Agent for SAP HANA in the AWS documentation.



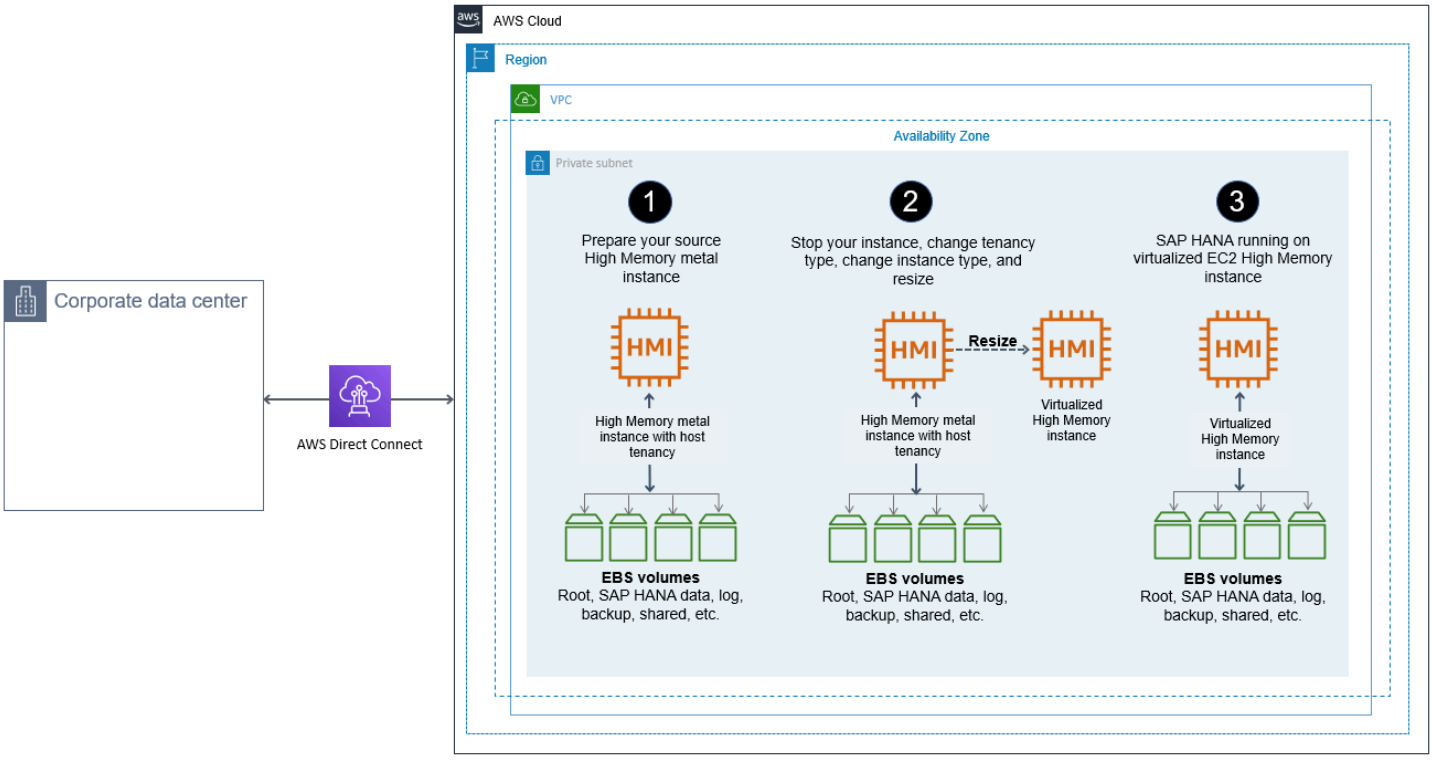
4. Stop your source system, complete any additional post-migration steps, like updating DNS and checking the connectivity between your SAP application servers and the new SAP HANA instance.
5. Configure your SAP HANA system for high availability on the EC2 High Memory instance with SAP HANA HSR and clustering software, and test it.
6. Complete post-migration tasks to ensure that you are not charged.

    - Review and confirm if you need to cancel reservations once migration is complete.
    - Review and confirm if you need to release Amazon EC2 Dedicated Hosts through the console. Once a reservation is cancelled, on-demand charging begins for the dedicated hosts until they are released from the console.

Option 3: Migrating from Amazon EC2 High Memory metal instance with Virtualized High Memory Host Tenancy

If your existing Amazon EC2 High Memory metal instance (`u*-tb1.metal`) is running with host tenancy, you can easily migrate it to virtualized high memory instance (`u-*tb*.56xlarge` or `u-*tb*.112xlarge`). Stop your instance to change the tenancy and instance type, and then resize

it to the desired target virtualized High Memory instance size. This architecture of this option is as shown in the following image.



1. Verify that your source system is running on a supported operating system version. If not, you might have to upgrade your operating system before resizing to an EC2 High Memory instance.
2. If you built your source High Memory metal instance with AWS Marketplace based image, such as SLES for SAP or RHEL for SAP, ensure that your target virtualized High Memory instance size is listed as supported instance type in the AWS Marketplace product page of your chosen image.
3. When you are ready to migrate, make sure that you have a good backup of your source system.
4. Stop the source instance in the Amazon EC2 console or by using the AWS CLI.
5. Change the tenancy type from host to default using AWS CLI. For more information, see Tenancy conversion.
6. Change your instance type to target High Memory instance type, such as `u-*tb*.56xlarge` or `u-*tb*.112xlarge` through the AWS CLI or AWS Console.
7. When you increase the memory of your SAP HANA system, you might need to adjust the storage size of SAP HANA data, log, shared, and backup volumes as well to accommodate data growth and to get improved performance. For details, see SAP HANA on AWS Operations Guide.

8. Enable Amazon EC2 automatic recovery to recover your instance when a system status check failure occurs. For more information, see Recover your instance.
9. Start your SAP HANA database and perform your validation.

> **Note**
>
> If necessary, complete any SAP HANA-specific post-migration activities.

10. Check the connectivity between your SAP application servers and the new SAP HANA instance.
11. If necessary, complete any AWS-specific post-migration activities, such as setting up Amazon CloudWatch, AWS Config, and AWS CloudTrail.
12. Complete post-migration tasks to ensure that you are not charged.

    - Review and confirm if you need to cancel reservations once migration is complete.
    - Review and confirm if you need to release Amazon EC2 Dedicated Hosts through the console. Once a reservation is cancelled, on-demand charging begins for the dedicated hosts until they are released from the console.

Security

In the AWS Cloud Adoption Framework (CAF), security is a perspective that focuses on subjects such as account governance, account ownership, control frameworks, change and access management, and other security best practices. We recommend that you become familiar with these security processes when planning any type of migration. In some cases, you might need to get sign-off from your internal IT audit and security teams before you start your migration project or during migration. See the CAF security whitepaper for a deeper dive into each of these topic areas.

Additionally, there are AWS services that help you secure your systems in AWS. For example, AWS CloudTrail, Amazon CloudWatch, and AWS Config can help you secure your AWS environment.

See the following AWS blog posts for help analyzing and evaluating architectures and design patterns for the VPC setup and configuration of your SAP landscape.

- VPC Subnet Zoning Patterns for SAP on AWS, Part 1: Internal-Only Access
- VPC Subnet Zoning Patterns for SAP on AWS, Part 2: Network Zoning
- VPC Subnet Zoning Patterns for SAP on AWS, Part 3: Internal and External Access

Beyond VPC and network security, SAP HANA systems require routine maintenance to remain secure, reliable, and available; see the SAP HANA operations overview for specific recommendations in this topic area.

Document history

Date	Change
April, 2023	Updates to the entire guide
May, 2021	Added new migration scenario for EC2 High Memory instances
June, 2019	Added new migration scenario for EC2 High Memory instances, and updated to reflect latest information
August, 2018	Initial publication

SAP HANA Environment Setup on AWS

Last updated: December 2022

This guide is part of a content series that provides detailed information about hosting, configuring, and using SAP technologies in the AWS Cloud. For the other guides in the series, ranging from overviews to advanced topics, see the SAP on AWS Technical Documentation home page.

This document provides guidance on how to set up AWS resources and configure SUSE Linux Enterprise Server (SLES) and Red Hat Enterprise Linux (RHEL) operating systems to deploy SAP HANA on Amazon Elastic Compute Cloud (Amazon EC2) instances in an existing virtual private cloud (VPC). It includes instructions for configuring storage for scale-up and scale-out workloads with Amazon Elastic Block Store (Amazon EBS), Amazon Elastic File System (Amazon EFS), and Amazon FSx for NetApp ONTAP (FSx for ONTAP).

This document follows AWS best practices to ensure that your system meets all key performance indicators (KPIs) that are required for Tailored Data Center Integration (TDI)–based SAP HANA implementations on AWS. In addition, this document also follows recommendations provided by SAP, SUSE, and Red Hat for SAP HANA in the following SAP OSS Notes (requires SAP portal access).

- 1944799 - SAP HANA Guidelines for SLES Operating System Installation
- 2205917 - SAP HANA DB: Recommended OS settings for SLES 12 / SLES for SAP Applications 12
- 2684254 - SAP HANA DB: Recommended OS settings for SLES 15 / SLES for SAP Applications 15
- 2009879 - SAP HANA Guidelines for Red Hat Enterprise Linux (RHEL) Operating System
- 2292690 - SAP HANA DB: Recommended OS settings for RHEL 7
- 2777782 - SAP HANA DB: Recommended OS Settings for RHEL 8

> **Note**
>
> SAP, SUSE, and Red Hat regularly updates these OSS notes. Review the latest version of the OSS notes for up-to-date information before proceeding.

This guide is intended for users with a good understanding of AWS services, network concepts, the Linux operating system and SAP HANA administration to successfully launch and configure the resources that are required for SAP HANA.

AWS Launch Wizard for SAP is a service that guides you through the sizing, configuration and deployment of SAP HANA based applications on AWS, and follows the best practices from AWS, SAP, and operating system vendors, including SUSE and Red Hat. AWS Launch Wizard for SAP supports a wide range of deployment models, including SAP HANA database in a scale-up and scale-out mode with cross-Availability Zone high availability. AWS Launch Wizard for SAP enables you to setup your SAP HANA based systems in a few hours with minimal manual intervention. For more information, see AWS Launch Wizard for SAP.

If your organization can't use AWS Launch Wizard for SAP for the deployment and you require additional customization to meet internal policies, you can follow the steps in this document to manually set up AWS resources such as Amazon EC2, Amazon EBS, Amazon EFS, and FSx for ONTAP by using the AWS Command Line Interface (AWS CLI) or the AWS Management Console.

This document doesn't provide guidance on how to set up network and security constructs such as Amazon VPC, subnets, route tables, access control lists (ACLs), NAT Gateway, AWS Identity and Access Management (IAM) roles, security groups, etc. Instead, this document focuses on configuring compute, storage, and operating system resources for SAP HANA deployment on AWS.

Prerequisites

Specialized Knowledge

If you are new to AWS, see Getting Started with AWS.

Technical Requirements

1. If necessary, request a service limit increase for the instance type that you're planning to use for your SAP HANA system. If you already have an existing deployment that uses this instance type, and you think you might exceed the default limit with this deployment, you will need to request an increase. For details, see Amazon EC2 Service Limits in the AWS documentation.
2. Ensure that you have a key pair that you can use to launch your Amazon EC2 instance. If you need to create or import a key pair, refer to Amazon EC2 Key Pairs in the AWS documentation.
3. Ensure that you have the network details of the VPC, such as VPC ID and subnet ID, where you plan to launch the Amazon EC2 instance that will host SAP HANA.
4. Ensure that you have a security group to attach to the Amazon EC2 instance that will host SAP HANA and that the required ports are open. If needed, create a new security group that allows

the traffic for SAP HANA ports. For additional details on the list of ports, see Security groups in AWS Launch Wizard for SAP.

5. If you intend to use AWS CLI to launch your instances, ensure that you have installed and configured AWS CLI with the necessary credentials. For details, see Installing the AWS Command Line Interface in the AWS documentation.
6. If you intend to use the console to launch your instances, ensure that you have credentials and permissions to launch and configure Amazon EC2, Amazon EBS, and other services. For details, see Access Management in the AWS documentation.

Plan the deployment

Consider the following when planning your SAP HANA deployment.

Topics

- Compute
- Operating System
- Amazon Machine Image (AMI)
- Storage
- Network

Compute

AWS provides multiple instance families with different sizes to run SAP HANA workloads. See the SAP Certified and Supported SAP HANA Hardware Directory and the Amazon EC2 Instance Types for SAP page to find the list of certified Amazon EC2 instances. For your production workloads, ensure that you choose an instance type that has been certified by SAP. You can run your non-production workloads on any size of a particular certified instance family to save costs.

Operating System

You can deploy your SAP HANA workload on SLES, SLES for SAP, RHEL for SAP with high availability and Update Services or RHEL for SAP Solutions.

SLES for SAP and RHEL for SAP with high availability and US products are available in AWS Marketplace under an hourly or an annual subscription model.

SLES for SAP

SLES for SAP provides additional benefits, including Extended Service Pack Overlap Support (ESPOS), configuration and tuning packages for SAP applications, and High Availability Extensions (HAE). For details, see the SUSE SLES for SAP product page to learn more about the benefits of using SLES for SAP. We strongly recommend using SLES for SAP instead of SLES for all your SAP workloads.

If you plan to use Bring Your Own Subscription (BYOS) images provided by SUSE, ensure that you have the registration code required to register your instance with SUSE to access repositories for software updates.

RHEL for SAP

RHEL for SAP with high availability and Update services provides access to Red Hat Pacemaker cluster software for High Availability, extended update support, and the libraries that are required to run SAP HANA. For details, see the RHEL for SAP Offerings on AWS FAQ in the Red Hat Knowledgebase.

If you plan to use the BYOS model with RHEL, either through the Red Hat Cloud Access program or another means, ensure that you have access to a RHEL for SAP Solutions subscription. For details, see Overview of Red Hat Enterprise Linux for SAP Solutions subscription in the Red Hat Knowledgebase.

Amazon Machine Image (AMI)

A base AMI is required to launch an Amazon EC2 instance. Depending on your choice of operating system, ensure that you have access to the appropriate AMI in your target region for the deployment.

If you plan to use the SLES for SAP or RHEL for SAP Amazon Machine Images (AMIs) offered in AWS Marketplace, ensure that you have completed the subscription process. You can search for *SLES for SAP* or *RHEL for SAP* in the AWS Marketplace, and follow the instructions to complete your subscription.

If you are using AWS CLI, you will need to provide the AMI ID when you launch the instance.

Storage

Deploying SAP HANA on AWS requires specific storage size and performance to ensure that SAP HANA data and log volumes both meet the SAP KPIs and sizing recommendations. Refer the SAP

HANA on AWS Operations Guide to understand the storage configuration details for different instance types. You need to configure your storage based on these recommendations during instance launch. If you plan to use FSx for ONTAP storage, see SAP HANA on AWS with FSx for ONTAP for more details.

Network

Ensure that your network constructs are set up to deploy resources related to SAP HANA. If you haven't already set up network components such as Amazon VPC, subnets, route table, etc., you can use the AWS Modular and Scalable VPC reference deployment to easily deploy a scalable VPC architecture in minutes. For details, see the reference deployment guide.

Configure the operating system

This section includes instructions for configuring your operating system for SAP HANA.

Topics

- Configure SLES 12/15 for SAP
- Configure RHEL 7/8/9 for SAP

> **Note**
>
> For scale-out workloads, you must repeat these steps for every node in the cluster.

Configure SLES 12/15 for SAP

> **Important**
>
> In the following steps, you need to update several configuration files. We recommend taking a backup of the files before you modify them. This will help you to revert to the previous configuration if needed.

To configure SLES 12/15 for SAP

1. After your instance is up and running, connect to the instance by using Secure Shell (SSH) and the key pair that you used to launch the instance.

> **Note**
>
> Depending on your network and security settings, you might have to first connect by using SSH to a bastion host before accessing your SAP HANA instance, or you might have to add IP addresses or ports to the security group to allow SSH access.

2. Switch to root user.

   Alternatively, you can use `sudo` to execute the following commands as ec2-user.

3. Set a hostname and fully qualified domain name (FQDN) for your instance by executing the `hostnamectl` command and updating the `/etc/hostname` file.

```
# hostnamectl set-hostname --static your_hostname
# echo your_hostname.example.com > /etc/hostname
```

   Open a new session to verify the hostname change.

4. Ensure that the `DHCLIENT_SET_HOSTNAME` parameter is set to **no** to prevent DHCP from changing the hostname during restart.

```
# grep DHCLIENT_SET_HOSTNAME /etc/sysconfig/network/dhcp
```

5. Set the `preserve_hostname` parameter to true to ensure your hostname is preserved during restart.

```
# sed -i '/preserve_hostname/ c\preserve_hostname: true' /etc/cloud/cloud.cfg
```

6. Add an entry to the `/etc/hosts` file with the new hostname and IP address.

```
ip_address hostname.example.com hostname
```

7. If you are using a BYOS SLES for SAP image, register your instance with SUSE. Ensure that your subscription is for SLES for SAP.

```
# SUSEConnect -r Your_Registration_Code
# SUSEConnect -s
```

8. Ensure that the following packages are installed:

`systemd`, `tuned`, `saptune`, `libgcc_s1`, `libstdc++6`, `cpupower`, `autofs`, `nvme-cli`, `libssh2-1`, `libopenssl1_0_0`

You can use the `rpm` command to check whether a package is installed.

```
# rpm -qi package_name
```

You can then use the zypper install command to install the missing packages.

```
# zypper install package_name
```

> **Note**
>
> If you are importing your own SLES image, additional packages might be required to ensure that your instance is optimally setup. For the latest information, refer to the Package List section in the SLES for SAP Application Configuration Guide for SAP HANA, which is attached to SAP OSS Note 1944799

9. Ensure that your instance is running on a kernel version that is recommended in SAP OSS Note 2205917 or 2684254 depending on your version. If needed, update your system to meet the minimum kernel version. You can check the version of the kernel and other packages by using the following command:

```
# rpm -qi kernel*
```

10. Start `saptune daemon` and use the following command to set it to automatically start when the system reboots.

```
# saptune daemon start
```

11. Check whether the `force_latency` parameter is set in the `saptune` configuration file.

```
# grep force_latency /usr/lib/tuned/saptune/tuned.conf
```

If the parameter is set, skip the next step and proceed with activating the HANA profile with `saptune`.

12. Update the `saptune` HANA profile according to SAP OSS Note 2205917, and then run the following commands to create a custom profile for SAP HANA. This step is not required if the `force_latency` parameter is already set.

```
# mkdir /etc/tuned/saptune
# cp /usr/lib/tuned/saptune/tuned.conf /etc/tuned/saptune/tuned.conf
# sed -i "/\[cpu\]/ a force_latency=70" /etc/tuned/saptune/tuned.conf
# sed -i "s/script.sh/\/usr\/lib\/tuned\/saptune\/script.sh/"
```

13. Switch the `tuned` profile to HANA and verify that all settings are configured appropriately.

```
# saptune solution apply HANA
# saptune solution verify HANA
```

14. Configure and start the Network Time Protocol (NTP) service. You can adjust the NTP server pool based on your requirements; for example:

> **Note**
>
> Remove any existing invalid NTP server pools from `/etc/ntp.conf` before adding the following.

```
# echo "server 0.pool.ntp.org" >> /etc/ntp.conf
# echo "server 1.pool.ntp.org" >> /etc/ntp.conf
# echo "server 2.pool.ntp.org" >> /etc/ntp.conf
# echo "server 3.pool.ntp.org" >> /etc/ntp.conf
# systemctl enable ntpd.service
# systemctl start ntpd.service
```

> **Tip**
>
> Instead of connecting to the global NTP server pool, you can connect to your internal NTP server if needed. Or you can use Amazon Time Sync Service to keep your system time in sync.

15. Set the clocksource to `tsc` by updating the `current_clocksource` file and the GRUB2 boot loader.

```
# echo "tsc" > /sys/devices/system/clocksource/*/current_clocksource
# cp /etc/default/grub /etc/default/grub.backup
# sed -i '/GRUB_CMDLINE_LINUX/ s|"| clocksource=tsc"|2' /etc/default/grub
# grub2-mkconfig -o /boot/grub2/grub.cfg
```

16. Reboot your system for the changes to take effect.
17. Continue with storage configuration for SAP HANA.

Configure RHEL 7/8/9 for SAP

Important

In the following steps, you need to update several configuration files. We recommend taking a backup of the files before you modify them. This will help you to revert to the previous configuration if needed.

To configure RHEL 7/8/9 for SAP

1. After your instance is up and running, connect to the instance by using Secure Shell (SSH) and the key pair that you used to launch the instance.

   Note

   Depending on your network and security settings, you might have to first connect by using SSH to a bastion host before accessing your SAP HANA instance, or you might have to add IP addresses or ports to the security group to allow SSH access.

2. Switch to root user.

   Alternatively, you can use sudo to execute the following commands as ec2-user.

3. Set a hostname for your instance by executing the `hostnamectl` command and update the `/etc/cloud/cloud.cfg` file to ensure that your hostname is preserved during system reboots.

```
# hostnamectl set-hostname --static your_hostname
# echo "preserve_hostname: true" >> /etc/cloud/cloud.cfg
```

Open a new session to verify the hostname change.

4. Add an entry to the `/etc/hosts` file with the new hostname and IP address.

```
ip address hostname.example.com hostname
```

Ensure that the packages listed in the following SAP Notes (SAP portal access required) are installed:

- SAP Note 2002167 - Red Hat Enterprise Linux 7.x: Installation and Upgrade
- SAP Note 2772999 - Red Hat Enterprise Linux 8.x: Installation and Configuration
- SAP Note 3108316 - Red Hat Enterprise Linux 9.x: Installation and Configuration

Note that your instance should have access to the SAP HANA channel to install libraries requires for SAP HANA installations.

You can use the `rpm` command to check whether a package is installed:

```
# rpm -qi package_name
```

You can then install any missing packages by using the `yum -y install` command.

```
# yum -y install package name
```

> **Note**
>
> Depending on your base RHEL image, additional packages might be required to ensure that your instance is optimally setup. (You can skip this step if you are using the RHEL for SAP with HA & US image.) For the latest information, refer to the RHEL configuration guide that is attached to SAP OSS Note 2009879. Review the packages in the Install Additional Required Packages section and the Appendix–Required Packages for SAP HANA on RHEL 7 section.

5. Ensure that your instance is running on a kernel version that is recommended in SAP OSS Note 2292690, 2777782, and 3108302. If needed, update your system to meet the minimum kernel version. You can check the version of the kernel and other packages using the following command.

```
  # rpm -qi kernel*
```

6. Start `tuned` daemon and use the following commands to set it to automatically start when the system reboots.

```
  # systemctl start tuned

  # systemctl enable tuned
```

7. Configure the `tuned` HANA profile to optimize your instance for SAP HANA workloads.

   Check whether the `force_latency` parameter is already set in the `/usr/lib/tuned/sap-hana/tuned.conf` file. If the parameter is set, execute the following commands to apply and activate the `sap-hana` profile.

```
  # tuned-adm profile sap-hana
  # tuned-adm active
```

   If the `force_latency` parameter is not set, execute the following steps to modify and activate the `sap-hana` profile.

```
  # mkdir /etc/tuned/sap-hana
  # cp /usr/lib/tuned/sap-hana/tuned.conf /etc/tuned/sap-hana/tuned.conf
  # sed -i '/force_latency/ c\force_latency=70' /etc/tuned/sap-hana/tuned.conf
  # tuned-adm profile sap-hana
  # tuned-adm active
```

8. Disable Security-Enhanced Linux (SELinux) by running the following command. (Skip this step if you are using the RHEL for SAP with HA & US image.)

```
  # sed -i 's/\(SELINUX=enforcing\|SELINUX=permissive\)/SELINUX=disabled/g' \/etc/
selinux/config
```

9. Disable Transparent Hugepages (THP) at boot time by adding the following to the line that starts with GRUB_CMDLINE_LINUX in the `/etc/default/grub` file. Execute the following commands to add the required parameter and to re-configure grub (Skip this step if you are using the RHEL for SAP with HA & US image.)

```
  # sed -i '/GRUB_CMDLINE_LINUX/ s|"| transparent_hugepage=never"|2' /etc/default/
grub
  # cat /etc/default/grub
  # grub2-mkconfig -o /boot/grub2/grub.cfg
```

10. Add symbolic links by executing following commands. (Skip this step if you are using the RHEL for SAP with HA & US image.)

```
  # ln -s /usr/lib64/libssl.so.10 /usr/lib64/libssl.so.1.0.1
  # ln -s /usr/lib64/libcrypto.so.10 /usr/lib64/libcrypto.so.1.0.1
```

11. Configure and start the Network Time Protocol (NTP) service. You can adjust the NTP server pool based on your requirements. The following is just an example.

> **Note**
>
> Remove any existing invalid NTP server pools from `/etc/ntp.conf` before adding the following.

```
  # echo "server 0.pool.ntp.org" >> /etc/ntp.conf
  # echo "server 1.pool.ntp.org" >> /etc/ntp.conf
  # echo "server 2.pool.ntp.org" >> /etc/ntp.conf
  # echo "server 3.pool.ntp.org" >> /etc/ntp.conf
  # systemctl enable ntpd.service
  # systemctl start ntpd.service
  # systemctl restart systemd-timedated.service
```

> **Tip**
>
> Instead of connecting to the global NTP server pool, you can connect to your internal NTP server if needed. Alternatively, you can also use Amazon Time Sync Service to keep your system time in sync.

12. Set clocksource to `tsc` by the updating the `current_clocksource` file and the GRUB2 boot loader.

```
  # echo "tsc" > /sys/devices/system/clocksource/*/current_clocksource
```

```
  # cp /etc/default/grub /etc/default/grub.backup
  # sed -i '/GRUB_CMDLINE_LINUX/ s|"| clocksource=tsc"|2' /etc/default/grub
  # grub2-mkconfig -o /boot/grub2/grub.cfg
```

13. For RHEL9 only, disable the LVM device persistence using the following commands.

```
# sed -i'.bkp' -e 's/# use_devicesfile = 0/use_devicesfile = 1/g' /etc/lvm/lvm.conf
# mv /etc/lvm/devices/system.devices /etc/lvm/devices/system.devices.bkp
```

14. Reboot your system for the changes to take effect.
15. After the reboot, log in as root and execute the `tuned-adm` command to verify that all SAP recommended settings are in place.

```
  # tuned-adm verify

  “tuned-adm verify” creates a log file under /var/log/tuned/tuned.log Review this
 log file and ensure that  all checks have passed.
```

16. Continue with storage configuration.

Configure storage

This section includes instructions for configuring your storage for SAP HANA.

Topics

- Storage architecture
- Configure storage (Amazon EBS)
- Configure storage (FSx for ONTAP)
- Configure storage (Amazon EFS)

Storage architecture

This section includes architecture diagrams for scale-up and scale-out environments for SAP HANA.

Topics

- Amazon EBS

- Amazon FSx for NetApp ONTAP

Amazon EBS

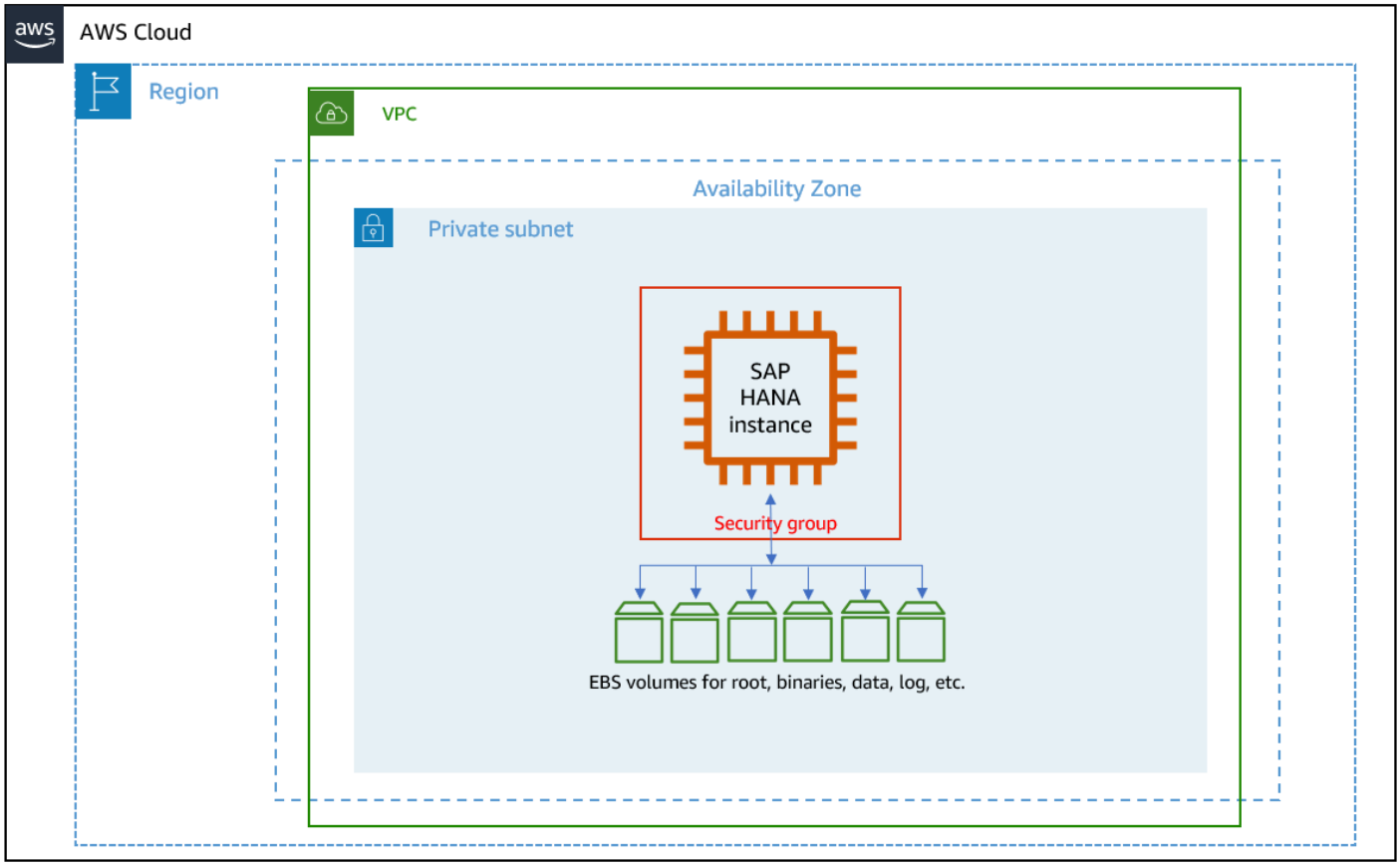
The following architecture diagrams show scale-up and scale-out environments for SAP HANA workloads using Amazon EBS volumes.

Topics

- Scale-up environment
- Scale-out environment

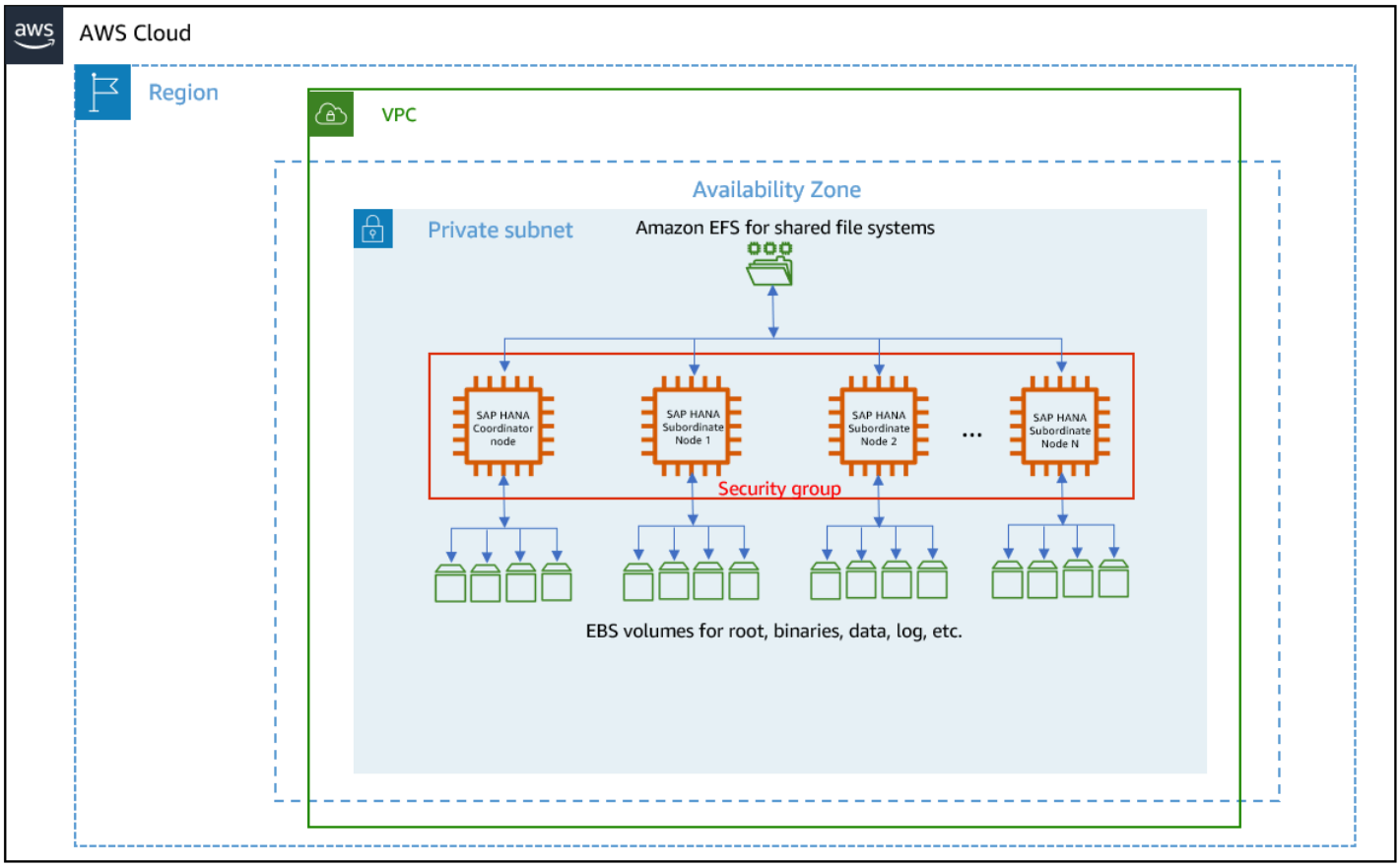
Scale-up environment

The following architecture diagram shows a scale-up environment for SAP HANA workloads using Amazon EBS volumes.



Scale-out environment

The following architecture diagram shows a scale-out environment for SAP HANA workloads using Amazon EBS volumes.



Amazon FSx for NetApp ONTAP

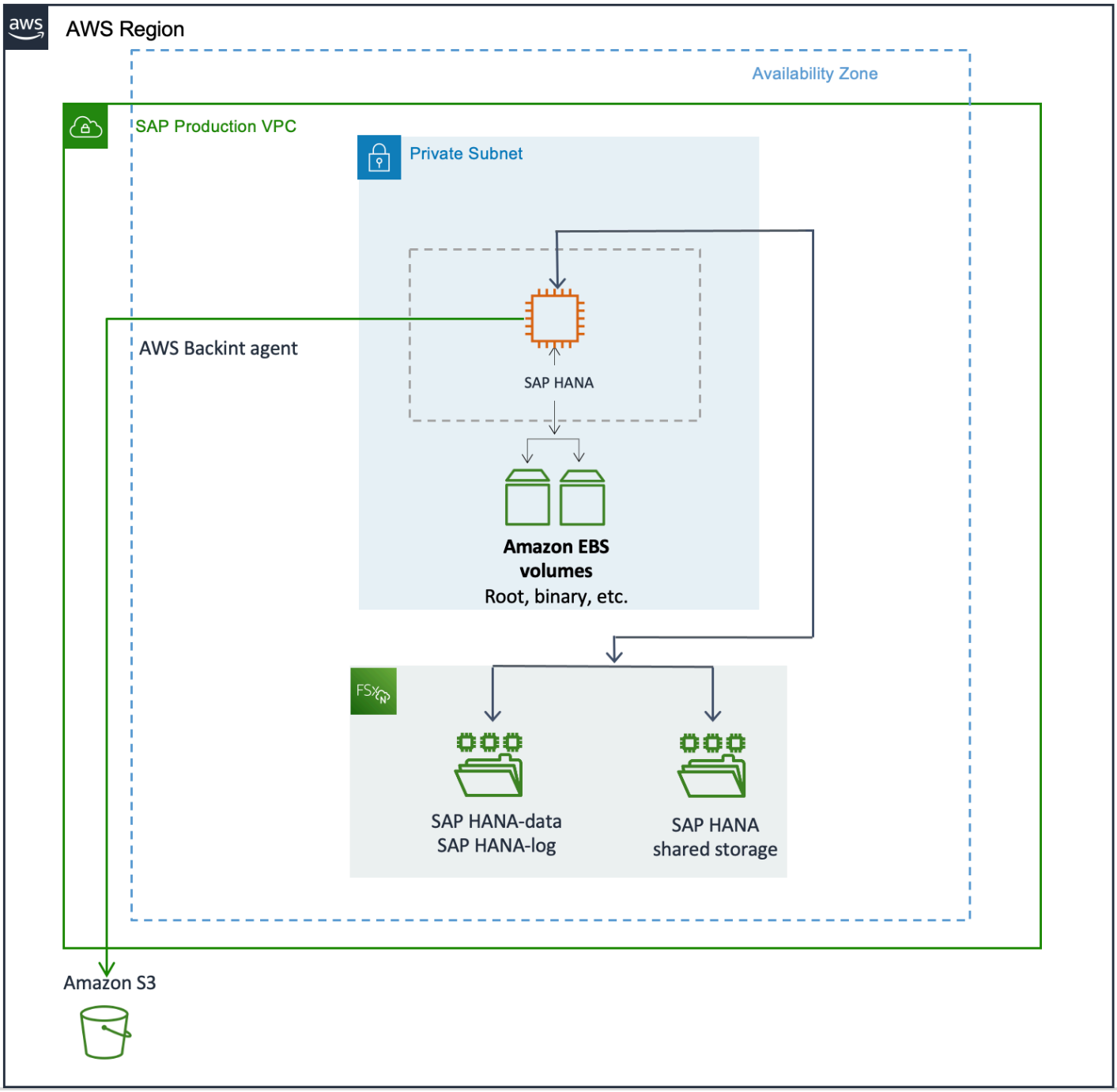
The following architecture diagrams show scale-up and scale-out environments for SAP HANA workloads using Amazon FSx for NetApp ONTAP.

Topics

- Scale-up environment
- Scale-out environment

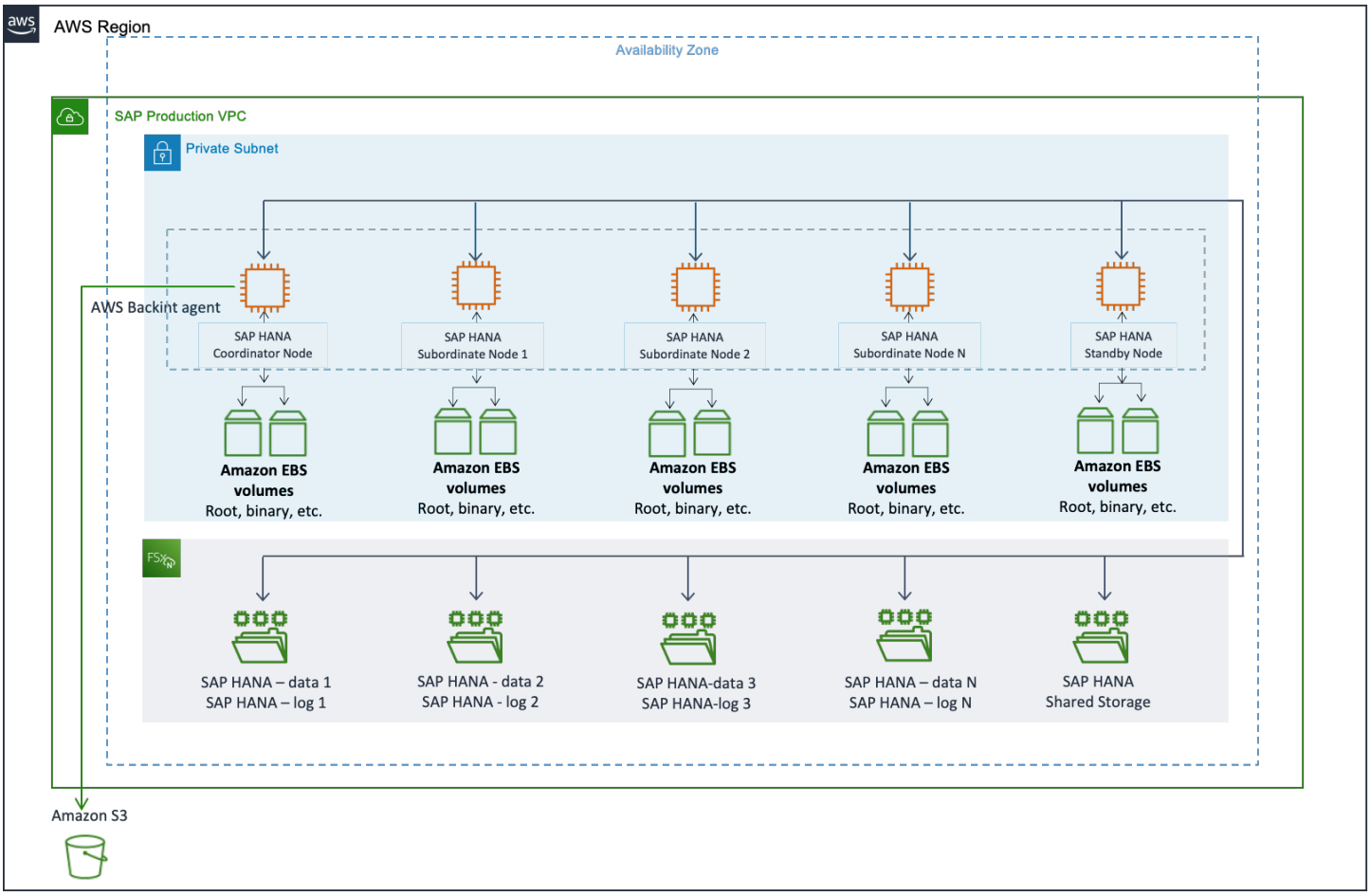
Scale-up environment

The following architecture diagram shows a scale-up environment for SAP HANA workloads using FSx for ONTAP.



Scale-out environment

The following architecture diagram shows a scale-out environment for SAP HANA workloads using FSx for ONTAP.



Configure storage (Amazon EBS)

This section explains how to deploy and configure scale-up and scale-out workloads with Amazon EBS.

Topics

- Deploy scale-up and scale-out workloads with Amazon EBS
- Configure Amazon EBS storage for SAP HANA

Deploy scale-up and scale-out workloads with Amazon EBS

This topic explains how to deploy scale-up and scale-out workloads with Amazon EBS.

Choose one of the following methods.

Console

1. Log in to the console with appropriate permissions and ensure that you have the right Region selected.
2. Choose **Services**, and then choose **EC2** (under **Compute**).
3. Choose **Launch Instance**.
4. Search for the image that you want to use:

   - Choose **AWS Marketplace** to search for RHEL for SAP and SLES for SAP images.
   - Choose **My AMIs** to search for your BYOS or custom AMI ID.

   When you find the image, choose **Select**, and then confirm to continue.
5. On the **Choose an Instance Type** page, select the instance type that you identified when planning the deployment, and choose **Configure Instance Details** to proceed with instance configuration.
6. On the **Configure Instance Details** page, do the following:

   a. Enter the number of instances (typically 1). For scale-out workloads, specify the number of nodes.

   b. Select the VPC ID and subnet for the network.

   c. Turn off the **Auto-assign Public IP** option.

   d. Select **Add instance to placement group** if needed (recommended for scale-out workloads; for details, see the AWS documentation).

   e. Select any IAM role that you want to assign to the instance to access AWS services from the instance.

   f. Select **Stop** for **Shutdown behavior**.

   g. Enable termination protection if needed (strongly recommended).

   h. Enable Amazon CloudWatch detailed monitoring (strongly recommended; for details, see the AWS documentation).

   i. Select the **Tenancy** or proceed with the default (**Shared**). For dedicated hosts, select the **Dedicated host** option.

   j. Choose **Add Storage** to proceed with storage configuration.
7. On the **Add Storage** page, choose **Add New Volume** to add volumes required for SAP HANA with the appropriate device, size, volume type, IOPS (for io1 only), and the **Delete**

on Termination flag. Ensure that you follow the storage guidance discussed earlier in this document. Add volumes for SAP HANA data, log, shared, backup, and binaries.

Figure 3 shows the storage configuration for x1.32xlarge instance type with io1 volume type for SAP HANA data and log.

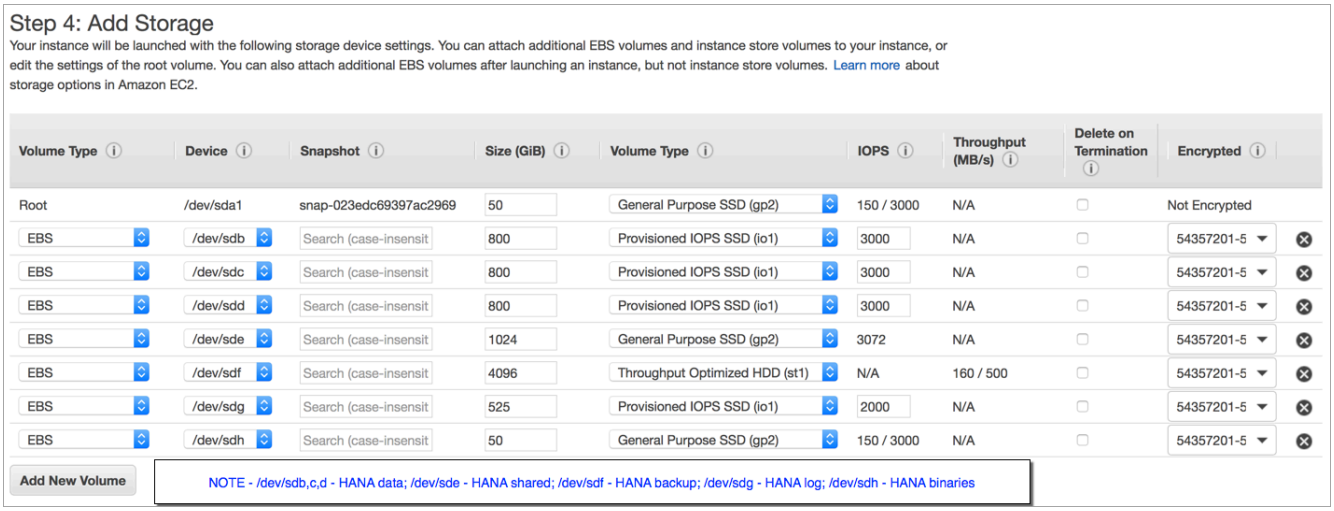


Figure 3: SAP HANA Storage Configuration with the console

> **Note**
>
> If you are planning to deploy scale-out workloads, you don't have to include Amazon EBS volumes for SAP HANA shared and backup volumes. You can use Amazon EFS with NFS to mount the HANA shared and backup volumes to your master and worker nodes.

Choose **Add Tags** to proceed with configuring tags

8. Choose **Add Tag** and add the key-value pair to track and manage your resources. We recommend adding `Name` as a minimum key to easily identify your resources.

   Next, choose **Configure Security Group**.

9. Choose **Select an existing security group** and select a security group, if you have one, to attach to your instance. Otherwise, choose **Create a new security group** and configure the **Type**, **Protocol**, **Port Range**, and the **Source IP address** from where you want to allow traffic to your SAP HANA instance. Refer to Security groups in AWS Launch Wizard for SAP

for a list of ports that we recommend. You can change the port as needed to meet your security requirements.

10. Choose **Review and Launch** to review your selections, and then choose **Launch**.
11. Select an existing key pair if you have one. Otherwise, create a new key pair, acknowledge it, and choose **Launch Instances**.
12. Your instance should be launching now with the selected configuration. After the instance is launched, you can proceed with the operating system and storage configuration steps.

> **Note**
>
> Amazon EBS volumes are presented as NVME block devices on Nitro-based instances. You need to perform additional mapping at the operating system level when you configure these volumes.

AWS CLI

Step 1. Prepare Storage Configuration for SAP HANA

Use the editor of your choice to create a .json file that contains block device mapping details similar to the following example, and save your file in a temporary directory. The example shows the block device mapping details for the x1.32xlarge instance type with io1 volumes for HANA data and log. Change the details depending on instance and storage type that you intend to use for your deployment. For more information, see SAP HANA on AWS Operations Guide .

```
[
  {"DeviceName":"/dev/sdb","Ebs":
{"VolumeSize":800,"VolumeType":"io1","Iops":3000,"Encrypted":true,"DeleteOnTermination":fals
  {"DeviceName":"/dev/sdc","Ebs":
{"VolumeSize":800,"VolumeType":"io1","Iops":3000,"Encrypted":true,"DeleteOnTermination":fals
  {"DeviceName":"/dev/sdd","Ebs":
{"VolumeSize":800,"VolumeType":"io1","Iops":3000,"Encrypted":true,"DeleteOnTermination":fals
  {"DeviceName":"/dev/sde","Ebs":
{"VolumeSize":1024,"VolumeType":"gp2","Encrypted":true,"DeleteOnTermination":false}},
  {"DeviceName":"/dev/sdf","Ebs":
{"VolumeSize":4096,"VolumeType":"st1","Encrypted":true,"DeleteOnTermination":false}},
  {"DeviceName":"/dev/sdh","Ebs":
{"VolumeSize":525,"VolumeType":"io1","Iops":2000,"Encrypted":true,"DeleteOnTermination":fals
```

```
  {"DeviceName":"/dev/sdr","Ebs":
{"VolumeSize":50,"VolumeType":"gp2","Encrypted":true,"DeleteOnTermination":false}}
]
```

Important

If the `DeleteOnTermination` flag is set to false, Amazon EBS volumes are not deleted when you terminate your Amazon EC2 instance. This helps preserve your data from accidental termination of your Amazon EC2 instance. When you terminate the instance, you need to manually delete the Amazon EBS volumes that are associated with the terminated instance to stop incurring storage cost.

Note

If you plan to deploy scale-out workloads, you don't have to include Amazon EBS volumes for SAP HANA shared and backup volumes. You can use Amazon EFS and Network File System (NFS) to mount the SAP HANA shared and backup volumes to your coordinator and subordinate nodes.

Step 2. Launch the Amazon EC2 instance

Use AWS CLI to launch the Amazon EC2 instance for SAP HANA, including Amazon EBS storage, in the VPC in your target AWS Region by using the information you gathered during the preparation steps; for example:

Important

Be sure to enter the command on a single line.

```
  $ aws ec2 run-instances
--image-id ami-xxxxxxxx
--count 1
--instance-type x1.32xlarge
--region us-west-2
--key-name=my_key
```

```
--security-group-ids sg-xxxxxxxx
--subnet-id subnet-xxxxxxxx
--placement GroupName=My-PlacementGroup,Tenancy=default,HostId=My-DedicatedHostId
--block-device-mappings file:///tmp/ebs_hana.json
--tag-specifications 'ResourceType=instance,Tags=[{Key=Name,Value=MyHANA}]'
 'ResourceType=volume,Tags=[{Key=Name,Value=MyHANAVolumes}]'
```

Notes

- The `--placement` parameter is optional and needed only when you use a dedicated host with host tenancy or you want to place all your Amazon EC2 instances in close proximity. You may also pass additional parameters like `private-ip-address`, `disable-api-termination`, etc., as needed for your environment. For additional details, see run-instances in the AWS CLI Command Reference.
- After the instance and volumes are created, you can adjust the values of Amazon EBS volume tags to be more specific for ease of management. You can also add any additional tags that you need.
- For scale-out workloads, you can use the `--count` parameter to specify the total number of required nodes.
- Amazon EC2 High Memory Metal Instances (u-*tb1.metal) can be launched only through AWS CLI or APIs. After launch, however, you can manage them by using the console, AWS CLI, or APIs. You can use the AWS Management Console, AWS CLI, or APIs to launch virtualized high memory instances (u*tb1.*xlarge).

Configure Amazon EBS storage for SAP HANA

This topic explains how to configure scale-up and scale-out workloads with Amazon EBS.

To configure workloads with Amazon EBS

1. Amazon EBS volumes should have been created and attached when you launched the Amazon EC2 instance. Confirm that all the required volumes are attached to the instance by running the `lsblk` command, which returns a list of the storage devices that are attached to the instance.

> **Note**
>
> On Nitro-based instances, Amazon EBS volumes are presented as NVME block devices. You need to perform additional mapping when configuring these volumes.

Depending on the instance and storage volume types, your block device mapping will look similar to the following examples.

Example from a non-Nitro instance

```
# lsblk
NAME     MAJ:MIN  RM  SIZE RO TYPE MOUNTPOINT
xvda     202:0     0   50G  0 disk
##xvda1 202:1     0    1M  0 part
##xvda2 202:2     0   50G  0 part /
xvdb     202:16    0  800G  0 disk
xvdc     202:32    0  800G  0 disk
xvdd     202:48    0  800G  0 disk
xvde     202:64    0    1T  0 disk
xvdf     202:80    0    4T  0 disk
xvdh     202:112   0  525G  0 disk
xvdr     202:4352  0   50G  0 disk
#
```

Example from a Nitro instance

```
## lsblk
NAME          MAJ:MIN  RM  SIZE RO TYPE MOUNTPOINT
nvme0n1       259:0     0   50G 0  disk
##nvme0n1p1  259:1     0   50G 0  part /
nvme1n1       259:2     0    4T 0  disk
nvme2n1       259:3     0  800G 0  disk
nvme3n1       259:4     0  800G 0  disk
nvme4n1       259:5     0  800G 0  disk
nvme5n1       259:6     0  525G 0  disk
nvme6n1       259:7     0    1T 0  disk
nvme7n1       259:8     0   50G 0  disk
#
```

2. Initialize the volumes of SAP HANA data, log, and backup to use with Linux Logical Volume Manager (LVM).

> **Note**
>
> Ensure you are choosing the devices that are associated with the SAP HANA data, log, and backup volumes. The device names might be different in your environment.

Example from a non-Nitro instance

```
# pvcreate /dev/xvdb /dev/xvdc /dev/xvdd /dev/xvdf /dev/xvdh
  Physical volume "/dev/xvdb" successfully created.
  Physical volume "/dev/xvdc" successfully created.
  Physical volume "/dev/xvdd" successfully created.
  Physical volume "/dev/xvdf" successfully created.
  Physical volume "/dev/xvdh" successfully created.
#
```

Example from a Nitro instance

```
# pvcreate /dev/nvme2n1 /dev/nvme3n1 /dev/nvme4n1 /dev/nvme5n1 /dev/nvme1n1
  Physical volume "/dev/nvme2n1" successfully created.
  Physical volume "/dev/nvme3n1" successfully created.
  Physical volume "/dev/nvme4n1" successfully created.
  Physical volume "/dev/nvme5n1" successfully created.
  Physical volume "/dev/nvme1n1" successfully created.
#
```

3. Create volume groups for SAP HANA data, log, and backup. Ensure that device IDs are associated correctly with the appropriate volume group.

Example from a non-Nitro instance

```
# vgcreate vghanadata /dev/xvdb /dev/xvdc /dev/xvdd
  Volume group "vghanadata" successfully created
# vgcreate vghanalog /dev/xvdh
  Volume group "vghanalog" successfully created
# vgcreate vghanaback /dev/xvdf
  Volume group "vghanaback" successfully created
```

```
#
```

Example from a Nitro instance

```
# vgcreate vghanadata /dev/nvme2n1 /dev/nvme3n1 /dev/nvme4n1
  Volume group "vghanadata" successfully created
# vgcreate vghanalog /dev/nvme5n1
  Volume group "vghanalog" successfully created
# vgcreate vghanaback /dev/nvme1n1
  Volume group "vghanaback" successfully created
#
```

4. Create a logical volume for SAP HANA data.

   In the following command, `-i 3` represents stripes based on the number of volumes that are used for a HANA data volume group. Adjust the number depending on the number of volumes that are allocated to the HANA data volume group, based on instance and storage type.

```
# lvcreate -n lvhanadata -i 3 -I 256 -L 2350G vghanadata
  Rounding size 2.29 TiB (601600 extents) up to stripe boundary size 2.29 TiB
(601602 extents).
  Logical volume "lvhanadata" created.
#
```

5. Create a logical volume for SAP HANA log.

   In the following command, `-i 1` represents stripes based on the number of volumes that are used for a HANA log volume group. Adjust the number depending on the number of volumes that are allocated to the HANA log volume group, based on instance and storage type.

```
# lvcreate -n lvhanalog -i 1 -I 256 -L 512G vghanalog
  Ignoring stripesize argument with single stripe.
  Logical volume "lvhanalog" created.
#
```

6. Create a logical volume for SAP HANA backup.

```
# lvcreate -n lvhanaback -i 1 -I 256 -L 4095G vghanaback
  Ignoring stripesize argument with single stripe.
  Logical volume "lvhanaback" created.
#
```

7. Construct XFS file systems with the newly created logical volumes for HANA data, log, and backup by using the following commands:

```
# mkfs.xfs -f /dev/mapper/vghanadata-lvhanadata
# mkfs.xfs -f /dev/mapper/vghanalog-lvhanalog
# mkfs.xfs -f /dev/mapper/vghanaback-lvhanaback
```

8. Construct XFS file systems for HANA shared and HANA binaries.

```
# mkfs.xfs -f /dev/xvde -L HANA_SHARE
# mkfs.xfs -f /dev/xvdr -L USR_SAP
```

> **Note**
>
> On Nitro-based instance types, device names can change during instance restarts. To prevent file system mount issues, it is important to create labels for devices that aren't part of logical volumes so that the devices can be mounted by using labels instead of the actual device names.

9. Create directories for HANA data, log, backup, shared, and binaries.

```
# mkdir /hana /hana/data /hana/log /hana/shared /backup /usr/sap
```

10. Use the `echo` command to add entries to the `/etc/fstab` file with the following mount options to automatically mount these file systems during restart.

```
 # echo "/dev/mapper/vghanadata-lvhanadata /hana/data xfs
noatime,nodiratime,logbsize=256k 0 0" >> /etc/fstab
 # echo "/dev/mapper/vghanalog-lvhanalog /hana/log xfs
noatime,nodiratime,logbsize=256k 0 0" >> /etc/fstab
 # echo "/dev/mapper/vghanaback-lvhanaback /backup xfs
noatime,nodiratime,logbsize=256k 0 0" >> /etc/fstab
 # echo "/dev/disk/by-label/HANA_SHARE /hana/shared xfs
noatime,nodiratime,logbsize=256k 0 0" >> /etc/fstab
 # echo "/dev/disk/by-label/USR_SAP /usr/sap xfs noatime,nodiratime,logbsize=256k
0 0" >> /etc/fstab
```

11. Mount the file systems.

```
# mount -a
```

12. Check to make sure that all file systems are mounted appropriately; for example, here is the output from an x1.32xlarge system:

```
# df -h
Filesystem                          Size  Used Avail Use% Mounted on
/dev/xvda2                           50G  1.8G   49G    4% /
devtmpfs                            961G     0 961G    0% /dev
tmpfs                               960G     0 960G    0% /dev/shm
tmpfs                               960G   17M 960G    1% /run
tmpfs                               960G     0 960G    0% /sys/fs/cgroup
tmpfs                               192G     0 192G    0% /run/user/1000
/dev/mapper/vghanadata-lvhanadata   2.3T   34M 2.3T    1% /hana/data
/dev/mapper/vghanalog-lvhanalog     512G   33M 512G    1% /hana/log
/dev/mapper/vghanaback-lvhanaback   4.0T   33M 4.0T    1% /backup
/dev/xvde                           1.0T   33M 1.0T    1% /hana/shared
/dev/xvdr                            50G   33M 50G     1% /usr/sap
#
```

13. At this time, we recommend rebooting the system and confirming that all the file systems mount automatically after the restart.
14. If you are deploying a scale-out workload, follow the steps specified in Configure NFS for scale-out workloads to set up SAP HANA shared and backup NFS file systems with Amazon EFS.

    If you are not deploying a scale-out workload, you can now proceed with your SAP HANA software installation.

Configure storage (FSx for ONTAP)

Amazon FSx for NetApp ONTAP is a fully managed service that provides highly reliable, scalable, high-performing, and feature-rich file storage built on NetApp's popular ONTAP file system. You can now deploy and operate SAP HANA on AWS with Amazon FSx for NetApp ONTAP. For more information, see Amazon FSx for NetApp ONTAP.

SAP HANA stores and processes all of its data in memory and provides protection against data loss by saving the data in persistent storage locations. To achieve optimal performance, the storage solution used for SAP HANA data and log volumes must meet SAP's storage KPI. As a fully managed service, Amazon FSx for NetApp ONTAP makes it easier to launch and scale reliable, high-performing, and secure shared file storage in the cloud.

If you are a first-time user, see How Amazon FSx for NetApp ONTAP works.

This guide covers the following topics.

- the section called "Amazon EC2 instance type considerations"
- the section called "Set up FSx for ONTAP file system"
- the section called "Set up host"

For SAP specifications, refer to SAP Note 2039883 - FAQ: SAP HANA database and data snapshots and SAP Note 3024346 - Linux Kernel Settings for NetApp NFS.

Amazon EC2 instance type considerations

The following rules and limitations are applicable for deploying SAP HANA on AWS with Amazon FSx for NetApp ONTAP.

- FSx for ONTAP file systems for SAP HANA data and log volumes are only supported for single Availability Zone deployment.
- Amazon EC2 instances where you plan to deploy your SAP HANA workload and FSx for ONTAP file systems must be in the same subnet.
- Use separate storage virtual machines (SVM) for SAP HANA data and log volumes. This ensures that your I/O traffic flows through different IP addresses and TCP sessions.
- For SAP HANA scale-out with standby node, the `basepath_shared` must be set to *Yes*. You can locate it in the *Persistence* section of the `global.ini` file.
- SAP HANA on FSx for ONTAP is only supported with the NFSv4.1 protocol. SAP HANA volumes must be created and mounted using the NFSv4.1 protocol.
- SAP HANA on FSx for ONTAP is only supported on the following operating systems:
  - Red Hat Enterprise Linux 8.4 and above
  - SUSE Linux Enterprise Server 15 SP2 and above
- `/hana/data` and `/hana/log` must have their own FSx for ONTAP volumes. `/hana/shared`, and `/usr/sap` can share a volume.

Topics

- Supported Amazon EC2 instance types
- Sizing

- SAP HANA parameters

Supported Amazon EC2 instance types

Amazon FSx for NetApp ONTAP is certified by SAP for scale-up and scale-out (OLTP/OLAP) SAP HANA workloads in a single Availability Zone setup. You can use Amazon FSx for NetApp ONTAP as the primary storage for SAP HANA data, log, binary, and shared volumes. For a complete list of supported Amazon EC2 instances for SAP HANA, see SAP HANA certified instances.

Sizing

You can configure the throughput capacity of FSx for ONTAP when you create a new file system by scaling up to 4 GB/s of read throughput and 1000 MB/s of write throughput in a single Availability Zone deployment. For more information, see Amazon FSx for NetApp ONTAP performance.

SAP KPIs

SAP requires the following KPIs for SAP HANA volumes.

	Read	Write
Data	400 MB/s	250 MB/s
Log	250 MB/s	250 MB/s
Latency for log	Less than 1 millisecond write latency with 4K and 16K block sized I/O	

Minimum requirement

You must provision FSx for ONTAP volumes with sufficient capacity and performance, based on the requirements of your SAP HANA workload. To meet the storage KPIs for SAP HANA, you need a throughput capacity of at least **1,024 MB/s**. Lower throughput may be acceptable for non-production systems.

Sharing a file system between multiple SAP HANA nodes is supported when the file system meets the requirements of all SAP HANA nodes. When sharing a file system, you can use the quality of service feature for consistent performance and reduced interference between competing workloads. For more information, see Using Quality of Service in Amazon FSx for NetApp ONTAP.

Higher throughput

If you require higher throughput, you can do one of the following:

- Create separate data and log volumes on different FSx for ONTAP file systems.
- Create additional data volume partitions across multiple FSx for ONTAP file systems.

The following table summarizes the throughput limits available with different scaling options.

		Data		Log
	read	***write***	***read***	***write***
Shared file system	1,000 read/750 write, shared by multiple SAP HANA databases			
Dedicated file system	1,000 read/750 write			
Separate data and log volumes	1,000	750	1,000	750
Second data volume partition	1,000	750	1,000	750
	1,000	750		

Note: The numbers in this table are based on file systems with 1,024 MB/s throughput capacity.

To learn more about FSx for ONTAP performance, see Performance details.

SAP HANA parameters

Set the following SAP HANA database parameters in the `global.ini` file.

```
[fileio]
max_parallel_io_requests=128
async_read_submit=on
async_write_submit_active=on
async_write_submit_blocks=all
```

Use the following SQL commands to set these parameters on SYSTEM level.

```
ALTER SYSTEM ALTER CONFIGURATION ('global.ini', 'SYSTEM') SET ('fileio',
 'max_parallel_io_requests') = '128' WITH RECONFIGURE;
ALTER SYSTEM ALTER CONFIGURATION ('global.ini', 'SYSTEM') SET ('fileio',
 'async_read_submit') = 'on' WITH RECONFIGURE;
ALTER SYSTEM ALTER CONFIGURATION ('global.ini', 'SYSTEM') SET ('fileio',
 'async_write_submit_active') = 'on' WITH RECONFIGURE;
ALTER SYSTEM ALTER CONFIGURATION ('global.ini', 'SYSTEM') SET ('fileio',
 'async_write_submit_blocks') = 'all' WITH RECONFIGURE;
```

Set up FSx for ONTAP file system, SVMs, and volumes

Before you create FSx for ONTAP file system, determine the total storage space you need for your SAP HANA workload. You can increase the storage size later. To decrease the storage size, you must create a new file system.

To create a FSx for ONTAP file system, see Step 1: Create an Amazon FSx for NetApp ONTAP file system. For more information, see Managing FSx for ONTAP file systems.

> **Note**
>
> Only single Availability Zone file systems are supported for SAP HANA workloads.

Topics

- Create storage virtual machines (SVM)
- Volume configuration
- Volume layout
- File system setup
- Disable snapshots
- Quality of Service (QoS)

Create storage virtual machines (SVM)

You get one SVM per FSx for ONTAP file system by default. You can create additional SVMs at any time. For optimal performance, mount data and log volumes using different IP addresses. You can

achieve this using separate SVMs for data and log volumes. If you plan to use NetApp SnapCenter, all SVMs used for SAP HANA must have unique names. You don't need to join your file system to Active Directory for SAP HANA. For more information, see Managing FSx for ONTAP storage virtual machines.

Volume configuration

The storage capacity of your file system should align with the needs of `/hana/shared`, `/hana/data`, and `/hana/log` volumes. You must also consider the capacity required for snapshots, if applicable.

We recommend creating separate FSx for ONTAP volumes for each of SAP HANA data, log, shared, and binary volumes. The following table lists the recommended minimum sizes per volume.

Volume	Recommended size for scale-up	Recommended size for scale-out
`/usr/sap`	50 GiB	50 GiB
`/hana/shared`	Minimum of 1 x memory of your Amazon EC2 instance or 1TB	1 x memory of your Amazon EC2 instance for every 4 subordinate nodes*
`/hana/data`	At least 1.2 x memory of your Amazon EC2 instance	At least 1.2 x memory of your Amazon EC2 instance
`/hana/log`	Minimum of 0.5 x memory of your Amazon EC2 instance or 600 GiB	Minimum of 0.5 x memory of your Amazon EC2 instance or 600 GiB

*For example, if you have 2-4 scale-out nodes, you need 1 x memory of your single Amazon EC2 instance. If you have 5-8 scale-out nodes, you need 2 x memory of your single Amazon EC2 instance.

The following limitations apply when you create a FSx for ONTAP file system for SAP HANA.

- *Storage Efficiency* is not supported for SAP HANA and must be **disabled**.
- *Capacity Pool Tiering* is not supported for SAP HANA and must be set to **None**.

- *Daily automatic backups* must be **disabled** for SAP HANA. Default FSx for ONTAP backups are not application-aware and cannot be used to restore SAP HANA to a consistent state.

Volume layout

Topics

- SAP HANA scale-up
- SAP HANA scale-out

SAP HANA scale-up

The following table presents an example of volume and mount point configuration for scale-up setup. It includes a single host. `HDB` is the SAP HANA system ID. To place the home directory of the `hdbadm` user on the central storage, the `/usr/sap/HDB` file system must be mounted from the `HDB_shared` volume.

Volume name	Junction path	Directory	Mount point
HDB_data_mnt00001	HDB_data_mnt00001	-	/hana/data/HDB/mnt00001
HDB_log_mnt00001	HDB_log_mnt00001	-	/hana/log/HDB/mnt00001
HDB_shared	HDB_shared	usr-sap	/usr/sap/HDB
		shared	/hana/shared

SAP HANA scale-out

You must mount all the data, log, and shared volumes in every node, including the standby node.

The following table presents an example of volume and mount point configuration for a scale-out setup. It includes four active and one standby host. `HDB` is the SAP HANA system ID. The home (`/usr/sap/HDB`) and shared ((`/hana/shared`) directories of every host are stored in the `HDB_shared` volume. To place the home directory of the `hdbadm` user on the central storage, the `/usr/sap/HDB` file system must be mounted from the `HDB_shared` volume.

Volume name	Junction path	Directory	Mount point	Note
HDB_data_mnt00001	HDB_data_mnt00001	N/A	/hana/data/HDB/mnt00001	Mounted on all hosts
HDB_log_mnt00001	HDB_log_mnt00001	N/A	/hana/log/HDB/mnt00001	Mounted on all hosts
HDB_data_mnt00002	HDB_data_mnt00002	N/A	/hana/data/HDB/mnt00002	Mounted on all hosts
HDB_log_mnt00002	HDB_log_mnt00002	N/A	/hana/log/HDB/mnt00002	Mounted on all hosts
HDB_data_mnt00003	HDB_data_mnt00003	N/A	/hana/data/HDB/mnt00003	Mounted on all hosts
HDB_log_mnt00003	HDB_log_mnt00003	N/A	/hana/log/HDB/mnt00003	Mounted on all hosts
HDB_data_mnt00004	HDB_data_mnt00004	N/A	/hana/data/HDB/mnt00004	Mounted on all hosts
HDB_log_mnt00004	HDB_log_mnt00004	N/A	/hana/log/HDB/mnt00004	Mounted on all hosts
HDB_shared	HDB_shared	HDB_shared	/hana/shared/HDB	Mounted on all hosts
HDB_shared	HDB_shared	usr-sap-host1	/usr/sap/HDB	Mounted on host 1
HDB_shared	HDB_shared	usr-sap-host2	/usr/sap/HDB	Mounted on host 2
HDB_shared	HDB_shared	usr-sap-host3	/usr/sap/HDB	Mounted on host 3

Volume name	Junction path	Directory	Mount point	Note
HDB_shared	HDB_shared	usr-sap-host4	/usr/sap/HDB	Mounted on host 4
HDB_shared	HDB_shared	usr-sap-host5	/usr/sap/HDB	Mounted on host 5

File system setup

After creating a FSx for ONTAP file system, you must complete additional file system setup.

Set administrative password

If you did not create an administrative password during FSx for ONTAP file system creation, you must set an ONTAP administrative password for `fsxadmin` user.

The administrative password enables you to access the file system via SSH, the ONTAP CLI, and REST API. To use tools like NetApp SnapCenter, you must have an administrative password.

Sign in to the management endpoint via SSH

Get the DNS name of the management endpoint from AWS console. Sign in to the management endpoint via SSH, using the `fsxadmin` user and administrative password.

```
ssh fsxadmin@management.<file-system-id>.fsx.<aws-region>.amazonaws.com Password:
```

Set TCP max transfer size

We recommend a TCP max transfer size of 262,144 for your SAP HANA workloads. Elevate the privilege level to *advanced* and use the following command on each SVM.

```
set advanced
nfs modify -vserver <svm> -tcp-max-xfer-size 262144
set admin
```

Set the lease time on NFSv4 protocol

This task applies to SAP HANA scale-out with standby node setup.

Lease period refers to the time in which ONTAP irrevocably grants a lock to a client. It is set to 30 seconds by default. You can have faster server recovery by setting shorter lease time.

You can change the lease time with the following command.

```
set advanced

nfs modify -vserver <svm> -v4-lease-seconds 10
set admin
```

> **Note**
>
> Starting with SAP HANA 2.0 SPS4, SAP provides parameters to control failover behavior. NetApp recommends using these parameters instead of setting the lease time at the SVM level. For more details, see.

Disable snapshots

FSx for ONTAP automatically enables a snapshot policy for volumes that take hourly snapshots. The default policy offers limited value to SAP HANA due to missing application awareness. We recommend disabling the automatic snapshots by setting the policy to none. You can disable snapshots during volume creation or by using the following command.

```
volume modify -vserver <vserver-name> -volume <volume-name> -snapshot-policy none
```

Data volume

The automatic FSx for ONTAP snapshots do not have application awareness. A database-consistent snapshot of the SAP HANA data volume must be prepared by creating a data snapshot. For more information, see Create a Data Snapshot.

Log volume

The log volume is automatically backed up every 15 minutes by SAP HANA. An hourly volume snapshot does not offer any additional value in terms of RPO reduction.

The high frequency of changes on the log volume can rapidly increase the total capacity used for snapshots. This can cause the log volume to run out of capacity, making the SAP HANA workload unresponsive.

Quality of Service (QoS)

Quality of Service (QoS) enables FSx for ONTAP to consistently deliver predictable performance to multiple applications, and eliminate noisy neighbor applications. When sharing a file system, you can use the quality of service feature for consistent performance and reduced interference between competing workloads. For more information, see Using Quality of Service in Amazon FSx for NetApp ONTAP.

QoS is configured by creating a QoS policy group, setting ceiling or floor performance levels (minimum or maximum performance), and assigning the policy to an SVM or volume. Performance can be specified in either IOPS or throughput.

Example

You are creating a test system, based on a snapshot from production, on the same file system as your production SAP HANA database. You want to ensure that the test system does not impact the performance of the production system. You create a QoS policy group (`qos-test`) and define an upper limit of 200 MB/s for data and log volumes (`vol-data` and `vol-log`), which share the same SVM (`svm-test`).

```
# Create QoS policy group
qos policy-group create -policy-group qos-test -vserver svm-test -is-shared false -max-
throughput 200MBs

# Assign QoS policy group to data on log volumes
volume modify -vserver svm-test -volume vol-data -qos-policy-group qos-test
volume modify -vserver svm-test -volume vol-log -qos-policy-group qos-test
```

Set up host

This section walks you through an example host setup for deploying SAP HANA scale-up and scale-out systems on AWS using Amazon FSx for NetApp ONTAP as the primary storage solution.

You must configure your Amazon EC2 instance on an operating system level to use FSx for ONTAP with SAP HANA on AWS.

> **Note**
>
> The following examples apply to an SAP HANA workload with SAP System ID `HDB`. The operating system user is `hdbadm`.

Topics

- SAP HANA scale-up
- SAP HANA scale-out

SAP HANA scale-up

The following section is an example host setup for SAP HANA scale-up deployment with FSx for ONTAP.

Topics

- Linux kernel parameters
- Network File System (NFS)
- Create subdirectories
- Create mount points
- Mount file systems
- Data volume partitions

Linux kernel parameters

Create a file named `91-NetApp-HANA.conf` with the following configurations in the `/etc/sysctl.d` directory.

```
net.core.rmem_max = 16777216
net.core.wmem_max = 16777216
net.ipv4.tcp_rmem = 4096 131072 16777216
net.ipv4.tcp_wmem = 4096 16384  16777216
net.core.netdev_max_backlog = 300000
net.ipv4.tcp_slow_start_after_idle=0
net.ipv4.tcp_no_metrics_save = 1
net.ipv4.tcp_moderate_rcvbuf = 1
net.ipv4.tcp_window_scaling = 1
net.ipv4.tcp_timestamps = 1
net.ipv4.tcp_sack = 1
```

Increase the max sessions slots for NFSv4 to 180.

```
echo options nfs max_session_slots=180 > /etc/modprobe.d/nfsclient.conf
```

You must reboot your instance for the kernel parameters and NFS settings to take effect.

Network File System (NFS)

Network File System (NFS) version 4 and higher requires user authentication. You can authenticate with Lightweight Directory Access Protocol (LDAP) server or local user accounts.

If you are using local user accounts, the NFSv4 domain must be set to the same value on all Linux servers and SVMs. You can set the domain parameter (`Domain = <domain name>`) in the `/etc/idmapd.conf` file on the Linux hosts.

To identify the domain setting of the SVM, use the following command:

```
nfs show -vserver hana-data -fields v4-id-domain
```

The following is example output:

```
vserver   v4-id-domain
--------- ------------
hana-data ec2.internal
```

Create subdirectories

Mount the `/hana/shared` volume, create `shared` and `usr-sap` subdirectories, and unmount.

```
mkdir /mnt/tmp
mount -t nfs -o sec=sys,vers=4.1 <svm-shared>:/HDB-shared /mnt/tmp
cd /mnt/tmp
mkdir shared
mkdir usr-sap
cd
umount /mnt/tmp
```

Create mount points

On single-host systems, create the following mount points on your Amazon EC2 instance.

```
mkdir -p /hana/data/HDB/mnt00001
mkdir -p /hana/log/HDB/mnt00001
mkdir -p /hana/shared
mkdir -p /usr/sap/HDB
```

Mount file systems

The created file systems must be mounted as NFS file systems on Amazon EC2. The following table is an example recommendation of NFS options for different SAP HANA file systems.

File systems	NFS mount options			
	Common	**NFS version**	**NFS transfer size**	**nconnect**
SAP HANA data	rw,bg,hard,timeo=600,noatime,	vers=4,minorversion=1,lock,	rsize=262144,wsize=262144,	nconnect=4
SAP HANA log	rw,bg,hard,timeo=600,noatime,	vers=4,minorversion=1,lock,	rsize=262144,wsize=262144,	nconnect=2
SAP HANA shared	rw,bg,hard,timeo=600,noatime,	vers=4,minorversion=1,lock,	rsize=262144,wsize=262144,	nconnect=2
SAP HANA binary	rw,bg,hard,timeo=600,noatime,	vers=4,minorversion=1,lock,	rsize=262144,wsize=262144,	nconnect=2

- Changes to the `nconnect` parameter take effect only if the NFS file system is unmounted and mounted again.
- Client systems must have unique host names when accessing FSx for ONTAP. If there are systems with the same name, the second system may not be able to access FSx for ONTAP.

Example

Add the following lines to `/etc/fstab` to preserve mounted file systems during an instance reboot. You can then run `mount -a` to mount the NFS file systems.

```
<svm-data>:/HDB_data_mnt00001 /hana/data/HDB/mnt00001 nfs
 rw,bg,hard,timeo=600,noatime,vers=4,minorversion=1,lock,rsize=262144,wsize=262144,nconnect=4
```

```
<svm-log>:/HDB_log_mnt00001 /hana/log/HDB/mnt00001 nfs
 rw,bg,hard,timeo=600,noatime,vers=4,minorversion=1,lock,rsize=262144,wsize=262144,nconnect=2
<svm-shared>:/HDB_shared/usr-sap /usr/sap/HDB nfs
 rw,bg,hard,timeo=600,noatime,vers=4,minorversion=1,lock,rsize=262144,wsize=262144,nconnect=2
<svm-shared>:/HDB_shared/shared /hana/shared nfs
 rw,bg,hard,timeo=600,noatime,vers=4,minorversion=1,lock,rsize=262144,wsize=262144,nconnect=2
```

Data volume partitions

With SAP HANA 2.0 SPS4, additional data volume partitions allow configuring two or more file system volumes for the DATA volume of an SAP HANA tenant database in a single-host or multi-host system. Data volume partitions enable SAP HANA to scale beyond the size and performance limits of a single volume. You can add additional data volume partitions at any time. For more information, see Host configuration.

Topics

- Host preparation
- Enabling data volume partitioning
- Adding additional data volume partition

Host preparation

Additional mount points and `/etc/fstab` entries must be created and the new volumes must be mounted.

- Create additional mount points and assign the required permissions, group, and ownership.

```
mkdir -p /hana/data2/HDB/mnt00001
chmod -R 777 /hana/data2/HDB/mnt00001
```

- Add additional file systems to `/etc/fstab`.

```
<data2>:/data2 /hana/data/HDB/mnt00001 nfs mount options
```

- Set the permissions to 777. This is required to enable SAP HANA to add a new data volume in the subsequent step. SAP HANA sets more restrictive permissions automatically during data volume creation.

Enabling data volume partitioning

To enable data volume partitions, add the following entry in the `global.ini` file in the SYSTEMDB configuration.

```
[customizable_functionalities]
persistence_datavolume_partition_multipath = true
```

```
ALTER SYSTEM ALTER CONFIGURATION ('global.ini', 'SYSTEM')
SET ('customizable_functionalities', 'PERSISTENCE_DATAVOLUME_PARTITION_MULTIPATH') =
 'true'
WITH RECONFIGURE;
```

> **Note**
>
> You must restart your database after updating the `global.ini` file.

Adding additional data volume partition

Run the following SQL statement against the tenant database to add an additional data volume partition to your tenant database.

```
ALTER SYSTEM ALTER DATAVOLUME ADD PARTITION PATH '/hana/data/HDB/mnt00002/';
```

Adding a data volume partition is quick. The new data volume partitions are empty after creation. Data is distributed equally across data volumes over time.

After you configure and mount FSx for ONTAP file systems, you can install and setup your SAP HANA workload on AWS. For more information, see SAP HANA Environment Setup on AWS.

SAP HANA scale-out

The following section is an example host setup for SAP HANA scale-out with standby node on AWS using FSx for ONTAP as the primary storage solution. You can use SAP HANA host auto failover, an automated solution provided by SAP, for recovering from a failure on your SAP HANA host. For more information, see SAP HANA - Host Auto-Failover.

Topics

- Linux kernel parameters

- Network File System (NFS)
- Create subdirectories
- Create mount points
- Mount file systems
- Set ownership for directories
- SAP HANA parameters
- Data volume partitions
- Testing host auto failover

Linux kernel parameters

Create a file named `91-NetApp-HANA.conf` with the following configurations in the `/etc/sysctl.d` directory on all the nodes.

```
net.core.rmem_max = 16777216
net.core.wmem_max = 16777216
net.ipv4.tcp_rmem = 4096 131072 16777216
net.ipv4.tcp_wmem = 4096 16384  16777216
net.core.netdev_max_backlog = 300000
net.ipv4.tcp_slow_start_after_idle=0
net.ipv4.tcp_no_metrics_save = 1
net.ipv4.tcp_moderate_rcvbuf = 1
net.ipv4.tcp_window_scaling = 1
net.ipv4.tcp_timestamps = 1
net.ipv4.tcp_sack = 1
```

Increase the max sessions slots for NFSv4 to 180.

```
echo options nfs max_session_slots=180 > /etc/modprobe.d/nfsclient.conf
```

You must reboot your instance for the kernel parameters and NFS settings to take effect.

Network File System (NFS)

> **Important**
>
> For SAP HANA scale-out systems, FSx for ONTAP only supports NFS version 4.1.

Network File System (NFS) version 4 and higher requires user authentication. You can authenticate with Lightweight Directory Access Protocol (LDAP) server or local user accounts.

If you are using local user accounts, the NFSv4 domain must be set to the same value on all Linux servers and SVMs. You can set the domain parameter (`Domain = <domain name>`) in the `/etc/idmapd.conf` file on the Linux hosts.

To identify the domain setting of the SVM, use the following command:

```
nfs show -vserver hana-data -fields v4-id-domain
```

The following is example output:

```
vserver   v4-id-domain
--------- ------------
hana-data ec2.internal
```

Create subdirectories

Mount the `/hana/shared` volume and create `shared` and `usr-sap` subdirectories for each host. The following example command applies to 4+1 SAP HANA scale-out systems.

```
mkdir /mnt/tmp
mount -t nfs -o sec=sys,vers=4.1 <svm-shared>:/HDB-shared /mnt/tmp
cd /mnt/tmp
mkdir shared
mkdir usr-sap-host1
mkdir usr-sap-host2
mkdir usr-sap-host3
mkdir usr-sap-host4
mkdir usr-sap-host5
cd
umount /mnt/tmp
```

Create mount points

On scale-out systems, create the following mount points on all the subordinate and standby nodes. The following example command applies to 4+1 SAP HANA scale-out systems.

```
mkdir -p /hana/data/HDB/mnt00001
mkdir -p /hana/log/HDB/mnt00001
```

```
mkdir -p /hana/data/HDB/mnt00002
mkdir -p /hana/log/HDB/mnt00002
mkdir -p /hana/data/HDB/mnt00003
mkdir -p /hana/log/HDB/mnt00003
mkdir -p /hana/data/HDB/mnt00004
mkdir -p /hana/log/HDB/mnt00004
mkdir -p /hana/shared
mkdir -p /usr/sap/HDB
```

Mount file systems

The created file systems must be mounted as NFS file systems on Amazon EC2. The following table is an example recommendation of NFS options for different SAP HANA file systems.

File systems	NFS mount options			
	Common	**NFS version**	**NFS transfer size**	**nconnect**
SAP HANA data	rw,bg,hard,timeo=600,noatime,	vers=4,minorversion=1,lock,	rsize=262144,wsize=262144,	nconnect=4
SAP HANA log	rw,bg,hard,timeo=600,noatime,	vers=4,minorversion=1,lock,	rsize=262144,wsize=262144,	nconnect=2
SAP HANA shared	rw,bg,hard,timeo=600,noatime,	vers=4,minorversion=1,lock,	rsize=262144,wsize=262144,	nconnect=2
SAP HANA binary	rw,bg,hard,timeo=600,noatime,	vers=4,minorversion=1,lock,	rsize=262144,wsize=262144,	nconnect=2

- Changes to the `nconnect` parameter take effect only if the NFS file system is unmounted and mounted again.
- Client systems must have unique host names when accessing FSx for ONTAP. If there are systems with the same name, the second system may not be able to access FSx for ONTAP.

Example - mount shared volumes

Add the following lines to `/etc/fstab` on **all** the hosts to preserve mounted file systems during an instance reboot. You can then run `mount -a` to mount the NFS file systems.

```
<svm-data_1>:/HDB_data_mnt00001 /hana/data/HDB/mnt00001 nfs
 rw,bg,hard,timeo=600,noatime,vers=4,minorversion=1,lock,rsize=262144,wsize=262144,nconnect=4
<svm-log_1>:/HDB_log_mnt00001 /hana/log/HDB/mnt00001 nfs
 rw,bg,hard,timeo=600,noatime,vers=4,minorversion=1,lock,rsize=262144,wsize=262144,nconnect=2
<svm-data_2>:/HDB_data_mnt00002 /hana/data/HDB/mnt00002 nfs
 rw,bg,hard,timeo=600,noatime,vers=4,minorversion=1,lock,rsize=262144,wsize=262144,nconnect=4
<svm-log_2>:/HDB_log_mnt00002 /hana/log/HDB/mnt00002 nfs
 rw,bg,hard,timeo=600,noatime,vers=4,minorversion=1,lock,rsize=262144,wsize=262144,nconnect=2
<svm-data_3>:/HDB_data_mnt00003 /hana/data/HDB/mnt00003 nfs
 rw,bg,hard,timeo=600,noatime,vers=4,minorversion=1,lock,rsize=262144,wsize=262144,nconnect=4
<svm-log_3>:/HDB_log_mnt00003 /hana/log/HDB/mnt00003 nfs
 rw,bg,hard,timeo=600,noatime,vers=4,minorversion=1,lock,rsize=262144,wsize=262144,nconnect=2
<svm-data_4>:/HDB_data_mnt00004 /hana/data/HDB/mnt00004 nfs
 rw,bg,hard,timeo=600,noatime,vers=4,minorversion=1,lock,rsize=262144,wsize=262144,nconnect=4
<svm-log_4>:/HDB_log_mnt00004 /hana/log/HDB/mnt00004 nfs
 rw,bg,hard,timeo=600,noatime,vers=4,minorversion=1,lock,rsize=262144,wsize=262144,nconnect=2
<svm-shared>:/HDB_shared/shared /hana/shared nfs
 rw,bg,hard,timeo=600,noatime,vers=4,minorversion=1,lock,rsize=262144,wsize=262144,nconnect=2
```

Example - mount host-specific volumes

Add the host-specific line to `/etc/fstab` of **each** host to preserve mounted file systems during an instance reboot. You can then run `mount -a` to mount the NFS file systems.

Host	Line
Host 1	`<svm-shared>:/HDB_shared/usr-sap-host1 /usr/sap/HDB nfs rw,bg,hard,timeo=600,noatime,vers=4,minorversion=1,lock,rsize=262144,wsize=262144,nconnect=2`
Host 2	`<svm-shared>:/HDB_shared/usr-sap-host2 /usr/sap/HDB nfs rw,bg,hard,timeo=600,noatime,vers=4,minorversion=1,lock,rsize=262144,wsize=262144,nconnect=2`

Host	Line
Host 3	`<svm-shared>:/HDB_shared/usr-sap-host3 /usr/sap/HDB nfs rw,bg,hard,timeo=600,noatime,vers=4,minorversion=1,lock,rsize=262144,wsize=262144,nconnect=2`
Host 4	`<svm-shared>:/HDB_shared/usr-sap-host4 /usr/sap/HDB nfs rw,bg,hard,timeo=600,noatime,vers=4,minorversion=1,lock,rsize=262144,wsize=262144,nconnect=2`
Host 5 (standby host)	`<svm-shared>:/HDB_shared/usr-sap-host5 /usr/sap/HDB nfs rw,bg,hard,timeo=600,noatime,vers=4,minorversion=1,lock,rsize=262144,wsize=262144,nconnect=2`

Set ownership for directories

Use the following command to set the hdbadm ownership on SAP HANA data and log directories.

```
sudo chown hdbadm:sapsys /hana/data/HDB
sudo chown hdbadm:sapsys /hana/log/HDB
```

SAP HANA parameters

Install your SAP HANA system with the required configuration, and then set the following parameters. For more information on SAP HANA installation, see SAP HANA Server Installation and Update Guide.

Topics

- Optimal performance
- NFS lock lease

Optimal performance

For optimal performance, set the following parameters in the `global.ini` file.

```
[fileio]
max_parallel_io_requests=128
async_read_submit=on
async_write_submit_active=on
async_write_submit_blocks=all
```

The following SQL commands can be used to set these parameters on SYSTEM level.

```
ALTER SYSTEM ALTER CONFIGURATION ('global.ini', 'SYSTEM') SET ('fileio',
 'max_parallel_io_requests') = '128' WITH RECONFIGURE;
ALTER SYSTEM ALTER CONFIGURATION ('global.ini', 'SYSTEM') SET ('fileio',
 'async_read_submit') = 'on' WITH RECONFIGURE;
ALTER SYSTEM ALTER CONFIGURATION ('global.ini', 'SYSTEM') SET ('fileio',
 'async_write_submit_active') = 'on' WITH RECONFIGURE;
ALTER SYSTEM ALTER CONFIGURATION ('global.ini', 'SYSTEM') SET ('fileio',
 'async_write_submit_blocks') = 'all' WITH RECONFIGURE;
```

NFS lock lease

Starting with SAP HANA 2.0 SPS4, SAP HANA provides parameters to control the failover behavior. It is recommended to use these parameters instead of setting the lease time at the SVM level. The following parameters are configured in the `namerserver.ini` file.

Section	Parameter	Value
failover	normal_retries	9
distributed_watchdog	deactivation_retries	11
distributed_watchdog	takeover_retries	9

The following SQL commands can be used to set these parameters on SYSTEM level.

```
ALTER SYSTEM ALTER CONFIGURATION ('nameserver.ini', 'SYSTEM') SET ('failover',
 'normal_retries') = '9' WITH RECONFIGURE;
ALTER SYSTEM ALTER CONFIGURATION ('nameserver.ini', 'SYSTEM') SET
 ('distributed_watchdog', 'deactivation_retries') = '11' WITH RECONFIGURE;
ALTER SYSTEM ALTER CONFIGURATION ('nameserver.ini', 'SYSTEM') SET
 ('distributed_watchdog', 'takeover_retries') = '9' WITH RECONFIGURE;
```

Data volume partitions

With SAP HANA 2.0 SPS4, additional data volume partitions allow configuring two or more file system volumes for the DATA volume of an SAP HANA tenant database in a single-host or multi-host system. Data volume partitions enable SAP HANA to scale beyond the size and performance limits of a single volume. You can add additional data volume partitions at any time. For more information, see Host configuration.

Topics

- Host preparation
- Enabling data volume partitioning
- Adding additional data volume partition

Host preparation

Additional mount points and `/etc/fstab` entries must be created and the new volumes must be mounted.

- Create additional mount points and assign the required permissions, group, and ownership.

```
mkdir -p /hana/data2/HDB/mnt00001
chmod -R 777 /hana/data2/HDB/mnt00001
```

- Add additional file systems to `/etc/fstab`.

```
<data2>:/data2 /hana/data2/HDB/mnt00001 nfs mount options
```

- Set the permissions to 777. This is required to enable SAP HANA to add a new data volume in the subsequent step. SAP HANA sets more restrictive permissions automatically during data volume creation.

Enabling data volume partitioning

To enable data volume partitions, add the following entry in the `global.ini` file in the SYSTEMDB configuration.

```
[customizable_functionalities]
persistence_datavolume_partition_multipath = true
```

```
ALTER SYSTEM ALTER CONFIGURATION ('global.ini', 'SYSTEM')
SET ('customizable_functionalities', 'PERSISTENCE_DATAVOLUME_PARTITION_MULTIPATH') =
 'true'
WITH RECONFIGURE;
```

Note

You must restart your database after updating the `global.ini` file.

Adding additional data volume partition

Run the following SQL statement against the tenant database to add an additional data volume partition to your tenant database.

```
ALTER SYSTEM ALTER DATAVOLUME ADD PARTITION PATH '/hana/data2/HDB/';
```

Adding a data volume partition is quick. The new data volume partitions are empty after creation. Data is distributed equally across data volumes over time.

Testing host auto failover

We recommend testing your SAP HANA host auto failover scenarios. For more information, see SAP HANA - Host Auto-Failover.

Some words have been redacted and replaced by inclusive terms. These words may appear different in your product, system code or table. For additional details, see Inclusive Language at SAP.

The following table presents the expected results of different test scenarios.

Scenario	Expected result
SAP HANA subordinate node failure using `echo b > /proc/sysrq-trigger`	Subordinate node failover to standby node
SAP HANA coordinator node failure using HDB kill	SAP HANA service failover to standby node (other candidate for coordinator node)

Scenario	Expected result
SAP HANA coordinator node failure while other coordinator nodes act as subordinate nodes	Coordinator node failover to standby node while other coordinator nodes act as subordinate nodes

Test scenarios

- SAP HANA subordinate node failure
- SAP HANA coordinator node failure
- SAP HANA coordinator node failure while other coordinator nodes act as subordinate nodes

SAP HANA subordinate node failure

Check the status of the landscape before testing.

```
hdbadm@hana:/usr/sap/HDB/HDB00/exe/python_support> python
 landscapeHostConfiguration.py
| Host    | Host   | Host   | Failover | Remove | Storage   | Storage   | Failover |
 Failover | NameServer | NameServer | IndexServer | IndexServer | Host    | Host    |
 Worker  | Worker  |
|         | Active | Status | Status   | Status | Config    | Actual    | Config   |
 Actual   | Config     | Actual     | Config      | Actual      | Config  | Actual  |
 Config  | Actual  |
|         |        |        |          |        | Partition | Partition | Group    |
 Group    | Role       | Role       | Role        | Role        | Roles   | Roles   |
 Groups  | Groups  |
| ------- | ------ | ------ | -------- | ------ | --------- | --------- | -------- |
 -------- | ---------- | ---------- | ----------- | ----------- | ------- | ------- |
 ------- | ------- |
| hana    | yes    | ok     |          |        |         1 |         1 | default  |
 default  | coordinator 1   | coordinator     | worker      | coordinator      | worker
  | worker  | default | default |
| hanaw01 | yes    | ok     |          |        |         2 |         2 | default  |
 default  | subordinate      | subordinate      | worker      | subordinate       |
 worker  | worker  | default | default |
| hanaw02 | yes    | ok     |          |        |         3 |         3 | default  |
 default  | subordinate      | subordinate      | worker      | subordinate       |
 worker  | worker  | default | default |
```

```
| hanaw03 | yes    | ok     |          |        |         4 |         4 | default
 | default  | coordinator 3   | subordinate      | worker      | subordinate       |
 worker  | worker  | default | default |
| hanaw04 | yes    | ignore |          |        |         0 |         0 | default  |
 default  | coordinator 2   | subordinate      | standby     | standby     | standby |
 standby | default | -       |

overall host status: ok
```

Run the following command on the subordinate node as `root` to simulate a node crash. In this case, the subordinate node is `hanaw01`.

```
echo b > /proc/sysrq-trigger
```

```
hdbadm@hana:/usr/sap/HDB/HDB00/exe/python_support> python
 landscapeHostConfiguration.py
| Host    | Host   | Host   | Failover | Remove | Storage   | Storage   | Failover |
 Failover | NameServer | NameServer | IndexServer | IndexServer | Host    | Host    |
 Worker  | Worker  |
|         | Active | Status | Status   | Status | Config    | Actual    | Config   |
 Actual   | Config     | Actual     | Config      | Actual      | Config  | Actual  |
 Config  | Actual  |
|         |        |        |          |        | Partition | Partition | Group    |
 Group    | Role       | Role       | Role        | Role        | Roles   | Roles   |
 Groups  | Groups  |
| ------- | ------ | ------ | -------- | ------ | --------- | --------- | -------- |
 -------- | ---------- | ---------- | ----------- | ----------- | ------- | ------- |
 ------- | ------- |
| hana    | yes    | ok     |          |        |         1 |         1 | default  |
 default  | coordinator 1   | coordinator     | worker      | coordinator      | worker
  | worker  | default | default |
| hanaw01 | no     | info   |          |        |         2 |         0 | default  |
 default  | subordinate      | subordinate      | worker      | standby     | worker  |
 standby | default | -       |
| hanaw02 | yes    | ok     |          |        |         3 |         3 | default  |
 default  | subordinate      | subordinate      | worker      | subordinate       |
 worker  | worker  | default | default |
| hanaw03 | yes    | ok     |          |        |         4 |         4 | default
 | default  | coordinator 3   | subordinate      | worker      | subordinate       |
 worker  | worker  | default | default |
| hanaw04 | yes    | info   |          |        |         0 |         2 | default
 | default  | coordinator 2   | subordinate      | standby     | subordinate       |
 standby | worker  | default | default |
```

```
overall host status: info
hdbadm@hana:/usr/sap/HDB/HDB00/exe/python_support>
```

SAP HANA coordinator node failure

Check the status of the landscape before crashing the node.

```
hdbadm@hana:/usr/sap/HDB/HDB00/exe/python_support> python
 landscapeHostConfiguration.py
| Host    | Host   | Host   | Failover | Remove | Storage   | Storage   | Failover |
 Failover | NameServer | NameServer | IndexServer | IndexServer | Host    | Host    |
 Worker  | Worker  |
|         | Active | Status | Status   | Status | Config    | Actual    | Config   |
 Actual   | Config     | Actual     | Config      | Actual      | Config  | Actual  |
 Config  | Actual  |
|         |        |        |          |        | Partition | Partition | Group    |
 Group    | Role       | Role       | Role        | Role        | Roles   | Roles   |
 Groups  | Groups  |
| ------- | ------ | ------ | -------- | ------ | --------- | --------- | -------- |
 -------- | ---------- | ---------- | ----------- | ----------- | ------- | ------- |
 ------- | ------- |
| hana    | yes    | ok     |          |        |         1 |         1 | default  |
 default  | coordinator 1   | coordinator     | worker      | coordinator      | worker
  | worker  | default | default |
| hanaw01 | yes    | ok     |          |        |         2 |         2 | default  |
 default  | subordinate      | subordinate      | worker      | subordinate       |
 worker  | worker  | default | default |
| hanaw02 | yes    | ok     |          |        |         3 |         3 | default  |
 default  | subordinate      | subordinate      | worker      | subordinate       |
 worker  | worker  | default | default |
| hanaw03 | yes    | ok     |          |        |         4 |         4 | default
 | default  | coordinator 3   | subordinate      | worker      | subordinate       |
 worker  | worker  | default | default |
| hanaw04 | yes    | ignore |          |        |         0 |         0 | default  |
 default  | coordinator 2   | subordinate      | standby     | standby     | standby |
 standby | default | -       |

overall host status: ok
hdbadm@hana:/usr/sap/HDB/HDB00/exe/python_support>
```

Use the following command to simulate failure, by interrupting SAP HANA processes, on the coordinator node. In this case, the coordinator node is hana.

```
hdbadm@hana:/usr/sap/HDB/HDB00/exe/python_support> HDB kill
```

```
hdbadm@hana:/usr/sap/HDB/HDB00/exe/python_support> python
 landscapeHostConfiguration.py
nameserver hana:30001 not responding.
| Host    | Host   | Host   | Failover | Remove | Storage   | Storage   | Failover |
 Failover | NameServer | NameServer | IndexServer | IndexServer | Host    | Host    |
 Worker  | Worker  |
|         | Active | Status | Status   | Status | Config    | Actual    | Config   |
 Actual   | Config     | Actual     | Config      | Actual      | Config  | Actual  |
 Config  | Actual  |
|         |        |        |          |        | Partition | Partition | Group    |
 Group    | Role       | Role       | Role        | Role        | Roles   | Roles   |
 Groups  | Groups  |
| ------- | ------ | ------ | -------- | ------ | --------- | --------- | -------- |
 -------- | ---------- | ---------- | ---------- | ---------- | ------- | ------- |
 ------- | ------- |
| hana    | no     | info   |          |        |         1 |         0 | default  |
 default  | coordinator 1   | subordinate      | worker      | standby     | worker  |
 standby | default | -       |
| hanaw01 | yes    | ok     |          |        |         2 |         2 | default  |
 default  | subordinate      | subordinate      | worker      | subordinate       |
 worker  | worker  | default | default |
| hanaw02 | yes    | ok     |          |        |         3 |         3 | default  |
 default  | subordinate      | subordinate      | worker      | subordinate       |
 worker  | worker  | default | default |
| hanaw03 | yes    | ok     |          |        |         4 |         4 | default
 | default  | coordinator 3   | subordinate      | worker      | subordinate       |
 worker  | worker  | default | default |
| hanaw04 | yes    | info   |          |        |         0 |         1 | default
 | default  | coordinator 2   | coordinator     | standby     | coordinator      |
 standby | worker  | default | default |

overall host status: info
hdbadm@hana:/usr/sap/HDB/HDB00/exe/python_support>
```

SAP HANA coordinator node failure while other coordinator nodes act as subordinate nodes

Check the status of the landscape before testing.

```
hdbadm@hana:/usr/sap/HDB/HDB00/exe/python_support> python
 landscapeHostConfiguration.py
```

```
| Host    | Host   | Host   | Failover | Remove | Storage   | Storage   | Failover |
 Failover | NameServer | NameServer | IndexServer | IndexServer | Host    | Host    |
 Worker  | Worker  |
|         | Active | Status | Status   | Status | Config    | Actual    | Config   |
 Actual   | Config     | Actual     | Config      | Actual      | Config  | Actual  |
 Config  | Actual  |
|         |        |        |          |        | Partition | Partition | Group    |
 Group    | Role       | Role       | Role        | Role        | Roles   | Roles   |
 Groups  | Groups  |
| ------- | ------ | ------ | -------- | ------ | --------- | --------- | -------- |
 -------- | ---------- | ---------- | ----------- | ----------- | ------- | ------- |
 ------- | ------- |
| hana    | yes    | ok     |          |        |         1 |         2 | default
 | default  | coordinator 1   | subordinate      | worker      | subordinate       |
 worker  | worker  | default | default |
| hanaw01 | yes    | info   |          |        |         2 |         0 | default  |
 default  | subordinate      | subordinate      | worker      | standby     | worker  |
 standby | default | -       |
| hanaw02 | yes    | ok     |          |        |         3 |         4 | default  |
 default  | subordinate      | subordinate      | worker      | subordinate       |
 worker  | worker  | default | default |
| hanaw03 | yes    | ok     |          |        |         4 |         3 | default
 | default  | coordinator 3   | subordinate      | worker      | subordinate       |
 worker  | worker  | default | default |
| hanaw04 | yes    | info   |          |        |         0 |         1 | default
 | default  | coordinator 2   | coordinator     | standby     | coordinator      |
 standby | worker  | default | default |

overall host status: info
hdbadm@hana:/usr/sap/HDB/HDB00/exe/python_support>
```

Use the following command to simulate failure, by interrupting SAP HANA processes, on the coordinator node. In this case, the coordinator node is `hana04`.

```
hdbadm@hanaw04:/usr/sap/HDB/HDB00> HDB kill
```

```
hdbadm@hana:/usr/sap/HDB/HDB00/exe/python_support> python
 landscapeHostConfiguration.py
| Host    | Host     | Host    | Failover         | Remove | Storage   | Storage   |
 Failover | Failover | NameServer | NameServer | IndexServer | IndexServer | Host    |
 Host    | Worker  | Worker  |
```

```
|         | Active   | Status  | Status           | Status | Config    | Actual    |
 Config   | Actual   | Config     | Actual     | Config      | Actual      | Config  |
 Actual  | Config  | Actual  |
|         |          |         |                  |        | Partition | Partition |
 Group    | Group    | Role       | Role       | Role        | Role        | Roles   |
 Roles   | Groups  | Groups  |
| ------- | -------- | ------- | ---------------- | ------ | --------- | --------- |
 -------- | -------- | ---------- | ---------- | ----------- | ----------- | ------- |
 ------- | ------- | ------- |
| hana    | starting | warning |                  |        |         1 |         1 |
 default  | default  | coordinator 1   | coordinator     | worker      | coordinator
   | worker  | worker  | default | default |
| hanaw01 | starting | warning |                  |        |         2 |         2 |
 default  | default  | subordinate      | subordinate      | worker      | subordinate
      | worker  | worker  | default | default |
| hanaw02 | yes      | ok      |                  |        |         3 |         3 |
 default  | default  | subordinate      | subordinate      | worker      | subordinate
      | worker  | worker  | default | default |
| hanaw03 | yes      | ok      |                  |        |         4 |         4 |
 default  | default  | coordinator 3   | subordinate      | worker      | subordinate
     | worker  | worker  | default | default |
| hanaw04 | no       | warning | failover to hana |        |         0 |         0 |
 default  | default  | coordinator 2   | subordinate      | standby     | standby     |
 standby | standby | default | -       |

overall host status: warning
hdbadm@hana:/usr/sap/HDB/HDB00/exe/python_support> python
 landscapeHostConfiguration.py
| Host    | Host   | Host   | Failover | Remove | Storage   | Storage   | Failover |
 Failover | NameServer | NameServer | IndexServer | IndexServer | Host    | Host    |
 Worker  | Worker  |
|         | Active | Status | Status   | Status | Config    | Actual    | Config   |
 Actual   | Config     | Actual     | Config      | Actual      | Config  | Actual  |
 Config  | Actual  |
|         |        |        |          |        | Partition | Partition | Group    |
 Group    | Role       | Role       | Role        | Role        | Roles   | Roles   |
 Groups  | Groups  |
| ------- | ------ | ------ | -------- | ------ | --------- | --------- | -------- |
 -------- | ---------- | ---------- | ---------- | ---------- | ------- | ------- |
 ------- | ------- |
| hana    | yes    | ok     |          |        |         1 |         1 | default  |
 default  | coordinator 1   | coordinator     | worker      | coordinator      | worker
  | worker  | default | default |
```

```
| hanaw01 | yes    | ok     |          |        |         2 |         2 | default  |
 default  | subordinate      | subordinate      | worker      | subordinate       |
 worker  | worker  | default | default |
| hanaw02 | yes    | ok     |          |        |         3 | default  |
 default  | subordinate      | subordinate      | worker      | subordinate       |
 worker  | worker  | default | default |
| hanaw03 | yes    | ok     |          |        |         4 | default
 | default  | coordinator 3   | subordinate      | worker      | subordinate       |
 worker  | worker  | default | default |
| hanaw04 | no     | ignore |          |        |         0 |         0 | default  |
 default  | coordinator 2   | subordinate      | standby     | standby     | standby |
 standby | default | -       |

overall host status: ok
hdbadm@hana:/usr/sap/HDB/HDB00/exe/python_support>
```

Configure storage (Amazon EFS)

> **Note**
>
> If you plan to use FSx for ONTAP storage for your deployment, refer to SAP HANA on AWS with Amazon FSx for NetApp ONTAP guide, and skip the Amazon EFS configuration steps detailed further here.

Amazon EFS provides easy-to-set-up, scalable, and highly available shared file systems that can be mounted with the NFSv4 client. For scale-out workloads, we recommend using Amazon EFS for SAP HANA shared and backup volumes. You can choose between different performance options for your file systems depending on your requirements. We recommend starting with the General Purpose and Provisioned Throughput options, with approximately 100 MiB/s to 200 MiB/s throughput. To set up your file systems, do the following:

1. Install the `nfs-utils` package in all the nodes in your scale-out cluster.
   - For RHEL, use `yum install nfs-utils`.
   - For SLES, use `zypper install nfs-utils`.
2. Create two Amazon EFS file systems and target mounts for SAP HANA shared and backup in your target VPC and subnet. For detailed steps, follow the instructions specified in the AWS documentation.

3. After the file systems are created, mount the newly created file systems in all the nodes by using the following commands:

```
  # mount -t nfs -o
 nfsvers=4.1,rsize=1048576,wsize=1048576,hard,timeo=600,retrans=2 EFS DNS Name:/ /
hana/shared

  # mount -t nfs -o
 nfsvers=4.1,rsize=1048576,wsize=1048576,hard,timeo=600,retrans=2 EFS DNS Name:/ /
backup
```

> **Note**
>
> If you have trouble mounting the NFS file systems, you might need to adjust your security groups to allow access to port 2049. For details, see Security Groups for Amazon EC2 Instances and Mount Targets in the AWS documentation.

4. Add NFS mount entries to the `/etc/fstab` file in all the nodes to automatically mount these file systems during system restart; for example:

```
  # echo “nfsvers=4.1,rsize=1048576,wsize=1048576,hard,timeo=600,retrans=2 EFS DNS
 Name:/ /hana/shared” >> /etc/fstab
  # echo “nfsvers=4.1,rsize=1048576,wsize=1048576,hard,timeo=600,retrans=2 EFS DNS
 Name:/ /backup” >> /etc/fstab
```

5. Set appropriate permissions and ownership for your target mount points.

Configure ENA Express

SAP HANA scale-out systems require a minimum of 9 Gbps of single flow network bandwidth between nodes. Amazon EC2 instances now support ENA Express, allowing a single flow bandwidth of up to 25 Gbps between instances, without requiring a cluster placement group. For more information, see Improve network performance with ENA Express on Linux instances.

Prerequisites

Before setting up ENA Express for SAP HANA scale-out systems or SAP NetWeaver workloads, verify the following prerequisites.

- Verify that your chosen instance type is certified for SAP HANA or supported for SAP NetWeaver.
  - For **SAP HANA scale-out workloads**, you can enable ENA Express on a certified and supported Amazon EC2 instance. For information on supported instances, see Supported instance types for ENA Express. For information on certified instances, see Certified and Supported SAP HANA Hardware. If an Amazon EC2 instance is certified for scale-out but doesn't support ENA Express, you can continue to use cluster placement group to obtain upto 10 Gbps of single flow network bandwidth.
  - For **SAP NetWeaver workloads**, you can use ENA Express with all of the SAP certified Amazon EC2 instances that support ENA Express. For more information, see the following resources.
    - SAP NetWeaver supported instances
    - SAP Note 1656099 – SAP Applications on AWS: Supported DB/OS and Amazon EC2 products
- Ensure that you are using the minimum required operating system version with the latest kernel version.
  - RHEL for SAP 8.4 and above
  - SLES 12 SP5 for SAP or SLES 15 SP2 for SAP and above

> **Note**
>
> Verify that your chosen operating system is certified for SAP HANA. For more information, see Certified and Supported SAP HANA Hardware.

Configure operating system

You must configure some of the network related parameters at the operating system level to ensure that ENA Express works effectively. This includes configuring the correct maximum transmission unit (mtu) required for ENA Express, and other parameters. For more information, see Prerequisites for ENA Express.

You can also use the check-ena-express-settings.sh script to check the operating system prerequisites. You can run the script from AWS Systems Manager against multiple instances simultaneously. To run the script with Systems Manager, you must ensure that your system has AWS Systems Manager Agent installed. Use the following steps to run the script.

1. Go to https://console.aws.amazon.com/systems-manager/.
2. Select **Node Management** > **Run Command**.

3. Select **Run a command**, and search for **`AWS-RunRemoteScript`**.
4. Choose **`AWS-RunRemoteScript`**, and input the following parameters.

   - **Source Type** – GitHub
   - **Source Info** – `{ "owner": "amzn", "repository": "amzn-ec2-ena-utilities", "path": "ena-express", "getOptions": "branch: main" }`
   - **Command Line** – `check-ena-express-settings.sh eth0`

   > **Note**
   >
   > You must repeat this check for all elastic network interfaces, such as `eth1`, `eth2`, etc.

5. In **Target selection**, specify the instances against which you want to run the script.
6. Select **Run**.

Once the command has completed running, you can review the output, and take corrective actions, if required.

ENA Express settings

After configuring your operating system, you can enable ENA Express for your target instance via AWS Management Console or AWS CLI. For more information, see Configure ENA Express settings. This setting must be repeated on all nodes in scale-out setup.

You do not need a cluster placement group to obtain minimum required single flow network throughput for SAP HANA scale-out systems after successfully enabling ENA Express. To remove a placement group, see Working with placement groups.

Check SAP HANA scale-out performance

After enabling ENA Express, you can use SAP HANA Hardware and Cloud Measurement Tools to check its performance. For additional details, see Measure System Configuration and Performance - Scale-out Systems.

Post Deployment Steps

1. Complete the steps required to connect your instance to your corporate directory service, such as Microsoft Active Directory, if needed.
2. Set up any monitoring required for your environment.
3. Set up a CloudWatch alarm and Amazon EC2 automatic recovery to automatically recover your instance from hardware failures. For details, see Recover Your Instance in the AWS documentation. You can also refer to the Knowledge Center video for detailed instructions.

   > **Note**
   >
   > Automatic recovery is not supported for Amazon EC2 instances running in dedicated hosts.

4. Create an AMI of your newly deployed system to take a full backup of your instance. For details, see Create an AMI from an Amazon EC2 Instance in the AWS documentation.
5. If you have deployed an SAP HANA scale-out cluster, consider adding additional elastic network interfaces and security groups to logically separate network traffic for client, inter-node, and optional SAP HANA System Replication (HSR) communications. For details, see the SAP HANA on AWS Operations Guide.

SAP HANA on AWS Operations Guide

SAP specialists, Amazon Web Services

Last updated: February 2022

Amazon Web Services offers you the ability to run your SAP HANA systems of various sizes and operating systems. Running SAP systems on AWS is very similar to running SAP systems in your data center. To a SAP Basis or NetWeaver administrator, there are minimal differences between the two environments. There are a number of AWS Cloud considerations relating to security, storage, compute configurations, management, and monitoring that will help you get the most out of your SAP HANA implementation on AWS.

This technical article provides the best practices for deployment, operations, and management of SAP HANA systems on AWS. The target audience is SAP Basis and NetWeaver administrators who have experience running SAP HANA systems in an on-premises environment and want to run their SAP HANA systems on AWS.

> **Note**
>
> You must have SAP portal access to view the SAP Notes. For more information, see the SAP Support website.

About this Guide

This guide is part of a content series that provides detailed information about hosting, configuring, and using SAP technologies in the AWS Cloud. For the other guides in the series, ranging from overviews to advanced topics, see the SAP on AWS Technical Documentation home page.

Introduction

This guide provides best practices for operating SAP HANA systems that have been deployed on AWS. This guide is not intended to replace any of the standard SAP documentation. See the following SAP guides and notes:

- SAP Library (help.sap.com) - SAP HANA Administration Guide

- SAP installation guides (SAP portal access required)
- SAP notes (SAP portal access required)

This guide assumes that you have a basic knowledge of AWS. If you are new to AWS, see the following on the AWS website before continuing:

- AWS Getting Started Resource Center
- What is Amazon EC2?

In addition, see the following SAP on AWS guides:

- SAP on AWS Implementation and Operations Guide provides best practices for achieving optimal performance, availability, and reliability, and lower total cost of ownership (TCO) while running SAP solutions on AWS.
- SAP on AWS High Availability Guide explains how to configure SAP systems on Amazon Elastic Compute Cloud (Amazon EC2) to protect your application from various single points of failure.
- SAP on AWS Backup and Recovery Guide explains how to back up SAP systems running on AWS, in contrast to backing up SAP systems on traditional infrastructure.

Administration

This section provides guidance on common administrative tasks required to operate an SAP HANA system, including information about starting, stopping, and cloning systems.

Starting and Stopping EC2 Instances Running SAP HANA Hosts

At any time, you can stop one or multiple SAP HANA hosts. Before stopping the EC2 instance of an SAP HANA host, first stop SAP HANA on that instance.

When you resume the instance, it will automatically start with the same IP address, network, and storage configuration as before. You also have the option of using the EC2 Scheduler to schedule starts and stops of your EC2 instances. The EC2 Scheduler relies on the native shutdown and start-up mechanisms of the operating system. These native mechanisms will invoke the orderly shutdown and startup of your SAP HANA instance. Here is an architectural diagram of how the EC2 Scheduler works:

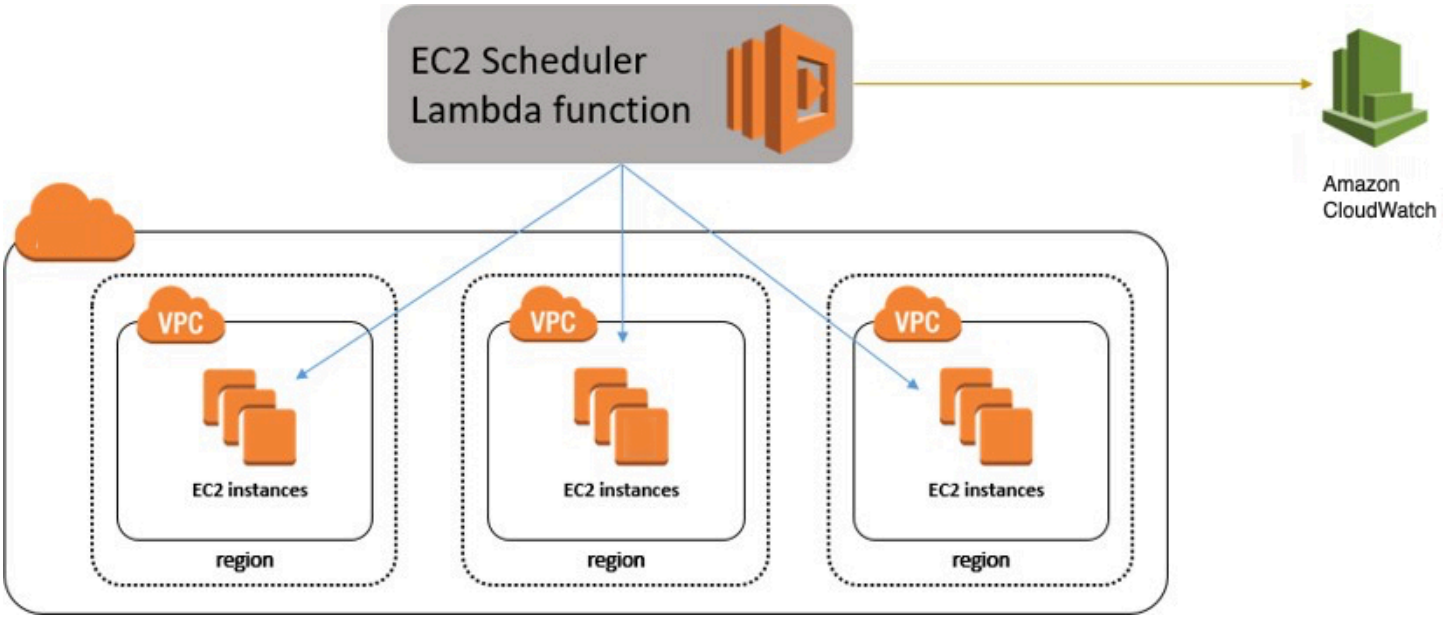


Figure 1: EC2 Scheduler

Tagging SAP Resources on AWS

Tagging your SAP resources on AWS can significantly simplify identification, security, manageability, and billing of those resources. You can tag your resources using the AWS Management Console or by using the `create-tags` functionality of the AWS Command Line Interface (AWS CLI). This table lists some example tag names and tag values:

Tag name	Tag value
Name	SAP server's virtual (host) name
Environment	SAP server's landscape role; for example: SBX, DEV, QAT, STG, PRD.
Application	SAP solution or product; for example: ECC, CRM, BW, PI, SCM, SRM, EP
Owner	SAP point of contact
Service level	Known uptime and downtime schedule

After you have tagged your resources, you can apply specific security restrictions such as access control, based on the tag values. Here is an example of such a policy from the AWS Security blog:

```
{
   "Version" : "2012-10-17",
   "Statement" : [
      {
         "Sid" : "LaunchEC2Instances", "Effect" : "Allow",
         "Action" : [
            "ec2:Describe*", "ec2:RunInstances"
         ],
         "Resource" : [
            "*"
         ]
      },
      {
         "Sid" : "AllowActionsIfYouAreTheOwner",
         "Effect" : "Allow",
         "Action" : [
            "ec2:StopInstances",
            "ec2:StartInstances",
            "ec2:RebootInstances",
            "ec2:TerminateInstances"
         ],
         "Condition" : {
            "StringEquals" : {
               "ec2:ResourceTag/PrincipalId" : "${aws:userid}"
            }
         },
         "Resource" : [
            "*"
         ]
      }
   ]
}
```

The AWS Identity and Access Management (IAM) policy allows only specific permissions based on the tag value. In this scenario, the current user ID must match the tag value in order for the user to be granted permissions. For more information on tagging, see the AWS documentation and AWS blog.

Monitoring

You can use various AWS, SAP, and third-party solutions to monitor your SAP workloads. Here are some of the core AWS monitoring services:

- Amazon CloudWatch – CloudWatch is a monitoring service for AWS resources. It's critical for SAP workloads where it's used to collect resource utilization logs and to create alarms to automatically react to changes in AWS resources.
- AWS CloudTrail – CloudTrail keeps track of all API calls made within your AWS account. It captures key metrics about the API calls and can be useful for automating trail creation for your SAP resources.

Configuring CloudWatch detailed monitoring for SAP resources is mandatory for getting AWS and SAP support. You can use native AWS monitoring services in a complementary fashion with the SAP Solution Manager. You can find third-party monitoring tools in AWS Marketplace.

Automation

AWS offers multiple options for programmatically scripting your resources to operate or scale them in a predictable and repeatable manner. You can use AWS CloudFormation to automate and operate SAP systems on AWS. Here are some examples for automating your SAP environment on AWS:

Area	Activities	AWS services
Infrastructure deployment	Provision new SAP environment SAP system cloning	AWS CloudFormation AWS CLI
Capacity management	Automate scale-up/scale-out of SAP application servers	AWS Lambda AWS CloudFormation
Operations	SAP backup automation (see the backup example)	Amazon CloudWatch AWS Systems Manager

	Performing monitoring and visualization	

Patching

There are two ways for you to patch your SAP HANA database, with options for minimizing cost and/or downtime. With AWS, you can provision additional servers as needed to minimize downtime for patching in a cost-effective manner. You can also minimize risks by creating on-demand copies of your existing production SAP HANA databases for lifelike production readiness testing.

This table summarizes the tradeoffs of the two patching methods:

Patching method	Benefits	Tradeoff	Technologies available
Patch an existing server	No costs for additional on-demand instances Lowest levels of relative complexit y and setup tasks involved	Need to patch the existing operating system and database Longest downtime to the existing server and database	Native OS patching tools Patch Manager Native SAP HANA patching tools
Provision and patch a new server	Leverage latest AMIs (only database patch is required) Shortest downtime on the existing server and database Option to patch and test the operating system and database separately or together	More costs for additional on-demand instances More complexity and setup tasks involved	Amazon Machine Image (AMI) AWS CLI AWS CloudFormation SAP HANA System Replication SAP HANA System Cloning SAP HANA backups

Patching method	Benefits	Tradeoff	Technologies available
			SAP Notes: 1984882 - Using HANA System Replication for Hardware Exchange with minimum/zero downtime 1913302 - HANA: Suspend DB connections for short maintenance tasks

The first method (patch an existing server) involves patching the operating system (OS) and database (DB) components of your SAP HANA server. The goal of this method is to minimize any additional server costs and to avoid any tasks needed to set up additional systems or tests. This method may be most appropriate if you have a well-defined patching process and are satisfied with your current downtime and costs. With this method you must use the correct operating system (OS) update process and tools for your Linux distribution. See this SUSE blog and Red Hat FAQ, or check each vendor's documentation for their specific processes and procedures.

In addition to patching tools provided by our Linux partners,AWS offers a free of charge patching service called Patch Manager. Patch Manager is an automated tool that helps you simplify your OS patching process. You can scan your EC2 instances for missing patches and automatically install them, select the timing for patch rollouts, control instance reboots, and many other tasks. You can also define auto-approval rules for patches with an added ability to black-list or white-list specific patches, control how the patches are deployed on the target instances (e.g., stop services before applying the patch), and schedule the automatic rollout through maintenance windows.

The second method (provision and patch a new server) involves provisioning a new EC2 instance that will receive a copy of your source system and database. The goal of the method is to minimize downtime, minimize risks (by having production data and executing production-like testing), and have repeatable processes. This method may be most appropriate if you are looking for higher degrees of automation to enable these goals and are comfortable with the trade- offs. This method

is more complex and has a many more options to fit your requirements. Certain options are not exclusive and can be used together. For example, your AWS CloudFormation template can include the latest Amazon Machine Images (AMIs), which you can then use to automate the provisioning, set up, and configuration of a new SAP HANA server.

For more information, see Automated patching.

Backup and Recovery

This section provides an overview of the AWS services used in the backup and recovery of SAP HANA systems and provides an example backup and recovery scenario. This guide does not include detailed instructions on how to execute database backups using native HANA backup and recovery features or third- party backup tools. Please refer to the standard OS, SAP, and SAP HANA documentation or the documentation provided by backup software vendors. In addition, backup schedules, frequency, and retention periods might vary with your system type and business requirements. See the following standard SAP documentation for guidance on these topics.

> **Note**
>
> For a discussion of both general and advanced backup and recovery concepts for SAP systems on AWS, see the SAP on AWS Backup and Recovery Guide.

SAP Note	Description
1642148	FAQ: SAP HANA Database Backup & Recovery
1821207	Determining required recovery files
1869119	Checking backups using hdbbackupcheck
1873247	Checking recoverability with hdbbackupdiag --check
1651055	Scheduling SAP HANA Database Backups in Linux
2484177	Scheduling backups for multi-tenant SAP HANA Cockpit 2.0

Creating an Image of an SAP HANA System

You can use the AWS Management Console or the command line to create your own AMI based on an existing instance. For more information, see the AWS documentation. You can use an AMI of your SAP HANA instance for the following purposes:

- **To create a full offline system backup** (of the OS /usr/sap, HANA shared, backup, data, and log files) – AMIs are automatically saved in multiple Availability Zones within the same AWS Region.
- **To move a HANA system from one AWS Region to another** – You can create an image of an existing EC2 instance and move it to another AWS Region by following the instructions in the AWS documentation. When the AMI has been copied to the target AWS Region, you can launch the new instance there.
- **To clone an SAP HANA system** – You can create an AMI of an existing SAP HANA system to create an exact clone of the system. See the next section for additional information.

> **Note**
>
> See Restoring SAP HANA Backups and Snapshots later in this whitepaper to view the recommended restoration steps for production environments.

> **Tip**
>
> The SAP HANA system should be in a consistent state before you create an AMI. To do this, stop the SAP HANA instance before creating the AMI or by following the instructions in SAP Note 1703435.

AWS Services and Components for Backup Solutions

AWS provides a number of services and options for storage and backup, including Amazon Simple Storage Service (Amazon S3), AWS Identity and Access Management (IAM), and S3 Glacier.

Amazon S3

Amazon S3 is the center of any SAP backup and recovery solution on AWS. It provides a highly durable storage infrastructure designed for mission-critical and primary data storage. It is designed

to provide 99.999999999% durability and 99.99% availability over a given year. See the Amazon S3 documentation for detailed instructions on how to create and configure an S3 bucket to store your SAP HANA backup files.

IAM

With IAM, you can securely control access to AWS services and resources for your users. You can create and manage AWS users and groups and use permissions to grant user access to AWS resources. You can create roles in IAM and manage permissions to control which operations can be performed by the entity, or AWS service, that assumes the role. You can also define which entity is allowed to assume the role.

During the deployment process, AWS CloudFormation creates an IAM role that allows access to get objects from and/or put objects into Amazon S3. That role is subsequently assigned to each EC2 instance that is hosting SAP HANA master and worker nodes at launch time as they are deployed.

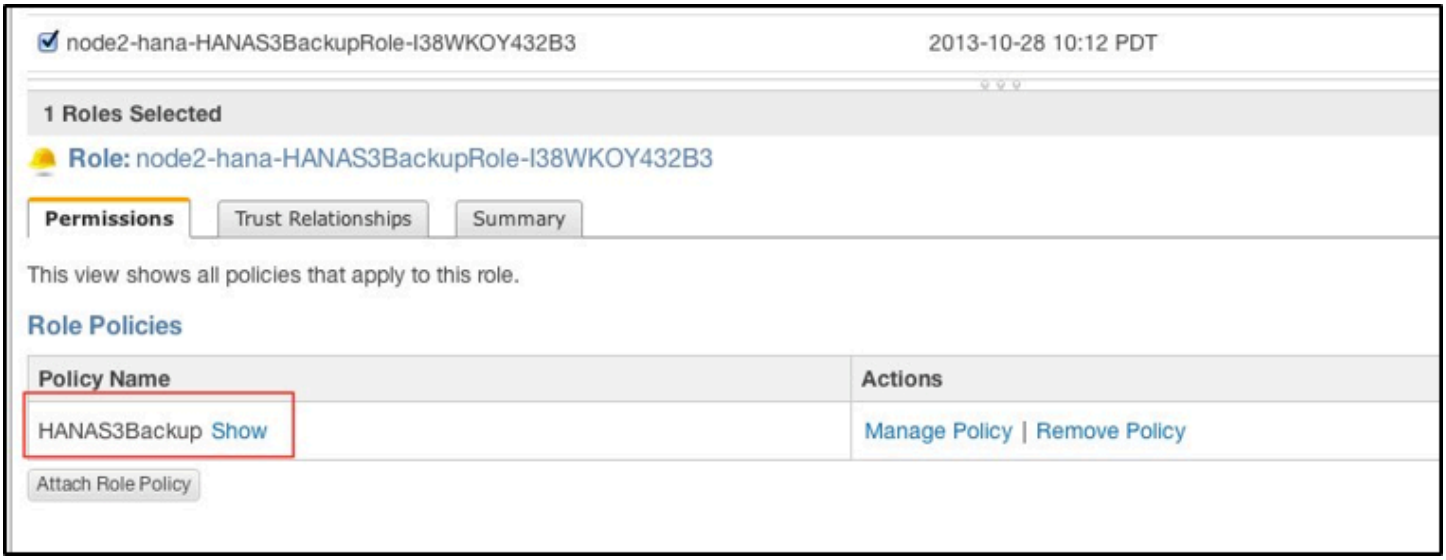


Figure 2: IAM role example

To ensure security that applies the principle of least privilege, permissions for this role are limited only to actions that are required for backup and recovery.

```
{"Statement":[
   {"Resource":"arn:aws:s3::: <your-s3-bucket-name>/*",
      "Action":["s3:GetObject","s3:PutObject","s3:DeleteObject",
"s3:ListBucket","s3:Get*","s3:List*"], "Effect":"Allow"},
```

```
{"Resource":"*","Action":["s3:List*","ec2:Describe*","ec2:Attach NetworkInterface",

"ec2:AttachVolume","ec2:CreateTags","ec2:CreateVolume","ec2:RunI nstances",
   "ec2:StartInstances"],"Effect":"Allow"}]}
```

To add functions later, you can use the AWS Management Console to modify the IAM role.

S3 Glacier

S3 Glacier is an extremely low-cost service that provides secure and durable storage for data archiving and backup. S3 Glacier is optimized for data that is infrequently accessed and provides multiple options such as expedited, standard, and bulk methods for data retrieval. With standard and bulk retrievals, data is available in 3-5 hours or 5-12 hours, respectively.

However, with expedited retrieval, S3 Glacier provides you with an option to retrieve data in 3-5 minutes, which can be ideal for occasional urgent requests. With S3 Glacier, you can reliably store large or small amounts of data for as little as $0.01 per gigabyte per month, a significant savings compared to on-premises solutions. You can use lifecycle policies, as explained in the *Amazon S3 Developer Guide*, to push SAP HANA backups to S3 Glacier for long-term archiving.

Backup Destination

The primary difference between backing up SAP systems on AWS compared with traditional on-premises infrastructure is the backup destination. Tape is the typical backup destination used with on-premises infrastructure. On AWS, backups are stored in Amazon S3. Amazon S3 has many benefits over tape, including the ability to automatically store backups offsite from the source system, since data in Amazon S3 is replicated across multiple facilities within the AWS Region.

SAP HANA systems provisioned with AWS Launch Wizard for SAP are configured with a set of EBS volumes to be used as an initial local backup destination. HANA backups are first stored on these local EBS volumes and then copied to Amazon S3 for long-term storage.

You can use SAP HANA Studio, SQL commands, or the DBA Cockpit to start or schedule SAP HANA data backups. Log backups are written automatically unless disabled. The /backup file system is configured as part of the deployment process.

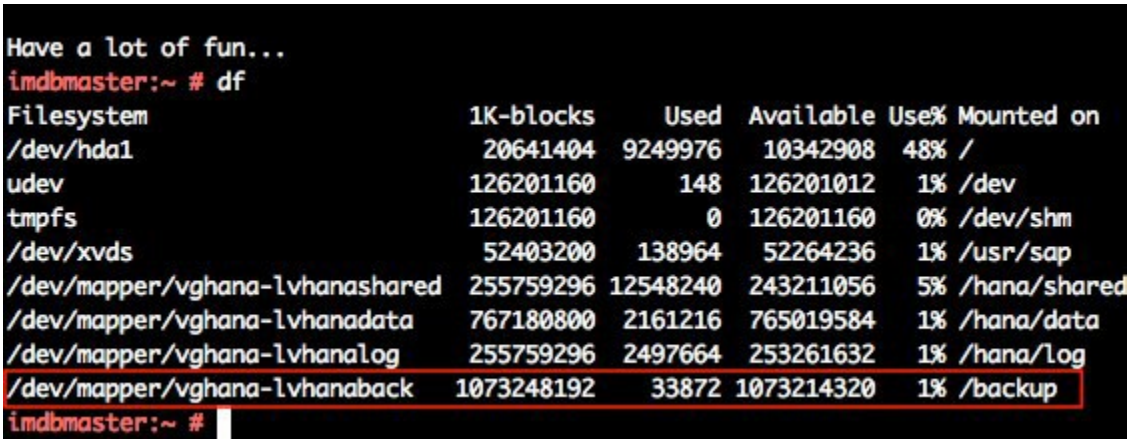


Figure 3: SAP HANA file system layout

The SAP HANA global.ini configuration file has been customized for database backups to go directly to `/backup/data/<SID>`, while automatic log archival files go to `/backup/log/<SID>`.

```
[persistence]
basepath_shared = no
savepoint_intervals = 300
basepath_datavolumes = /hana/data/<SID>
basepath_logvolumes = /hana/log/<SID>
basepath_databackup = /backup/data/<SID>
basepath_logbackup = /backup/log/<SID>
```

Some third-party backup tools like Commvault, NetBackup, and IBM Tivoli Storage Manager (IBM TSM) are integrated with Amazon S3 capabilities and can be used to trigger and save SAP HANA backups directly into Amazon S3 without needing to store the backups on EBS volumes first.

AWS CLI

The AWS Command Line Interface (AWS CLI), which is a unified tool to manage AWS services, is installed as part of the base image. Using various commands, you can control multiple AWS services from the command line directly and automate them through scripts. Access to your S3 bucket is available through the IAM role assigned to the instance (as discussed earlier). Using the AWS CLI commands for Amazon S3, you can list the contents of the previously created bucket, back up files, and restore files, as explained in the AWS CLI documentation.

```
imdbmaster:/backup # aws s3 ls --region=us-east-1 s3://node2- hana-s3bucket-
gcynh5v2nqs3
```

```
Bucket: node2-hana-s3bucket-gcynh5v2nqs3
Prefix:
      LastWriteTime      Length      Name
      -------------      ------      ----
```

Backup Example

Here are the steps you can take for a typical backup task:

1. In the SAP HANA Backup Editor, choose **Open Backup Wizard**. You can also open the Backup Wizard by right-clicking the system that you want to back up and choosing **Back Up**.
    1. Select the destination type **File**. This will back up the database to files in the specified file system.
    2. Specify the backup destination (`/backup/data/<SID>`) and the backup prefix.

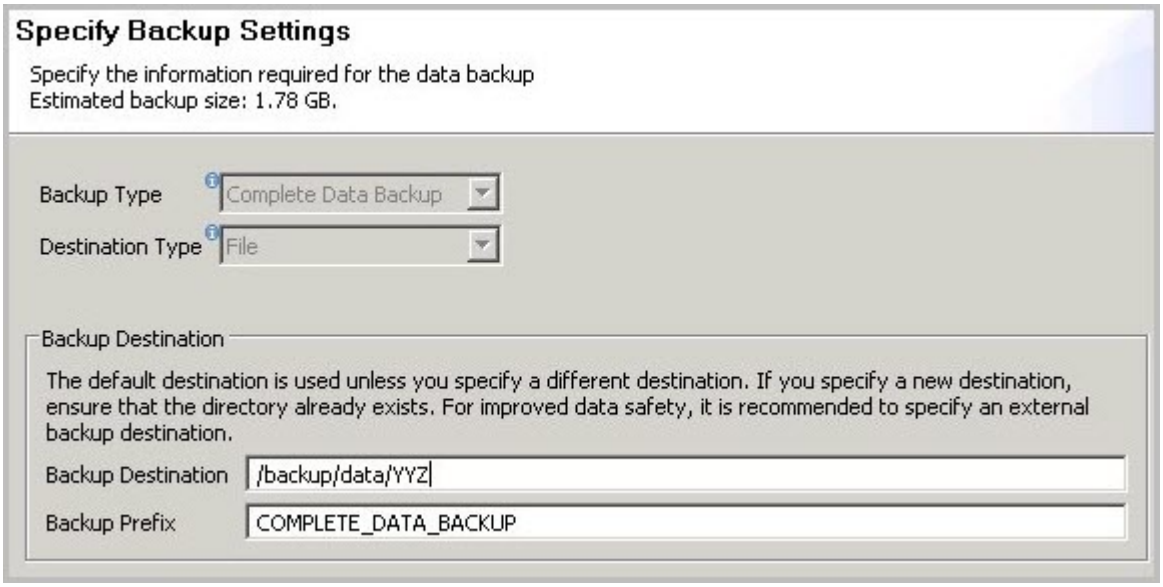


Figure 4: SAP HANA backup example

3. Choose **Next** and then **Finish**. A confirmation message will appear when the backup is complete.
4. Verify that the backup files are available at the OS level. The next step is to push or synchronize the backup files from the /backup file system to Amazon S3 by using the aws s3 sync command.

```
imdbmaster:/ # aws s3 sync backup s3://node2-hana-s3bucket- gcynh5v2nqs3 --
region=us-east-1
```

2. Use the AWS Management Console to verify that the files have been pushed to Amazon S3. You can also use the aws s3 ls command shown previously in the AWS Command Line Interface section.

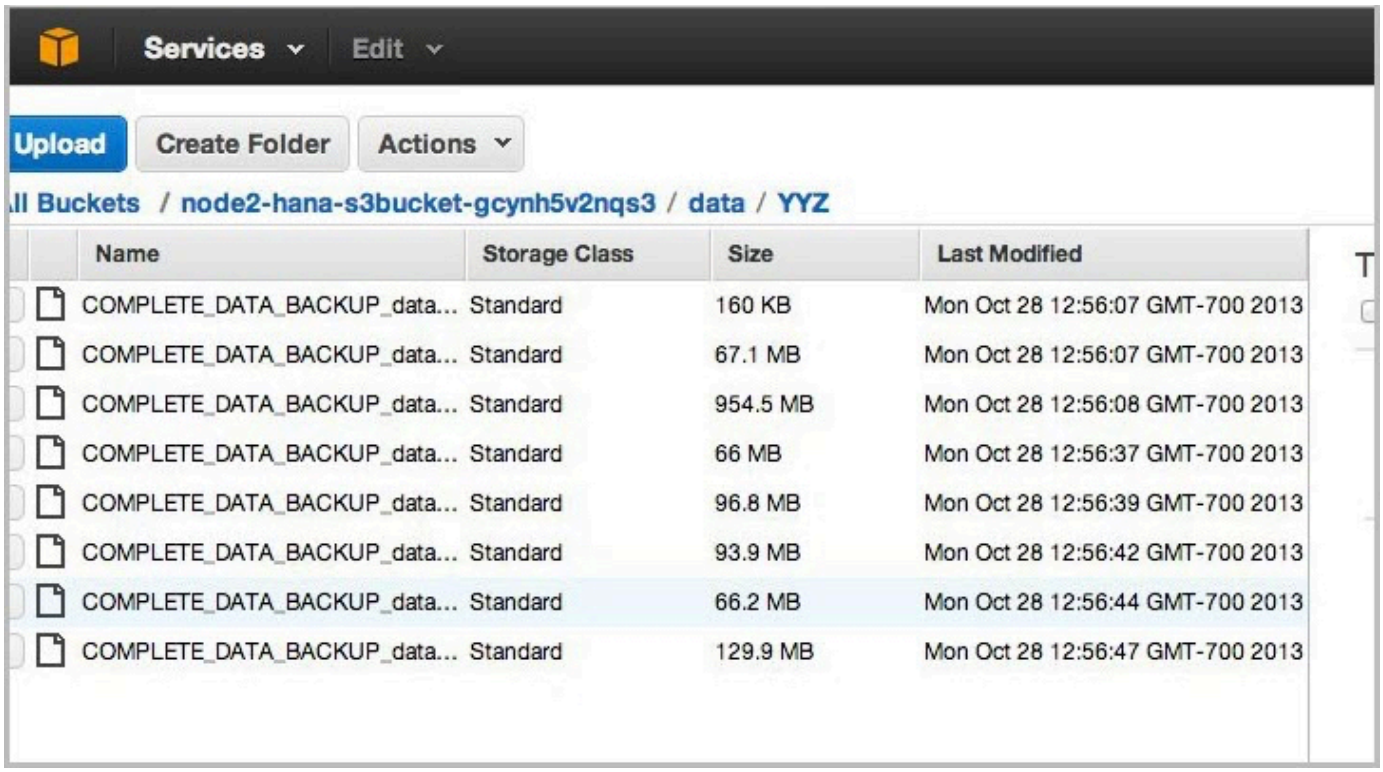


Figure 5: Amazon S3 bucket contents after backup

Tip

The `aws s3 sync` command will only upload new files that don't exist in Amazon S3. Use a periodically scheduled `cron` job to sync, and then delete files that have been uploaded. See SAP Note 1651055 for scheduling periodic backup jobs in Linux, and extend the supplied scripts with `aws s3 sync` commands.

Scheduling and Executing Backups Remotely

You can use the AWS Systems Manager Run Command, along with Amazon CloudWatch Events, to schedule backups of your SAP HANA system remotely without the need to log in to the EC2 instances. You can also use `cron` or any other instance-level scheduling mechanism.

The Systems Manager Run Command lets you remotely and securely manage the configuration of your managed instances. A managed instance is any EC2 instance or on-premises machine in your hybrid environment that has been configured for Systems Manager. The Run Command enables you to automate common administrative tasks and perform ad hoc configuration changes at

scale. You can use the Run Command from the Amazon EC2 console, the AWS CLI, Windows PowerShell, or the AWS SDKs.

Systems Manager Prerequisites

Systems Manager has the following prerequisites.

Supported operating system (Linux)	Instances must run a supported version of Linux. 64-bit and 32-bit systems: • Amazon Linux 2014.09, 2014.03 or later • Ubuntu Server 16.04 LTS, 14.04 LTS, or 12.04 LTS • Red Hat Enterprise Linux (RHEL) 6.5 or later • CentOS 6.3 or later 64-bit systems only: • Amazon Linux 2015.09, 2015.03 or later • Red Hat Enterprise Linux (RHEL) 7.x or later • CentOS 7.1 or later • SUSE Linux Enterprise Server (SLES) 12 or higher

	For the latest information about supported operating systems, see the AWS Systems Manager documentation.
Roles for Systems Manager	Systems Manager requires an IAM role for instances that will process commands and a separate role for users who are executing commands. Both roles require permissio n policies that enable them to communica te with the Systems Manager API. You can choose to use Systems Manager managed policies or you can create your own roles and specify permissions. For more information, see Configuring Security Roles for Systems Manager in the AWS documentation. If you are configuring on-premises servers or virtual machines (VMs) that you want to configure using Systems Manager, you must also configure an IAM service role. For more information, see Create an IAM Service Role in the AWS documentation.
SSM Agent (EC2 Linux instances)	AWS Systems Manager Agent (SSM Agent) processes Systems Manager requests and configures your machine as specified in the request. You must download and install SSM Agent to your EC2 Linux instances. For more information, see Installing SSM Agent on Linux in the AWS documentation.

To schedule remote backups, follow these high-level steps:

1. Install and configure SSM Agent on the EC2 instance. For detailed installation steps, see the AWS Systems Manager documentation.

2. Provide SSM access to the EC2 instance role that is assigned to the SAP HANA instance. For detailed information on how to assign SSM access to a role, see the AWS Systems Manager documentation.

3. Create an SAP HANA backup script. You can use the following sample script as a starting point and modify it to meet your requirements.

```
#!/bin/sh
set -x
S3Bucket_Name=<Name of the S3 bucket where backup files will be copied>
TIMESTAMP=$(date +\%F\_%H\%M)
exec 1>/backup/data/${SAPSYSTEMNAME}/${TIMESTAMP}_backup_log.out 2>&1
echo "Starting to take backup of Hana Database and Upload the backup files to S3"
echo "Backup Timestamp for $SAPSYSTEMNAME is $TIMESTAMP" BACKUP_PREFIX=
${SAPSYSTEMNAME}_${TIMESTAMP}
echo $BACKUP_PREFIX
# source HANA environment
source $DIR_INSTANCE/hdbenv.sh
# execute command with user key
hdbsql -U BACKUP "backup data using file ('$BACKUP_PREFIX')" echo "HANA Backup is
 completed"
echo "Continue with copying the backup files in to S3" echo $BACKUP_PREFIX
sudo -u root /usr/local/bin/aws s3 cp --recursive
/backup/data/${SAPSYSTEMNAME}/ s3://${S3Bucket_Name}/bkps/${SAPSYSTEMNAME}/data/ --
exclude "*" --include "${BACKUP_PREFIX}*"
echo "Copying HANA Database log files in to S3"
sudo -u root /usr/local/bin/aws s3 sync
/backup/log/${SAPSYSTEMNAME}/ s3://${S3Bucket_Name}/bkps/${SAPSYSTEMNAME}/log/ --
exclude "*" --include "log_backup*"
sudo -u root /usr/local/bin/aws s3 cp
/backup/data/${SAPSYSTEMNAME}/${TIMESTAMP}_backup_log.out
s3://${S3Bucket_Name}/bkps/${SAPSYSTEMNAME}
```

> **Note**
>
> This script takes into consideration that `hdbuserstore` has a key named `Backup`.

4. Test a one-time backup by executing an `ssm` command directly.

> **Note**
>
> For this command to execute successfully, you will have to enable `<sid>adm login` using `sudo`.

```
aws ssm send-command --instance-ids <HANA master instance ID> --document-name AWS-
RunShellScript
--parameters commands="sudo - u <HANA_SID>adm TIMESTAMP=$(date +\%F\_%H\%M)
 SAPSYSTEMNAME=<HANA_SID>
DIR_INSTANCE=/hana/shared/${SAPSYSTEMNAME}/HDB00 -i /usr/sap/HDB/HDB00/
hana_backup.sh"
```

5. Using CloudWatch Events, you can schedule backups remotely at any desired frequency. Navigate to the CloudWatch Events page and create a rule.

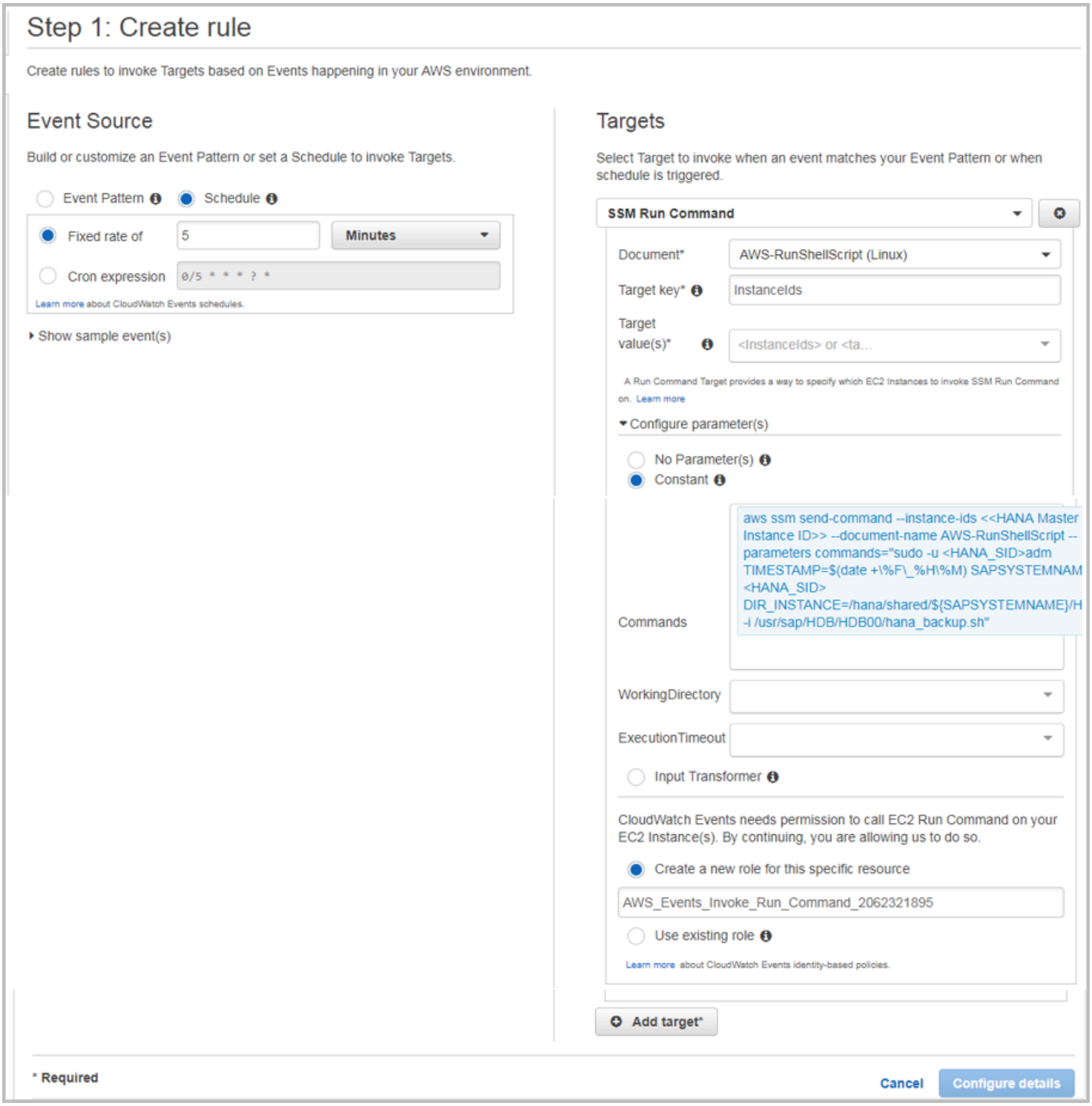


Figure 6: Creating Amazon CloudWatch Events rules

When configuring the rule:

1. Choose **Schedule**.

2. Select **SSM Run Command** as the target.
3. Select **AWS-RunShellScript (Linux)** as the document type.
4. Choose **InstanceIds** or **Tags** as the target key.
5. Choose **Constant** under **Configure Parameters**, and type the `run` command.

Restoring SAP HANA Backups and Snapshots

Restoring SAP Backups

To restore your SAP HANA database from a backup, perform the following steps:

1. If the backup files are not already available in the /backup file system but are in Amazon S3, restore the files from Amazon S3 by using the aws s3 cp command. This command has the following syntax:

   ```
   aws --region <region> cp <s3-bucket/path> --recursive <backup- prefix>*.
   ```

   For example:

   ```
   imdbmaster:/backup/data/YYZ # aws --region us-east-1 s3 cp s3://node2-hana-
   s3bucket-gcynh5v2nqs3/data/YYZ . --recursive -- include COMPLETE*
   ```

2. Recover the SAP HANA database by using the Recovery Wizard as outlined in the SAP HANA Administration Guide. Specify **File** as the destination type and enter the correct backup prefix.

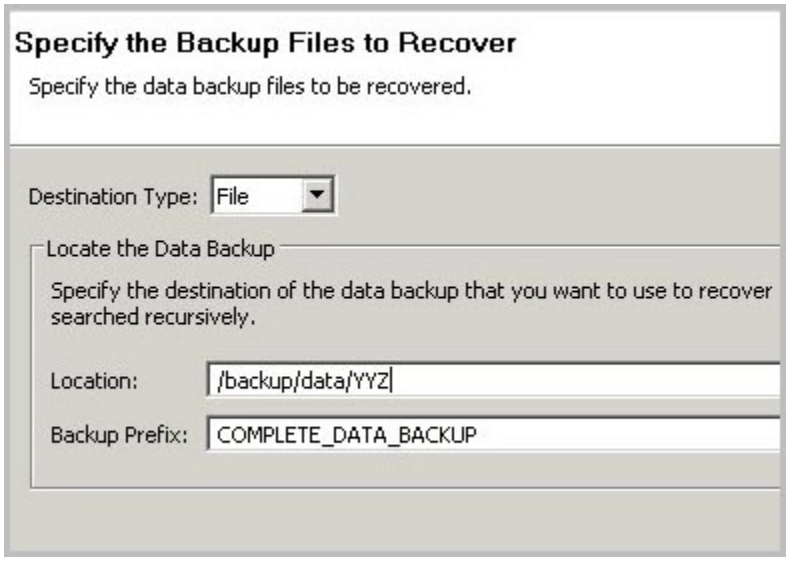


Figure 7: Restore example

3. When the recovery is complete, you can resume normal operations and clean up backup files from the `/backup/<SID>/*` directories.

Restoring EBS Snapshots

To restore EBS snapshots, perform the following steps:

1. Create a new volume from the snapshot:

```
aws ec2 create-volume --region us-west-2 --availability-zone us- west-2a --
snapshot-id snap-1234abc123a12345a --volume-type gp2
```

2. Attach the newly created volume to your EC2 host:

```
aws ec2 attach-volume --region=us-west-2 --volume-id vol- 4567c123e45678dd9 --
instance-id i-03add123456789012 --device /dev/sdf
```

3. Mount the logical volume associated with SAP HANA data on the host:

```
mount /dev/sdf /hana/data
```

4. Start your SAP HANA instance.

Note

For large mission-critical systems, we highly recommend that you execute the volume initialization command on the database data and log volumes after restoring the AMI but before starting the database. Executing the volume initialization command will help you avoid extensive wait times before the database is available. Here is the sample `fio` command that you can use:

```
sudo fio –filename=/dev/xvdf –rw=read –bs=128K –iodepth=32 –
 ioengine=libaiodirect=1 –name=volume-initialize
```

For more information about initializing Amazon EBS volumes, see the AWS documentation.

Restoring AMI Snapshots

You can restore your SAP HANA AMI snapshots through the AWS Management Console. Open the Amazon EC2 console, and choose **AMIs** in the navigation pane.

Choose the AMI that you want to restore, expand **Actions**, and then choose **Launch**.

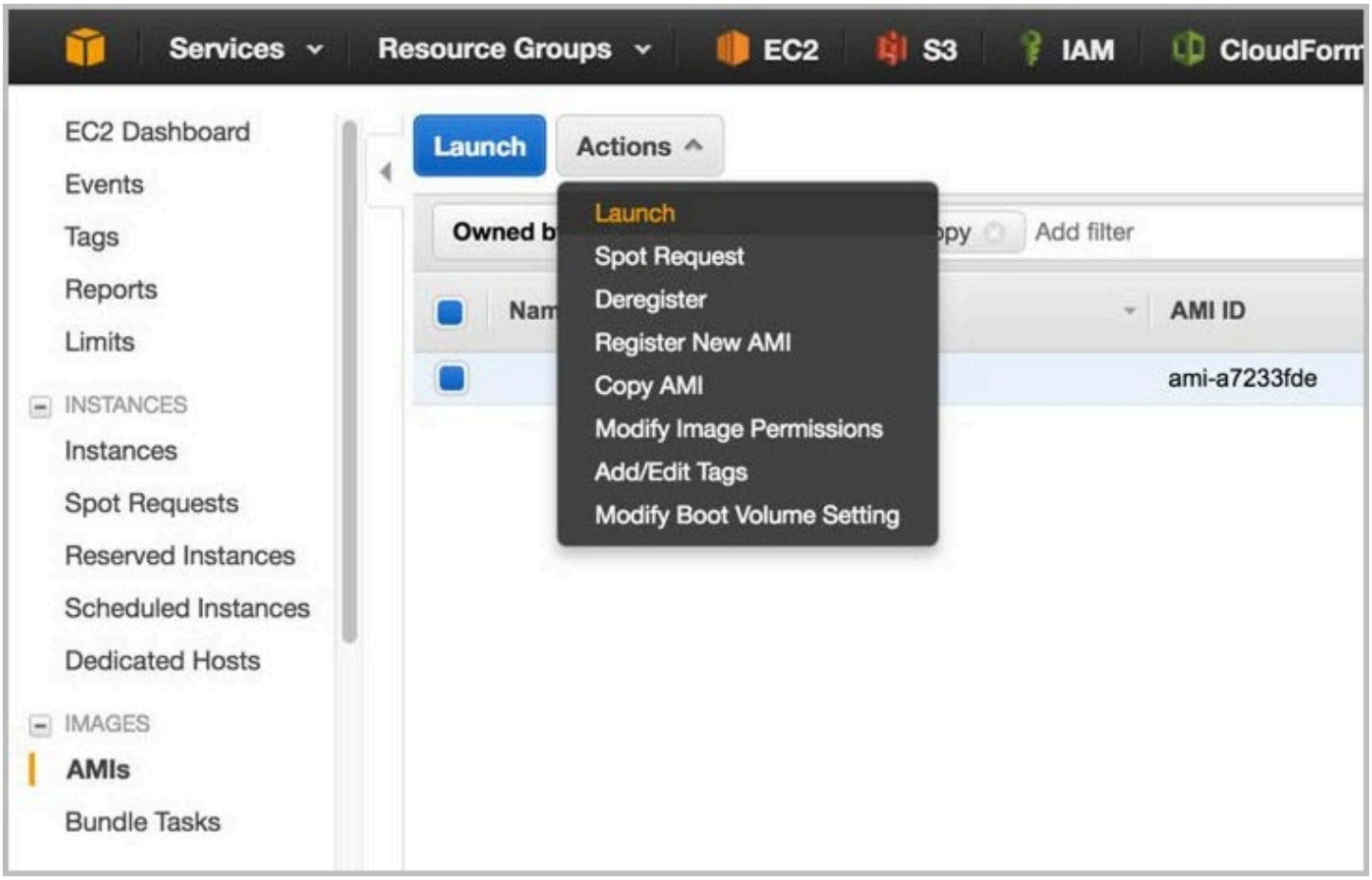


Figure 8: Restoring an AMI snapshot

Automated patching for SAP HANA

Maintaining the SAP HANA database software version keeps the database on with supported software versions, and enables you to stay updated with security fixes and software improvements.

This section provides information about automating the update of your SAP HANA database software version with AWS Systems Manager. You must have a good understanding of SAP HANA

patching processes, paths, and prerequisites. Apart from SAP HANA, you must also keep all other components of an SAP system updates with an SAP-supported version.

Topics

- SAP references
- Architecture
- Prerequisites
- SSM automation document
- AWS services
- Prepare to run the SSM automation document
- Troubleshoot
- SAP HANA version reporting

SAP references

It is recommended that you familiarize yourself with the following SAP documents to understand SAP HANA patching processes, paths, and prerequisites.

You must have SAP portal access to view the SAP Notes.

- SAP Note : 2115815 - FAQ: SAP HANA Database Patches and Upgrades
- SAP Note : 1948334 - SAP HANA Database Update Paths for SAP HANA Maintenance Revisions
- SAP Note : 2378962 - SAP HANA 2.0 Revision and Maintenance Strategy
- SAP HANA Master Guide : Updating an SAP HANA System Landscape

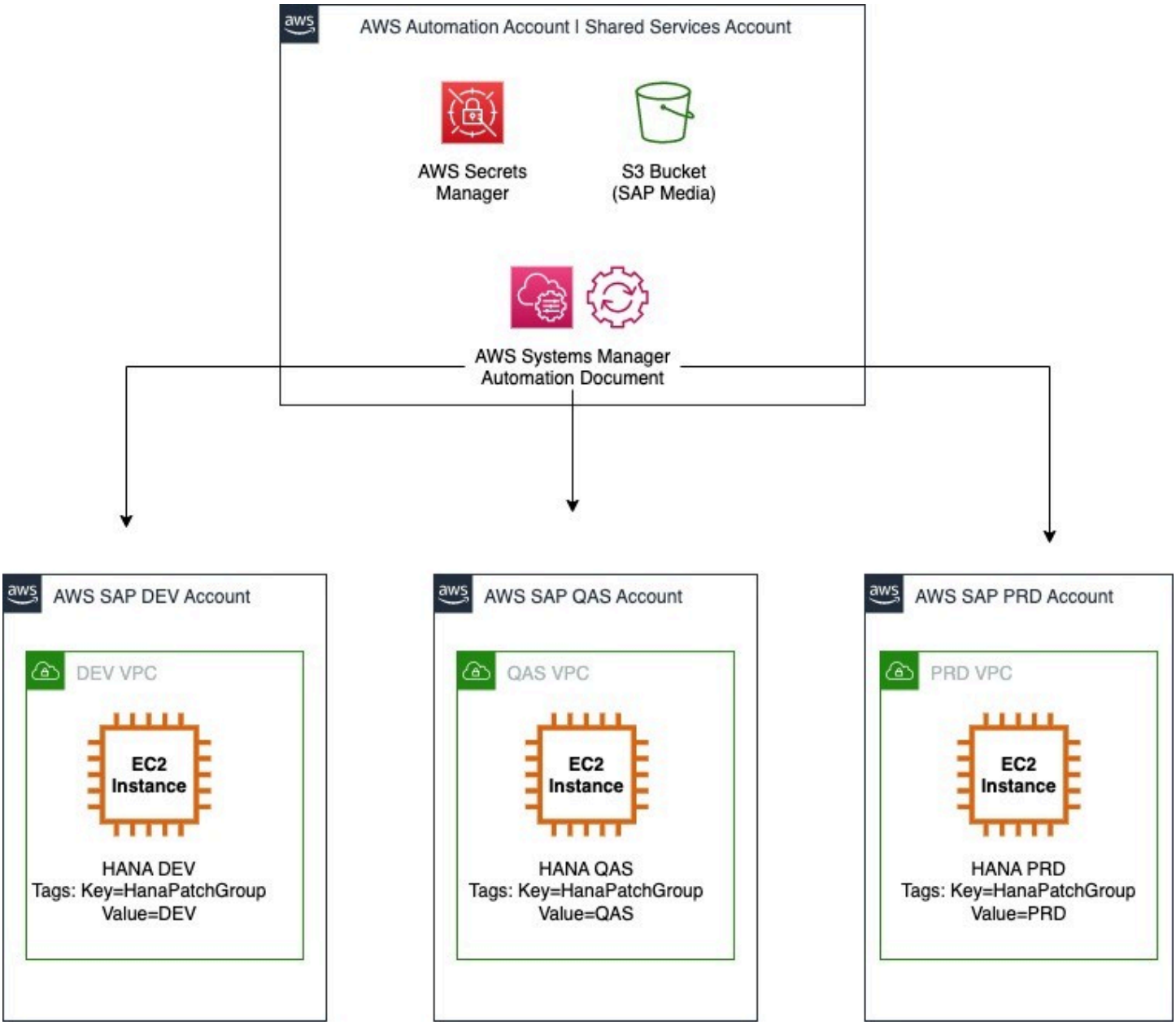
Architecture

Based on your governance strategy, you can centralize AWS SSM automation document into a Shared Services account or an automation account. For more information, see Infrastructure OU - Shared Services account.

A Shared Services account is used in this document. The AWS SSM automation document is stored in this account. It is connected to the child AWS accounts that host Amazon EC2 instances running SAP HANA workloads. The Shared Services account also hosts the Amazon S3 bucket containing

the SAP HANA media software, and specific parameters stored in AWS Secrets Manager. These parameters are required for the automation document to run.

The automation account can be a production account running SAP workloads or a dedicated account for only running SSM automation documents. A Shared Services account for automation reduces the administrative overhead by maintaining the automation document and its dependencies in the same account.

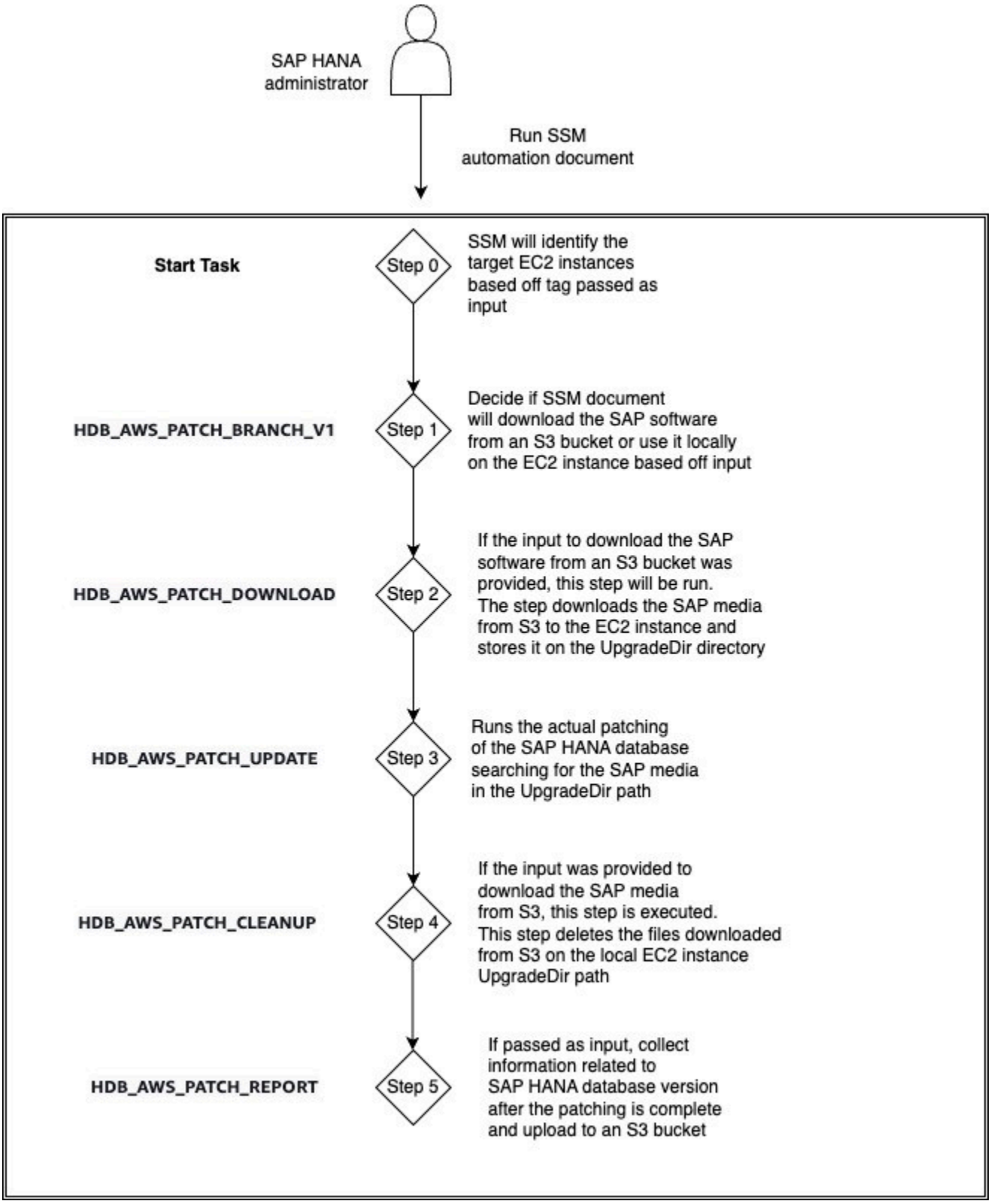


Prerequisites

- You must setup IAM permissions in the Shared Services account as well as the connected child accounts. This is to enable AWS Systems Manager to run automation documents from the Shared Services account to connected accounts. For more information, see Running automations in multiple AWS Regions and accounts.
- You must set up your Amazon EC2 instance running SAP workloads to be managed by AWS Systems Manager. For more information, see Working with SSM Agent on Amazon EC2 instances on Linux.

SSM automation document

You can find the code for the SSM automation document on AWS Samples GitHub repository. For more information, see sap-hana-patch-sample.yml. The following diagram illustrates the steps run by the SSM automation document.


SAP HANA
administrator
Run SSM
automation document
Start Task
Step 0
SSM will identify the
target EC2 instances
based off tag passed as
input
HDB_AWS_PATCH_BRANCH_V1
Step 1
Decide if SSM document
will download the SAP software
from an S3 bucket or use it locally
on the EC2 instance based off input
HDB_AWS_PATCH_DOWNLOAD
Step 2
If the input to download the SAP
software from an S3 bucket was
provided, this step will be run.
The step downloads the SAP media
from S3 to the EC2 instance and
stores it on the UpgradeDir directory
HDB_AWS_PATCH_UPDATE
Step 3
Runs the actual patching
of the SAP HANA database
searching for the SAP media
in the UpgradeDir path
HDB_AWS_PATCH_CLEANUP
Step 4
If the input was provided to
download the SAP media
from S3, this step is executed.
This step deletes the files downloaded
from S3 on the local EC2 instance
UpgradeDir path
HDB_AWS_PATCH_REPORT
Step 5
If passed as input, collect
information related to
SAP HANA database version
after the patching is complete
and upload to an S3 bucket

AWS services

The sample code interacts with the following AWS services to run the SSM automation documents.

Services

Amazon S3

You have the following three options to store the SAP HANA software media.

- Amazon EBS volume attached to your Amazon EC2 instance
- NFS mount point – Amazon EFS or Amazon FSx for NetApp ONTAP
- Amazon S3 bucket

An Amazon S3 bucket can be used to store all the SAP HANA software media containing different versions. The target software version to be used in the SSM automation document can be selected from here.

Store the SAP media in a compressed `.SAR` file. The SSM automation document extracts information from this file when you choose to download SAP HANA media from Amazon S3.

The bucket can reside in a Shared Services account and can be shared with all AWS accounts that run SAP HANA workloads. The following table provides an example structure of the SAP HANA software media in Amazon S3.

Software	Version	Revision	Patch	Amazon S3 path
SAP HANA database software	2	SP04	48	S3://<Your SAP software bucket>/linuxx86/hanadb/2.0/SP04/48
SAP HANA database software	2	SP05	59	S3://<Your SAP software bucket>/linuxx86/hanadb/2.0/SP05/59
SAP HANA database software	2	SP05	59.5	S3://<Your SAP software bucket>/linuxx86/hanadb/2.0/SP05/59p5

SAP HANA database software	2	SP06	60	S3://<Your SAP software bucket>/linuxx86/hanadb/2.0/SP06/60
SAP HANA database software	2	SP06	64	S3://<Your SAP software bucket>/linuxx86/hanadb/2.0/SP06/64

Amazon S3 bucket policies

The Amazon S3 bucket containing the SAP HANA software media must be accessible to all Amazon EC2 instances running SAP HANA workloads in all of your AWS accounts. Use Amazon S3 bucket policies to grant limited access to Amazon S3 buckets and their contents only to specific authorized entities. For more information, see the following documents.

- Policies and Permissions in Amazon S3
- Security best practices for Amazon S3

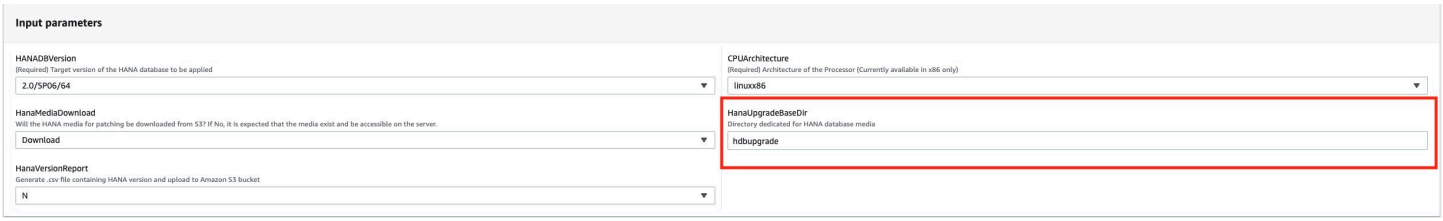
The following policy is an example Amazon S3 bucket policy that grants access to a specific role on a specific account to download all files from an Amazon S3 bucket.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AddPerm",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::{account_id}:role/service-role/{ec2_role}"
            },
            "Action": [
                "s3:GetObject",
                "s3:GetObjectVersion",
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::{bucket_name}/*",
                "arn:aws:s3:::{bucket_name}"
            ]
```

```
        }
    ]
}
```

Dedicated Linux file system

If the SAP HANA database software is stored in an Amazon S3 bucket, it is downloaded to the local Linux directory on Amazon EC2. It is recommended to have at least 30 GB of free space when downloading the SAP HANA software media files from Amazon S3 bucket to a local Linux directory. The directory path must be specified in the input parameters of the SSM automation document, as shown in the following image.



The files must be present in the specified directory on Amazon EC2 instance. The files must be unzipped, and stored in the following structure, based on the AWS SSM automation document code.

```
/{{HanaUpgradeBaseDir}}/x-sap-lnx-patch-hanadb/{{HANADBVersion}}/SAP_HANA_DATABASE/
```

The downloaded files are removed from the local directory once the SSM automation document has completed updating the SAP HANA database.

Amazon EC2

Your Amazon EC2 instance running SAP HANA workloads requires two tags to support the SSM automation document code. For more information, see Tag your Amazon EC2 resources.

The `DBSid:{SID}` and `HanaPatchGroup:{Usage}` tags are accessed by AWS Secrets Manager. Both of these tags are depicted in the arc the section called “Architecture”.

The `HanaPatchGroup` tag is used to filter different Amazon Resource Names (ARNs) that are retrieved from AWS Secrets Manager for the SAP HANA database user. The following is an example of `HanaPatchGroup` tag values.

```
DBSid = HDB
HanaPatchGroup = DEV
```

```
HanaPatchGroup = QAS
HanaPatchGroup = PRD
HanaPatchGroup = SBX
```

You can customize the tags based on your strategy for user and password management of the database user that is going to perform the SAP HANA update process.

AWS Identity and Access Management

AWS Systems Manager must be able to manage Amazon EC2 instance running SAP HANA workload. For more information, see Create an IAM instance profile for Systems Manager.

If your SAP HANA database instance is provisioned via AWS Launch Wizard for SAP, this permission is included in the deployment. For more information, see AWS Launch Wizard for SAP.

AWS Secrets Manager

AWS Secrets Manager is used to store the parameters of the SAP HANA database that are required to run the SSM automation document. AWS Secrets Manager enables sharing of secrets across multiple accounts. With this flexibility, you can manage the parameters in one location, and outside of the code.

Sharing the secrets across different accounts requires additional permissions. For more information, see How do I share AWS Secrets Manager secrets between AWS accounts?

The following table shows the example secrets created in the Shared Services account to run the sample code.

Secret name	Secret key	Secret value
zsap/hana/upgrade/user	User	<HANA Upgrade User ID>
zsap/hana/upgrade/password/DEV	Password	<HANA DEV Upgrade User Password>
zsap/hana/upgrade/password/QAS	Password	<HANA QAS Upgrade User Password>
zsap/hana/upgrade/password/PRD	Password	<HANA PRD Upgrade User Password>

zsap/hana/upgrade/password/SBX	Password	<HANA SBX Upgrade User Password>
zsap/hana/upgrade/bucket	Amazon S3 bucket	<Amazon S3 bucket for SAP HANA software>
zsap/sap/bucket/version_repo	Amazon S3 bucket	<Amazon S3 bucket for SAP HANA version repository>

Note

The sample code has references to the Amazon Resource Names of the secrets. This is required as the secrets are stored in a different account. The AWS account that contains the Amazon EC2 instance running the SAP HANA workload is different.

Policies for AWS Secrets Manager

The secrets created in AWS Secrets Manager must be set up to be accessible to target AWS accounts. For more information, see Resource-based policies.

The following is an example policy that is assigned to a Secret, granting access from a different AWS account.

```
{
  "Version" : "2012-10-17",
  "Statement" : [ {
    "Effect" : "Allow",
    "Principal" : {
      "AWS" : "arn:aws:iam::{sap_workloads_account_id}:role/service-role/{ec2_role}"
    },
    "Action" : "secretsmanager:GetSecretValue",
    "Resource" : "arn:aws:secretsmanager:{region}:{automation_account_id}:
    {secret_ARN}"
  } ]
}
```

> **Note**
>
> **SAP HANA database user ID**
>
> A valid user in the SAP HANA database SYSTEMDB with the required authorization to make the SAP HANA update is required.

In the sample code, the user and password are stored in AWS Secrets Manager as a secret. Follow the principle of granting least privilege, and use a user with the required authorizations. For more details, see Create a Lesser-Privileged Database User for Update.

AWS Key Management Service

The sample code uses AWS Secrets Manager to share secrets across different AWS accounts. As AWS Secrets Manager encrypts the contents of the parameters, a KMS key is used for encryption and decryption operations. The KMS key must be accessible to all of your AWS accounts. For more information, see Creating keys.

Prepare to run the SSM automation document

Before running the SSM automation document, you must ensure that a valid backup of the SAP HANA database exists, and that the applications connecting to the SAP HANA database are properly stopped. For more details, see Administration.

For SAP HANA databases managed by an operation system or a third-party cluster software, the cluster must be placed in maintenance mode before initiating automated patching. The SSM automation document must run on the secondary node first.

For more details on SAP HANA clustered environments, see SAP HANA on AWS: High Availability Configuration Guide for SLES and RHEL. For more details on updating SAP HANA databases with SAP HANA System Replication enabled, see Update SAP HANA Systems Running in a System Replication Setup.

Concurrency enables you to define how many SAP HANA databases should be updated in parallel. For more information, see Control automations at scale.

Troubleshoot

Follow these steps to see the status of each SSM automation.

1. Open console.aws.amazon.com.
2. Go to AWS Systems Manager.
3. On the left navigation pane, select **Automation**.
4. Select **Configure preferences** > **Executions**.
5. You can see the status of your SSM automations in the **Automation executions** section.

AWS Management Console enables you to drill into each execution, review the steps executed, and the result for each step. You can understand the failures that occur *before* SSM automation. For troubleshooting *after* the SSM automation has been initiated, review the logs. You can find the SSM logs on Amazon EC2 at the following path.

```
/var/lib/amazon/ssm/{instance-id}/document/orchestration/
{automation_step_execution_id}/awsrunShellScript/0.awsrunShellScript
```

You can send the output of each SSM automation to Amazon CloudWatch Logs. For more information, see Configuring Amazon CloudWatch Logs for Run Command.

SAP HANA version reporting

You can use Amazon QuickSight to create server less BI dashboards that can serve as a repository for your SAP HANA software versions. With Amazon QuickSight, you can review all of your SAP HANA database versions xin all of your AWS accounts. For more information, see Maintain an SAP landscape inventory with AWS Systems Manager and Amazon Athena.

The `HDB_Report_Version` step in the sample code gathers SAP HANA version information, and uploads that data into an Amazon S3 bucket. (In the sample code, the Amazon S3 bucket has a `/HANA` folder that contains the SAP HANA version information.) You can use the data in this bucket as source dataset to feed Amazon QuickSight dashboards. For more information, see Creating a dataset using Amazon S3 files. You can ensure accuracy of the data by scheduling automatic refreshes. For more information, see Refreshing SPICE data.

You must set up IAM permissions for the Amazon S3 bucket. The following is a sample Amazon S3 bucket policy for storing SAP HANA version information.

```
{
    "Version": "2012-10-17",
```

```
    "Statement": [
        {
            "Sid": "AddPerm",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::{account_id}:role/service-role/{ec2_role}"
            },
            "Action": "s3:PutObject",
            "Resource": [
                "arn:aws:s3:::{bucket_name}/*",
                "arn:aws:s3:::{bucket_name}"
            ]
        }
    ]
}
```

Storage Configuration for SAP HANA

SAP HANA stores and processes all or most of its data in memory, and provides protection against data loss by saving the data in persistent storage locations. To achieve optimal performance, the storage solution used for SAP HANA data and log volumes should meet SAP's storage KPI. AWS has worked with SAP to certify both Amazon EBS General Purpose SSD (gp2 and gp3) and Provisioned IOPS SSD (io1, io2, and io2 Block Express) storage solutions for SAP HANA workloads.

gp2 and gp3 volumes balance price and performance for a variety of workloads, while io1, io2, and io2 Block Express volumes provide the highest performance for mission-critical applications. From these options, you can choose the best storage solution that meets your performance and cost requirements. We recommend the io2 or io2 Block Express configuration for mission-critical SAP HANA production workloads.

For multi-node deployments, storage volumes for SAP HANA data and logs are provisioned in the master and worker nodes.

In the following configurations, we intentionally kept the same storage configuration for SAP HANA data and log volumes for all R3, certain R4 and R5, and smaller X1e/X2iedn instance types so you can scale up from smaller instances to larger instances without having to reconfigure your storage.

> **Note**
>
> The X1, X1e, X2idn, and X2iedn instance types include instance storage but should not be used to persist any SAP HANA related files.

gp2 and gp3 for HANA

gp2 for HANA data

Instance type	Memory (GiB)	vCPUs / logical processors*	General Purpose SSD (gp2) storage with LVM	Total maximum throughput (MiB/s)	Total baseline IOPS	Total burst IOPS
Certified for production use						
u-24tb1.112xlarge	24,576	448	6 x 4,800 GiB	1,500	86,400	N/A
u-24tb1.metal	24,576	448	6 x 4,800 GiB	1,500	86,400	N/A
u-18tb1.112xlarge	18,432	448	6 x 3,600 GiB	1,500	64,800	N/A
u-18tb1.metal	18,432	448	6 x 3,600 GiB	1,500	64,800	N/A
u-12tb1.112xlarge	12,288	448	6 x 2,400 GiB	1,500	43,200	N/A
u-12tb1.metal	12,288	448	6 x 2,400 GiB	1,500	43,200	N/A
u-9tb1.112xlarge	9,216	448	6 x 1,800 GiB	1,500	32,400	N/A

Instance type	Memory (GiB)	vCPUs / logical processors*	General Purpose SSD (gp2) storage with LVM	Total maximum throughput (MiB/s)	Total baseline IOPS	Total burst IOPS
u-9tb1.metal	9,216	448	6 x 1,800 GiB	1,500	32,400	N/A
u-6tb1.112xlarge	6,144	448	6 x 1,200 GiB	1,500	21,600	N/A
u-6tb1.56xlarge	6,144	224	6 x 1,200 GiB	1,500	21,600	N/A
u-6tb1.metal	6,144	448	6 x 1,200 GiB	1,500	21,600	N/A
u-3tb1.56xlarge	3,072	224	3 x 1,200 GiB	750	10,800	N/A
x2iedn.32xlarge	4,096	128	3 x 1,600 GiB	750	14,400	N/A
x2iedn.24xlarge	3,072	96	3 x 1,200 GiB	750	10,800	N/A
x2idn.32xlarge	2,048	128	3 x 800 GiB	750	7,200	9,000
x2idn.24xlarge	1,536	96	3 x 600 GiB	750	5,400	9,000
x2idn.16xlarge	1,024	64	3 x 400 GiB	750	3,600	9,000
x1e.32xlarge	3,904	128	3 x 1,600 GiB	750	14,400	N/A

Instance type	Memory (GiB)	vCPUs / logical processors*	General Purpose SSD (gp2) storage with LVM	Total maximum throughput (MiB/s)	Total baseline IOPS	Total burst IOPS
x1.32xlarge	1,952	128	3 x 800 GiB	750	7,200	9,000
x1.16xlarge	976	64	3 x 400 GiB	750	3,600	9,000
r7i.48xlarge	1,536	192	3 x 600 GiB	750	5,400	9,000
r7i.24xlarge	768	96	3 x 400 GiB	750	3,600	9,000
r7i.16xlarge	512	64	3 x 225 GiB	750	2,025	9,000
r7i.12xlarge	384	48	3 x 225 GiB	750	2,025	9,000
r7i.8xlarge	256	32	3 x 225 GiB	750	2,025	9,000
r6i.32xlarge	1,024	128	3 x 400 GiB	750	3,600	9,000
r6i.24xlarge	768	96	3 x 400 GiB	750	3,600	9,000
r6i.16xlarge	512	64	3 x 225 GiB	750	2,025	9,000
r6i.12xlarge	384	48	3 x 225 GiB	750	2,025	9,000

Instance type	Memory (GiB)	vCPUs / logical processors*	General Purpose SSD (gp2) storage with LVM	Total maximum throughput (MiB/s)	Total baseline IOPS	Total burst IOPS
r6i.8xlarge	256	32	3 x 225 GiB	750	2,025	9,000
r5.24xlarge	768	96	3 x 400 GiB	750	3,600	9,000
r5.16xlarge	512	64	3 x 225 GiB	750	2,025	9,000
r5.12xlarge	384	48	3 x 225 GiB	750	2,025	9,000
r5.8xlarge	256	32	3 x 225 GiB	750	2,025	9,000
r5.metal	768	96	3 x 400 GiB	750	3,600	9,000
r5b.24xlarge	768	96	3 x 400 GiB	750	3,600	9,000
r5b.16xlarge	512	64	3 x 225 GiB	750	2,025	9,000
r5b.12xlarge	384	48	3 x 225 GiB	750	2,025	9,000
r5b.8xlarge	256	32	3 x 225 GiB	750	2,025	9,000
r5b.metal	768	96	3 x 400 GiB	750	3,600	9,000

Instance type	Memory (GiB)	vCPUs / logical processors*	General Purpose SSD (gp2) storage with LVM	Total maximum throughput (MiB/s)	Total baseline IOPS	Total burst IOPS
r4.16xlarge	488	64	3 x 225 GiB	750	2,025	9,000
r4.8xlarge	244	32	3 x 225 GiB	750	2,025	9,000
r3.8xlarge	244	32	3 x 225 GiB	750	2,025	9,000
Supported for nonproduction use only						
x2iedn.4xlarge	512	16	3 x 225 GiB	750	2,025	9,000
x2iedn.2xlarge	256	8	3 x 225 GiB	750	2,025	9,000
x2iedn.xlarge	128	4	3 x 225 GiB	750	2,025	9,000
x1e.4xlarge	488	16	3 x 225 GiB	750**	2,025	9,000
x1e.2xlarge	244	8	3 x 225 GiB	750**	2,025	9,000
x1e.xlarge	122	4	3 x 225 GiB	750**	2,025	9,000
r7i.4xlarge	128	16	3 x 225 GiB	750	2,025	9,000

Instance type	Memory (GiB)	vCPUs / logical processors*	General Purpose SSD (gp2) storage with LVM	Total maximum throughput (MiB/s)	Total baseline IOPS	Total burst IOPS
r7i.2xlarge	64	8	3 x 225 GiB	750	2,025	9,000
r6i.4xlarge	128	16	3 x 225 GiB	750	2,025	9,000
r6i.2xlarge	64	8	3 x 225 GiB	750	2,025	9,000
r5.4xlarge	128	16	3 x 225 GiB	750**	2,025	9,000
r5.2xlarge	64	8	3 x 225 GiB	750**	2,025	9,000
r5b.4xlarge	128	16	3 x 225 GiB	750	2,025	9,000
r5b.2xlarge	64	8	3 x 225 GiB	750	2,025	9,000
r4.4xlarge	122	16	3 x 225 GiB	750**	2,025	9,000
r4.2xlarge	61	8	3 x 225 GiB	750**	2,025	9,000
r3.4xlarge	122	16	3 x 225 GiB	750**	2,025	9,000
r3.2xlarge	61	8	3 x 225 GiB	750**	2,025	9,000

* Each logical processor offered by Amazon EC2 High Memory Instances is a hyperthread on a physical CPU core.

** This value represents the maximum throughput that could be achieved when striping multiple EBS volumes. Actual throughput depends on the instance type. Every instance type has its own Amazon EBS throughput maximum. For details, see Amazon EBS-Optimized Instances in the AWS documentation.

***gp3 based configurations are only supported in production for Nitro based instances, not for Xen based instances as SAP HANA HCMT storage tests may not meet the minimum required KPI for log writes.

gp2 for HANA logs

Instance type	Memory (GiB)	vCPUs / logical processors*	General Purpose SSD (gp2) storage with LVM	Total maximum throughput (MiB/s)	Total baseline IOPS	Total burst IOPS
Certified for production use						
u-24tb1.112xlarge	24,576	448	2 x 300 GiB	500	1,800	6,000
u-24tb1.metal	24,576	448	2 x 300 GiB	500	1,800	6,000
u-18tb1.112xlarge	18,432	448	2 x 300 GiB	500	1,800	6,000
u-18tb1.metal	18,432	448	2 x 300 GiB	500	1,800	6,000
u-12tb1.112xlarge	12,288	448	2 x 300 GiB	500	1,800	6,000
u-12tb1.metal	12,288	448	2 x 300 GiB	500	1,800	6,000

Instance type	Memory (GiB)	vCPUs / logical processors*	General Purpose SSD (gp2) storage with LVM	Total maximum throughput (MiB/s)	Total baseline IOPS	Total burst IOPS
u-9tb1.112xlarge	9,216	448	2 x 300 GiB	500	1,800	6,000
u-9tb1.metal	9,216	448	2 x 300 GiB	500	1,800	6,000
u-6tb1.112xlarge	6,144	448	2 x 300 GiB	500	1,800	6,000
u-6tb1.56xlarge	6,144	224	2 x 300 GiB	500	1,800	6,000
u-6tb1.metal	6,144	448	2 x 300 GiB	500	1,800	6,000
u-3tb1.56xlarge	3,072	224	2 x 300 GiB	500	1,800	6,000
x2iedn.32xlarge	4,096	128	2 x 300 GiB	500	1,800	6,000
x2iedn.24xlarge	3,072	96	2 x 300 GiB	500	1,800	6,000
x2idn.32xlarge	2,048	128	2 x 300 GiB	500	1,800	6,000
x2idn.24xlarge	1,536	96	2 x 300 GiB	500	1,800	6,000
x2idn.16xlarge	1,024	64	2 x 300 GiB	500	1,800	6,000

Instance type	Memory (GiB)	vCPUs / logical processors*	General Purpose SSD (gp2) storage with LVM	Total maximum throughput (MiB/s)	Total baseline IOPS	Total burst IOPS
x1e.32xlarge	3,904	128	2 x 300 GiB	500	1,800	6,000
x1.32xlarge	1,952	128	2 x 300 GiB	500	1,800	6,000
x1.16xlarge	976	64	2 x 300 GiB	500	1,800	6,000
r7i.48xlarge	1,536	192	2 x 300 GiB	500	1,800	6,000
r7i.24xlarge	768	96	2 x 300 GiB	500	1,800	6,000
r7i.16xlarge	512	64	2 x 300 GiB	500	1,800	6,000
r7i.12xlarge	384	48	2 x 175 GiB	500**	1,050	6,000
r7i.8xlarge	256	32	2 x 175 GiB	500**	1,050	6,000
r6i.32xlarge	1,024	128	2 x 300 GiB	500	1,800	6,000
r6i.24xlarge	768	96	2 x 300 GiB	500	1,800	6,000
r6i.16xlarge	512	64	2 x 300 GiB	500	1,800	6,000

Instance type	Memory (GiB)	vCPUs / logical processors*	General Purpose SSD (gp2) storage with LVM	Total maximum throughput (MiB/s)	Total baseline IOPS	Total burst IOPS
r6i.12xlarge	384	48	2 x 300 GiB	500	1,800	6,000
r6i.8xlarge	256	32	2 x 175 GiB	500	1,050	6,000
r5.24xlarge	768	96	2 x 300 GiB	500	1,800	6,000
r5.16xlarge	512	64	2 x 300 GiB	500	1,800	6,000
r5.12xlarge	384	48	2 x 300 GiB	500	1,800	6,000
r5.8xlarge	256	32	2 x 300 GiB	500	1,800	6,000
r5.metal	768	96	2 x 300 GiB	500	1,800	6,000
r5b.24xlarge	768	96	2 x 300 GiB	500	1,800	6,000
r5b.16xlarge	512	64	2 x 300 GiB	500	1,800	6,000
r5b.12xlarge	384	48	2 x 300 GiB	500	1,800	6,000
r5b.8xlarge	256	32	2 x 300 GiB	500	1,800	6,000

Instance type	Memory (GiB)	vCPUs / logical processors*	General Purpose SSD (gp2) storage with LVM	Total maximum throughput (MiB/s)	Total baseline IOPS	Total burst IOPS
r5b.metal	768	96	2 x 300 GiB	500	1,800	6,000
r4.16xlarge	488	64	2 x 300 GiB	500	1,800	6,000
r4.8xlarge	244	32	2 x 300 GiB	500	1,800	6,000
r3.8xlarge	244	32	2 x 300 GiB	500	1,800	6,000
Supported for nonproduction use only						
x2iedn.4xlarge	512	16	2 x 175 GiB	500**	1,050	6,000
x2iedn.2xlarge	256	8	2 x 175 GiB	500**	1,050	6,000
x2iedn.xlarge	128	4	2 x 175 GiB	500**	1,050	6,000
x1e.4xlarge	488	16	2 x 175 GiB	500**	1,050	6,000
x1e.2xlarge	244	8	2 x 175 GiB	500**	1,050	6,000
x1e.xlarge	122	4	2 x 175 GiB	500**	1,050	6,000

Instance type	Memory (GiB)	vCPUs / logical processors*	General Purpose SSD (gp2) storage with LVM	Total maximum throughput (MiB/s)	Total baseline IOPS	Total burst IOPS
r7i.4xlarge	128	16	2 x 175 GiB	500**	1,050	6,000
r7i.2xlarge	64	8	2 x 175 GiB	500**	1,050	6,000
r6i.4xlarge	128	16	2 x 175 GiB	500	1,050	6,000
r6i.2xlarge	64	8	2 x 175 GiB	500	1,050	6,000
r5.4xlarge	128	16	2 x 175 GiB	500**	1,050	6,000
r5.2xlarge	64	8	2 x 175 GiB	500**	1,050	6,000
r5b.4xlarge	128	16	2 x 175 GiB	500	1,050	6,000
r5b.2xlarge	64	8	2 x 175 GiB	500	1,050	6,000
r4.4xlarge	122	16	2 x 175 GiB	500**	1,050	6,000
r4.2xlarge	61	8	2 x 175 GiB	500**	1,050	6,000
r3.4xlarge	122	16	2 x 175 GiB	500**	1,050	6,000

Instance type	Memory (GiB)	vCPUs / logical processors*	General Purpose SSD (gp2) storage with LVM	Total maximum throughput (MiB/s)	Total baseline IOPS	Total burst IOPS
r3.2xlarge	61	8	2 x 175 GiB	500**	1,050	6,000

* Each logical processor offered by Amazon EC2 High Memory Instances is a hyperthread on a physical CPU core.

** This value represents the maximum throughput that could be achieved when striping multiple EBS volumes. Actual throughput depends on the instance type. Every instance type has its own Amazon EBS throughput maximum. For details, see Amazon EBS-Optimized Instances in the AWS documentation.

***gp3 based configurations are only supported in production for Nitro based instances, not for Xen based instances as SAP HANA HCMT storage tests may not meet the minimum required KPI for log writes.

gp3 for HANA data

Instance type	Memory (GiB)	vCPUs / logical processors*	General Purpose SSD (gp3) storage with LVM	Configured throughput per volume (MiB/s)	Configured IOPS per volume	Total throughput (MiB/s)	Total IOPS
Certified for production use							
u-24tb1.112xlarge	24,576	448	2 x 14,400 GiB	1,000	9,000	2,000	18,000

Instance type	Memory (GiB)	vCPUs / logical processors*	General Purpose SSD (gp3) storage with LVM	Configured throughput per volume (MiB/s)	Configured IOPS per volume	Total throughput (MiB/s)	Total IOPS
u-24tb1.metal	24,576	448	2 x 14,400 GiB	1,000	9,000	2,000	18,000
u-18tb1.112xlarge	18,432	448	2 x 10,800 GiB	1,000	9,000	2,000	18,000
u-18tb1.metal	18,432	448	2 x 10,800 GiB	1,000	9,000	2,000	18,000
u-12tb1.112xlarge	12,228	448	2 x 7,200 GiB	1,000	6,000	2,000	12,000
u-12tb1.metal	12,228	448	2 x 7,200 GiB	1,000	6,000	2,000	12,000
u-9tb1.112xlarge	9,216	448	2 x 5,400 GiB	1,000	6,000	2,000	12,000
u-9tb1.metal	9,216	448	2 x 5,400 GiB	1,000	6,000	2,000	12,000

Instance type	Memory (GiB)	vCPUs / logical processors*	General Purpose SSD (gp3) storage with LVM	Configured throughput per volume (MiB/s)	Configured IOPS per volume	Total throughput (MiB/s)	Total IOPS
u-6tb1.112xlarge	6,114	448	2 x 3,600 GiB	1,000	6,000	2,000	12,000
u-6tb1.56xlarge	6,114	224	2 x 3,600 GiB	1,000	6,000	2,000	12,000
u-6tb1.metal	6,114	448	2 x 3,600 GiB	1,000	6,000	2,000	12,000
u-3tb1.56xlarge	3,072	224	2 x 1,800 GiB	750	4,500	1,500	9,000
x2iedn.32xlarge	4,096	128	2 x 2,400 GiB	750	4,500	1,500	9,000
x2iedn.24xlarge	3,072	96	2 x 1,800 GiB	750	4,500	1,500	9,000
x2idn.32xlarge	2,048	128	2 x 1,200 GiB	750	4,500	1,500	9,000
x2idn.24xlarge	1,536	96	2 x 900 GiB	750	4,500	1,500	9,000

Instance type	Memory (GiB)	vCPUs / logical processors*	General Purpose SSD (gp3) storage with LVM	Configured throughput per volume (MiB/s)	Configured IOPS per volume	Total throughput (MiB/s)	Total IOPS
x2idn.16xlarge	1,024	64	2 x 600 GiB	500	3,750	1,000	7,500
x1e.32xlarge ***	3,904	128	2 x 2,400 GiB	750	4,500	1,500	9,000
x1.32xlarge**	1,952	128	2 x 1,200 GiB	750	4,500	1,500	9,000
x1.16xlarge**	976	64	1 x 1,200 GiB	500	7,500	500	7,500
r7i.48xlarge	1,536	192	2 x 900 GiB	750	4,500	1,500	9,000
r7i.24xlarge	768	96	1 x 920 GiB	500	7,500	500	7,500
r7i.16xlarge	512	64	1 x 615 GiB	500	7,500	500	7,500
r7i.12xlarge	384	48	1 x 460 GiB	500	7,500	500	7,500
r7i.8xlarge	256	32	1 x 320 GiB	500	7,500	500	7,500

Instance type	Memory (GiB)	vCPUs / logical processors*	General Purpose SSD (gp3) storage with LVM	Configured throughput per volume (MiB/s)	Configured IOPS per volume	Total throughput (MiB/s)	Total IOPS
r6i.32xlarge	1,024	128	1 x 1,200 GiB	500	7,500	500	7,500
r6i.24xlarge	768	96	1 x 920 GiB	500	7,500	500	7,500
r6i.16xlarge	512	64	1 x 615 GiB	500	7,500	500	7,500
r6i.12xlarge	384	48	1 x 460 GiB	500	7,500	500	7,500
r6i.8xlarge	256	32	1 x 320 GiB	500	7,500	500	7,500
r5.24xlarge	768	96	1 x 920 GiB	500	7,500	500	7,500
r5.16xlarge	512	64	1 x 615 GiB	500	7,500	500	7,500
r5.12xlarge	384	48	1 x 460 GiB	500	7,500	500	7,500
r5.8xlarge	256	32	1 x 320 GiB	500	7,500	500	7,500
r5.metal	768	96	1 x 920 GiB	500	7,500	500	7,500

Instance type	Memory (GiB)	vCPUs / logical processors*	General Purpose SSD (gp3) storage with LVM	Configured throughput per volume (MiB/s)	Configured IOPS per volume	Total throughput (MiB/s)	Total IOPS
r5b.24xlarge	768	96	1 x 920 GiB	500	7,500	500	7,500
r5b.16xlarge	512	64	1 x 615 GiB	500	7,500	500	7,500
r5b.12xlarge	384	48	1 x 460 GiB	500	7,500	500	7,500
r5b.8xlarge	256	32	1 x 320 GiB	500	7,500	500	7,500
r5b.metal	768	96	1 x 920 GiB	500	7,500	500	7,500
r4.16xlarge* **	488	64	1 x 585 GiB	500	7,500	500	7,500
r4.8xlarge* **	244	32	1 x 300 GiB	500	7,500	500	7,500
r3.8xlarge* **	244	32	1 x 300 GiB	500	7,500	500	7,500
Supported for nonproduction use only							
x2iedn.4xlarge	512	16	1 x 585 GiB	125	3,000	125	3,000
x2iedn.2xlarge	256	8	1 x 295 GiB	125	3,000	125	3,000

Instance type	Memory (GiB)	vCPUs / logical processors*	General Purpose SSD (gp3) storage with LVM	Configured throughput per volume (MiB/s)	Configured IOPS per volume	Total throughput (MiB/s)	Total IOPS
x2iedn.xlarge	128	4	1 x 150 GiB	125	3,000	125	3,000
x1e.4xlarge	488	16	1 x 585 GiB	125	3,000	125	3,000
x1e.2xlarge	244	8	1 x 295 GiB	125	3,000	125	3,000
x1e.xlarge	122	4	1 x 150 GiB	125	3,000	125	3,000
r7i.4xlarge	128	16	1 x 150 GiB	125	3,000	125	3,000
r7i.2xlarge	64	8	1 x 80 GiB	125	3,000	125	3,000
r6i.4xlarge	128	16	1 x 150 GiB	125	3,000	125	3,000
r6i.2xlarge	64	8	1 x 80 GiB	125	3,000	125	3,000
r5.4xlarge	128	16	1 x 150 GiB	125	3,000	125	3,000
r5.2xlarge	64	8	1 x 80 GiB	125	3,000	125	3,000

Instance type	Memory (GiB)	vCPUs / logical processors*	General Purpose SSD (gp3) storage with LVM	Configured throughput per volume (MiB/s)	Configured IOPS per volume	Total throughput (MiB/s)	Total IOPS
r5b.4xlarge	128	16	1 x 150 GiB	125	3,000	125	3,000
r5b.2xlarge	64	8	1 x 80 GiB	125	3,000	125	3,000
r4.4xlarge	122	16	1 x 150 GiB	125	3,000	125	3,000
r4.2xlarge	61	8	1 x 80 GiB	125	3,000	125	3,000
r3.4xlarge	122	16	1 x 150 GiB	125	3,000	125	3,000
r3.2xlarge	61	8	1 x 80 GiB	125	3,000	125	3,000

* Each logical processor offered by Amazon EC2 High Memory Instances is a hyperthread on a physical CPU core.

** This value represents the maximum throughput that could be achieved when striping multiple EBS volumes. Actual throughput depends on the instance type. Every instance type has its own Amazon EBS throughput maximum. For details, see Amazon EBS-Optimized Instances in the AWS documentation.

***gp3 based configurations are only supported in production for Nitro based instances, not for Xen based instances as SAP HANA HCMT storage tests may not meet the minimum required KPI for log writes.

gp3 for HANA logs

Instance type	Memory (GiB)	vCPUs / logical processors*	General Purpose SSD (gp3) storage with LVM	Configured throughput per volume (MiB/s)	Configured IOPS per volume	Total throughput (MiB/s)	Total IOPS
Certified for production use							
u-24tb1.112xlarge	24,576	448	1 x 512 GiB	500	3,000	500	3,000
u-24tb1.metal	24,576	448	1 x 512 GiB	500	3,000	500	3,000
u-18tb1.112xlarge	18,432	448	1 x 512 GiB	500	3,000	500	3,000
u-18tb1.metal	18,432	448	1 x 512 GiB	500	3,000	500	3,000
u-12tb1.112xlarge	12,228	448	1 x 512 GiB	500	3,000	500	3,000
u-12tb1.metal	12,228	448	1 x 512 GiB	500	3000	500	3,000
u-9tb1.112xlarge	9,216	448	1 x 512 GiB	300	3,000	300	3,000
u-9tb1.metal	9,216	448	1 x 512 GiB	300	3,000	300	3,000
u-6tb1.112xlarge	6,114	448	1 x 512 GiB	300	3,000	300	3,000

Instance type	Memory (GiB)	vCPUs / logical processors*	General Purpose SSD (gp3) storage with LVM	Configured throughput per volume (MiB/s)	Configured IOPS per volume	Total throughput (MiB/s)	Total IOPS
u-6tb1.56xlarge	6,114	224	1 x 512 GiB	300	3,000	300	3,000
u-6tb1.metal	6,114	448	1 x 512 GiB	300	3,000	300	3,000
u-3tb1.56xlarge	3,072	224	1 x 512 GiB	300	3,000	300	3,000
x2iedn.32xlarge	4,096	128	1 x 512 GiB	300	3,000	300	3,000
x2iedn.24xlarge	3,072	96	1 x 512 GiB	300	3,000	300	3,000
x2idn.32xlarge	2,048	128	1 x 512 GiB	300	3,000	300	3,000
x2idn.24xlarge	1,536	96	1 x 512 GiB	300	3,000	300	3,000
x2idn.16xlarge	1,024	64	1 x 512 GiB	300	3,000	300	3,000
x1e.32xlarge ***	3,904	128	1 x 512 GiB	300	3,000	300	3,000
x1.32xlarge***	1,952	128	1 x 512 GiB	300	3,000	300	3,000

Instance type	Memory (GiB)	vCPUs / logical processors*	General Purpose SSD (gp3) storage with LVM	Configured throughput per volume (MiB/s)	Configured IOPS per volume	Total throughput (MiB/s)	Total IOPS
x1.16xlarge* **	976	64	1 x 512 GiB	300	3,000	300	3,000
r7i.48xlarge	1,536	192	1 x 512 GiB	300	3,000	300	3,000
r7i.24xlarge	768	96	1 x 512 GiB	300	3,000	300	3,000
r7i.16xlarge	512	64	1 x 256 GiB	300	3,000	300	3,000
r7i.12xlarge	384	48	1 x 192 GiB	300	3,000	300	3,000
r7i.8xlarge	256	32	1 x 128 GiB	300	3,000	300	3,000
r6i.32xlarge	1,024	128	1 x 512 GiB	300	3,000	300	3,000
r6i.24xlarge	768	96	1 x 512 GiB	300	3,000	300	3,000
r6i.16xlarge	512	64	1 x 256 GiB	300	3,000	300	3,000
r6i.12xlarge	384	48	1 x 192 GiB	300	3,000	300	3,000

Instance type	Memory (GiB)	vCPUs / logical processors*	General Purpose SSD (gp3) storage with LVM	Configured throughput per volume (MiB/s)	Configured IOPS per volume	Total throughput (MiB/s)	Total IOPS
r6i.8xlarge	256	32	1 x 128 GiB	300	3,000	300	3,000
r5.24xlarge	768	96	1 x 512 GiB	300	3,000	300	3,000
r5.16xlarge	512	64	1 x 256 GiB	300	3,000	300	3,000
r5.12xlarge	384	48	1 x 192 GiB	300	3,000	300	3,000
r5.8xlarge	256	32	1 x 128 GiB	300	3,000	300	3,000
r5.metal	768	96	1 x 512 GiB	300	3,000	300	3,000
r5b.24xlarge	768	96	1 x 512 GiB	300	3,000	300	3,000
r5b.16xlarge	512	64	1 x 256 GiB	300	3,000	300	3,000
r5b.12xlarge	384	48	1 x 192 GiB	300	3,000	300	3,000
r5b.8xlarge	256	32	1 x 128 GiB	300	3,000	300	3,000

Instance type	Memory (GiB)	vCPUs / logical processors*	General Purpose SSD (gp3) storage with LVM	Configured throughput per volume (MiB/s)	Configured IOPS per volume	Total throughput (MiB/s)	Total IOPS
r5b.metal	768	96	1 x 512 GiB	300	3,000	300	3,000
r4.16xlarge ***	488	64	1 x 256 GiB	300	3,000	300	3,000
r4.8xlarge ***	244	32	1 x 128 GiB	300	3,000	300	3,000
r3.8xlarge ***	244	32	1 x 128 GiB	300	3,000	300	3,000
Supported for nonproduction use only							
x2iedn.4xlarge	512	16	1 x 245 GiB	125	3,000	125	3,000
x2iedn.2xlarge	256	8	1 x 125 GiB	125	3,000	125	3,000
x2iedn.xlarge	128	4	1 x 64 GiB	125	3,000	125	3,000
x1e.4xlarge	488	16	1 x 245 GiB	125	3,000	125	3,000
x1e.2xlarge	244	8	1 x 125 GiB	125	3,000	125	3,000
x1e.xlarge	122	4	1 x 64 GiB	125	3,000	125	3,000

Instance type	Memory (GiB)	vCPUs / logical processors*	General Purpose SSD (gp3) storage with LVM	Configured throughput per volume (MiB/s)	Configured IOPS per volume	Total throughput (MiB/s)	Total IOPS
r7i.4xlarge	128	16	1 x 64 GiB	125	3,000	125	3,000
r7i.2xlarge	64	8	1 x 32 GiB	125	3,000	125	3,000
r6i.4xlarge	128	16	1 x 64 GiB	125	3,000	125	3,000
r6i.2xlarge	64	8	1 x 32 GiB	125	3,000	125	3,000
r5.4xlarge	128	16	1 x 64 GiB	125	3,000	125	3,000
r5.2xlarge	64	8	1 x 32 GiB	125	3,000	125	3,000
r5b.4xlarge	128	16	1 x 64 GiB	125	3,000	125	3,000
r5b.2xlarge	64	8	1 x 32 GiB	125	3,000	125	3,000
r4.4xlarge	122	16	1 x 64 GiB	125	3,000	125	3,000
r4.2xlarge	61	8	1 x 32 GiB	125	3,000	125	3,000

Instance type	Memory (GiB)	vCPUs / logical processors*	General Purpose SSD (gp3) storage with LVM	Configured throughput per volume (MiB/s)	Configured IOPS per volume	Total throughput (MiB/s)	Total IOPS
r3.4xlarge	122	16	1 x 64 GiB	125	3,000	125	3,000
r3.2xlarge	61	8	1 x 32 GiB	125	3,000	125	3,000

* Each logical processor offered by Amazon EC2 High Memory Instances is a hyperthread on a physical CPU core.

** This value represents the maximum throughput that could be achieved when striping multiple EBS volumes. Actual throughput depends on the instance type. Every instance type has its own Amazon EBS throughput maximum. For details, see Amazon EBS-Optimized Instances in the AWS documentation.

***gp3 based configurations are only supported in production for Nitro based instances, not for Xen based instances as SAP HANA HCMT storage tests may not meet the minimum required KPI for log writes.

General Purpose SSD (gp2) volumes created or modified after 12/03/2018 have a throughput maximum between 128 MiB/s and 250 MiB/s depending on volume size. Volumes greater than 170 GiB and below 334 GiB deliver a maximum throughput of 250 MiB/s if burst credits are available. Volumes with 334 GiB and above deliver 250 MiB/s, irrespective of burst credits. For details, see Amazon EBS Volume Types in the AWS documentation.

General Purpose SSD gp3 volumes deliver a consistent baseline of 3,000 IOPS and 125 MiB/s. You can also purchase additional IOPS (up to 16,000) and throughput (up to 1,000 MiB/s). While we recommend you to use the configurations shown in this guide, gp3 volumes provide flexibility to customize SAP HANA's storage configuration (IOPS and throughput) according to your needs and usage.

The **minimum** gp3 configuration required to meet SAP HANA KPIs are the following:

Storage Area	IOPS	Throughput
SAP HANA Data	7,000	425 MiB/s
SAP HANA Logs	3,000	275 MiB/s

io1, io2, and io2 Block Express for HANA

io1 for HANA data

Instance type	Memory (GiB)	vCPUs / logical processors*	Provisioned IOPS SSD (io1/io2) storage with LVM	Total maximum throughput (MiB/s)	Provisioned IOPS per volume	Total provisioned IOPS
Certified for production use						
u-24tb1.112xlarge	24,576	448	6 x 4,800 GiB	3,000	3,000	18,000
u-24tb1.metal	24,576	448	6 x 4,800 GiB	3,000	3,000	18,000
u-18tb1.112xlarge	18,432	448	6 x 3,600 GiB	3,000	3,000	18,000
u-18tb1.metal	18,432	448	6 x 3,600 GiB	3,000	3,000	18,000
u-12tb1.112xlarge	12,288	448	6 x 2,400 GiB	3,000	2,000	12,000
u-12tb1.metal	12,288	448	6 x 2,400 GiB	3,000	2,000	12,000

Instance type	Memory (GiB)	vCPUs / logical processors*	Provisioned IOPS SSD (io1/io2) storage with LVM	Total maximum throughput (MiB/s)	Provisioned IOPS per volume	Total provisioned IOPS
u-9tb1.112xlarge	9,216	448	6 x 1,800 GiB	3,000	2,000	12,000
u-9tb1.metal	9,216	448	6 x 1,800 GiB	3,000	2,000	12,000
u-6tb1.112xlarge	6,144	448	6 x 1,200 GiB	3,000	2,000	12,000
u-6tb1.56xlarge	6,144	224	6 x 1,200 GiB	3,000	2,000	12,000
u-6tb1.metal	6,144	448	6 x 1,200 GiB	3,000	2,000	12,000
u-3tb1.56xlarge	3,072	224	3 x 1,200 GiB	1,500	3,000	9,000
x2iedn.32xlarge	4,096	128	2 x 2,400 GiB	1,000	4,500	9,000
x2iedn.24xlarge	3,072	96	2 x 1,800 GiB	1,000	4,500	9,000
x2idn.32xlarge	2,048	128	2 x 1,200 GiB	1,000	4,500	9,000
x2idn.24xlarge	1,536	96	2 x 900 GiB	1,000	4,500	9,000
x2idn.16xlarge	1,024	64	2 x 600 GiB	1,000	3,750	7,500

Instance type	Memory (GiB)	vCPUs / logical processors*	Provisioned IOPS SSD (io1/io2) storage with LVM	Total maximum throughput (MiB/s)	Provisioned IOPS per volume	Total provisioned IOPS
x1e.32xlarge	3,904	128	3 x 1,600 GiB	1,500	3,000	9,000
x1.32xlarge	1,952	128	3 x 800 GiB	1,500	3,000	9,000
x1.16xlarge	976	64	1 x 1,200 GiB	500	7,500	7,500
r7i.48xlarge	1,536	192	1 x 1,800 GiB	500	7,500	7,500
r7i.24xlarge	768	96	1 x 900 GiB	500	7,500	7,500
r7i.16xlarge	512	64	1 x 600 GiB	500	7,500	7,500
r7i.12xlarge	384	48	1 x 600 GiB	500	7,500	7,500
r7i.8xlarge	256	32	1 x 300 GiB	500	7,500	7,500
r6i.32xlarge	1,024	128	1 x 1,200 GiB	500	7,500	7,500
r6i.24xlarge	768	96	1 x 1,200 GiB	500	7,500	7,500
r6i.16xlarge	512	64	1 x 600 GiB	500	7,500	7,500

Instance type	Memory (GiB)	vCPUs / logical processors*	Provisioned IOPS SSD (io1/io2) storage with LVM	Total maximum throughput (MiB/s)	Provisioned IOPS per volume	Total provisioned IOPS
r6i.12xlarge	384	48	1 x 600 GiB	500	7,500	7,500
r6i.8xlarge	256	32	1 x 300 GiB	500	7,500	7,500
r5.24xlarge	768	96	1 x 1,200 GiB	500	7,500	7,500
r5.16xlarge	512	64	1 x 600 GiB	500	7,500	7,500
r5.12xlarge	384	48	1 x 600 GiB	500	7,500	7,500
r5.8xlarge	256	32	1 x 300 GiB	500	7,500	7,500
r5.metal	768	96	1 x 1,200 GiB	500	7,500	7,500
r5b.24xlarge	768	96	1 x 1,200 GiB	500	7,500	7,500
r5b.16xlarge	512	64	1 x 600 GiB	500	7,500	7,500
r5b.12xlarge	384	48	1 x 600 GiB	500	7,500	7,500
r5b.8xlarge	256	32	1 x 300 GiB	500	7,500	7,500

Instance type	Memory (GiB)	vCPUs / logical processors*	Provisioned IOPS SSD (io1/io2) storage with LVM	Total maximum throughput (MiB/s)	Provisioned IOPS per volume	Total provisioned IOPS
r5b.metal	768	96	1 x 1,200 GiB	500	7,500	7,500
r4.16xlarge	488	64	1 x 600 GiB	500	7,500	7,500
r4.8xlarge	244	32	1 x 300 GiB	500	7,500	7,500
r3.8xlarge	244	32	1 x 300 GiB	500	7,500	7,500
Supported for nonproduction use only						
x2iedn.4xlarge	512	16	1 x 600 GiB	500	2,000	2,000
x2iedn.2xlarge	256	8	1 x 300 GiB	500	2,000	2,000
x2iedn.xlarge	128	4	1 x 300 GiB	500	2,000	2,000
x1e.4xlarge	488	16	1 x 600 GiB	500**	2,000	2,000
x1e.2xlarge	244	8	1 x 300 GiB	500**	2,000	2,000
x1e.xlarge	122	4	1 x 300 GiB	500**	2,000	2,000

Instance type	Memory (GiB)	vCPUs / logical processors*	Provisioned IOPS SSD (io1/io2) storage with LVM	Total maximum throughput (MiB/s)	Provisioned IOPS per volume	Total provisioned IOPS
r7i.4xlarge	128	16	1 x 300 GiB	500	7,500	7,500
r7i.2xlarge	64	8	1 x 300 GiB	500	7,500	7,500
r6i.4xlarge	128	16	1 x 300 GiB	500	2,000	2,000
r6i.2xlarge	64	8	1 x 300 GiB	500	2,000	2,000
r5.4xlarge	128	16	1 x 300 GiB	500	2,000	2,000
r5.2xlarge	64	8	1 x 300 GiB	500	2,000	2,000
r5b.4xlarge	128	16	1 x 300 GiB	500	2,000	2,000
r5b.2xlarge	64	8	1 x 300 GiB	500	2,000	2,000
r4.4xlarge	122	16	1 x 300 GiB	500**	2,000	2,000
r4.2xlarge	61	8	1 x 300 GiB	500**	2,000	2,000
r3.4xlarge	122	16	1 x 300 GiB	500**	2,000	2,000

Instance type	Memory (GiB)	vCPUs / logical processors*	Provisioned IOPS SSD (io1/io2) storage with LVM	Total maximum throughput (MiB/s)	Provisioned IOPS per volume	Total provisioned IOPS
r3.2xlarge	61	8	1 x 300 GiB	500**	2,000	2,000

* Each logical processor offered by Amazon EC2 High Memory Instances is a hyperthread on a physical CPU core.

** This value represents the maximum throughput that could be achieved when striping multiple EBS volumes. Actual throughput depends on the instance type. Every instance type has its own Amazon EBS throughput maximum. For details, see Amazon EBS-Optimized Instances in the AWS documentation.

io1 for HANA logs

Instance type	Memory (GiB)	vCPUs / logical processors*	Provisioned IOPS SSD (io1/io2) storage with LVM	Total maximum throughput (MiB/s)	Provisioned IOPS per volume	Total provisioned IOPS
Certified for production use						
u-24tb1.112xlarge	24,576	448	1 x 525 GiB	500	2,000	2,000
u-24tb1.metal	24,576	448	1 x 525 GiB	500	2,000	2,000
u-18tb1.112xlarge	18,432	448	1 x 525 GiB	500	2,000	2,000

Instance type	Memory (GiB)	vCPUs / logical processors*	Provisioned IOPS SSD (io1/io2) storage with LVM	Total maximum throughput (MiB/s)	Provisioned IOPS per volume	Total provisioned IOPS
u-18tb1.metal	18,432	448	1 x 525 GiB	500	2,000	2,000
u-12tb1.112xlarge	12,288	448	1 x 525 GiB	500	2,000	2,000
u-12tb1.metal	12,288	448	1 x 525 GiB	500	2,000	2,000
u-9tb1.112xlarge	9,216	448	1 x 525 GiB	500	2,000	2,000
u-9tb1.metal	9,216	448	1 x 525 GiB	500	2,000	2,000
u-6tb1.112xlarge	6,144	448	1 x 525 GiB	500	2,000	2,000
u-6tb1.56xlarge	6,144	224	1 x 525 GiB	500	2,000	2,000
u-6tb1.metal	6,144	448	1 x 525 GiB	500	2,000	2,000
u-3tb1.56xlarge	3,072	224	1 x 525 GiB	500	2,000	2,000
x2iedn.32xlarge	4,096	128	1 x 525 GiB	500	2,000	2,000
x2iedn.24xlarge	3,072	96	1 x 525 GiB	500	2,000	2,000

Instance type	Memory (GiB)	vCPUs / logical processors*	Provisioned IOPS SSD (io1/io2) storage with LVM	Total maximum throughput (MiB/s)	Provisioned IOPS per volume	Total provisioned IOPS
x2idn.32xlarge	2,048	128	1 x 525 GiB	500	2,000	2,000
x2idn.24xlarge	1,536	96	1 x 525 GiB	500	2,000	2,000
x2idn.16xlarge	1,024	64	1 x 525 GiB	500	2,000	2,000
x1e.32xlarge	3,904	128	1 x 525 GiB	500	2,000	2,000
x1.32xlarge	1,952	128	1 x 525 GiB	500	2,000	2,000
x1.16xlarge	976	64	1 x 525 GiB	500	2,000	2,000
r7i.48xlarge	1,536	192	1 x 525 GiB	500	2,000	2,000
r7i.24xlarge	768	96	1 x 525 GiB	500	2,000	2,000
r7i.16xlarge	512	64	1 x 260 GiB	500	2,000	2,000
r7i.12xlarge	384	48	1 x 260 GiB	500	2,000	2,000
r7i.8xlarge	256	32	1 x 260 GiB	500	2,000	2,000

Instance type	Memory (GiB)	vCPUs / logical processors*	Provisioned IOPS SSD (io1/io2) storage with LVM	Total maximum throughput (MiB/s)	Provisioned IOPS per volume	Total provisioned IOPS
r6i.32xlarge	1,024	128	1 x 525 GiB	500	2,000	2,000
r6i.24xlarge	768	96	1 x 525 GiB	500	2,000	2,000
r6i.16xlarge	512	64	1 x 260 GiB	500	2,000	2,000
r6i.12xlarge	384	48	1 x 260 GiB	500	2,000	2,000
r6i.8xlarge	256	32	1 x 260 GiB	250	1,000	1,000
r5.24xlarge	768	96	1 x 525 GiB	500	2,000	2,000
r5.16xlarge	512	64	1 x 260 GiB	500	2,000	2,000
r5.12xlarge	384	48	1 x 260 GiB	500	2,000	2,000
r5.8xlarge	256	32	1 x 260 GiB	500	2,000	2,000
r5.metal	768	96	1 x 525 GiB	500	2,000	2,000
r5b.24xlarge	768	96	1 x 525 GiB	500	2,000	2,000

Instance type	Memory (GiB)	vCPUs / logical processors*	Provisioned IOPS SSD (io1/io2) storage with LVM	Total maximum throughput (MiB/s)	Provisioned IOPS per volume	Total provisioned IOPS
r5b.16xlarge	512	64	1 x 260 GiB	500	2,000	2,000
r5b.12xlarge	384	48	1 x 260 GiB	500	2,000	2,000
r5b.8xlarge	256	32	1 x 260 GiB	500	2,000	2,000
r5b.metal	768	96	1 x 525 GiB	500	2,000	2,000
r4.16xlarge	488	64	1 x 260 GiB	500	2,000	2,000
r4.8xlarge	244	32	1 x 260 GiB	500	2,000	2,000
r3.8xlarge	244	32	1 x 260 GiB	500	2,000	2,000
Supported for nonproduction use only						
x2iedn.4xlarge	512	16	1 x 260 GiB	250	1,000	1,000
x2iedn.2xlarge	256	8	1 x 260 GiB	250	1,000	1,000
x2iedn.xlarge	128	4	1 x 260 GiB	250	1,000	1,000

Instance type	Memory (GiB)	vCPUs / logical processors*	Provisioned IOPS SSD (io1/io2) storage with LVM	Total maximum throughput (MiB/s)	Provisioned IOPS per volume	Total provisioned IOPS
x1e.4xlarge	488	16	1 x 260 GiB	250**	1,000	1,000
x1e.2xlarge	244	8	1 x 260 GiB	250**	1,000	1,000
x1e.xlarge	122	4	1 x 260 GiB	250**	1,000	1,000
r7i.4xlarge	128	16	1 x 260 GiB	500	2,000	2,000
r7i.2xlarge	64	8	1 x 260 GiB	250	1,000	1,000
r6i.4xlarge	128	16	1 x 260 GiB	250	1,000	1,000
r6i.2xlarge	64	8	1 x 260 GiB	250	1,000	1,000
r5.4xlarge	128	16	1 x 260 GiB	250	1,000	1,000
r5.2xlarge	64	8	1 x 260 GiB	250	1,000	1,000
r5b.4xlarge	128	16	1 x 260 GiB	250	1,000	1,000
r5b.2xlarge	64	8	1 x 260 GiB	250	1,000	1,000

Instance type	Memory (GiB)	vCPUs / logical processors*	Provisioned IOPS SSD (io1/io2) storage with LVM	Total maximum throughput (MiB/s)	Provisioned IOPS per volume	Total provisioned IOPS
r4.4xlarge	122	16	1 x 260 GiB	250	1,000	1,000
r4.2xlarge	61	8	1 x 260 GiB	250**	1,000	1,000
r3.4xlarge	122	16	1 x 260 GiB	250	1,000	1,000
r3.2xlarge	61	8	1 x 260 GiB	250**	1,000	1,000

* Each logical processor offered by Amazon EC2 High Memory Instances is a hyperthread on a physical CPU core.

** This value represents the maximum achievable throughput when striping multiple EBS volumes. Actual throughput depends on the instance type. Every instance type has its own Amazon EBS throughput maximum. For more information, see Amazon EBS-Optimized Instances.

io2 for HANA data

Instance type	Memory (GiB)	vCPUs / logical processors*	Provisioned IOPS SSD (io1/io2) storage with LVM	Total maximum throughput (MiB/s)	Provisioned IOPS per volume	Total provisioned IOPS
Certified for production use						

Instance type	Memory (GiB)	vCPUs / logical processors*	Provisioned IOPS SSD (io1/io2) storage with LVM	Total maximum throughput (MiB/s)	Provisioned IOPS per volume	Total provisioned IOPS
x1e.32xlarge	3,904	128	3 x 1,600 GiB	1,500	3,000	9,000
x1.32xlarge	1,952	128	3 x 800 GiB	1,500	3,000	9,000
x1.16xlarge	976	64	1 x 1,200 GiB	500	7,500	7,500
r4.16xlarge	488	64	1 x 600 GiB	500	7,500	7,500
r4.8xlarge	244	32	1 x 300 GiB	500	7,500	7,500
r3.8xlarge						
Supported for nonproduction use only						
x1e.4xlarge	488	16	1 x 600 GiB	500**	2,000	2,000
x1e.2xlarge	244	8	1 x 300 GiB	500**	2,000	2,000
x1e.xlarge	122	4	1 x 300 GiB	500**	2,000	2,000
r4.4xlarge	122	16	1 x 300 GiB	500**	2,000	2,000
r4.2xlarge	61	8	1 x 300 GiB	500**	2,000	2,000

Instance type	Memory (GiB)	vCPUs / logical processors*	Provisioned IOPS SSD (io1/io2) storage with LVM	Total maximum throughput (MiB/s)	Provisioned IOPS per volume	Total provisioned IOPS
r3.4xlarge	122	16	1 x 300 GiB	500**	2,000	2,000
r3.2xlarge	61	8	1 x 300 GiB	500**	2,000	2,000

* Each logical processor offered by Amazon EC2 High Memory Instances is a hyperthread on a physical CPU core.

** This value represents the maximum throughput that could be achieved when striping multiple EBS volumes. Actual throughput depends on the instance type. Every instance type has its own Amazon EBS throughput maximum. For details, see Amazon EBS-Optimized Instances in the AWS documentation.

io2 for HANA logs

Instance type	Memory (GiB)	vCPUs / logical processors*	Provisioned IOPS SSD (io1/io2) storage with LVM	Total maximum throughput (MiB/s)	Provisioned IOPS per volume	Total provisioned IOPS
Certified for production use						
x1e.32xlarge	3,904	128	1 x 525 GiB	500	2,000	2,000
x1.32xlarge	1,952	128	1 x 525 GiB	500	2,000	2,000

Instance type	Memory (GiB)	vCPUs / logical processors*	Provisioned IOPS SSD (io1/io2) storage with LVM	Total maximum throughput (MiB/s)	Provisioned IOPS per volume	Total provisioned IOPS
x1.16xlarge	976	64	1 x 525 GiB	500	2,000	2,000
r4.16xlarge	488	64	1 x 260 GiB	500	2,000	2,000
r4.8xlarge	244	32	1 x 260 GiB	500	2,000	2,000
r3.8xlarge						
Supported for nonproduction use only						
x1e.4xlarge	488	16	1 x 260 GiB	250**	1,000	1,000
x1e.2xlarge	244	8	1 x 260 GiB	250**	1,000	1,000
x1e.xlarge	122	4	1 x 260 GiB	250**	1,000	1,000
r4.4xlarge	122	16	1 x 260 GiB	250	1,000	1,000
r4.2xlarge	61	8	1 x 260 GiB	250**	1,000	1,000
r3.4xlarge	122	16	1 x 260 GiB	250	1,000	1,000
r3.2xlarge	61	8	1 x 260 GiB	250**	1,000	1,000

* Each logical processor offered by Amazon EC2 High Memory Instances is a hyperthread on a physical CPU core.

** This value represents the maximum achievable throughput when striping multiple EBS volumes. Actual throughput depends on the instance type. Every instance type has its own Amazon EBS throughput maximum. For more information, see Amazon EBS-Optimized Instances.

io2 Block Express for HANA data

Instance type	Memory (GiB)	vCPUs / logical processors*	Provisioned IOPS SSD (io1/io2) storage with LVM	Total maximum throughput (MiB/s)	Provisioned IOPS per volume	Total provisioned IOPS
Certified for production use						
u-24tb1.112xlarge	24,576	448	2 x 14,400 GiB	4,500	9,000	18,000
u-24tb1.metal	24,576	448	2 x 14,400 GiB	4,500	9,000	18,000
u-18tb1.112xlarge	18,432	448	2 x 10,800 GiB	4,500	9,000	18,000
u-18tb1.metal	18,432	448	2 x 10,800 GiB	4,500	9,000	18,000
u-12tb1.112xlarge	12,288	448	2 x 7,200 GiB	3,000	6,000	12,000
u-12tb1.metal	12,288	448	2 x 7,200 GiB	3,000	6,000	12,000
u-9tb1.112xlarge	9,216	448	2 x 5,400 GiB	3,000	6,000	12,000

Instance type	Memory (GiB)	vCPUs / logical processors*	Provisioned IOPS SSD (io1/io2) storage with LVM	Total maximum throughput (MiB/s)	Provisioned IOPS per volume	Total provisioned IOPS
u-9tb1.metal	9,216	448	2 x 5,400 GiB	3,000	6,000	12,000
u-6tb1.112xlarge	6,144	448	2 x 3,600 GiB	3,000	6,000	12,000
u-6tb1.56xlarge	6,144	224	2 x 3,600 GiB	3,000	6,000	12,000
u-6tb1.metal	6,144	448	2 x 3,600 GiB	3,000	6,000	12,000
u-3tb1.56xlarge	3,072	224	2 x 1,800 GiB	2,250	4,500	9,000
x2iedn.32xlarge	4,096	128	2 x 2,400 GiB	2,250	4,500	9,000
x2iedn.24xlarge	3,072	96	2 x 1,800 GiB	2,250	4,500	9,000
x2idn.32xlarge	2,048	128	2 x 1,200 GiB	2,250	4,500	9,000
x2idn.24xlarge	1,536	96	2 x 900 GiB	1,875	3,750	7,500
x2idn.16xlarge	1,024	64	2 x 600 GiB	1,875	3,750	7,500
r7i.48xlarge	1,536	192	1 x 1,800 GiB	1,875	7,500	7,500

Instance type	Memory (GiB)	vCPUs / logical processors*	Provisioned IOPS SSD (io1/io2) storage with LVM	Total maximum throughput (MiB/s)	Provisioned IOPS per volume	Total provisioned IOPS
r7i.24xlarge	768	96	1 x 900 GiB	1,875	7,500	7,500
r7i.16xlarge	512	64	1 x 600 GiB	1,875	7,500	7,500
r7i.12xlarge	384	48	1 x 300 GiB	1,875	7,500	7,500
r7i.8xlarge	256	32	1 x 300 GiB	1,875	7,500	7,500
r6i.32xlarge	1,024	128	1 x 1,200 GiB	1,875	7,500	7,500
r6i.24xlarge	768	96	1 x 1,200 GiB	1,875	7,500	7,500
r6i.16xlarge	512	64	1 x 600 GiB	1,875	7,500	7,500
r6i.12xlarge	384	48	1 x 600 GiB	1,875	7,500	7,500
r6i.8xlarge	256	32	1 x 300 GiB	1,875	7,500	7,500
r5.24xlarge	768	96	1 x 1,200 GiB	1,875	7,500	7,500
r5.16xlarge	512	64	1 x 600 GiB	1,875	7,500	7,500

Instance type	Memory (GiB)	vCPUs / logical processors*	Provisioned IOPS SSD (io1/io2) storage with LVM	Total maximum throughput (MiB/s)	Provisioned IOPS per volume	Total provisioned IOPS
r5.12xlarge	384	48	1 x 600 GiB	1,875	7,500	7,500
r5.8xlarge	256	32	1 x 300 GiB	1,875	7,500	7,500
r5.metal	768	96	1 x 1,200 GiB	1,875	7,500	7,500
r5b.24xlarge	768	96	1 x 1,200 GiB	1,875	7,500	7,500
r5b.16xlarge	512	64	1 x 600 GiB	1,875	7,500	7,500
r5b.12xlarge	384	48	1 x 600 GiB	1,875	7,500	7,500
r5b.8xlarge	256	32	1 x 300 GiB	1,875	7,500	7,500
r5b.metal	768	96	1 x 1,200 GiB	1,875	7,500	7,500
Supported for nonproduction use only						
x2iedn.4xlarge	512	16	1 x 300 GiB	500	2,000	2,000
x2iedn.2xlarge	256	8	1 x 300 GiB	500	2,000	2,000

Instance type	Memory (GiB)	vCPUs / logical processors*	Provisioned IOPS SSD (io1/io2) storage with LVM	Total maximum throughput (MiB/s)	Provisioned IOPS per volume	Total provisioned IOPS
x2iedn.xlarge	128	4	1 x 300 GiB	500	2,000	2,000
r7i.4xlarge	128	16	1 x 300 GiB	500	2,000	2,000
r7i.2xlarge	64	8	1 x 300 GiB	500	2,000	2,000
r6i.4xlarge	128	16	1 x 300 GiB	500	2,000	2,000
r6i.2xlarge	64	8	1 x 300 GiB	500	2,000	2,000
r5.4xlarge	128	16	1 x 300 GiB	500	2,000	2,000
r5.2xlarge	64	8	1 x 300 GiB	500	2,000	2,000
r5b.4xlarge	128	16	1 x 300 GiB	500	2,000	2,000
r5b.2xlarge	64	8	1 x 300 GiB	500	2,000	2,000

* Each logical processor offered by Amazon EC2 High Memory Instances is a hyperthread on a physical CPU core.

** This value represents the maximum throughput that could be achieved when striping multiple EBS volumes. Actual throughput depends on the instance type. Every instance type has

its own Amazon EBS throughput maximum. For details, see Amazon EBS-Optimized Instances in the AWS documentation.

io2 Block Express for HANA logs

Instance type	Memory (GiB)	vCPUs / logical processors*	Provisioned IOPS SSD (io1/io2) storage with LVM	Total maximum throughput (MiB/s)	Provisioned IOPS per volume	Total provisioned IOPS
Certified for production use						
u-24tb1.112xlarge	24,576	448	1 x 525 GiB	500	2,000	2,000
u-24tb1.metal	24,576	448	1 x 525 GiB	500	2,000	2,000
u-18tb1.112xlarge	18,432	448	1 x 525 GiB	500	2,000	2,000
u-18tb1.metal	18,432	448	1 x 525 GiB	500	2,000	2,000
u-12tb1.112xlarge	12,288	448	1 x 525 GiB	500	2,000	2,000
u-12tb1.metal	12,288	448	1 x 525 GiB	500	2,000	2,000
u-9tb1.112xlarge	9,216	448	1 x 525 GiB	500	2,000	2,000
u-9tb1.metal	9,216	448	1 x 525 GiB	500	2,000	2,000
u-6tb1.112xlarge	6,144	448	1 x 525 GiB	500	2,000	2,000

Instance type	Memory (GiB)	vCPUs / logical processors*	Provisioned IOPS SSD (io1/io2) storage with LVM	Total maximum throughput (MiB/s)	Provisioned IOPS per volume	Total provisioned IOPS
u-6tb1.56xlarge	6,144	224	1 x 525 GiB	500	2,000	2,000
u-6tb1.metal	6,144	448	1 x 525 GiB	500	2,000	2,000
u-3tb1.56xlarge	3,072	224	1 x 525 GiB	500	2,000	2,000
x2iedn.32xlarge	4,096	128	1 x 525 GiB	500	2,000	2,000
x2iedn.24xlarge	3,072	96	1 x 525 GiB	500	2,000	2,000
x2idn.32xlarge	2,048	128	1 x 525 GiB	500	2,000	2,000
x2idn.24xlarge	1,536	96	1 x 525 GiB	500	2,000	2,000
x2idn.16xlarge	1,024	64	1 x 525 GiB	500	2,000	2,000
r7i.48xlarge	1,536	192	1 x 525 GiB	500	2,000	2,000
r7i.24xlarge	768	96	1 x 525 GiB	500	2,000	2,000
r7i.16xlarge	512	64	1 x 260 GiB	500	2,000	2,000

Instance type	Memory (GiB)	vCPUs / logical processors*	Provisioned IOPS SSD (io1/io2) storage with LVM	Total maximum throughput (MiB/s)	Provisioned IOPS per volume	Total provisioned IOPS
r7i.12xlarge	384	48	1 x 260 GiB	500	2,000	2,000
r7i.8xlarge	256	32	1 x 260 GiB	500	2,000	2,000
r6i.32xlarge	1,024	128	1 x 525 GiB	500	2,000	2,000
r6i.24xlarge	768	96	1 x 525 GiB	500	2,000	2,000
r6i.16xlarge	512	64	1 x 260 GiB	500	2,000	2,000
r6i.12xlarge	384	48	1 x 260 GiB	500	2,000	2,000
r6i.8xlarge	256	32	1 x 260 GiB	500	2,000	2,000
r5.24xlarge	768	96	1 x 525 GiB	500	2,000	2,000
r5.16xlarge	512	64	1 x 260 GiB	500	2,000	2,000
r5.12xlarge	384	48	1 x 260 GiB	500	2,000	2,000
r5.8xlarge	256	32	1 x 260 GiB	500	2,000	2,000

Instance type	Memory (GiB)	vCPUs / logical processors*	Provisioned IOPS SSD (io1/io2) storage with LVM	Total maximum throughput (MiB/s)	Provisioned IOPS per volume	Total provisioned IOPS
r5.metal	768	96	1 x 525 GiB	500	2,000	2,000
r5b.24xlarge	768	96	1 x 525 GiB	500	2,000	2,000
r5b.16xlarge	512	64	1 x 260 GiB	500	2,000	2,000
r5b.12xlarge	384	48	1 x 260 GiB	500	2,000	2,000
r5b.8xlarge	256	32	1 x 260 GiB	500	2,000	2,000
r5b.metal	768	96	1 x 525 GiB	500	2,000	2,000
Supported for nonproduction use only						
x2iedn.4xlarge	512	16	1 x 260 GiB	250	1,000	1,000
x2iedn.2xlarge	256	8	1 x 260 GiB	250	1,000	1,000
x2iedn.xlarge	128	4	1 x 260 GiB	250	1,000	1,000
r7i.4xlarge	128	16	1 x 260 GiB	250	1,000	1,000

Instance type	Memory (GiB)	vCPUs / logical processors*	Provisioned IOPS SSD (io1/io2) storage with LVM	Total maximum throughput (MiB/s)	Provisioned IOPS per volume	Total provisioned IOPS
r7i.2xlarge	64	8	1 x 260 GiB	250	1,000	1,000
r6i.4xlarge	128	16	1 x 260 GiB	250	1,000	1,000
r6i.2xlarge	64	8	1 x 260 GiB	250	1,000	1,000
r5.4xlarge	128	16	1 x 260 GiB	250	1,000	1,000
r5.2xlarge	64	8	1 x 260 GiB	250	1,000	1,000
r5b.4xlarge	128	16	1 x 260 GiB	250	1,000	1,000
r5b.2xlarge	64	8	1 x 260 GiB	250	1,000	1,000

* Each logical processor offered by Amazon EC2 High Memory Instances is a hyperthread on a physical CPU core.

** This value represents the maximum throughput that could be achieved when striping multiple EBS volumes. Actual throughput depends on the instance type. Every instance type has its own Amazon EBS throughput maximum. For details, see Amazon EBS-Optimized Instances in the AWS documentation.

Note

io2 Block Express volume supports up to 4000 MiB/s throughput per volume with 16,000 IOPS at 256 KiB I/O size or with 64,000 IOPS at 16 KiB I/O size. The maximum throughput value represented in the *Total maximum throughput* column = Total provisioned IOPS * 256 KiB I/O. To increase the throughput, increase the provisioned IOPS.

Root, binaries, shared, and backup volumes

In addition to the SAP HANA data and log volumes, we recommend the following storage configuration for root, SAP binaries, and SAP HANA shared and backup volumes:

Instance type	Memory (GiB)	vCPUs / logical processors*	Root volume (gp2/gp3)	SAP binaries (gp2/gp3)	SAP HANA shared** (gp2/gp3)	SAP HANA backup*** (st1)
Certified for production use						
u-24tb1.112xlarge	24,576	448	1 x 50 GiB	1 x 50 GiB	1 x 1,024 GiB	2 x 16,384 GiB
u-24tb1.metal	24,576	448	1 x 50 GiB	1 x 50 GiB	1 x 1,024 GiB	2 x 16,384 GiB
u-18tb1.112xlarge	18,432	448	1 x 50 GiB	1 x 50 GiB	1 x 1,024 GiB	2 x 16,384 GiB
u-18tb1.metal	18,432	448	1 x 50 GiB	1 x 50 GiB	1 x 1,024 GiB	2 x 16,384 GiB
u-12tb1.112xlarge	12,288	448	1 x 50 GiB	1 x 50 GiB	1 x 1,024 GiB	1 x 16,384 GiB
u-12tb1.metal	12,288	448	1 x 50 GiB	1 x 50 GiB	1 x 1,024 GiB	1 x 16,384 GiB

Instance type	Memory (GiB)	vCPUs / logical processors*	Root volume (gp2/gp3)	SAP binaries (gp2/gp3)	SAP HANA shared** (gp2/gp3)	SAP HANA backup*** (st1)
u-9tb1.112xlarge	9,216	448	1 x 50 GiB	1 x 50 GiB	1 x 1,024 GiB	1 x 16,384 GiB
u-9tb1.metal	9,216	448	1 x 50 GiB	1 x 50 GiB	1 x 1,024 GiB	1 x 16,384 GiB
u-6tb1.112xlarge	6,144	448	1 x 50 GiB	1 x 50 GiB	1 x 1,024 GiB	1 x 12,288 GiB
u-6tb1.56xlarge	6,144	224	1 x 50 GiB	1 x 50 GiB	1 x 1,024 GiB	1 x 12,288 GiB
u-6tb1.metal	6,144	448	1 x 50 GiB	1 x 50 GiB	1 x 1,024 GiB	1 x 12,288 GiB
u-3tb1.56xlarge	3,072	224	1 x 50 GiB	1 x 50 GiB	1 x 1,024 GiB	1 x 6,144 GiB
x2iedn.32xlarge	4,096	128	1 x 50 GiB	1 x 50 GiB	1 x 1,024 GiB	1 x 8,192 GiB
x2iedn.24xlarge	3,072	96	1 x 50 GiB	1 x 50 GiB	1 x 1,024 GiB	1 x 6,144 GiB
x2idn.32xlarge	2,048	128	1 x 50 GiB	1 x 50 GiB	1 x 1,024 GiB	1 x 4,096 GiB
x2idn.24xlarge	1,536	96	1 x 50 GiB	1 x 50 GiB	1 x 1,024 GiB	1 x 3,096 GiB
x2idn.16xlarge	1,024	64	1 x 50 GiB	1 x 50 GiB	1 x 1,024 GiB	1 x 2,048 GiB

Instance type	Memory (GiB)	vCPUs / logical processors*	Root volume (gp2/gp3)	SAP binaries (gp2/gp3)	SAP HANA shared** (gp2/gp3)	SAP HANA backup*** (st1)
x1e.32xlarge	3,904	128	1 x 50 GiB	1 x 50 GiB	1 x 1,024 GiB	1 x 8,192 GiB
x1.32xlarge	1,952	128	1 x 50 GiB	1 x 50 GiB	1 x 1,024 GiB	1 x 4,096 GiB
x1.16xlarge	976	64	1 x 50 GiB	1 x 50 GiB	1 x 1,024 GiB	1 x 2,048 GiB
r7i.48xlarge	1,536	192	1 x 50 GiB	1 x 50 GiB	1 x 1,024 GiB	1 x 3,096 GiB
r7i.24xlarge	768	96	1 x 50 GiB	1 x 50 GiB	1 x 1,024 GiB	1 x 2,048 GiB
r7i.16xlarge	512	64	1 x 50 GiB	1 x 50 GiB	1 x 512 GiB	1 x 1,024 GiB
r7i.12xlarge	384	48	1 x 50 GiB	1 x 50 GiB	1 x 512 GiB	1 x 1,024 GiB
r7i.8xlarge	256	32	1 x 50 GiB	1 x 50 GiB	1 x 300 GiB	1 x 1,024 GiB
r6i.32xlarge	1,024	128	1 x 50 GiB	1 x 50 GiB	1 x 1,024 GiB	1 x 2,048 GiB
r6i.24xlarge	768	96	1 x 50 GiB	1 x 50 GiB	1 x 1,024 GiB	1 x 2,048 GiB
r6i.16xlarge	512	64	1 x 50 GiB	1 x 50 GiB	1 x 512 GiB	1 x 1,024 GiB

Instance type	Memory (GiB)	vCPUs / logical processors*	Root volume (gp2/gp3)	SAP binaries (gp2/gp3)	SAP HANA shared** (gp2/gp3)	SAP HANA backup*** (st1)
r6i.12xlarge	384	48	1 x 50 GiB	1 x 50 GiB	1 x 512 GiB	1 x 1,024 GiB
r6i.8xlarge	256	32	1 x 50 GiB	1 x 50 GiB	1 x 300 GiB	1 x 1,024 GiB
r5.24xlarge	768	96	1 x 50 GiB	1 x 50 GiB	1 x 1,024 GiB	1 x 2,048 GiB
r5.16xlarge	512	64	1 x 50 GiB	1 x 50 GiB	1 x 512 GiB	1 x 1,024 GiB
r5.12xlarge	384	48	1 x 50 GiB	1 x 50 GiB	1 x 512 GiB	1 x 1,024 GiB
r5.8xlarge	256	32	1 x 50 GiB	1 x 50 GiB	1 x 300 GiB	1 x 1,024 GiB
r5.metal	768	96	1 x 50 GiB	1 x 50 GiB	1 x 1,024 GiB	1 x 2,048 GiB
r5b.24xlarge	768	96	1 x 50 GiB	1 x 50 GiB	1 x 1,024 GiB	1 x 2,048 GiB
r5b.16xlarge	512	64	1 x 50 GiB	1 x 50 GiB	1 x 512 GiB	1 x 1,024 GiB
r5b.12xlarge	384	48	1 x 50 GiB	1 x 50 GiB	1 x 512 GiB	1 x 1,024 GiB
r5b.8xlarge	256	32	1 x 50 GiB	1 x 50 GiB	1 x 300 GiB	1 x 1,024 GiB

Instance type	Memory (GiB)	vCPUs / logical processors*	Root volume (gp2/gp3)	SAP binaries (gp2/gp3)	SAP HANA shared** (gp2/gp3)	SAP HANA backup*** (st1)
r5b.metal	768	96	1 x 50 GiB	1 x 50 GiB	1 x 1,024 GiB	1 x 2,048 GiB
r4.16xlarge	488	64	1 x 50 GiB	1 x 50 GiB	1 x 512 GiB	1 x 1,024 GiB
r4.8xlarge	244	32	1 x 50 GiB	1 x 50 GiB	1 x 300 GiB	1 x 1,024 GiB
r3.8xlarge						
Supported for nonproduction use only						
x2iedn.4xlarge	512	16	1 x 50 GiB	1 x 50 GiB	1 x 512 GiB	1 x 1,024 GiB
x2iedn.2xlarge	256	8	1 x 50 GiB	1 x 50 GiB	1 x 300 GiB	1 x 512 GiB
x2iedn.xlarge	128	4	1 x 50 GiB	1 x 50 GiB	1 x 300 GiB	1 x 512 GiB
x1e.4xlarge	488	16	1 x 50 GiB	1 x 50 GiB	1 x 512 GiB	1 x 1,024 GiB
x1e.2xlarge	244	8	1 x 50 GiB	1 x 50 GiB	1 x 300 GiB	1 x 512 GiB
x1e.xlarge	122	4	1 x 50 GiB	1 x 50 GiB	1 x 300 GiB	1 x 512 GiB
r7i.4xlarge	128	16	1 x 50 GiB	1 x 50 GiB	1 x 300 GiB	1 x 512 GiB
r7i.2xlarge	64	8	1 x 50 GiB	1 x 50 GiB	1 x 300 GiB	1 x 512 GiB

Instance type	Memory (GiB)	vCPUs / logical processors*	Root volume (gp2/gp3)	SAP binaries (gp2/gp3)	SAP HANA shared** (gp2/gp3)	SAP HANA backup*** (st1)
r6i.4xlarge	128	16	1 x 50 GiB	1 x 50 GiB	1 x 300 GiB	1 x 512 GiB
r6i.2xlarge	64	8	1 x 50 GiB	1 x 50 GiB	1 x 300 GiB	1 x 512 GiB
r5.4xlarge	128	16	1 x 50 GiB	1 x 50 GiB	1 x 300 GiB	1 x 512 GiB
r5.2xlarge	64	8	1 x 50 GiB	1 x 50 GiB	1 x 300 GiB	1 x 512 GiB
r5b.4xlarge	128	16	1 x 50 GiB	1 x 50 GiB	1 x 300 GiB	1 x 512 GiB
r5b.2xlarge	64	8	1 x 50 GiB	1 x 50 GiB	1 x 300 GiB	1 x 512 GiB
r4.4xlarge	122	16	1 x 50 GiB	1 x 50 GiB	1 x 300 GiB	1 x 512 GiB
r4.2xlarge	61	8	1 x 50 GiB	1 x 50 GiB	1 x 300 GiB	1 x 512 GiB
r3.4xlarge	122	16	1 x 50 GiB	1 x 50 GiB	1 x 300 GiB	1 x 512 GiB
r3.2xlarge	61	8	1 x 50 GiB	1 x 50 GiB	1 x 300 GiB	1 x 512 GiB

** Each logical processor offered by Amazon EC2 High Memory Instances is a hyperthread on a physical CPU core.*

*** In a multi-node architecture, the SAP HANA NFS shared volume is provisioned only once on the master node.*

**** In a multi-node architecture, the SAP HANA backup volume can be deployed as NFS or Amazon EFS. The size of the SAP HANA NFS backup volume is multiplied by the number of nodes. The SAP HANA backup volume is provisioned only once on the master node, and NFS is mounted on the worker nodes. There is no provision needed for Amazon EFS as it is built to scale on demand, growing and shrinking automatically as files are added and removed.*

Backup options

For SAP HANA backup, you can choose file-based backup with storage configuration recommended in this guide or AWS Backint for SAP HANA to backup your database on Amazon S3. AWS Backint Agent for SAP HANA is an SAP-certified backup and restore solution for SAP HANA workloads running on Amazon EC2 instances. With AWS Backint for SAP HANA as your backup solution, provisioning additional Amazon EBS storage volumes or Amazon EFS file systems becomes optional. For more details, see AWS Backint Agent for SAP HANA.

For Disaster Recovery (DR) purposes, you can also automate the creation of application-consistent EBS snapshots for SAP HANA using Amazon Data Lifecycle Manager and the AWS Systems Manager document for SAP HANA. EBS snapshots make it easy to maintain a copy of your SAP HANA databases in another Region or account. Restoring an entire SAP HANA database from an EBS snapshot can take longer than other backups. However, you can reduce the restore time by enabling the EBS snapshots for Amazon EBS fast snapshot restore. We recommend that you use EBS snapshots to supplement your existing backups with AWS Backint Agent, and to use Amazon Data Lifecycle Manager to automate the copying and retention of EBS snapshots in DR Regions as needed. For more information, see Amazon EBS snapshots for SAP HANA.

For single-node deployment, we recommend using Amazon EBS Throughput Optimized HDD (`st1`) volumes for SAP HANA to perform file-based backup. This volume type provides low-cost magnetic storage designed for large sequential workloads. SAP HANA uses sequential I/O with large blocks to back up the database, so `st1` volumes provide a low-cost, high-performance option for this scenario. To learn more about `st1` volumes, see Amazon EBS Volume Types.

The SAP HANA backup volume size is designed to provide optimal baseline and burst throughput as well as the ability to hold several backup sets. Holding multiple backup sets inthe backup volume makes it easier to recover your database if necessary. You may resize your SAP HANA backup volume after initial setup if needed. To learn more about resizing your Amazon EBS volumes, see Expanding the Storage Size of an EBS Volume on Linux.

For multi-node deployment, we recommend using Amazon EFS for SAP HANA to perform file-based backup. It can support performance over 10 GB/sec and over 500,000 IOPS.

The configurations recommended in this guide are used by AWS Launch Wizard for SAP.

Networking

SAP HANA components communicate over the following logical network zones:

- Client zone – to communicate with different clients such as SQL clients, SAP Application Server, SAP HANA Extended Application Services (XS), and SAP HANA Studio
- Internal zone – to communicate with hosts in a distributed SAP HANA system as well as for SAP HSR
- Storage zone – to persist SAP HANA data in the storage infrastructure for resumption after start or recovery after failure

Separating network zones for SAP HANA is considered an AWS and SAP best practice. It enables you to isolate the traffic required for each communication channel.

In a traditional, bare-metal setup, these different network zones are set up by having multiple physical network cards or virtual LANs (VLANs). Conversely, on the AWS Cloud, you can use elastic network interfaces combined with security groups to achieve this network isolation. Amazon EBS-optimized instances can also be used for further isolation for storage I/O.

EBS-Optimized Instances

Many newer Amazon EC2 instance types such as the X1 use an optimized configuration stack and provide additional, dedicated capacity for Amazon EBS I/O. These are called EBS-optimized instances. This optimization provides the best performance for your EBS volumes by minimizing contention between Amazon EBS I/O and other traffic from your instance.

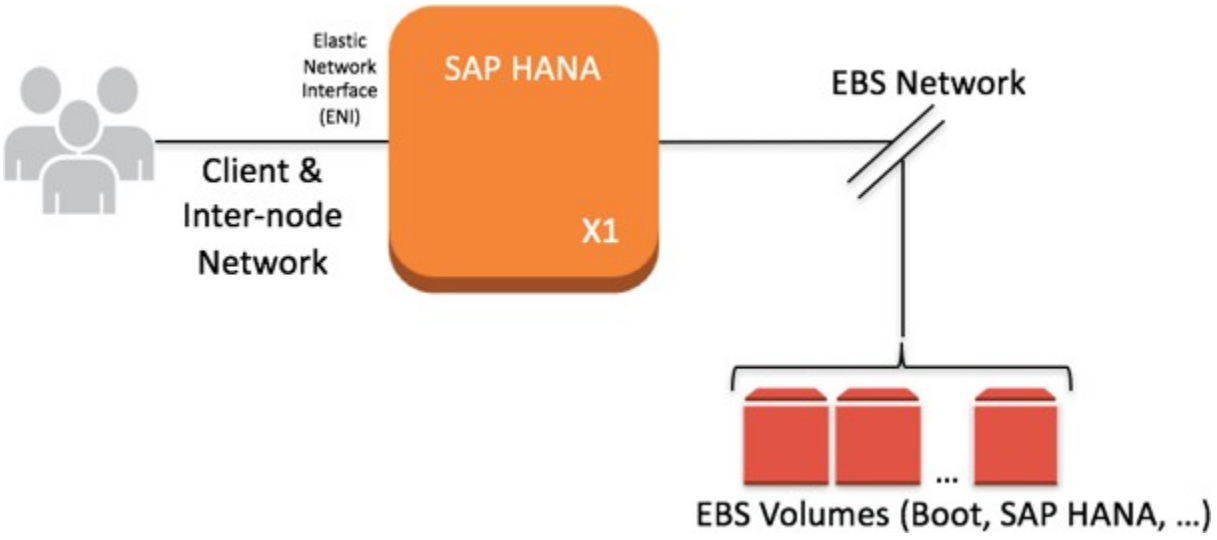


Figure 9: EBS-optimized instances

Elastic Network Interfaces

An elastic network interface is a virtual network interface that you can attach to an EC2 instance in an Amazon Virtual Private Cloud (Amazon VPC). With an elastic network interface (referred to as *network interface* in the remainder of this guide), you can create different logical networks by specifying multiple private IP addresses for your instances.

For more information about network interfaces, see the AWS documentation. In the following example, two network interfaces are attached to each SAP HANA node as well as in a separate communication channel for storage.

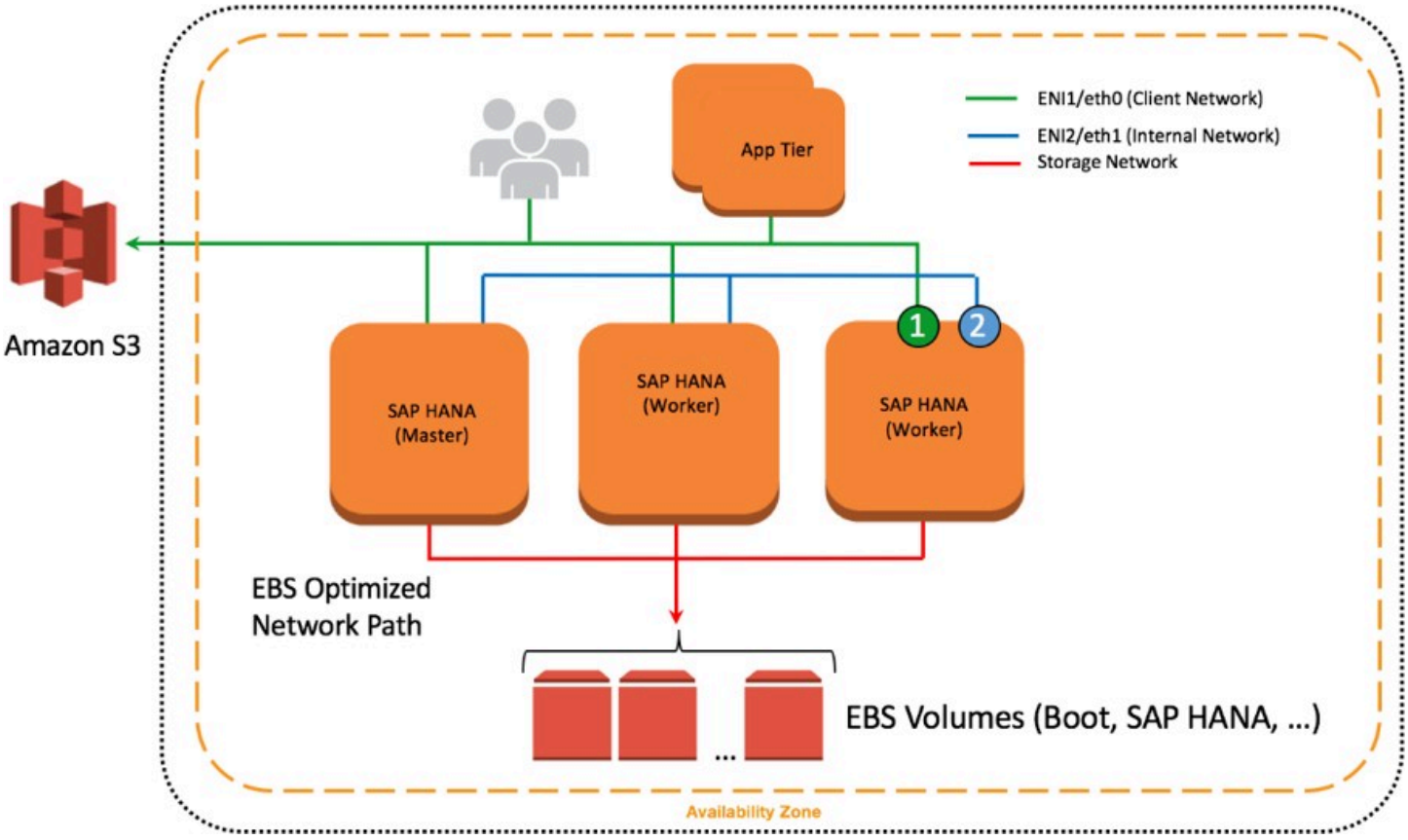


Figure 10: Network interfaces attached to SAP HANA nodes

Security Groups

A security group acts as a virtual firewall that controls the traffic for one or more instances. When you launch an instance, you associate one or more security groups with the instance. You add rules to each security group that allow traffic to or from its associated instances. You can modify the rules for a security group at any time. The new rules are automatically applied to all instances that are associated with the security group. To learn more about security groups, see the AWS documentation. In the following example, ENI-1 of each instance shown is a member of the same security group that controls inbound and outbound network traffic for the client network.

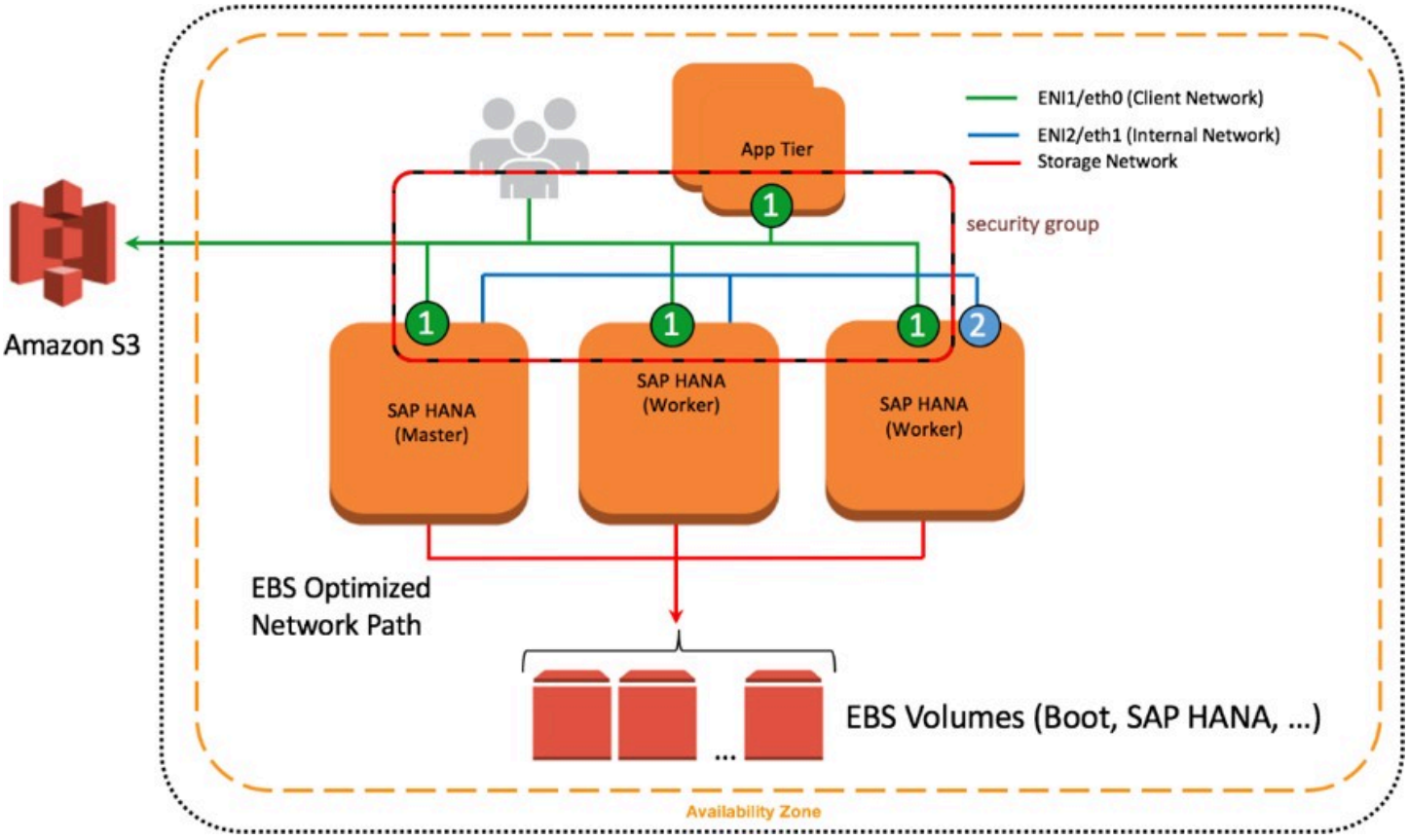


Figure 11: Network interfaces and security groups

Network Configuration for SAP HANA System Replication (HSR)

You can configure additional network interfaces and security groups to further isolate inter-node communication as well as SAP HSR network traffic. In Figure 10, ENI-2 is has its own security group (not shown) to secure client traffic from inter-node communication. ENI-3 is configured to secure SAP HSR traffic to another Availability Zone within the same Region. In this example, the target SAP HANA cluster would be configured with additional network interfaces similar to the source environment, and ENI-3 would share a common security group.

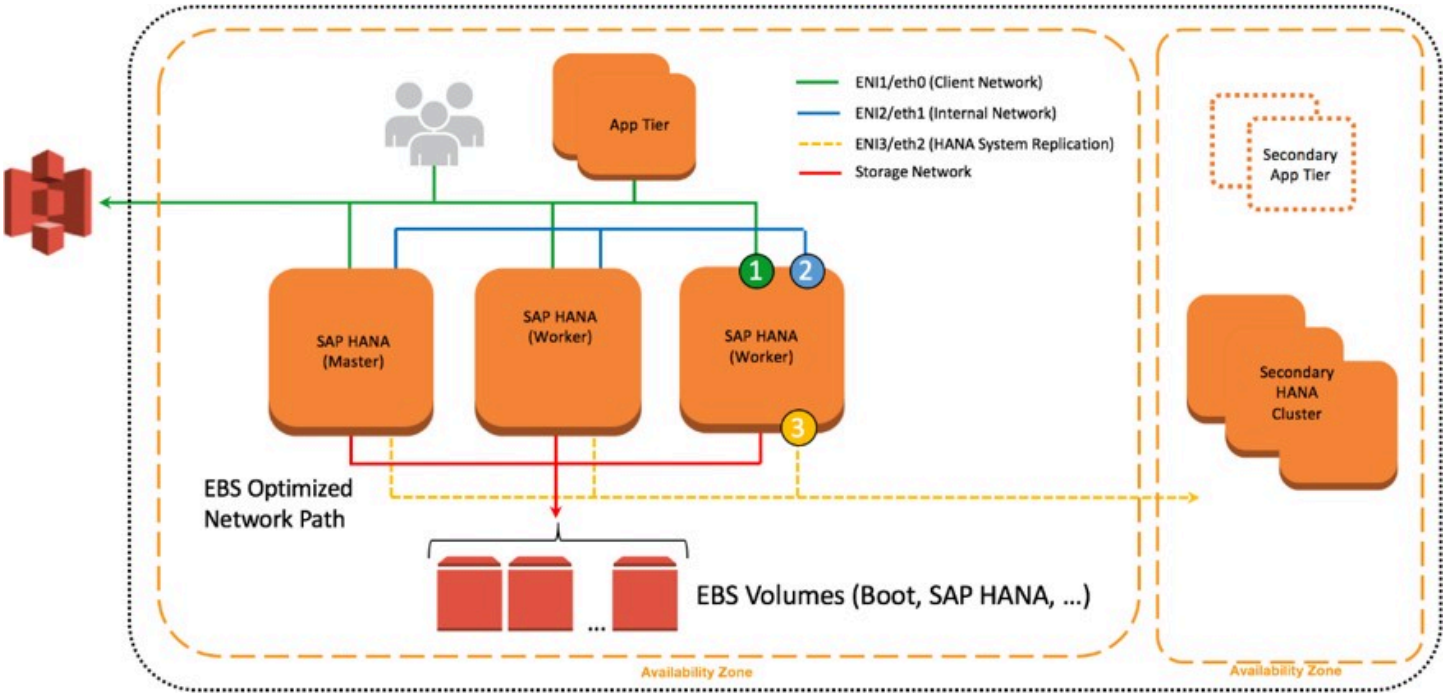


Figure 12: Further isolation with additional ENIs and security groups

Configuration Steps for Logical Network Separation

To configure your logical network for SAP HANA, follow these steps:

1. Create new security groups to allow for isolation of client, internal communication, and, if applicable, SAP HSR network traffic. See Ports and Connections in the SAP HANA documentation to learn about the list of ports used for different network zones. For more information about how to create and configure security groups, see the AWS documentation.
2. Use Secure Shell (SSH) to connect to your EC2 instance at the OS level. Follow the steps described in the appendix to configure the OS to properly recognize and name the Ethernet devices associated with the new network interfaces you will be creating.
3. Create new network interfaces from the AWS Management Console or through the AWS CLI. Make sure that the new network interfaces are created in the subnet where your SAP HANA instance is deployed. As you create each new network interface, associate it with the appropriate security group you created in step 1. For more information about how to create a new network interface, see the AWS documentation.

4. Attach the network interfaces you created to your EC2 instance where SAP HANA is installed. For more information about how to attach a network interface to an EC2 instance, see the AWS documentation.
5. Create virtual host names and map them to the IP addresses associated with client, internal, and replication network interfaces. Ensure that host name-to-IP-address resolution is working by creating entries in all applicable host files or in the Domain Name System (DNS). When complete, test that the virtual host names can be resolved from all SAP HANA nodes and clients.
6. For scale-out deployments, configure SAP HANA inter-service communication to let SAP HANA communicate over the internal network. To learn more about this step, see Configuring SAP HANA Inter-Service Communication in the SAP HANA documentation.
7. Configure SAP HANA hostname resolution to let SAP HANA communicate over the replication network for SAP HSR. To learn more about this step, see Configuring Hostname Resolution for SAP HANA System Replication in the SAP HANA documentation.

SAP Support Access

In some situations it may be necessary to allow an SAP support engineer to access your SAP HANA systems on AWS. The following information serves only as a supplement to the information contained in the “Getting Support” section of the SAP HANA Administration Guide.

A few steps are required to configure proper connectivity to SAP. These steps differ depending on whether you want to use an existing remote network connection to SAP, or you are setting up a new connection directly with SAP from systems on AWS.

Support Channel Setup with SAProuter on AWS

When setting up a direct support connection to SAP from AWS, consider the following steps:

1. For the SAProuter instance, create and configure a specific SAProuter security group, which only allows the required inbound and outbound access to the SAP support network. This should be limited to a specific IP address that SAP gives you to connect to, along with TCP port 3299. See the Amazon EC2 security group documentation for additional details about creating and configuring security groups.
2. Launch the instance that the SAProuter software will be installed on into a public subnet of the VPC and assign it an Elastic IP address.

3. Install the SAProuter software and create a saprouttab file that allows access from SAP to your SAP HANA system on AWS.
4. Set up the connection with SAP. For your internet connection, use **Secure Network Communication (SNC)**. For more information, see the SAP Remote Support – Help page.
5. Modify the existing SAP HANA security groups to trust the new SAProuter security group you have created.

> **Tip**
>
> For added security, shut down the EC2 instance that hosts the SAProuter service when it is not needed for support purposes

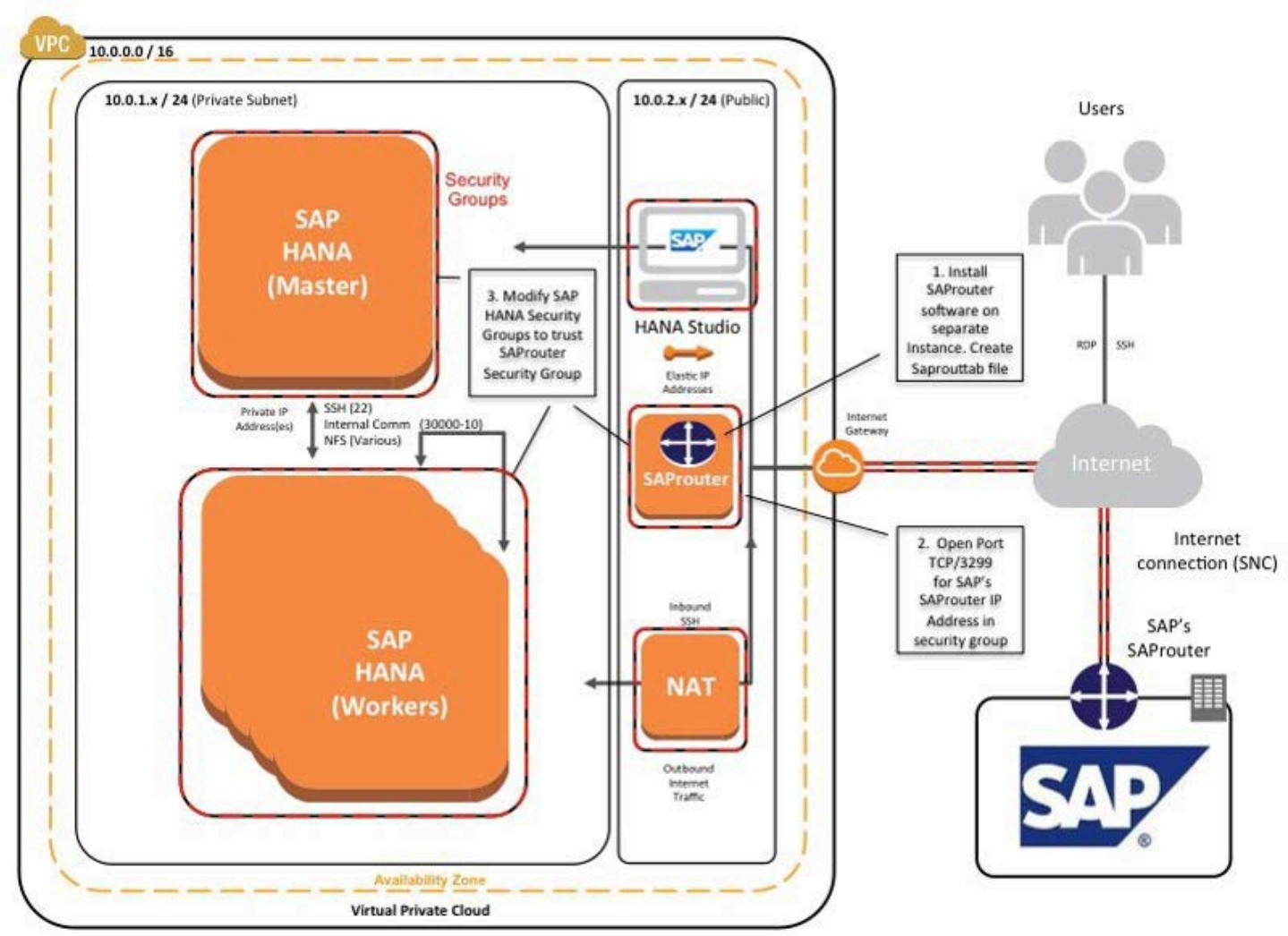


Figure 13: Support connectivity with SAProuter on AWS

Support Channel Setup with SAProuter on Premises

In many cases, you may already have a support connection configured between your data center and SAP. This can easily be extended to support SAP systems on AWS. This scenario assumes that connectivity between your data center and AWS has already been established, either by way of a secure VPN tunnel over the internet or by using AWS Direct Connect.

You can extend this connectivity as follows:

1. Ensure that the proper saprouttab entries exist to allow access from SAP to resources in the VPC.
2. Modify the SAP HANA security groups to allow access from the on- premises SAProuter IP address.
3. Ensure that the proper firewall ports are open on your gateway to allow traffic to pass over TCP port 3299.

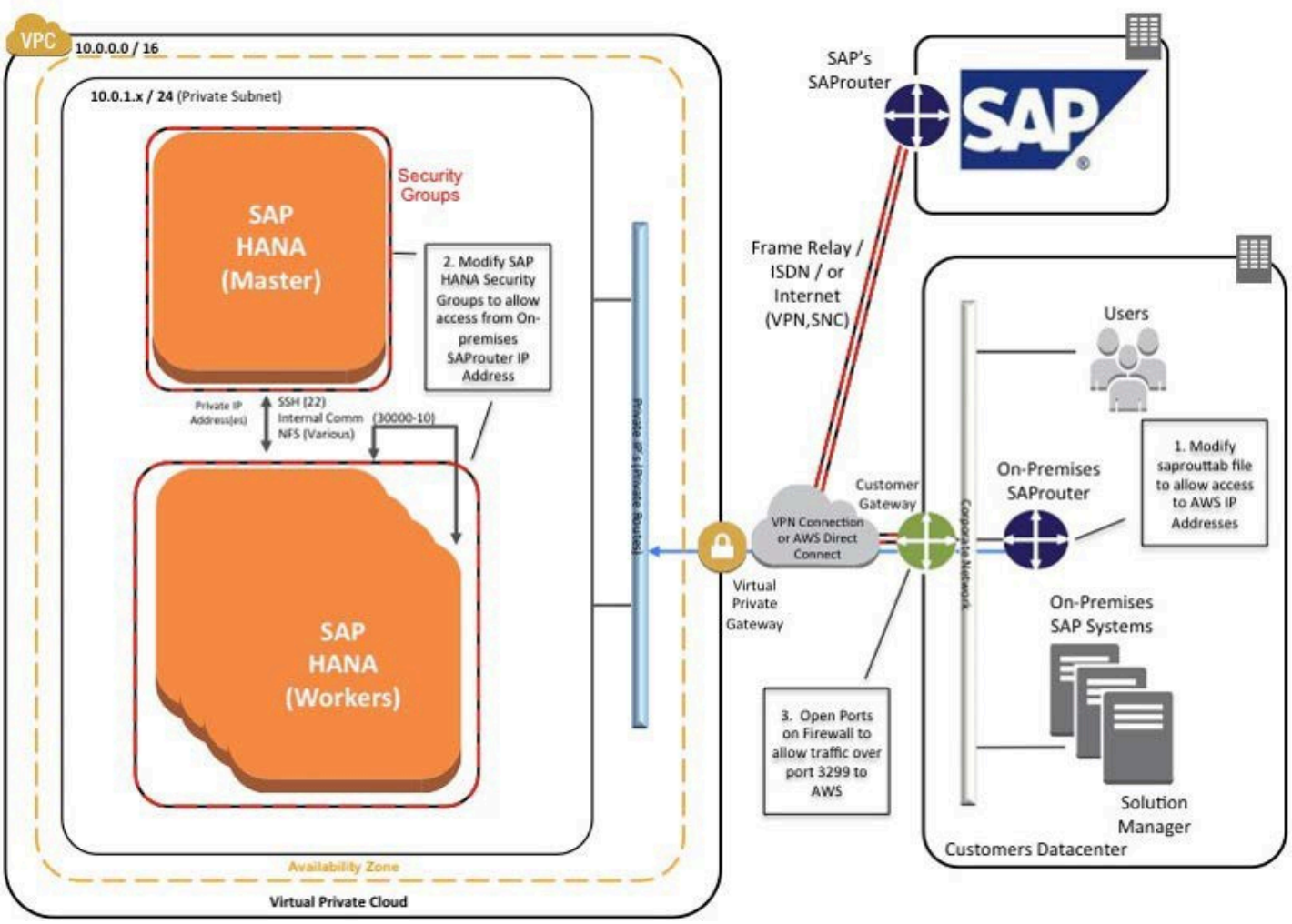


Figure 14: Support connectivity with SAProuter on premises

Security

Here are additional AWS security resources to help you achieve the level of security you require for your SAP HANA environment on AWS.

- AWS Cloud Security Center
- CIS AWS Foundation whitepaper
- Introduction to AWS Security
- AWS Cloud Security Best Practices whitepaper

OS Hardening

You may want to lock down the OS configuration further, for example, to avoid providing a DB administrator with root credentials when logging into an instance.

You can also refer to the following SAP notes:

- 1730999: *Configuration changes in HANA appliance*
- 1731000: *Unrecommended configuration changes*

Disabling HANA Services

HANA services such as HANA XS are optional and should be deactivated if they are not needed. For instructions, see SAP Note 1697613: *Remove XS Engine out of SAP HANA database*. In case of service deactivation, you should also remove the TCP ports from the SAP HANA AWS security groups for complete security.

API Call Logging

AWS CloudTrail is a web service that records AWS API calls for your account and delivers log files to you. The recorded information includes the identity of the API caller, the time of the API call, the source IP address of the API caller, the request parameters, and the response elements returned by the AWS service.

With CloudTrail, you can get a history of AWS API calls for your account, including API calls made via the AWS Management Console, AWS SDKs, command line tools, and higher-level AWS services (such as AWS CloudFormation). The AWS API call history produced by CloudTrail enables security analysis, resource change tracking, and compliance auditing.

Notifications on Access

You can use Amazon Simple Notification Service (Amazon SNS) or third-party applications to set up notifications on SSH login to your email address or mobile phone.

Architecture patterns for SAP HANA on AWS

This section provides information on architecture patterns that can be used as guidelines for deploying SAP HANA systems on AWS. For more information on the architecture patterns for SAP

NetWeaver-based applications on AWS, see Architecture guidance for availability and reliability of SAP on AWS.

You can change the patterns to fit your changing business requirements with minimum to no downtime, depending on the complexity of your chosen architecture pattern.

Topics

- SAP HANA System Replication
- Secondary SAP HANA instance
- Overview of patterns
- Single Region architecture patterns for SAP HANA
- Multi-Region architecture patterns for SAP HANA

SAP HANA System Replication

SAP HANA System Replication is a high availability solution provided by SAP for SAP HANA that can be used to reduce outage due to maintenance activities, faults, and disasters. It continuously replicates data on a secondary instance. The changes persist on the alternate instance in the event of a failure on the primary instance. For more information, see Configuring SAP HANA System Replication.

Secondary SAP HANA instance

In AWS Cloud, a secondary SAP HANA instance can exist in the same Region on a different Availability Zone or in a separate Region. For more information, see Architecture guidelines and decisions. The secondary instance can be deployed as a passive instance or an active (read-only) instance. When the secondary instance is deployed as a passive instance, you can reuse the Amazon EC2 instance capacity to accommodate a non-production SAP HANA workload.

Overview of patterns

The architecture patterns for SAP HANA are divided into the following two categories:

- the section called “Single Region patterns”
- the section called “Multi-Region patterns”

You must consider the risk and impact of each failure type, and the cost of mitigation when choosing a pattern. The following table provides a quick overview of the architecture patterns for SAP HANA systems on AWS.

Patterns	Business requirements			Solution characteristics			Implementation details	
	Resilience type	Recovery point objective [1]	Recovery time objective [2]	Cost	Complexity	Capacity re-use [3]	SAP HANA System Replication [4]	Amazon S3 replication [5]
Pattern 1	Single Region disaster recovery	Near zero	Low	Medium	Medium	Optional	2-tier	Same Region
Pattern 2		Near zero	Low	Medium	High	Yes	3-tier	
Pattern 3		Low	Medium	Low	Medium	Yes	2-tier	
Pattern 4		Medium	High	Very low	Very low	N/A	N/A	
Pattern 5	Multi-Region disaster recovery	Near zero	Low	Medium	Medium	Optional	2-tier	Cross Region
Pattern 6		Near zero	Low	High	High	Optional	3-tier	
Pattern 7		Near zero	Low	Very high	Very high	Optional	Multi-target	
Pattern 8		Medium	High	Low	Low	N/A	N/A	

[1]To achieve near zero recovery point objective, SAP HANA System Replication must be setup in sync mode for the SAP HANA instances within the same Region.

[2]To achieve the lowest recovery time objective, we recommend using a high availability setup with third-party cluster solutions in combination with SAP HANA System Replication.

[3]A production sized Amazon EC2 instance can be deployed as an MCOS installation to accommodate a non-production SAP HANA instance.

[4]SAP HANA System Replication and the number of SAP HANA instance copies as targets.

[5]Same-Region replication copies objects across Amazon S3 buckets in the same Region.

Single Region architecture patterns for SAP HANA

Single Region architecture patterns help you avoid network latency as your SAP workload components are located in a close proximity within the same Region. Every AWS Region generally has three Availability Zones. For more information, see AWS Global Infrastructure Map.

You can choose these patterns when you need to ensure that your SAP data resides within regional boundaries stipulated by the data sovereignty laws.

The following are the four single Region architecture patterns.

Patterns

- Pattern 1: Single Region with two Availability Zones for production
- Pattern 2: Single Region with two Availability Zones for production and production sized non-production in a third Availability Zone
- Pattern 3: Single Region with one Availability Zone for production and another Availability Zone for non-production
- Pattern 4: Single Region with one Availability Zone for production

Pattern 1: Single Region with two Availability Zones for production

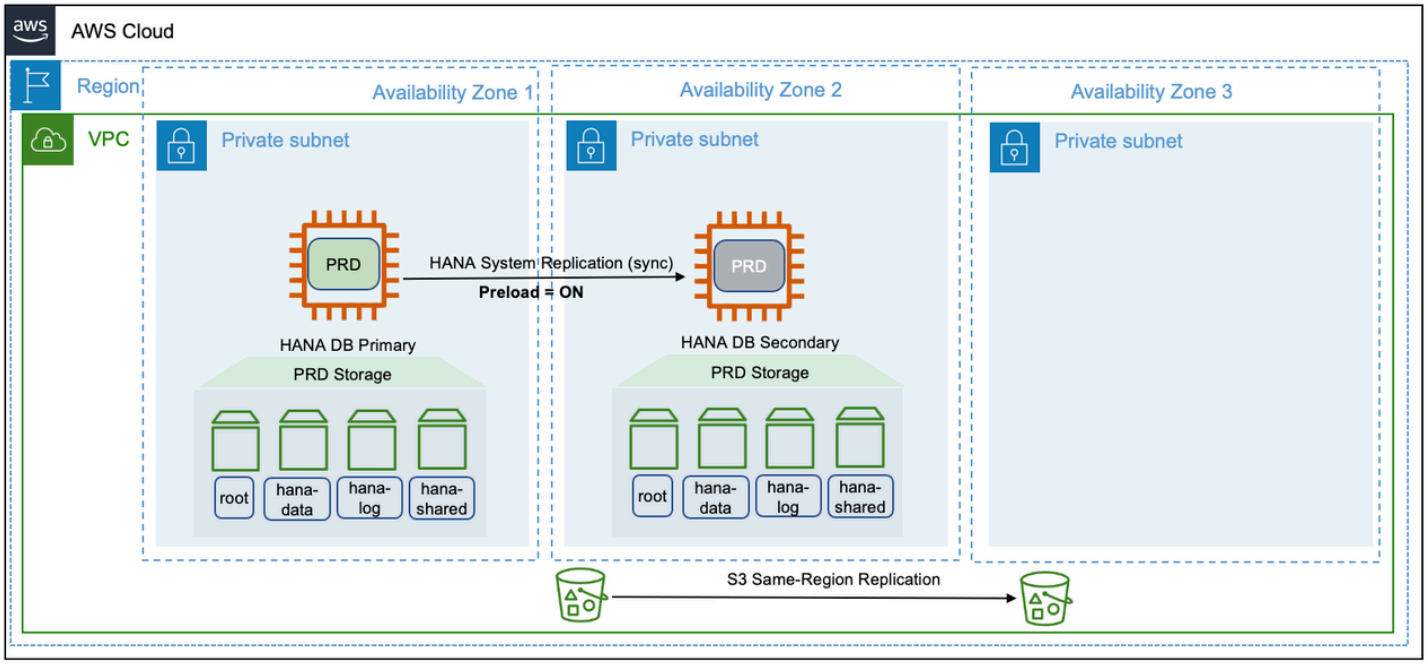
In this pattern, SAP HANA instance is deployed across two Availability Zones with SAP HANA System Replication configured on both the instances. The primary and secondary instances are of the same instance type. The secondary instance can be deployed in active/passive or active/active mode. We recommend using the sync mode of HANA System Replication for the low-latency

connectivity between the two Availability Zones. For more information, see Replication Modes for SAP HANA System Replication.

This pattern is foundational if you are looking for high availability cluster solutions for automated failover to fulfill near-zero recovery point and time objectives. SAP HANA System Replication with high availability cluster solutions for automated failover provides resiliency against failure scenarios. For more information, see Failure scenarios.

You need to consider the cost of licensing for third-party cluster solutions. If the secondary SAP HANA instance is not being used for read-only operations, then it is an idle capacity. Provisioning production equivalent instance type as standby adds to the total cost of ownership.

Your SAP HANA instance backups can be stored in Amazon S3 buckets using AWS Backint Agent for SAP HANA. Amazon S3 objects are automatically stored across multiple devices spanning a minimum of three Availability Zones across a Region. To protect against logical data loss, you can use the Same-Region Replication feature of Amazon S3. For more information, see Setting up replication.



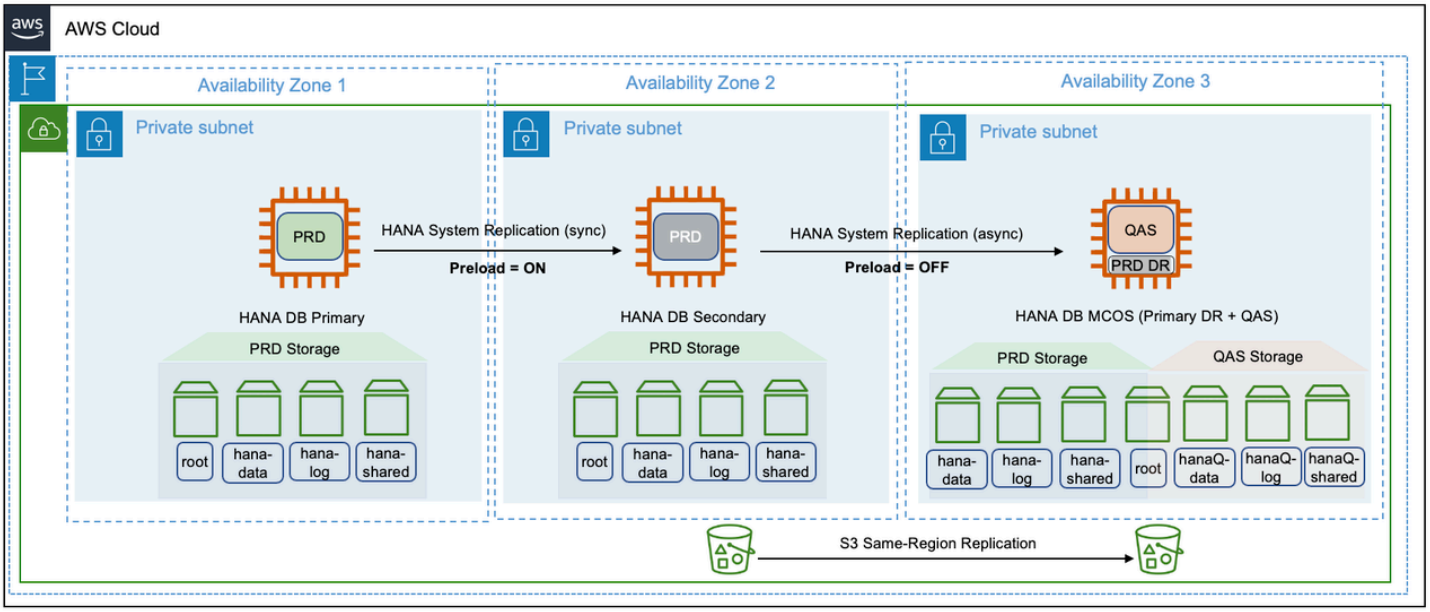
Pattern 2: Single Region with two Availability Zones for production and production sized non-production in a third Availability Zone

In this pattern, SAP HANA instance is deployed in a multi-tier SAP HANA System Replication across three Availability Zones. The primary and secondary SAP HANA instances are of the same instance type and can be configured in a highly available setup, using third-party cluster solutions. The

secondary SAP HANA instance can be deployed in an active/passive or active/active configuration. We recommend using the sync mode of SAP HANA System Replication for the low-latency connectivity between the two Availability Zones. The tertiary SAP HANA instance is deployed in a third Availability Zone, as a Multiple Components on One System (MCOS) installation. The production instance is co-hosted (on the same Amazon EC2 instance) along with a non-production SAP HANA instance.

This architectural pattern is cost-optimized. It aids disaster recovery in the unlikely event of losing connection to two Availability Zones at the same time. For disaster recovery, the non-production SAP HANA workload is stopped to make resources available for production workload. However, invoking disaster recovery (third Availability Zone) is a manual activity. As per the requirements of MCOS, you are required to provision the non-production SAP HANA instance with the same AWS instance type as that of the primary instance and it has to be located in a third Availability Zone. Also, operating an MCOS system requires additional storage for non-production workloads and detailed tested procedures to invoke a disaster recovery.

In comparison to pattern 1, pattern 2 further enhances the application availability. There is no restoration or recovery from backups required to invoke a disaster recovery. The additional cost of the third instance is justified as the idle capacity is being utilized for non-production workloads.

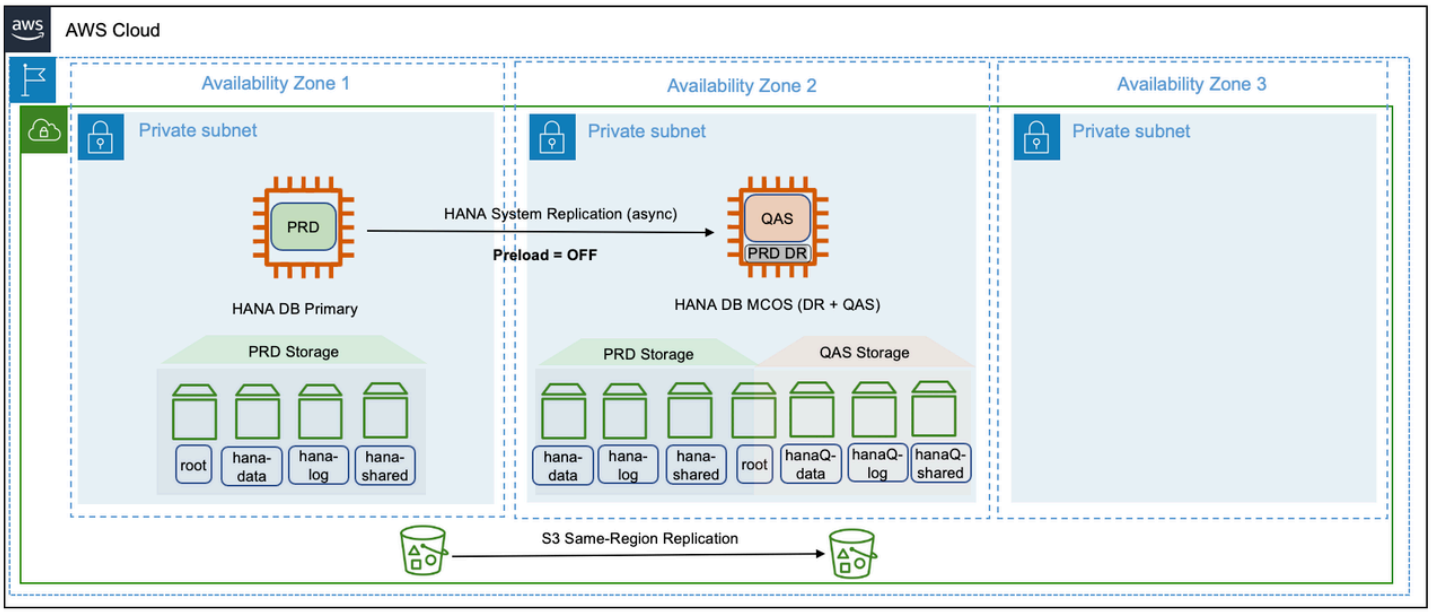


Pattern 3: Single Region with one Availability Zone for production and another Availability Zone for non-production

In this pattern, SAP HANA instance is deployed in a two-tier SAP HANA System Replication across two Availability Zones. The primary and secondary SAP HANA instances are of the same type and

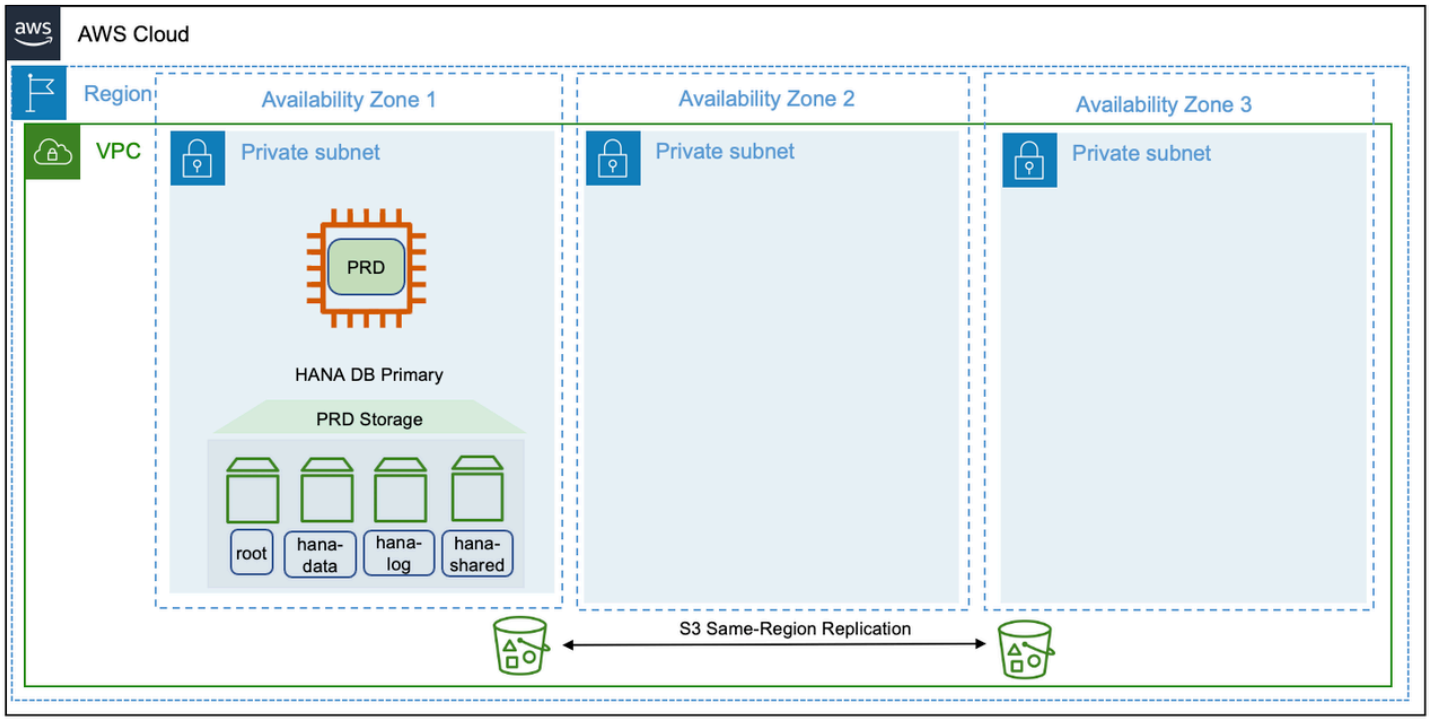
there is no idle capacity or high availability licensing requirement. Additional storage is required for the non-production SAP HANA workloads on the secondary instance.

The secondary instance is an MCOS installation and co-hosts a non-production SAP HANA workload. For more information, see SAP Note Multiple SAP HANA DBMSs (SIDs) on one SAP HANA system. This is a cost-optimized solution without high availability. In the event of a failure on the primary instance, the non-production SAP HANA workload is stopped and a takeover is performed on the secondary instance. Considering the time taken in recovering services on the secondary instance, this type of pattern is suitable for SAP HANA workloads that can have a higher recovery time objective and are functioning as disaster recovery systems.



Pattern 4: Single Region with one Availability Zone for production

In this pattern, SAP HANA instance is deployed as a standalone installation with no target systems to replicate data. This is the most basic and cost-efficient deployment option. However, this is the least resilient of all the architectures and is not recommended for business-critical SAP HANA workloads. The options available to restore business operations during a failure scenario are by Amazon EC2 auto recovery, in the event of an instance failure or by restoration and recovery from most recent and valid backups, in the event of a significant issue impacting the Availability Zone. The non-production SAP HANA workloads have no dependency on the production SAP HANA instance. They are free to be deployed in an Availability Zone within the Region and can be appropriately sized for its workload.



Multi-Region architecture patterns for SAP HANA

AWS Global Infrastructure spans across multiple Regions around the world and this footprint is constantly increasing. For the latest updates, see AWS Global Infrastructure. If you are looking for your SAP data to reside in multiple regions at any given point to ensure increased availability and minimal downtime in the event of failure, you should opt for multi-Region architecture patterns.

When deploying a multi-Region pattern, you can benefit from using an automated approach such as, cluster solution, for fail over between Availability Zones to minimize the overall downtime and remove the need for human intervention. Multi-Region patterns not only provide high availability but also disaster recovery, thereby lowering overall costs. Distance between the chosen regions have direct impact on latency and hence, in a multi-Region pattern, this has to be considered into the overall design of SAP HANA System Replication.

There are additional cost implications from cross-Region replication or data transfer that also need to be factored into the overall solution pricing. The pricing varies between Regions.

The following are the four multi-Region architecture patterns.

Patterns

- Pattern 5: Primary Region with two Availability Zones for production and secondary Region with a replica of backups/AMIs

- Pattern 6: Primary Region with two Availability Zones for production and secondary Region with compute and storage capacity deployed in a single Availability Zone
- Pattern 7: Primary Region with two Availability Zones for production and a secondary Region with compute and storage capacity deployed, and data replication across two Availability Zones
- Pattern 8: Primary Region with one Availability Zone for production and a secondary Region with a replica of backups/AMIs
- Summary

Pattern 5: Primary Region with two Availability Zones for production and secondary Region with a replica of backups/AMIs

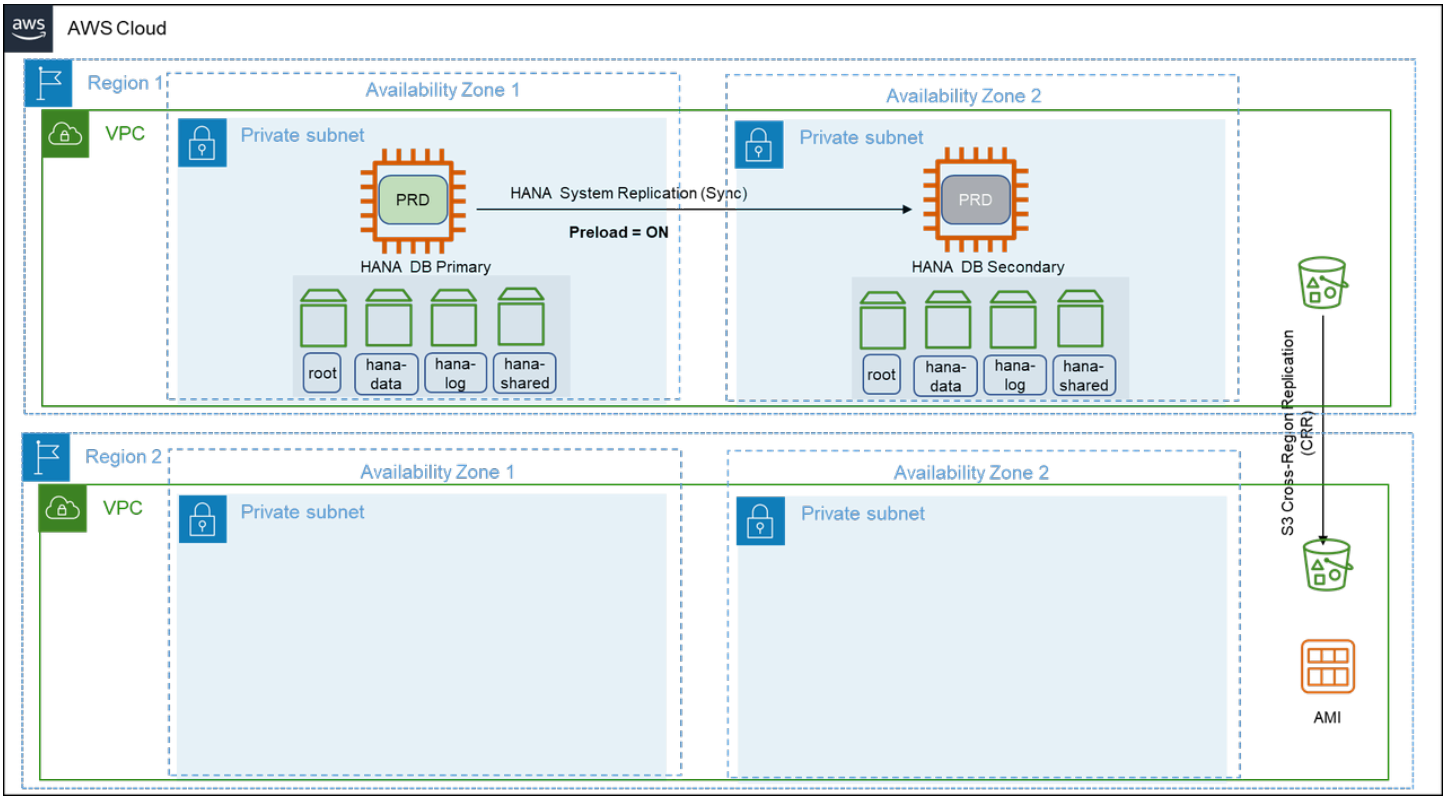
This pattern is similar to pattern 1 where your SAP HANA instance is highly available. You deploy your production SAP HANA instance across two Availability Zones in the primary Region using synchronous SAP HANA System Replication. You can restore your SAP HANA instance in a secondary Region with a replica of backups stores in Amazon S3, Amazon EBS, and Amazon Machine Images (AMIs).

With cross-Region replication of files stored in Amazon S3, the data stored in a bucket is automatically (asynchronously) copied to the target Region. Amazon EBS snapshots can be copied between Regions. For more information, see Copy an Amazon EBS snapshot. You can copy an AMI within or across Regions using AWS CLI, AWS Management Console, AWS SDKs or Amazon EC2 APIs. For more information, see Copy an AMI. You can also use AWS Backup to schedule and run snapshots and replications across Regions.

In the event of a complete Region failure, the production SAP HANA instance needs to be built in the secondary Region using AMI. You can use AWS CloudFormation templates to automate the launch of a new SAP HANA instance. Once your instance is launched, you can then download the last set of backup from Amazon S3 to restore your SAP HANA instance to a point-in-time before the disaster event. You can also use AWS Backint Agent to restore and recover your SAP HANA instance and redirect your client traffic to the new instance in the secondary Region.

This architecture provides you with the advantage of implementing your SAP HANA instance across multiple Availability Zones with the ability to failover instantly in the event of a failure. For disaster recovery that is outside the primary Region, recovery point objective is constrained by how often you store your SAP HANA backup files in your Amazon S3 bucket and the time it takes to replicate your Amazon S3 bucket to the target Region. You can use Amazon S3 replication time control for a time-bound replication. For more information, see Enabling Amazon S3 Replication Time Control.

Your recovery time objective depends on the time it takes to build the system in the secondary Region and restore operations from backup files. The amount of time will vary depending on the size of the database. Also, the time required to get the compute capacity for restore procedures may be more in the absence of a reserved instance capacity. This pattern is suitable when you need the lowest possible recovery time and point objectives within a region and high recovery point and time objectives for disaster recovery outside the primary Region.



Pattern 6: Primary Region with two Availability Zones for production and secondary Region with compute and storage capacity deployed in a single Availability Zone

In addition to the architecture of pattern 5, this pattern has an asynchronous SAP HANA System Replication is setup between the SAP HANA instance in the primary Region and an identical third instance in one of the Availability Zones in the secondary Region. We recommend using the asynchronous mode of SAP HANA System Replication when replicating between AWS Regions due to increased latency.

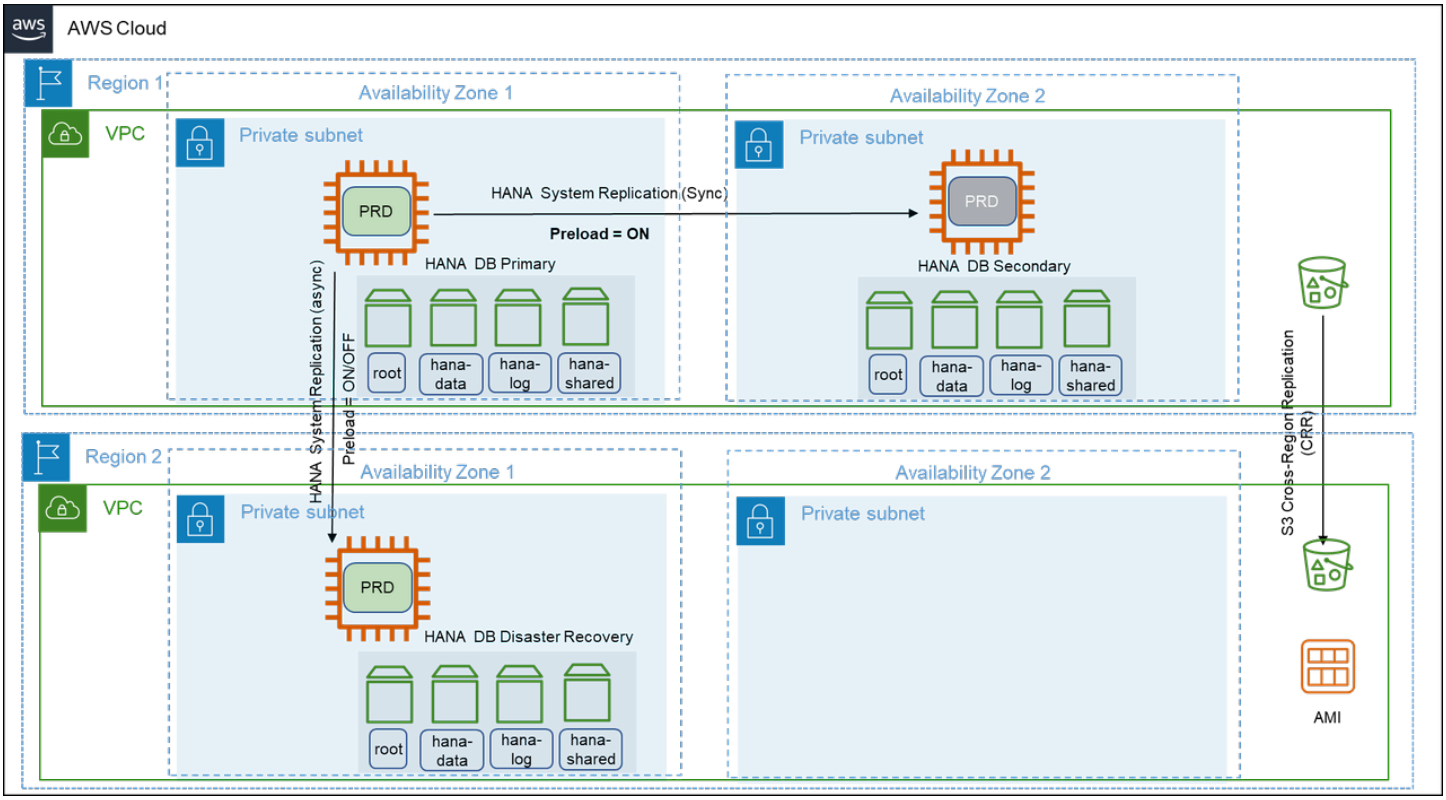
In the event of a failure in the primary Region, the production workloads are failed over to the secondary Region manually. This pattern ensures that your SAP systems are highly available and

are disaster-tolerant. This pattern provides a quicker failover and continuity of business operations with continuous data replication.

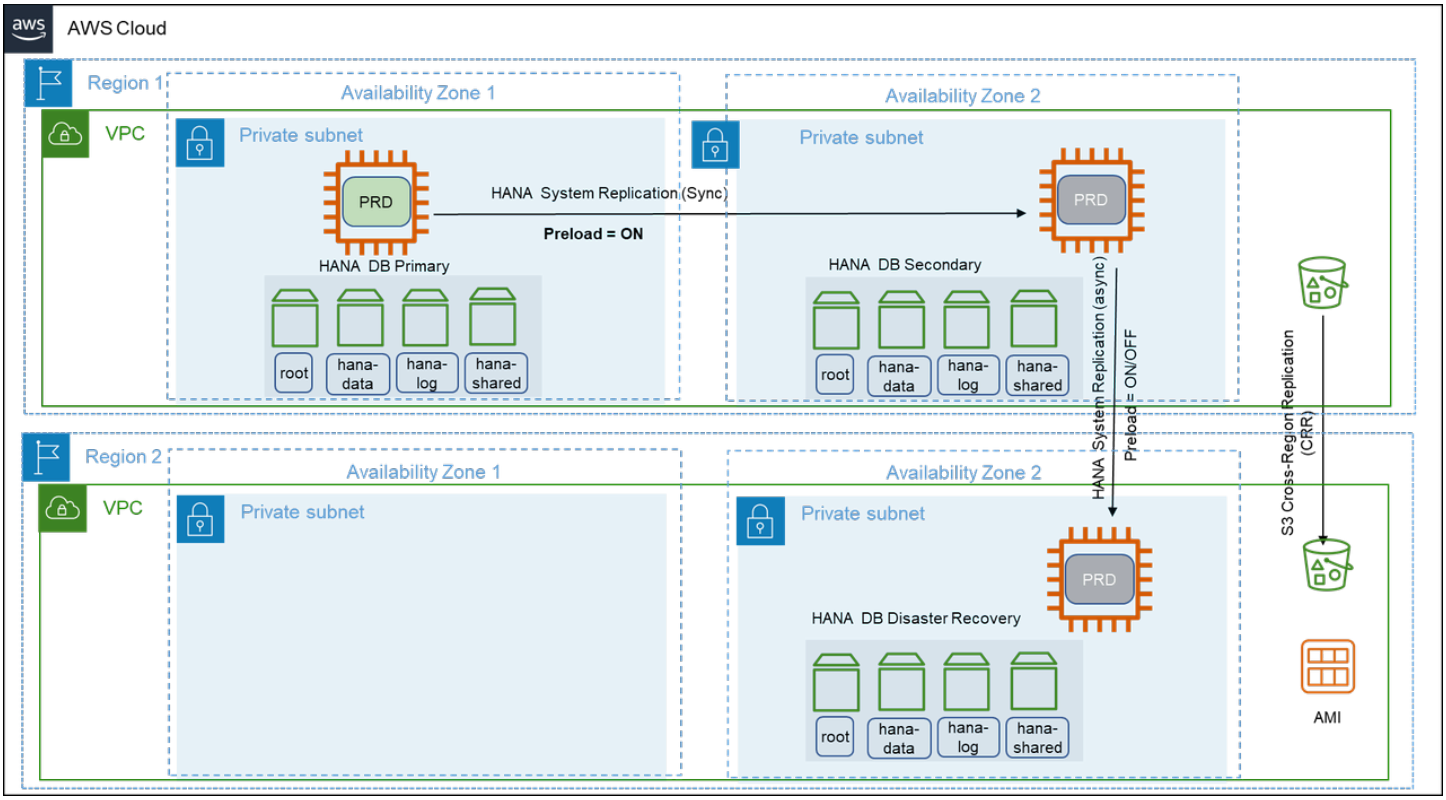
There is an increased cost of deploying the required compute and storage for the production SAP HANA instance in the secondary Region and of data transfers between Regions. This pattern is suitable when you require disaster recovery outside of the primary Region with low recovery point and time objectives.

This pattern can be deployed in a multi-tier as well as multi-target replication configuration.

The following diagram shows a multi-target replication where the primary SAP HANA instance is replicated on both Availability Zones within the same Region and also in the secondary Region.



The following diagram shows a multi-tier replication where the replication is configured in a chained fashion.



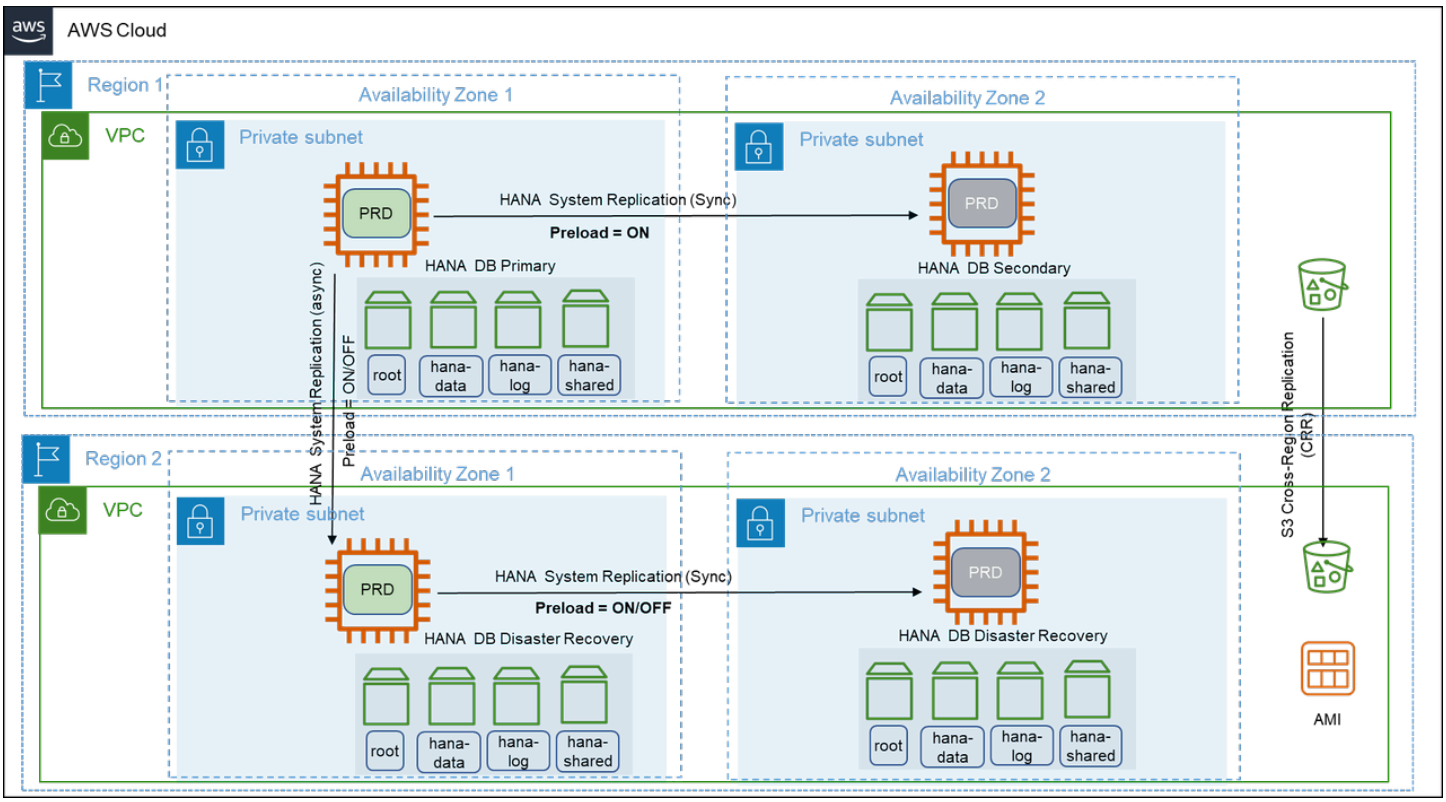
Pattern 7: Primary Region with two Availability Zones for production and a secondary Region with compute and storage capacity deployed, and data replication across two Availability Zones

In this pattern, two sets of two-tier SAP HANA System Replication is deployed across two AWS Regions. The two-tier SAP HANA System Replication is configured across two Availability Zones within the same Region and the replication outside of the primary Region is configured using SAP HANA Multi-target System Replication. This setup can be extended with high availability cluster solution for automatic failover capability on the primary Region. For more information, see SAP HANA Multi-target System Replication.

This pattern provides protection against failures in the Availability Zones and Regions. However, a cross-Region takeover of SAP HANA instance requires manual intervention. During a failover of the secondary Region, the SAP HANA instance continues to have SAP HANA System Replication up and running in the new Region without any manual intervention. This setup is applicable if you are looking for the highest application availability at all times and disaster recovery outside the primary Region with the least possible recovery point and time objectives. This pattern can withstand an extremely rare possibility of the failure of three Availability Zones spread across multiple Regions.

This pattern is highly suitable for you if you operate active/active (read-only) SAP HANA instances in the primary Region and plan to continue the same SAP HANA System Replication configuration with read-only capability. If you are looking for read-only capability across two Regions along with an existing read-only instance within the Region, you can configure multiple secondary systems supporting active/active (read-only) configuration. However, only one of the systems can be accessed via hint-based statement routing and the others must be accessed via direct connection.

With this pattern, the redundant compute and storage capacity deployed across two Availability Zones in two Regions and the cross-Region communication add to the total cost of ownership.



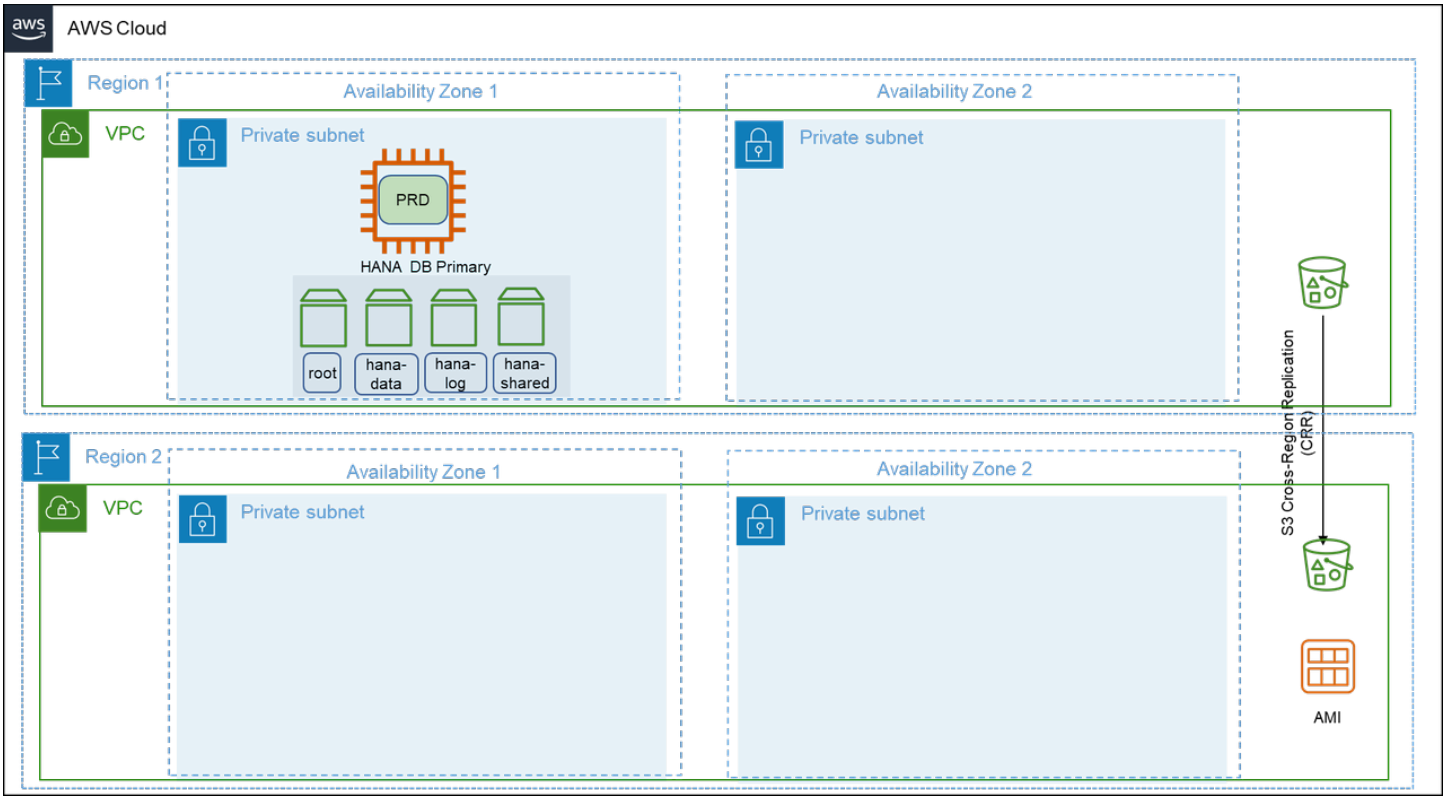
Pattern 8: Primary Region with one Availability Zone for production and a secondary Region with a replica of backups/AMIs

This pattern is similar to pattern 4 with additional disaster recovery in a secondary Region containing replicas of the SAP HANA instance backups stored in Amazon S3, Amazon EBS snapshots, and AMIs. In this pattern, the SAP HANA instance is deployed as a standalone installation in the primary Region in one Availability Zone with no target SAP HANA systems to replicate data.

With this pattern, your SAP HANA instance is not highly available. In the event of a complete Region failure, the production SAP HANA instance needs to be built in the secondary Region using

AMI. You can use AWS CloudFormation templates to automate the launch of a new SAP HANA instance. Once your instance is launched, you can then download the last set of backup from Amazon S3 to restore your SAP HANA instance to a point-in-time before the disaster event. You can also use AWS Backint Agent to restore recover your SAP HANA instance and redirect your client traffic to the new instance in the secondary Region.

For disaster recovery that is outside the primary Region, recovery point objective is constrained by how often you store your SAP HANA backup files in your Amazon S3 bucket and the time it takes to replicate your Amazon S3 bucket to the target Region. Your recovery time objective depends on the time it takes to build the system in the secondary Region and restore operations from backup files. The amount of time will vary depending on the size of the database. This pattern is suitable for non-production or non-critical production systems that can tolerate a downtime required to restore normal operations.



Summary

We highly recommend operating business critical SAP HANA instances across two Availability Zones. You can use a third-party cluster solution, such as, Pacemaker along with SAP HANA System Replication to ensure a highly availability setup.

A high availability setup with third-party cluster solution adds to the licensing cost and is still recommended as it can provide high resiliency architecture, a near-zero recovery time and point objectives.

High availability and disaster recovery

AWS provides multiple options for performing disaster recovery and making your SAP HANA systems highly available. This section provides information about these solutions. It also covers the support on AWS platform for native SAP HANA recovery features provided by SAP.

Topics

- Amazon EC2 recovery options
- SAP HANA service auto-restart
- SAP HANA backup/restore
- AWS Backint Agent for SAP HANA
- Amazon EBS snapshots
- Cluster solutions
- Pacemaker cluster
- AWS Launch Wizard for SAP
- AWS Application Migration Service and AWS Elastic Disaster Recovery
- SAP HANA system replication
- Testing SAP HANA high availability deployments
- Troubleshoot high availability SAP HANA deployments

Amazon EC2 recovery options

You can recover your SAP HANA databases running on Amazon EC2 instances with the following recovery options.

Simplified automatic recovery

> The default configuration of an Amazon EC2 instance enables automatic recovery of a supported instance due to hardware failure or a problem requiring the involvement of AWS. Automatic recovery of your Amazon EC2 instance increases the resiliency of your SAP workload. For more information, see Simplified automatic recovery based on instance configuration.

Amazon CloudWatch action based recovery

You can create a `StatusCheckFailed_System` CloudWatch alarm to monitor your Amazon EC2 instance. The system status check may fail due to the following reasons:

- Loss of network connectivity
- Loss of system power
- Software issues on the physical host
- Hardware issues on the physical host that impact network reachability

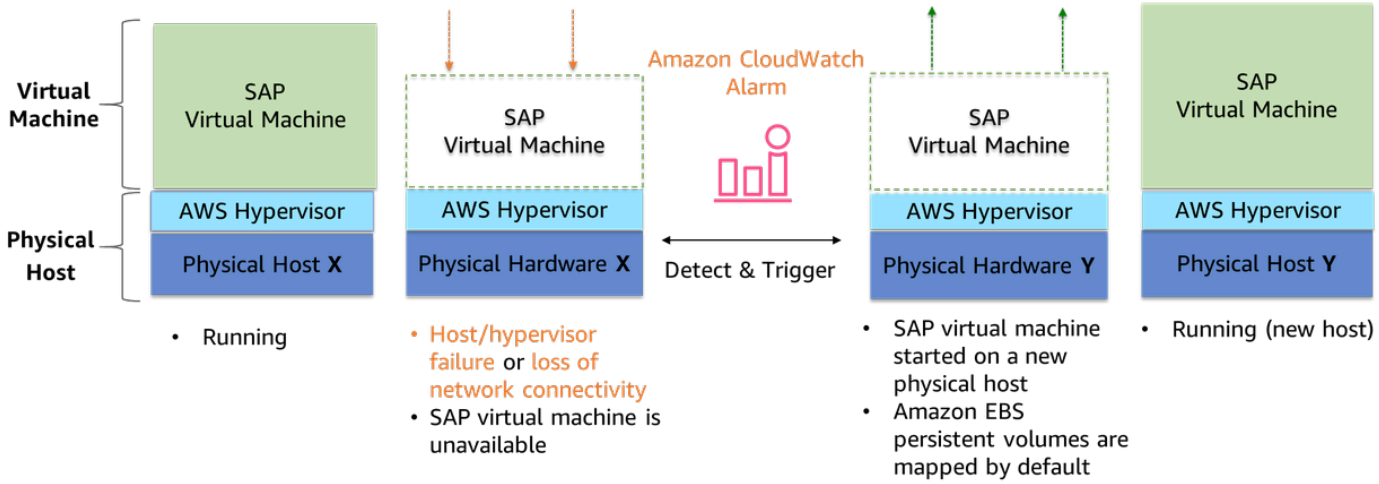
When the CloudWatch alarm detects this failure, recover action is initiated. A recovered instance is identical to the original instance, including the instance ID, private IP addresses, Elastic IP addresses, and all instance metadata. For more information, see Amazon CloudWatch action based recovery.

> **Tip**
>
> When you create the `StatusCheckFailed_System` CloudWatch alarm using AWS Management Console, associate it with Amazon SNS to receive email notifications. Alternatively, you can set up Amazon SNS notifications after creating the alarm. For more information, see Setting up Amazon SNS notifications.

Dedicated host recovery

Dedicated host auto recovery restarts your instances on to a new replacement host when there is a Dedicated Host failure due to system power or network connectivity events. For more information, see Host recovery.

We recommend configuring your Amazon EC2 instances, except instances in a third-party cluster solution, and dedicated hosts with automatic recovery to protect against hardware failure. The following diagram illustrates Amazon EC2 recovery options.



SAP HANA service auto-restart

SAP HANA service auto-restart is a fault recovery solution provided by SAP. SAP HANA has many configured services running all the time for various activities. When any of these services is disabled due to software failure or human error, the service is automatically restarted with the SAP HANA service auto-restart watchdog function. When the service is restarted, it loads all the necessary data back into memory and resumes its operation. SAP HANA service auto-restart solution works the same way on AWS as it does on any other platform. Using SAP HANA service auto-restart along with the section called "Amazon EC2 recovery options" is a robust disaster recovery solution.

SAP HANA backup/restore

Although SAP HANA is an in-memory database, it persists all changes in persistent storage to recover and resume from any failures, such as power outages. If the persistent storage is damaged or any logical errors occur, SAP HANA backups are required to restore the database. The SAP HANA database backup files can be regularly backed up to a remote location for disaster recovery purposes. SAP HANA backup/restore works the same way on AWS as it does on any other platform. For more information, see SAP HANA Administration Guide.

AWS Backint Agent for SAP HANA

AWS Backint Agent for SAP HANA (AWS Backint agent) is an SAP-certified backup and restore application for SAP HANA workloads running on Amazon EC2 instances in the cloud. AWS Backint agent runs as a standalone application that integrates with your existing workflows to back up your SAP HANA database to Amazon S3 and to restore it using SAP HANA Cockpit, SAP HANA Studio,

and SQL commands. AWS Backint agent supports full, incremental, and differential backup of SAP HANA databases. Additionally, you can back up log files and catalogs to Amazon S3. For more information, see AWS Backint Agent for SAP HANA.

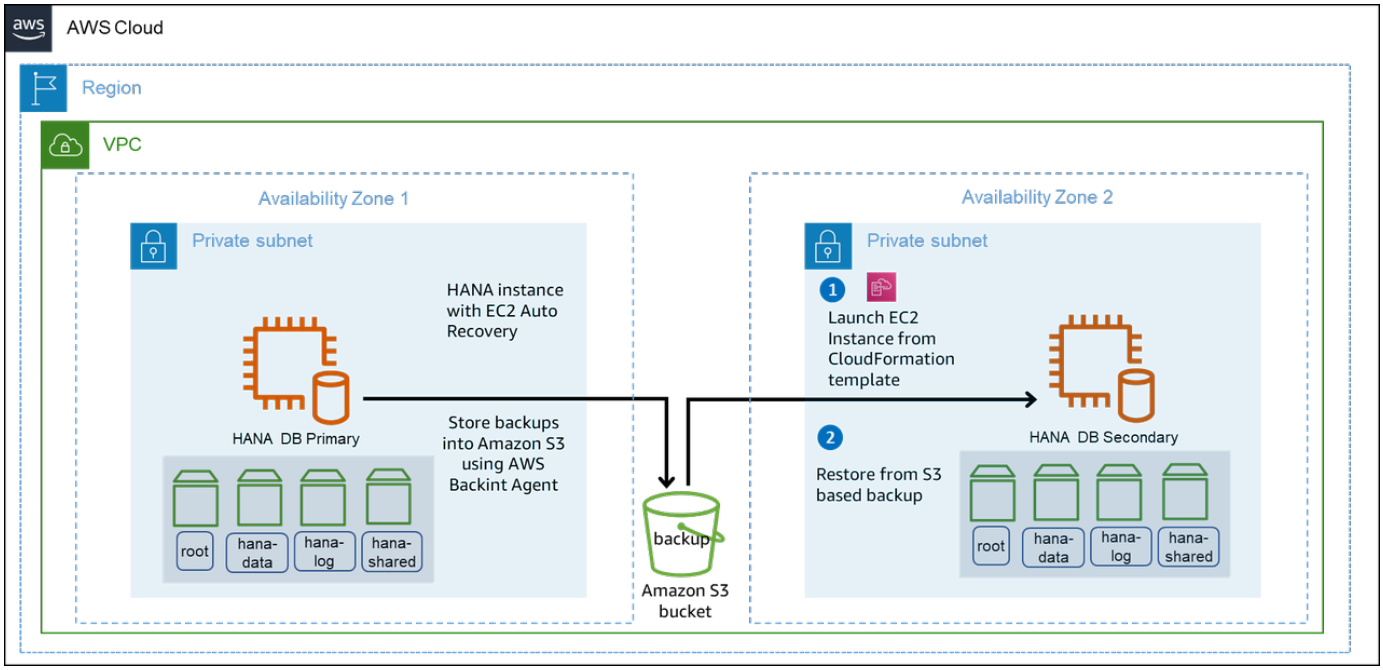
Topics

- Example scenario
- Time to back up
- Recovery time and point objectives

Example scenario

AWS Backint Agent for SAP HANA enables you to make your SAP HANA systems on AWS highly available and ready for disaster recovery. See the following example scenario to learn more.

1. Run your SAP HANA system on Amazon EC2 in Availability Zone 1.
2. Set up the `StatusCheckFailed_System` CloudWatch alarm to automatically recover your Amazon EC2 instance if the system check fails.
    a. Your instance is recovered within the same Availability Zone.
    b. You may not be able to access the instance when the Availability Zone becomes unavailable.
3. Launch a new Amazon EC2 instance using a AWS CloudFormation template in Availability Zone 2. For more information, see Launch an instance from a launch template.
4. Restore your SAP HANA database from Amazon S3 with AWS Backint agent. For more information, see Back up and restore your SAP HANA system with AWS Backint Agent for SAP HANA.
5. Redirect your client traffic to the new SAP HANA system on Amazon EC2 when it is operational.



In this scenario, you avoid the cost of a standby node. Using AWS multi-Availability Zone infrastructure and backup/restore with AWS Backint Agent for SAP HANA, you can quickly resume operations and significantly reduce downtime costs.

The elaborate recovery procedure makes this model suitable for a longer recovery time objective and a recovery point objective that is greater than zero. Your recovery point objective depends on how frequently you store your SAP HANA backup files in Amazon S3.

You can lower your recovery point objective with AWS Backint agent storing your SAP HANA system backups to Amazon S3. Additionally, you can quickly restore from the backup files in Amazon S3 without creating custom scripts to manually copy your SAP HANA backup files to and from Amazon S3.

Time to back up

The time taken to back up and restore your SAP HANA database on Amazon EC2 with AWS Backint agent depends on the configuration of your system. These include Amazon EC2 instance type, Amazon EBS volume type, and database size. The following are the key variables that impact the time taken to back up and restore your SAP HANA system.

- Storage throughput of the underlying Amazon EBS volume supporting the SAP HANA database
- Network throughput supporting the communication channel with Amazon S3

- Available CPU resources on the instance type

Recovery time and point objectives

We recommend you to perform various tests to identify the right system configuration that suit your business recovery time and point objectives. AWS Backint Agent for SAP HANA maximizes the available throughput by parallel processing the back up and restore processes. The recovery time objective is optimized for any given system configuration. For example, with SAP HANA scale-up node on r5.2xlarge, AWS Backint agent was able to upload 551GB of data in 4 minutes and 15 seconds, achieving an overall throughput of 2.16GB/s. Similarly, for a 4 node SAP HANA scale-out running on u6-tb1.metal instances, AWS Backint agent was able to upload 22.86TB of data in 23 minutes, achieving an overall throughput of 16.8GB/s.

Based on our testing, the time taken for restore operations using AWS Backint agent is normally 1.5 to 2 times the back up time. For more information, see Performance tuning.

Amazon EBS snapshots

You can back up your data on Amazon EBS volumes to Amazon S3 by taking point-in-time snapshots. Snapshots provide a fast backup process, regardless of the database size. They are stored in Amazon S3 and replicated across Availability Zones automatically.

Amazon EBS snapshots are incremental by default. Only the delta changes are stored since the last snapshot. Snapshots are also crash consistent. They contain the blocks of completed I/O operations. You can copy the snapshots across AWS Regions or share it with other AWS accounts. You can restore Amazon EBS volumes from the snapshot or create a new volume out of a snapshot in the same or different Availability Zone, and launch Amazon EC2 instances. Amazon EBS snapshots provide a simple and secure data protection solution that is designed to protect your block storage data, such as Amazon EBS volumes, boot volumes, and on-premises block data. For more information, see Amazon EBS snapshots.

Amazon EBS snapshots can also be used to enable disaster recovery, and migrate data across AWS Regions and accounts. Amazon EBS fast snapshot restore enables you to create a volume from a snapshot that is fully initialized at creation. This eliminates the latency of I/O operations on a block when it is accessed for the first time. Volumes that are created using fast snapshot restore instantly deliver all of their provisioned performance. Amazon EBS fast snapshot restore can be enabled on a snapshot while it is being created. It helps you achieve low recovery time objective. For more information, see Amazon EBS fast snapshot restore.

Cluster solutions

SAP HANA workloads on AWS are configured in a highly available and fault tolerant manner at the infrastructure layer. A failure still needs to be managed at the SAP HANA database layer. If a failure is detected at the hardware or software level, you can perform a manual failover process with SAP HANA cockpit, SAP HANA studio, or `hdbnsutil` command line tool. The manual processes can affect the availability of your business processes.

You can also use Python-based API included with SAP HANA to create your own high availability and disaster recovery provider or hooks. You can then integrate these hooks with SAP HANA system replication takeover process to automate tasks such as, restarting the primary node, IP redirection, DNS redirection, and shutdown of dev/QA systems in the secondary node. For more information, see Implementing a HA/DR Provider.

Based on the operating system of your SAP HANA database, you can implement a third-party high availability cluster solution. It can reduce downtime and automate failover steps. The following solutions include a pacemaker framework along with SAP HANA hooks that are certified by SAP and supported on AWS.

- SUSE Linux Enterprise Server (SLES) High Availability Extension (HAE)
- Red Hat Enterprise Linux (RHEL) for SAP high availability

For more information, see SAP HANA on AWS: High Availability Configuration Guide for SLES and RHEL.

Pacemaker cluster

SAP HANA high availability solution based on SAP HANA system replication is automated for failover between primary and secondary SAP HANA instances. The primary and secondary instances are configured together as a pacemaker cluster. The clustering software is at the operating system layer and is integrated with the SAP HANA database using SAP HANA hooks. The clustering software detects and automates the failover. The recovery time can be in minutes or less. For more information, see SAP HANA system replication.

The SAPHanaSR and SAPHANASR-Scale-out solutions from SUSE are based on pacemaker and corosync. These along with dedicated resource agents for SAP HANA are released as part of SLES for SAP Applications. For more information on how to set up a high availability cluster on SLES for SAP Applications on AWS, see High availability cluster configuration on SLES.

The high availability solution from RHEL also provides a pacemaker cluster framework and the resource agents required for automation failover process of SAP HANA system replication. For more information on how to set up a high availability cluster on RHEL on AWS, see High availability cluster configuration on RHEL. The following resources are available from Red Hat.

- Configuring SAP HANA Scale-Up System Replication with the RHEL HA Add-On on Amazon Web Services (AWS)
- Configuring SAP HANA Scale-Out System Replication with the RHEL HA Add-On on Amazon Web Services (AWS)

For automated deployment of SAP HANA system replication using AWS Launch Wizard for SAP, see the section called “AWS Launch Wizard for SAP”.

The pacemaker cluster uses a virtual IP address to connect to the master SAP HANA instance. The virtual IP address is migrated to the secondary instance during failover. The secondary instance is then promoted as active primary for traffic redirection. An overlay IP address is used for the networking configuration on AWS. It is a virtual IP address configured to point to the master SAP HANA instance whether it is on the primary node or secondary node. You can configure overlay IP routing with AWS Transit Gateway or Network Load Balancer. For more information, see SAP on AWS High Availability with Overlay IP Address Routing.

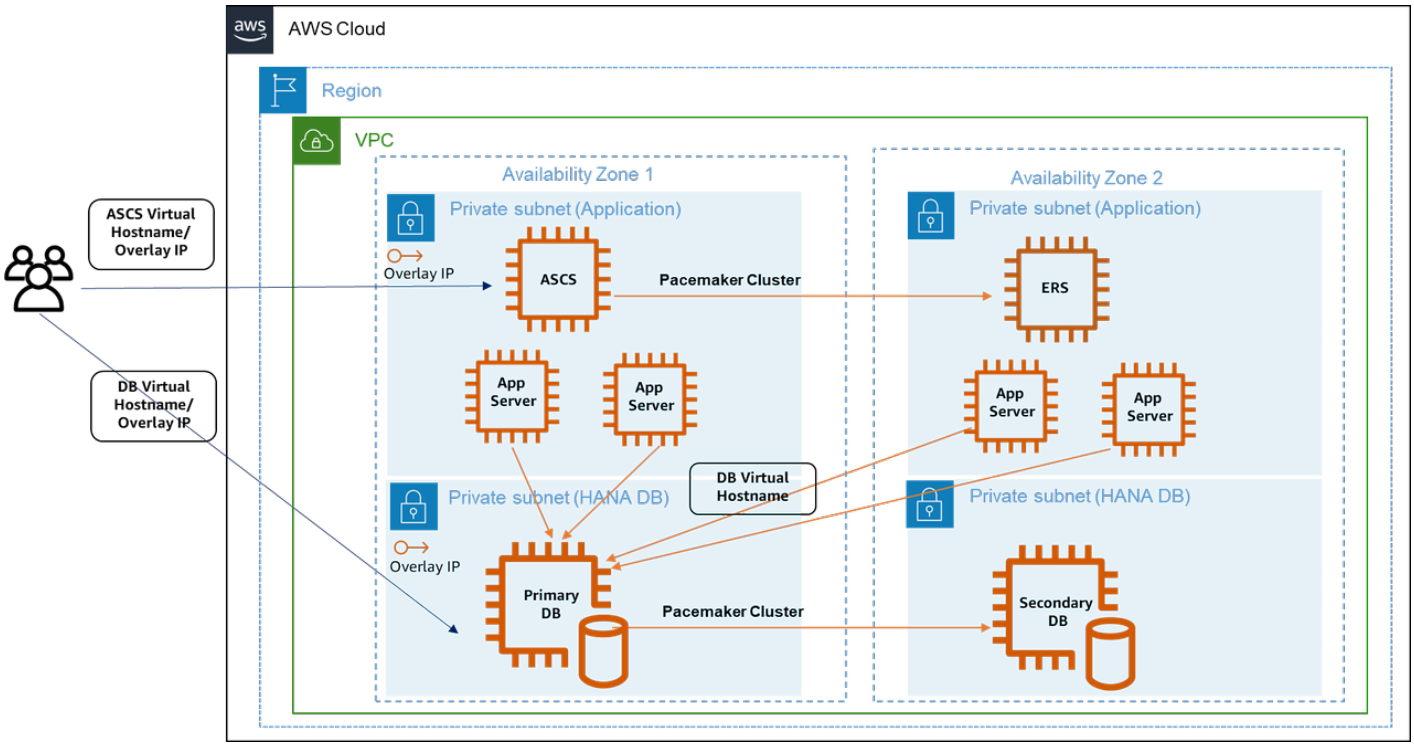
AWS Launch Wizard for SAP

AWS Launch Wizard for SAP offers guided deployment for production-ready applications on AWS with resource sizing, customizable deployments, application configuration, and cost estimation. These tools eliminate the complexity of high availability deployments. For more information, see AWS Launch Wizard for SAP.

AWS Launch Wizard for SAP fast-tracks your SAP HANA deployments on AWS. It requires minimal manual intervention. The following high availability automated deployment patterns for SAP HANA are supported by AWS Launch Wizard.

- **Cross-AZ SAP HANA database high availability setup**: Deploy SAP HANA with high availability configured across two Availability Zones.
- **Cross-AZ SAP NetWeaver system setup**: Deploy Amazon EC2 instances for ASCS/ERS and SAP HANA databases across two Availability Zones, and spread the deployment of application servers across them.

- **SUSE/RHEL cluster setup**: For SAP HANA and NetWeaver on HANA high availability deployments, Launch Wizard for SAP configures SUSE/RHEL clustering when you provide SAP software and specify the deployment of SAP database or application software. Clustering is enabled between the ASCS and ERS nodes for SAP HANA databases across two Availability Zones. See the following diagram.



> **Note**
>
> We strongly recommend that you validate the setup of your environment before using the high availability cluster for deployment. Run tests before deploying an application on your SAP HANA instance set up by Launch Wizard. The tests can ensure that failover and fail-back operations are working properly.

The following table summarizes the deployment patterns supported by AWS Launch Wizard for SAP.

Deployment pattern	Support
SAP HANA database on a single Amazon EC2 instance	Yes

Deployment pattern	Support
SAP NetWeaver on SAP HANA system on a single Amazon EC2 instance	Yes
SAP HANA database on multiple Amazon EC2 instances	Yes
SAP NetWeaver system on multiple Amazon EC2 instances	Yes
Cross-Availability Zone SAP HANA database high availability setup	Yes
Cross-Availability Zone SAP NetWeaver system setup	Yes
SUSE/RHEL cluster setup	Yes

For more information, see Supported deployments and features of AWS Launch Wizard.

AWS Application Migration Service and AWS Elastic Disaster Recovery

We recommend using AWS Application Migration Service to migrate your SAP HANA databases to AWS. For more information, see What is AWS Application Migration Service?

For disaster recovery, we recommend using AWS Elastic Disaster Recovery. It uses block level replication to continuously replicate data from source to target. It helps reduce the infrastructure costs and total cost of ownership. It provides sub-second recovery point objective and recovery time objective of minutes. For more information, see What is AWS Elastic Disaster Recovery?

Cloud Endure, an AWS company, also provides migration and disaster recovery services. Cloud Endure disaster recovery service is a business continuity offering that can be used for SAP and non-SAP workloads.

SAP HANA system replication

SAP HANA system replication is a highly available solution provided by SAP for SAP HANA. SAP HANA system replication is used to address SAP HANA outage reduction due to planned

maintenance, fault, and disasters. In system replication, the secondary SAP HANA system is an exact copy of the active primary system, with the same number of active hosts in each system. Each service in the primary system communicates with its counterpart in the secondary system, and operates in live replication mode to replicate and persist data and logs, and typically load data in the memory. SAP HANA system replication is fully supported on AWS.

Topics

- Architecture patterns
- Replication and operation modes
- Configuration scenarios
- Takeover considerations

Architecture patterns

AWS isolates facilities geographically, in Regions and Availability Zones. A multi-Availability Zone architecture reduces the risk of location failure while maintaining performance.

With single Region multi-Availabilty Zone pattern, the secondary system can be installed in a different Availability Zone in the same AWS Region as the primary system. This provides a rapid failover solution for planned downtime, managing storage corruption or any other local faults.

For disaster recovery, you can use a multi-Region architecture pattern where the secondary system is installed in a different AWS Region. You can choose the Region based on your business requirements, such as data residency limitations for compliance.

For more information, see Architecture patterns for SAP HANA on AWS.

Replication and operation modes

SAP HANA system replication offers the following replication and operation modes that are fully supported on AWS.

Replication modes

Different replication mode options for the replication of redo logs, including synchronous on disk, synchronous in-memory, and asynchronous, can be used depending on your recovery time and point objectives. Synchronous SAP HANA system replication is recommended for multi-Availability Zone deployments, ensuring near zero recovery point objectives. AWS provides low latency and high bandwidth connectivity between the different Availability Zones within a Region.

Asynchronous replication is recommended for system replication across AWS Regions. You can select a multi-Region architecture pattern if your business requirements are not impacted by potential network latency. You must also factor the cost of AWS services in different Regions and cross-Region data transfer.

Operation modes

Different operation modes can be used while registering the secondary SAP HANA system, such as `delta_datashipping`, `logreplay` or `logreplay_readaccess`. The database accordingly sends different types of data packages to the secondary system.

Configuration scenarios

SAP HANA system replication supports the following configuration scenarios that are fully supported on AWS.

Topics

- Active/Passive secondary system
- Active/Active (read enabled) secondary system
- SAP HANA secondary time travel
- SAP HANA replication scenarios in AWS
- SAP HANA multi-tier replication
- SAP HANA multi-target replication

Active/Passive secondary system

In this scenario, system replication does not allow read access or SQL querying on the secondary system until the active system is switched from the current primary to the secondary system by takeover. The secondary system acts as a hot standby with the `logreplay` operation mode.

Active/Active (read enabled) secondary system

In this scenario, system replication supports read access on the secondary system. It requires the `logreplay_readaccess` operation mode.

SAP HANA secondary time travel

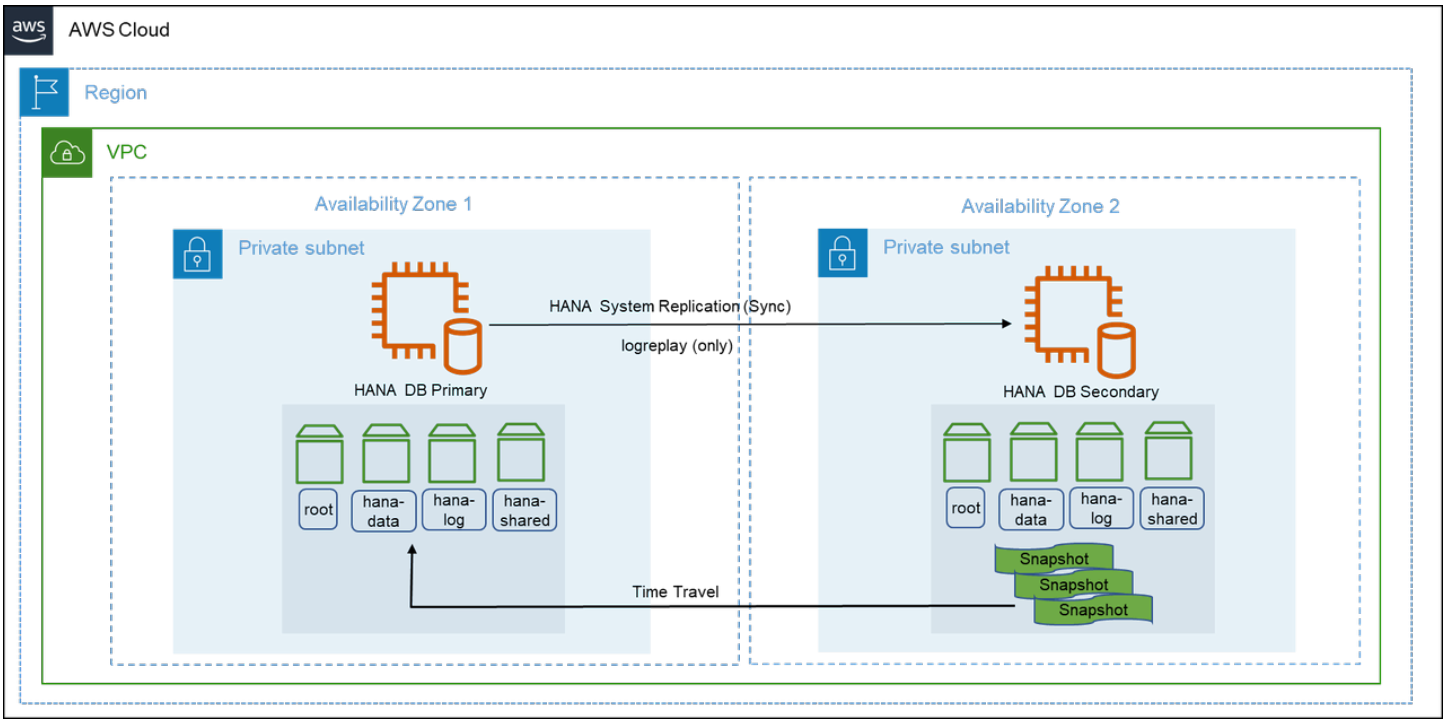
In this scenario, you can gain access to the data that was deleted in the primary system or intentionally delay the `logreplay` in secondary system to read older data while the replication

continues on the secondary system. You can recover from logical errors and have a faster recovery. You can use the secondary time travel configuration only with the `logreplay` operation mode.

You must properly size the secondary time travel memory instance for replication. The minimum memory requirement is to use row store size, column store memory size, and 50 GB of memory with preload for `logreplay` operation mode. For more information, see 1999880 - FAQ: SAP HANA System ReplicationThe following parameters are require for setup.

- `global.ini/[system_replication]/timetravel_max_retention_time` parameter must be configured on the secondary system. This parameter defines the time period to which the secondary system can be brought back in the past.
- `global.ini/[system_replication]/timetravel_snapshot_creation_interval` is an optional parameter. You can adjust the secondary system's snapshot creation. The secondary system can start retaining logs and snapshots.

The following diagram shows the SAP HANA secondary time travel configuration scenario.



SAP HANA replication scenarios in AWS

In a two-tier SAP HANA system replication, deployment on AWS is optimized based on performance or cost. For the fastest takeover time, use a secondary instance with the same size as the primary instance. This is a performance optimized deployment. A cost optimized deployment

can reduce overall costs with a compromise on the recovery time objective. Cost optimized scenarios are also referred to as pilot light disaster recovery. For more information, see Rapidly recover mission-critical systems in a disaster.

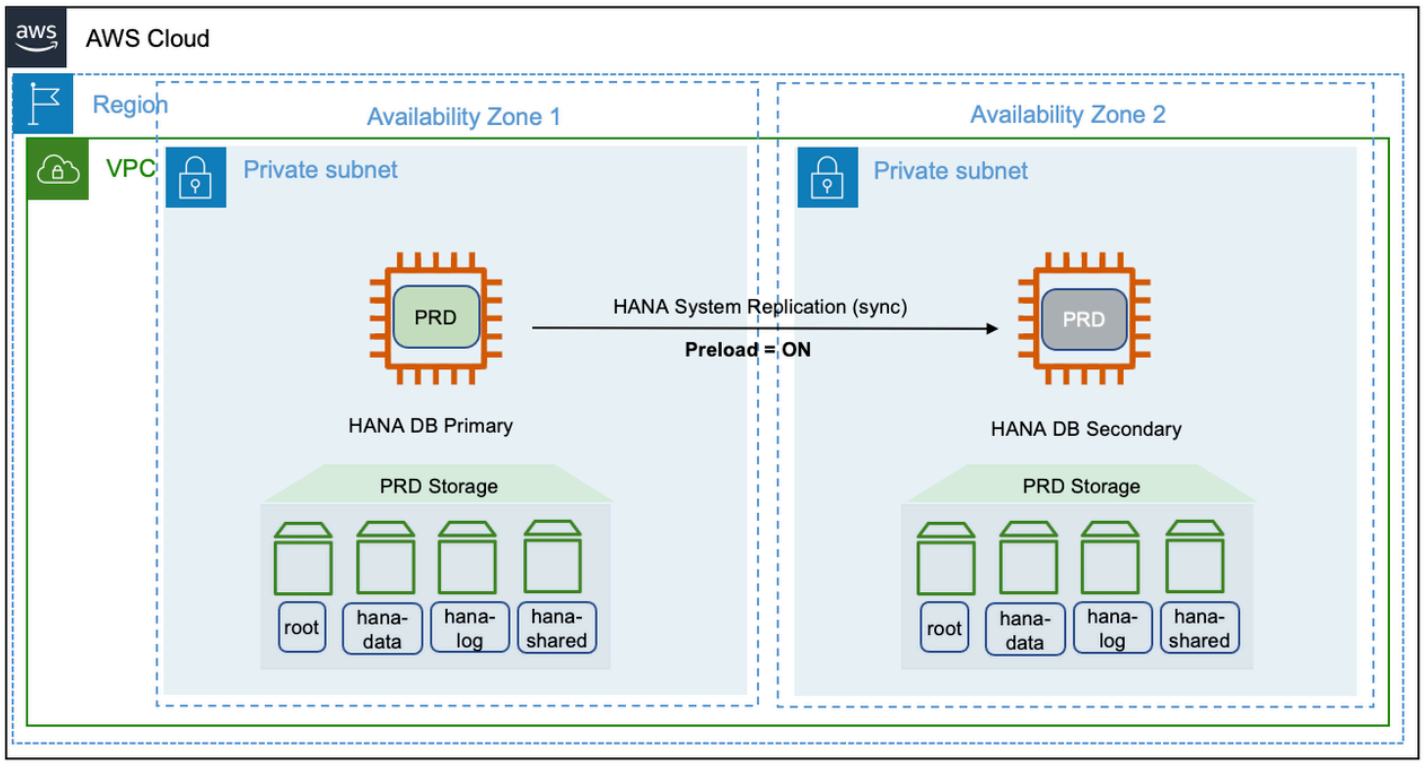
Scenarios

- Performance optimized
- Cost optimized

Performance optimized

SAP HANA database systems that are critical to business continuity require a near-zero recovery time objective during planned and unplanned outages. You can optimize performance with a secondary instance of the same size as primary. This configuration can accommodate preloaded column tables in-memory, and synchronous system replication. We do not recommend hosting your SAP HANA instances across AWS Regions in this setup. This is to avoid latency while replicating in a synchronous mode. This deployment protects your critical SAP HANA systems against failure of an Availability Zone, a rare occurrence.

You can set up a third-party cluster solution along with SAP HANA system replication to detect failure and automate failover. For more information, see Pacemaker cluster. The following diagram shows a performance optimized deployment.



Cost optimized

You can reduce costs by using a smaller or shared secondary SAP HANA system. In the smaller secondary option, the infrastructure is initially sized smaller than the primary and resized before performing a takeover. In the shared secondary option, the unused memory on the secondary system is used by a non-production or sacrificial instance.

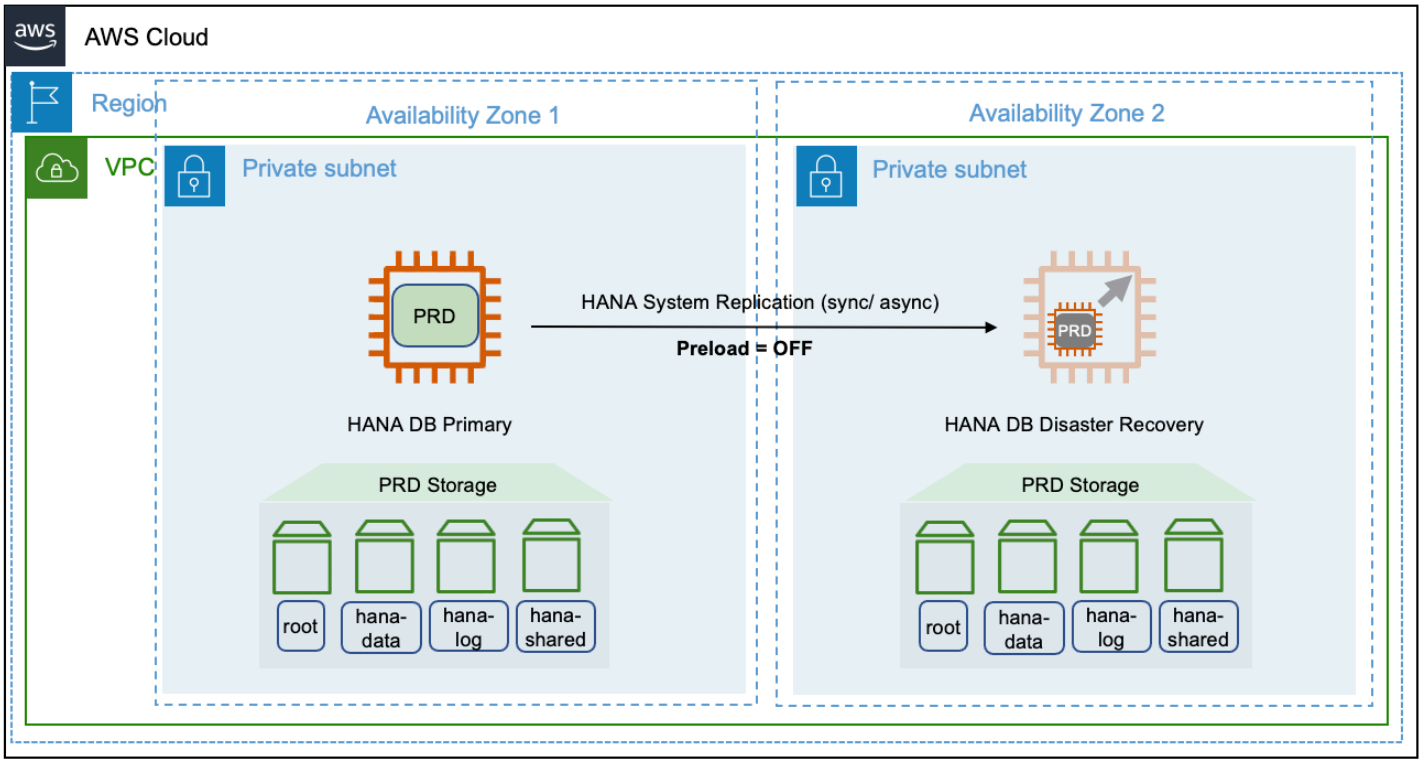
The `preload_column_tables` parameter is set to *false* for both, smaller and shared secondary options. You can find this parameter in the `global.ini` file located at `(/hana/shared/<SID>/global/hdb/custom/config`. Setting the parameter as *false* enables the secondary system to operate with reduced memory. However, the default value of the `preload_column_tables` is *true*.

> **Note**
>
> Before performing a takeover in a cost optimized deployment, you must set the `preload_column_tables` parameter to its default value of *true* and restart the SAP HANA system.

The size of your SAP HANA database impacts the time taken to load the column tables into main memory. This affects your overall recovery time objective. You can use SQL scripts to get a rough estimate of the minimum memory required for these tables. Refer to the *HANA_Tables_ColumnStore_Columns_LastTouchTime* section in SAP Note 1969700 – SQL Statement Collection for SAP HANA for more information.

Smaller secondary

The following diagram shows the deployment of a smaller secondary SAP HANA system in a different Availability Zones within the same AWS Region.



This deployment is also possible across multiple AWS Regions. We recommend using the asynchronous mode while replicating across Regions. Note that when you resize the secondary system before a takeover, there is no reserved capacity. The requirement of a production sized instance is subject to the current availability in your Availability Zone.

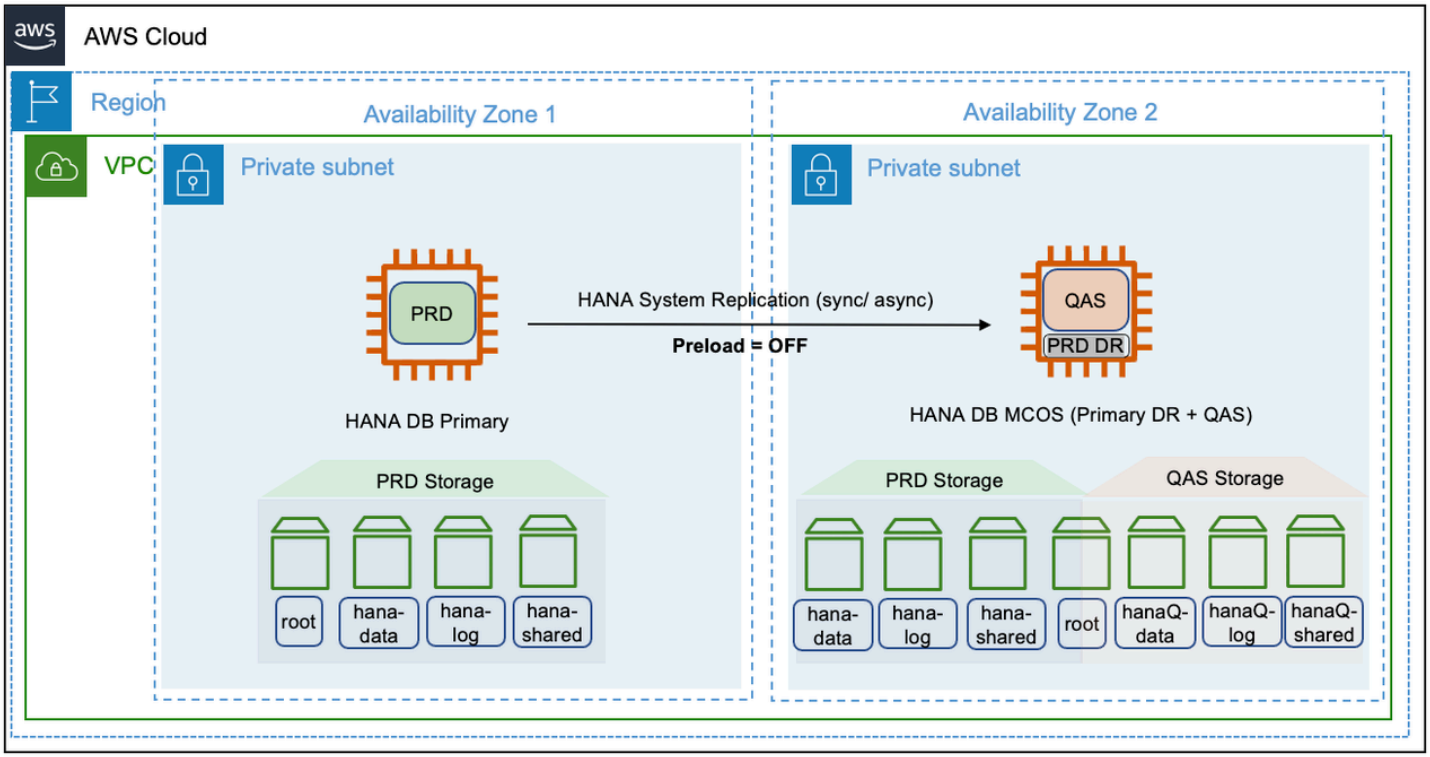
Shared secondary

Multiple components one system (MCOS) model is a common use case of the shared secondary deployment option. You can operate an active quality instance along with the secondary instance on the same host. This setup requires additional storage to operate the additional instances. During

a takeover, the instance with lower priority can be shutdown to make the underlying host resources available for production workloads.

You must set the `global_allocation_limit` for all instances running on the site. This ensures that no one instance with the `global_allocation_limit` set to `0` occupies the entire memory available on the host. For more information, see SAP Note: 1681092 – Multiple SAP HANA systems (SIDs) on the same underlying server(s).

The following diagram shows a shared secondary deployment on AWS.



Sizing considerations for cost optimized deployments

Despite disabling the preload of column tables, the actual memory usage on the secondary host is also dependent on the operation mode of system replication. For more information, see SAP Note: SAP Note : 1999880 – FAQ: SAP HANA System Replication.

Although the `preload_column_tables` parameter is set to *false*, the `logreplay` operation mode is also a contributing factor to the memory size. You should consider the size of column store tables with data modified in the previous 30 days from the current date of evaluation.

The `logreplay` operation mode may not be able to provide true cost optimization.
The `delta_datashipping` operation mode can be an alternative. However, the

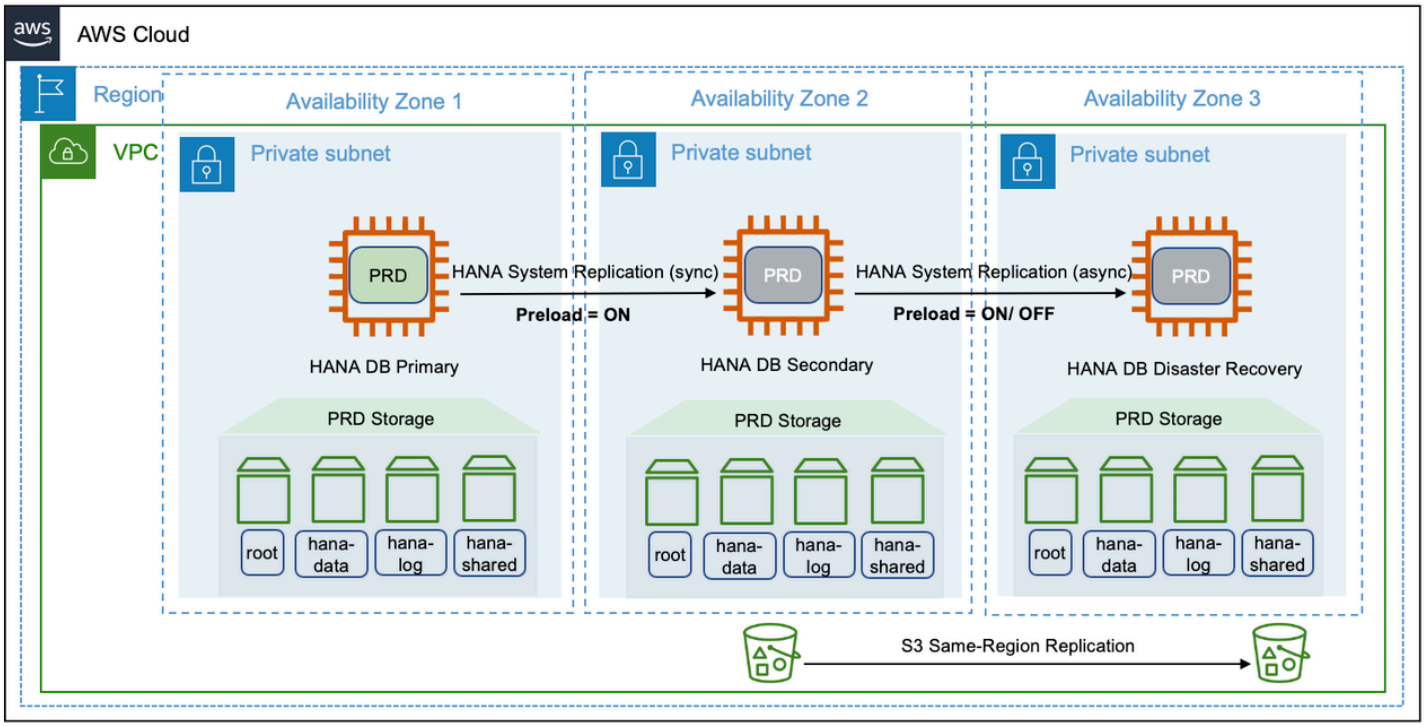
`delta_datashipping` has limitations. It can include a higher recovery time and an increase demand for network bandwidth between the replication sites. If your business requirements can afford higher network bandwidth and relaxed recovery times, `delta_datashipping` mode can be a viable option.

The potential cost savings is higher with larger database instances. The memory footprint on the secondary system has a minimum requirement of row store memory and buffer requirements, even for smaller database instances. Calculation the memory requirement and accordingly setting the `global_allocation_limit` is an iterative process. The column store demand for delta merge grows with the growing size of the production database. Therefore, memory allocations for all hosts on a site should be monitored periodically, and after mass data loads, go-lives, and SAP system specific lifecycle events.

SAP HANA multi-tier replication

This configuration scenario is suitable if you are looking for both, high availability and disaster recovery. This setup provides a chained replication model where a primary system can replicate to only one secondary system at any given point of time. For more information, see Setting Up SAP HANA Multi-tier System Replication.

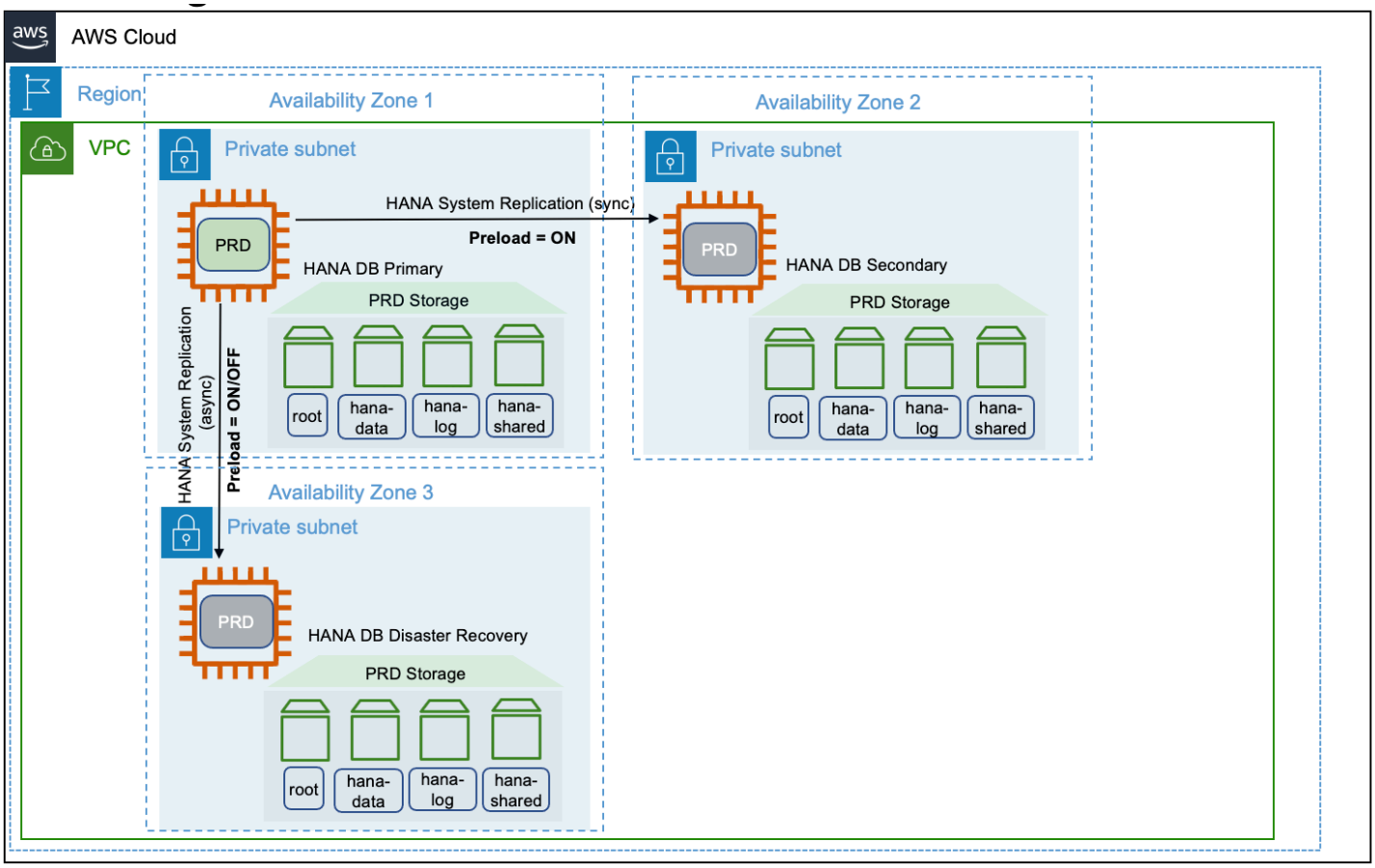
In this scenario, there can be a mix of performance and cost deployment options. The primary and secondary system can be deployed in a high availability setup using a pacemaker cluster. The tertiary or disaster recovery system can be a cost optimized deployment. An active non-production instance can run on the same node, as a multiple components one installation model. This setup is shown in the following diagram.



SAP HANA multi-target replication

In SAP HANA multi-tier scenario, replication happens sequentially, from primary to secondary system, and then from secondary to tertiary system. Starting with SPA HANA 2.0 SPS 03, SAP HANA provides multi-target system replication configuration for a single primary system to replicate to multiple secondary systems. For more information, see SAP HANA Multitarget System Replication.

The following diagram shows a multi-tier target replication configuration on AWS.



Replication mode

The primary, secondary, and tertiary systems can be placed on different Availability Zones within the same or across AWS Regions. Apart from the replication modes supported by SAP, SAP HANA systems deployed across different AWS Regions must choose the async mode of replication due to latency requirements. To see the replication modes supported by SAP, see Supported Replication Modes between Sites.

Operation mode

It is not possible to combine `logreplay` and `delta_datashipping` operation modes in a multi-tier or multi-target system replication. For example, if the primary and secondary systems use `logreplay` for system replication, then `delta_datashipping` cannot be used between the secondary and tertiary systems or vice-versa.

The `logreplay` operation mode is only supported in a multi-target system replication scenario. To implement a high availability pacemaker cluster solution along with multi-target replication, check the relevant resources from SUSE and RHEL.

The `logreplay_readaccess` operation mode is supported on an Active/Active (read enabled) configuration with multi-target system replication. However, in a multi-tier replication, only the secondary system can be used for read-only capability, and cannot be extended to the tertiary system.

Disaster recovery

The multi-target system replication offers automated re-registration of the secondary systems to a new primary source in case of failure on the primary. You can set this automation with the `register_secondaries_on_takeover` parameter. For more information, see Disaster Recovery Scenarios for Multitarget System Replication.

Takeover considerations

When there is a need for SAP HANA system replication takeover, you must trigger it in your secondary system by following the standard SAP HANA takeover process. You must decide if you want to wait for your system to be recovered in the primary Availability Zone before a takeover, if you have enabled automatic recovery. For more information, see SAP OSS Note 2063657.

Topics

- Client redirect options
- Client redirection for Active/Active high availability scenario

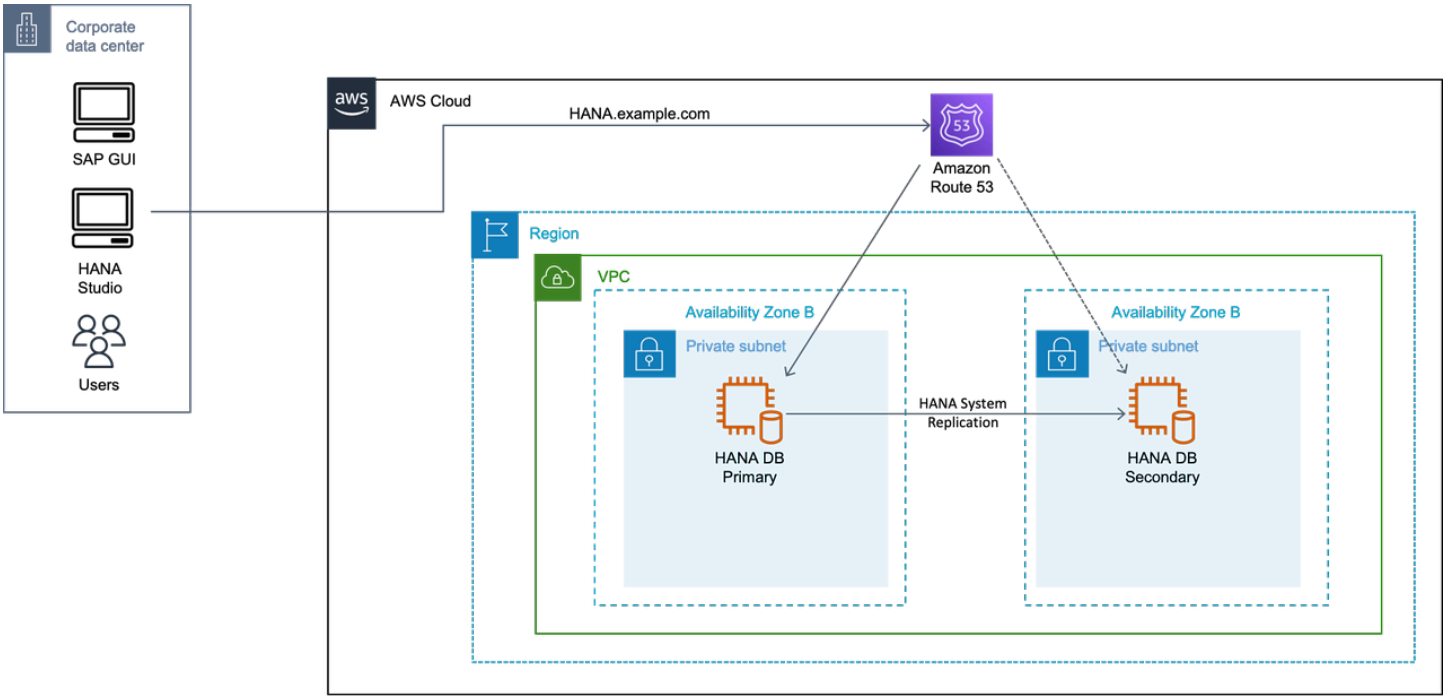
Client redirect options

In almost all scenarios, failover of the SAP HANA system alone does not guarantee business continuity. You must ensure that your client applications, such as NetWeaver application server, JDBC, ODBC, etc are able to connect to the SAP HANA system after the failover. Connection can be reestablished by redirecting your network-based IP or DNS. IP redirection can be processed faster in a script as compared to synchronizing changes in DNS entries over a global network. For more information, see the *Client Connection Recovery* section in the SAP HANA Administration Guide.

DNS redirection

You must set the IP address of the secondary system in the host name for a network-based DNS redirection. The DNS records must point to the active SAP HANA instance in the same Availability Zone. You can use a script as part of the takeover to modify the DNS records. You can also make the change to DNS records manually.

A vendor proprietary solution is required to modify DNS records. With AWS, you can use Amazon Route 53 to automate the modification of DNS records with AWS CLI or AWS API. For more information, see Configuring Amazon Route 53 as your DNS service.



IP redirection

With network-based IP redirection, a virtual IP address is assigned to the virtual host name. In case of a takeover, the virtual IP unbinds from the network adapter of the primary system and binds to the network adapter on the secondary system.

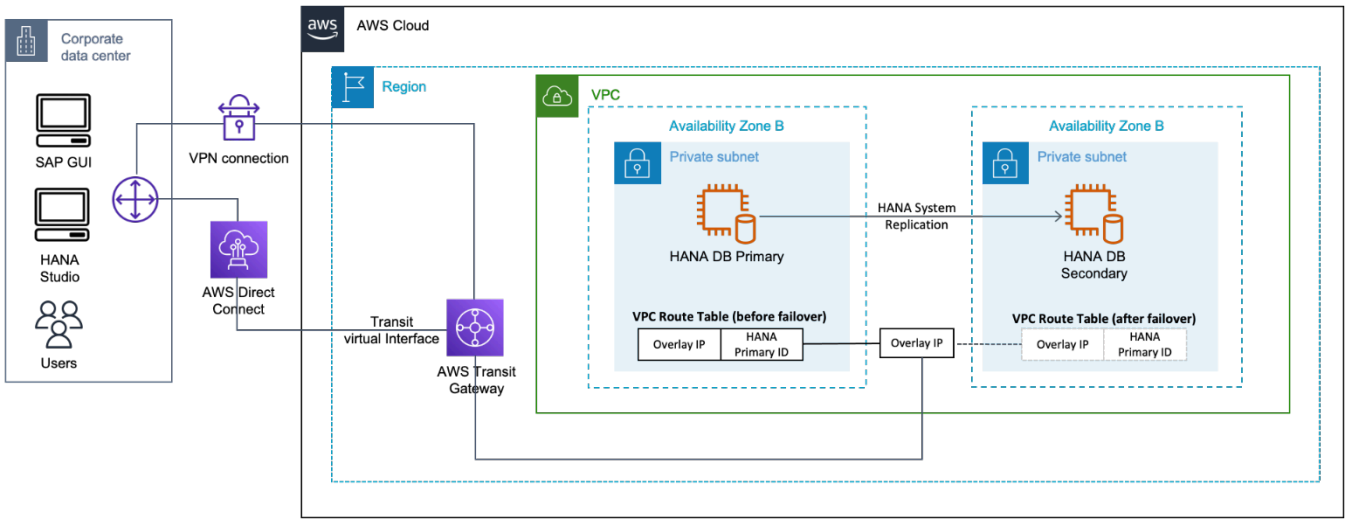
Amazon VPC setup includes assigning subnets to your primary and secondary nodes for the SAP HANA database. Each of these configured subnets has a classless inter-domain routing (CIDR) IP assignment from the Amazon VPC which resides entirely within one Availability Zone. This CIDR IP assignment cannot span multiple zones or be reassigned to the secondary instance in a different Availability Zone during a failover. For more information, see How Amazon VPC works.

Topics

- AWS Transit Gateway
- Network Load Balancer

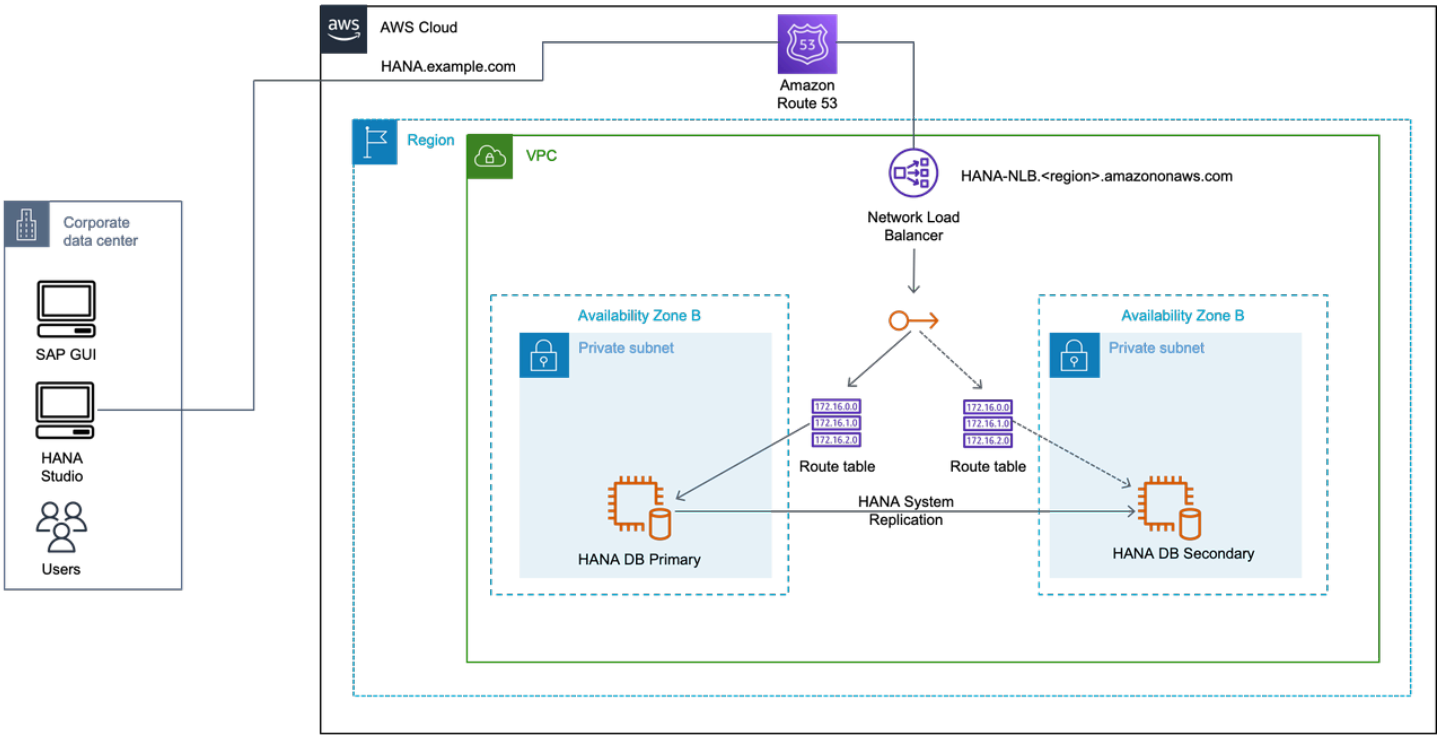
AWS Transit Gateway

With Transit Gateway, you use route table rules which allow the overlay IP address to communicate to the SAP instance without having to configure any additional components, like a Network Load Balancer or Route 53. You can connect to the overlay IP from another VPC, another subnet (not sharing the same route table where overlay IP address is maintained), over a VPN connection, or via an AWS Direct Connect connection from a corporate network. For more information, see What is a Transit Gateway?



Network Load Balancer

If you do not use Amazon Route 53 or AWS Transit Gateway, you can use Network Load Balancer for accessing the overlay IP address externally. The Network Load Balancer functions at the fourth layer of the Open Systems Interconnection (OSI) model. It can handle millions of requests per second. After the load balancer receives a connection request, it selects a target from the Network Load Balancer target group to route network connection request to a destination address which can be an overlay IP address. For more information, see What is a Network Load Balancer?



Client redirection for Active/Active high availability scenario

You use the additional overlay IP address for your secondary read-only system in this configuration. The IP address binds to the active secondary system as part of the cluster failover. The DNS records for the secondary system can be updated manually or by using a script during takeover.

An additional Network Load Balancer needs to be created for load balancing your secondary system.

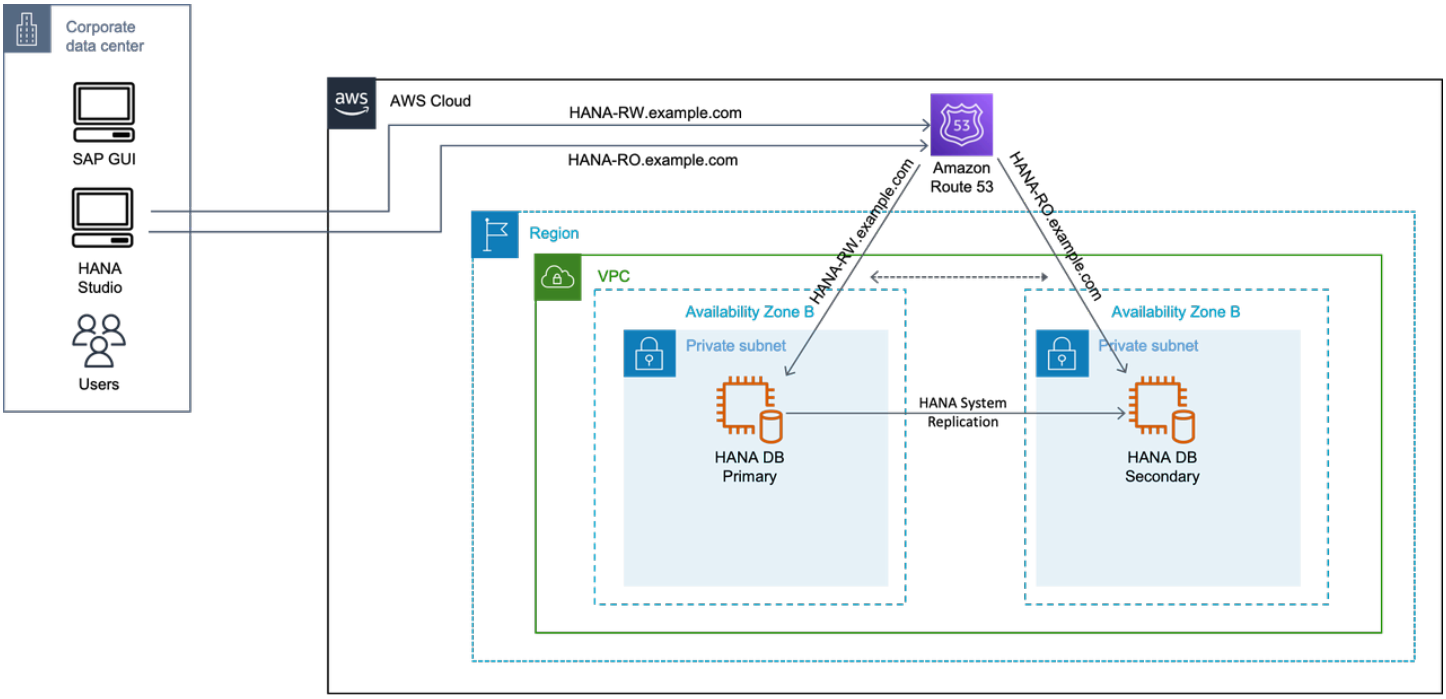
With Transit Gateway, you use an overlay IP address on your secondary system to connect with Amazon VPC and subnet where your secondary system will run.

Topics

- Active/Active scenario with DNS
- Active/Active scenario with AWS Transit Gateway
- Active/Active scenario with Network Load Balancer

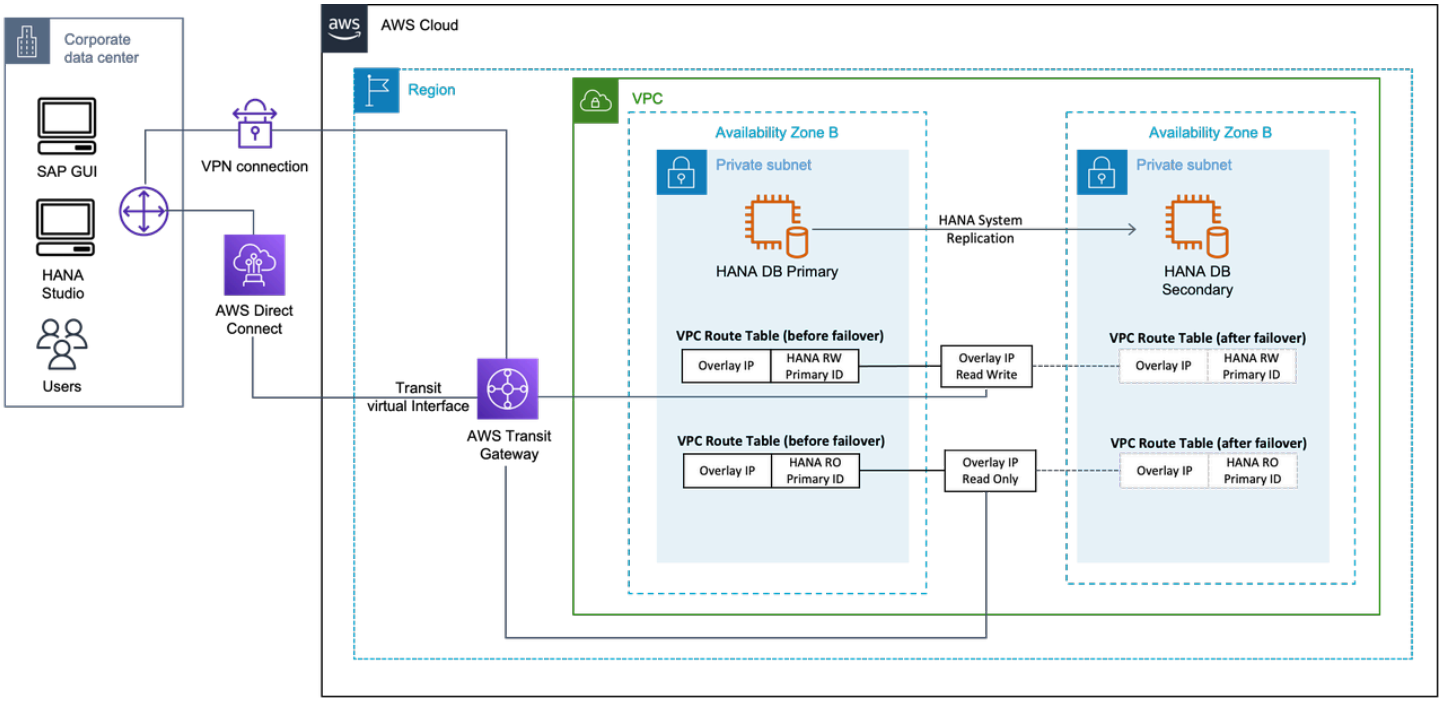
Active/Active scenario with DNS

In this scenario, you use two DNS records for SAP HANA read/write primary instance and SAP HANA read only secondary instance. In case of failover, the modification of DNS records can be automated or manual.



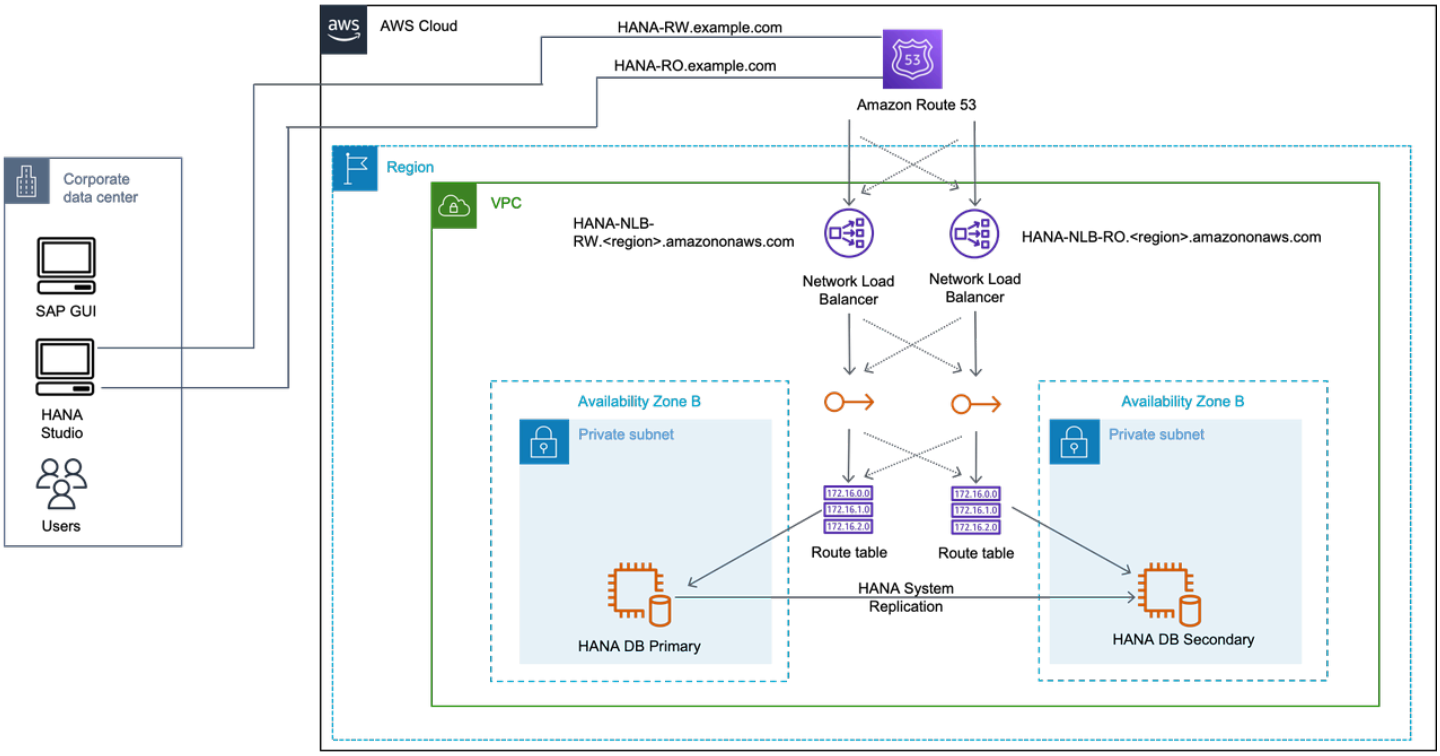
Active/Active scenario with AWS Transit Gateway

In this scenario, two overlay IP addresses for SAP HANA read/write primary instance and SAP HANA read only secondary instance. In case of failover, the route table is adjusted in its Availability Zone, and Transit Gateway reroutes the connections to these IP addresses. This applies to both overlay IP addresses.



Active/Active scenario with Network Load Balancer

In this scenario, two overlay IP addresses for SAP HANA read/write primary instance and SAP HANA read only secondary instance. In case of failover, the route table is adjusted in its Availability Zone, and Network Load Balancer for the read/write or read only endpoint points to the overlay IP address in its Availability Zone. This applies to both overlay IP addresses.



Testing SAP HANA high availability deployments

This section covers failure scenarios for backup, testing guidance and considerations for high availability and disaster recovery solutions, and disaster recovery mock exercise.

Topics

- Failure scenarios for backup and recommendations
- Testing guidance and considerations
- Disaster recovery mock exercise guidance

Failure scenarios for backup and recommendations

The following table provides an overview of different failures scenarios for the SAP HANA system, the risk of occurrence, potential data loss, and maximum outage. It is important to determine which failure scenario will require a recovery from backup. Note that the granularity of the scenarios, classification, and impact will vary depending on your requirements and architecture.

Data protectio	Failure scenarios	Comparati ve risk of occurrence	Potential data loss	Maximum outage	Impact

n/disaster recovery					
No high availability	Resource exhausted or compromised (high CPU utilization/ file system full/out of memory/storage issues)	Medium	~o (uncommitted transactions)	Avoidable	Region
High availability	Single point of failure (database)	Medium	~o (uncommitted transactions)	Time to detect failure and failover (automated)	Region
	Availability Zone/network failure	Low	~o (uncommitted transactions)	Time to detect failure and failover (automated)	Region
	Core service failure	Low	o	Dependent on failure	Region
Disaster recovery	Corruption/accidental deletion/ malicious activities/ faulty code deployment	Low	Last consistent restore point before failure	Time to detect failure and failover (manual)	Cross-Region

	Region failure	Very low	Replication delay	Time to detect failure and make a decision to invoke disaster recovery and takeover	Cross-Region

For SAP HANA systems without high availability implementation, the core critical components of failure for an instance at the infrastructure level are compute, memory, and storage. For compute or memory related failure scenarios, it could be a processor, memory failure or resource exhaustion, such as high CPU utilization, out of memory etc. We recommend the following approaches for recovery of SAP HANA system, in case of CPU or memory issue.

- Use Amazon EC2 automatic recovery or host recovery to bring the SAP HANA system up on new host. For more information, see Amazon EC2 recovery options.
- Create a full backup of your Amazon EC2 instance using Amazon Machine Image along with a snapshot of individual Amazon EBS volumes. Use this as golden image to launch a new instance in case of any failure.
- Implement a monitoring solution, such as Amazon CloudWatch to prevent failure scenarios involving CPU or memory resource exhaustion.

  You can resize or upgrade your Amazon EC2 instance to support a greater number of CPU cores or instance memory size. For more information, see Change the instance type.

For SAP HANA system, Amazon EBS volumes can be the primary storage for operating `root`, `data` or `log` volumes. There can be different failure scenarios, such as Amazon EBS volume failure, disk corruption, accidental deletion of data, malicious attack, faulty code deployment etc. We recommend the following options to safeguard your data.

- Use SAP HANA backup and restore to back up your SAP HANA database to Amazon S3 using AWS Backint Agent for SAP HANA.
- Take regular Amazon Machine Images/Amazon EBS snapshots of your servers on a regular basis.

Amazon S3 single-Region replication should be configured to protect against data loss in the same Region. For disaster recovery, we recommend using Amazon S3 cross-Region replication to save backups/snapshots in the secondary Region, in the event of a failure in the primary Region. You can restore the SAP HANA system in the secondary Region from the last set of backups/snapshots. Here the recovery point objective depends on the last consistent restore point before the failure.

Testing guidance and considerations

Pacemaker cluster can help you perform planned downtime tasks, such as patching SAP HANA database, by automating failover and failback of cluster members. Various unplanned or fault situations can arise during SAP HANA database operations. These can include but are not limited to the following.

- Hardware failures, such as memory module failures on bare-metal instances
- Software failures, such as process crashes due to out-of-memory issues
- Network outage

Most of these failure scenarios can be simulated using SAP HANA database and Linux operating system commands. The scenarios for AWS infrastructure can also be simulated on AWS Management Console or by using AWS APIs. For more information, see AWS APIs.

High availability cluster solutions constantly monitor the configured resources, to detect and react as per pre-defined thresholds, dependencies, and target state. SAP HANA pacemaker cluster configuration can vary, depending on factors such as, size of the database, application availability, and others. The following are some of the considerations for testing SAP HANA high availability deployments based on pacemaker cluster.

- SAP HANA high availability installation based on a pacemaker cluster must undergo planned and unplanned outage scenarios to verify stability.
- You can perform initial cluster tests without loading business data into SAP HANA database. The first iteration of testing verifies if the cluster behaves as intended during various fault scenarios. In this iteration, you can also perform initial cycle of test cases and find out about any product or configuration issues.
- The second iteration of testing can be performed with production size data loaded into the SAP HANA database. The main objective is to tune the cluster monitors for effective timeouts.

Large SAP HANA databases take more time to start and stop. If they are hosted on AWS bare-metal instances, the time taken to reboot can be longer. As these factors can impact the cluster behavior, the cluster timeout values have to be tuned accordingly.

- An SAP Application can have many single point of failures, and SAP HANA database is one of them. The availability of an SAP application is dependent on all single point of failures being resilient to failure situations. Include single point of failures in overall testing. For example, validate an AWS Availability Zone failure where both SAP Application/NetWeaver stack component (ASCS) and SAP HANA database are deployed in the same Availability Zone. The cluster solution must be able to failover pre-configured resources and the SAP application must be restored on the target Availability Zone.
- Test cases that comprise of planned and unplanned downtimes should be tested as a minimum validation. You can also include scenarios where single point of failures was observed in the past. For instance, a year-end consolidation jobs testing the instance memory limits, leading to database crashes.

  For SAP HANA high availability deployment with pacemaker cluster on **SLES** on AWS test cases, see Testing the cluster.

  For SAP HANA high availability deployment with pacemaker cluster on **RHEL** on AWS test cases, see Testing the cluster.
- Pacemaker cluster solution require virtual IP address configuration for client connections. With virtual IP addresses, the actual hardware where the SAP workloads run remain transparent to client applications. There is a seamless failover of connections in the event of a failure. You must verify that all the intended SAP or third-party interfaces are able to connect to the target SAP application post failover.

  You can start by preparing a client connections or interfaces list that includes all critical connections to the target SAP system. Identify the modifications required in your connection configuration to point to a virtual IP address or load balancing mechanism. During testing, each connection must be validated for connectivity, time taken to detect new connection, and loss of locks set by the application, before the cluster performs a failover. For more information, see Client redirect options.
- If you have high availability and disaster recovery on your SAP HANA workloads, you must take additional steps to perform cluster validations. A pacemaker cluster only has visibility into its cluster members(primary and secondary). The cluster software does not control disaster recovery operations (tier-3/tertiary).

When a failover is triggered in a multi-tier SAP HANA system replication setup and the secondary database takes over the role of primary, the replication continues on the tertiary system. However, once the fault with the original primary system is rectified and the system is made available again, manual intervention are be required to complete the reverse replication requirements from the new primary SAP HANA database to the original primary. These manual steps are needed for SAP HANA databases that do not support (lower than SAP HANA 2.0) multi-target replication. For more information, see SAP HANA multi-target replication.

After performing failback to the original primary, some manual steps have to be performed to re-enable the replication on the tertiary site. It is very important to validate the flow of these steps and the time taken for services to startup during each testing scenario before releasing the systems for productive usage.

- SAP HANA system replication can be configured in an Active/Active configuration. This configuration utilizes the secondary hardware for read-only purpose. The supported products include SAP S/4 HANA, BW on HANA, and BW4/HANA.

  SLES and RHEL support an Active/Active SAP HANA system replication setup using pacemaker cluster. Depending on the operating system version, additional steps may be required to set up an Active/Active configuration using the pacemaker cluster.

  The testing scenarios will vary to incorporate additional validation of failover and failback behavior of read-only virtual IP and the respective client connections being able to connect post failover and failback.

Disaster recovery mock exercise guidance

Your disaster recovery setup must be validated by performing a manual mock exercise. With a mock disaster recovery exercise, you can verify the recovery point and time objectives and the steps for invoking a disaster recovery. You can also identify ownership and tasks for various teams involved, and make a detailed plan of routing client connections as well as establishing connections to hub systems and third-party connections.

Invoking a disaster recovery system requires detailed planning and support from other teams, such as a dedicated network operations team. It also needs agreement on the performance requirements once these systems are started in the disaster recovery Region.

Disaster recovery mock exercise also involves validating cross-Region replication of Amazon EFS, Amazon S3, and other AWS services that are part of the overall disaster recovery plan. Any sync jobs scheduled for cross-Region replication of these services (for instance, Amazon EFS) must be adapted or paused. They tend to overwrite any new content created on the disaster recovery site. You might also have to perform tasks on the networking layer for SAP and third-party systems to inter-communicate in the disaster recovery Region, and for client connectivity. Post-recovery tasks, such as applying for new licenses must also be performed. End-user communication requirements along with guidance on how to connect to SAP HANA systems on a disaster recovery site must also be considered.

An in-depth disaster recovery mock exercise also involves testing the steps to resume SAP HANA systems on the original site (primary Region or Availability Zone). This task must be planned carefully to avoid any data loss. The steps for replication vary on a two-tier and multi-tier SAP HANA system replication setup. It requires an async replication mode.

Functional and technical teams must verify the SAP HANA systems for potential data loss before invoking a disaster recovery and failing back to the original site. With a mock disaster recovery exercise, you can also prepare standard operating procedures for business continuity, saving time during a real disaster and minimizing possible data loss.

Troubleshoot high availability SAP HANA deployments

This section provides guidance for troubleshooting SAP HANA high availability deployments.

A healthy status of SAP HANA system replication is a foundational requirement for the cluster solution to maintain stability. If SAP HANA system replication doesn't have any dependencies on cluster solution, it can be independently verified using SAP Note 2518979 - HANA : how to check system replication status.

For manual deployment, there must not be any underlying issues within the cluster member systems for their continuous system replication and takeover procedures. This must be independently verified before integrating a cluster solution for automation. SAP HANA system replication depends on various factors for functioning smoothly. To troubleshoot any issues, see Troubleshoot System Replication.

Alternatively, you can use guided troubleshooting provided by SAP. For more information, see SAP HANA Troubleshooting. You can also chat with experts or open an incident with SAP. For a speedy resolution, collect the relevant SAP HANA logs as per SAP Note 2934640 - HANA and Replication

- Collecting Support Data for Replication / Network related Tickets. The *fullsysteminfo-dumps* log must be collected from all the cluster member systems for a complete analysis.

For troubleshooting issues with AWS Launch Wizard, see Troubleshoot AWS Launch Wizard for SAP.

For troubleshooting issues with high availability SAP HANA setup on SLES, see Indepth HANA Cluster Debug Data Collection (PACEMAKER, SAP).

For troubleshooting issues with high availability SAP HANA setup on RHEL, see How can I debug the SAPHana and SAPHanaTopology resource agents in a Pacemaker cluster?

Appendix: Configuring Linux to Recognize Ethernet Devices for Multiple Network Interfaces

Follow these steps to configure the Linux operating system to recognize and name the Ethernet devices associated with the new elastic network interfaces created for logical network separation, which were discussed earlier in this paper.

1. Use SSH to connect to your SAP HANA host as `ec2-user`, and `sudo` to root.
2. Remove the existing `udev` rule; for example:

```
hanamaster:# rm -f /etc/udev/rules.d/70-persistent-net.rules
```

3. Create a new `udev` rule that writes rules based on MAC address rather than other device attributes. This will ensure that on reboot, `eth0` is still `eth0`, `eth1` is `eth1`, and so on. For example:

```
hanamaster:# cat <<EOF >/etc/udev/rules.d/75-persistent-net- generator.rules
# Copyright (C) 2012 Amazon.com, Inc. or its affiliates. # All Rights Reserved.
#
# Licensed under the Apache License, Version 2.0 (the "License").
# You may not use this file except in compliance with the License.
# A copy of the License is located at #
#     https://aws.amazon.com/apache2.0/ #
# or in the "license" file accompanying this file. This file is # distributed on an
 "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS
# OF ANY KIND, either express or implied. See the License for the
# specific language governing permissions and limitations under the
# License.
# these rules generate rules for persistent network device naming
```

```
SUBSYSTEM!="net", GOTO="persistent_net_generator_end"
 KERNEL!="eth*", GOTO="persistent_net_generator_end" ACTION!
="add", GOTO="persistent_net_generator_end" NAME=="?*",
 GOTO="persistent_net_generator_end"

# do not create rule for eth0
ENV{INTERFACE}=="eth0", GOTO="persistent_net_generator_end" # read MAC address
ENV{MATCHADDR}="\$attr{address}" # do not use empty address
ENV{MATCHADDR}=="00:00:00:00:00:00",
GOTO="persistent_net_generator_end"
# discard any interface name not generated by our rules ENV{INTERFACE_NAME}=="?*",
 ENV{INTERFACE_NAME}=""
# default comment
ENV{COMMENT}="elastic network interface" # write rule
 IMPORT{program}="write_net_rules"
# rename interface if needed ENV{INTERFACE_NEW}=="?*", NAME="\$env{INTERFACE_NEW}"
LABEL="persistent_net_generator_end" EOF
```

4. Ensure proper interface properties. For example:

```
hanamaster:# cd /etc/sysconfig/network/

hanamaster:# cat <<EOF >/etc/sysconfig/network/ifcfg-ethN
BOOTPROTO='dhcp4'
MTU="9000"
REMOTE_IPADDR=''
STARTMODE='onboot'
LINK_REQUIRED=no
LINK_READY_WAIT=5
EOF
```

5. Ensure that you can accommodate up to seven more Ethernet devices or network interaces, and restart `wicked`. For example:

```
hanamaster:# for dev in eth{1..7} ; do
ln -s -f ifcfg-ethN /etc/sysconfig/network/ifcfg-${dev} done

hanamaster:# systemctl restart wicked
```

6. Create and attach a new network interface to the instance.
7. Reboot.

8. Modify `/etc/iproute2/rt_tables`.

> **Important**
>
> Repeat the following for each ENI that you attach to your instance.

For example:

```
hanamaster:# cd /etc/iproute2
hanamaster:/etc/iproute2 # echo "2 eth1_rt" >> rt_tables
hanamaster:/etc/iproute2 # ip route add default via 172.16.1.122 dev eth1 table
 eth1_rt

hanamaster:/etc/iproute2 # ip rule
0: from all lookup local
32766: from all lookup main
32767: from all lookup default

hanamaster:/etc/iproute2 # ip rule add from <ENI IP Address>
lookup eth1_rt prio 1000

hanamaster:/etc/iproute2 # ip rule 0: from all lookup local
1000: from <ENI IP address> lookup eth1_rt
32766: from all lookup main
32767: from all lookup default
```

Document history

Date	Change
September 2022	High availability and disaster recovery for SAP HANA
July 2022	Architecture patterns for SAP HANA
December 2021	r6i instances updated on storage configuration for SAP HANA

Date	Change
July 2021	Storage configuration for SAP HANA
December 2017	Initial publication

SAP HANA Data Tiering on AWS Overview

Last updated: December 2022

This guide is part of a content series that provides detailed information about hosting, configuring, and using SAP technologies in the Amazon Web Services Cloud. For the other guides in the series, ranging from overviews to advanced topics, see SAP on AWS Technical Documentation home page.

Overview

This guide provides an overview of data tiering for SAP customers and partners who are considering implementing or migrating SAP environments or systems to the Amazon Web Services Cloud.

This guide is for users who architect, design, deploy, and support SAP systems directly and IT professionals that support these same functions for their SAP systems.

Prerequisites

Specialized Knowledge

You should have previous experience installing, migrating, and operating SAP environments and systems.

Technical Requirements

To access the SAP notes referenced in this guide, you must have an SAP One Support Launchpad account.

SAP Data Tiering

SAP data tiering is a data management strategy that's used to separate your data into different categories (hot, warm, and cold tiers) by various characteristics of the data. The most common characteristics used to assign the data to the correct categories are:

- data access frequency

- data update requirement
- data access performance requirement

Assigning your data to the correct category is a process that is specific to your business and IT requirements. Here are some ways to align these categories with your specific requirements.

Hot Tier: The hot tier is for storing data that is used (read or updated) frequently and that must be available in time. This hot data is critical and valuable to the business for its operational and analytical processes.

Warm Tier: The warm tier is for data that is read less often than hot data, has less stringent performance requirements, but must still be updatable. SAP HANA manages application access and update to data transparently regardless to where the data resides - hot or warm tier.

Cold Tier: The cold tier is for storing data that is infrequently accessed, does not require updates, can tolerate high access latency, and is not critical for daily operational or analytical processes.

The following table summarizes the data tiers and their characteristics.

Data tier characteristics

	Data access frequency	Data access performance	Data criticality	Data updatability
Hot	High	High	High	Required
Warm	Medium	Medium	Medium	Required
Cold	Low	Low	Low	N/A

After you have assigned the data to your preferred tiers, you can map your SAP product to the data tiering solution that is supported by SAP on AWS. For more information, see SAP HANA on AWS: Dynamic Tiering.

For the hot tier, all relevant data stays in memory. You must have an Amazon EC2 instance with adequate memory to meet your sizing requirement. For more information, see SAP HANA certified instances. You can choose a tier based on the type of your SAP product. See the following table to learn more.

Warm and cold tier options

	Native SAP HANA	SAP BW on HANA or SAP BW/4 HANA	SAP Business Suite on HANA or SAP S/4 HANA
Hot	Amazon EC2 instances certified for SAP HANA	Amazon EC2 instances certified for SAP HANA	Amazon EC2 instances certified for SAP HANA
Warm	SAP HANA dynamic tiering SAP HANA extension node SAP HANA native storage extension	SAP HANA extension node SAP HANA native storage extension for data tiering optimization (DTO)	Data aging SAP HANA native storage extension
Cold	Data Lifecycle Manager (DLM) with SAP Data Hub and Amazon S3 SAP Data Intelligence and Data Warehousing Foundation DLM with SAP HANA Spark Controller	SAP BW NLS with SAP IQ SAP BW NLS with Hadoop and Amazon S3 SAP BW/4 HANA Data Tiering Optimization (DTO) with SAP Data Hub and Amazon S3	ILM Store with SAP IQ Data archiving and Amazon S3

Warm Data Tiering Options

The following sections discuss the warm data tiering options you have on AWS.

SAP HANA native storage extension

SAP HANA Native Storage Extension (NSE) is a solution to store your warm data in SAP HANA. NSE manages warm data in a special area of SAP HANA memory (buffer cache) that is separate from the SAP HANA hot and working memory areas. The NSE solution is managed at the SAP HANA layer, making it independent of other warm data solutions (such as Data Aging). Refer to the following SAP Notes for more information about NSE (requires SAP portal access).

- SAP Note 2799997 - FAQ: SAP HANA Native Storage Extension (NSE)
- SAP Note 2816823 - Use of SAP HANA Native Storage Extension in SAP S/4HANA and SAP Business Suite powered by SAP HANA
- SAP Note 2973243 - Guidance for use of SAP HANA Native Storage Extension in SAP S/4HANA and SAP Business Suite powered by SAP HANA

SAP HANA Dynamic Tiering

SAP HANA dynamic tiering is an optional add-on to the SAP HANA database to manage historical data that can be used for your native SAP HANA use case. The purpose of SAP HANA dynamic tiering is to extend SAP HANA memory with a disk-centric columnar store (as opposed to SAP HANA's in-memory store) for managing less frequently accessed data. In this disk-centric solution, dynamic tiering service (extended storage service - esserver) runs on a separate dedicated server. The main use case for dynamic tiering is to offload less active data from SAP HANA memory to the dynamic tiering disk-backed store. As noted in the solution table, SAP HANA dynamic tiering:

- can only be used for native SAP HANA use cases.
- provides online data storage in extended store, available for both queries and updates.
- is fully validated and supported on the AWS Cloud beginning with SAP HANA 2 SPS 2.
- is an integrated component of the SAP HANA database and cannot be operated separately from the SAP HANA database.
- allows you to store up to 5 times more data in the warm tier than in your hot tier.

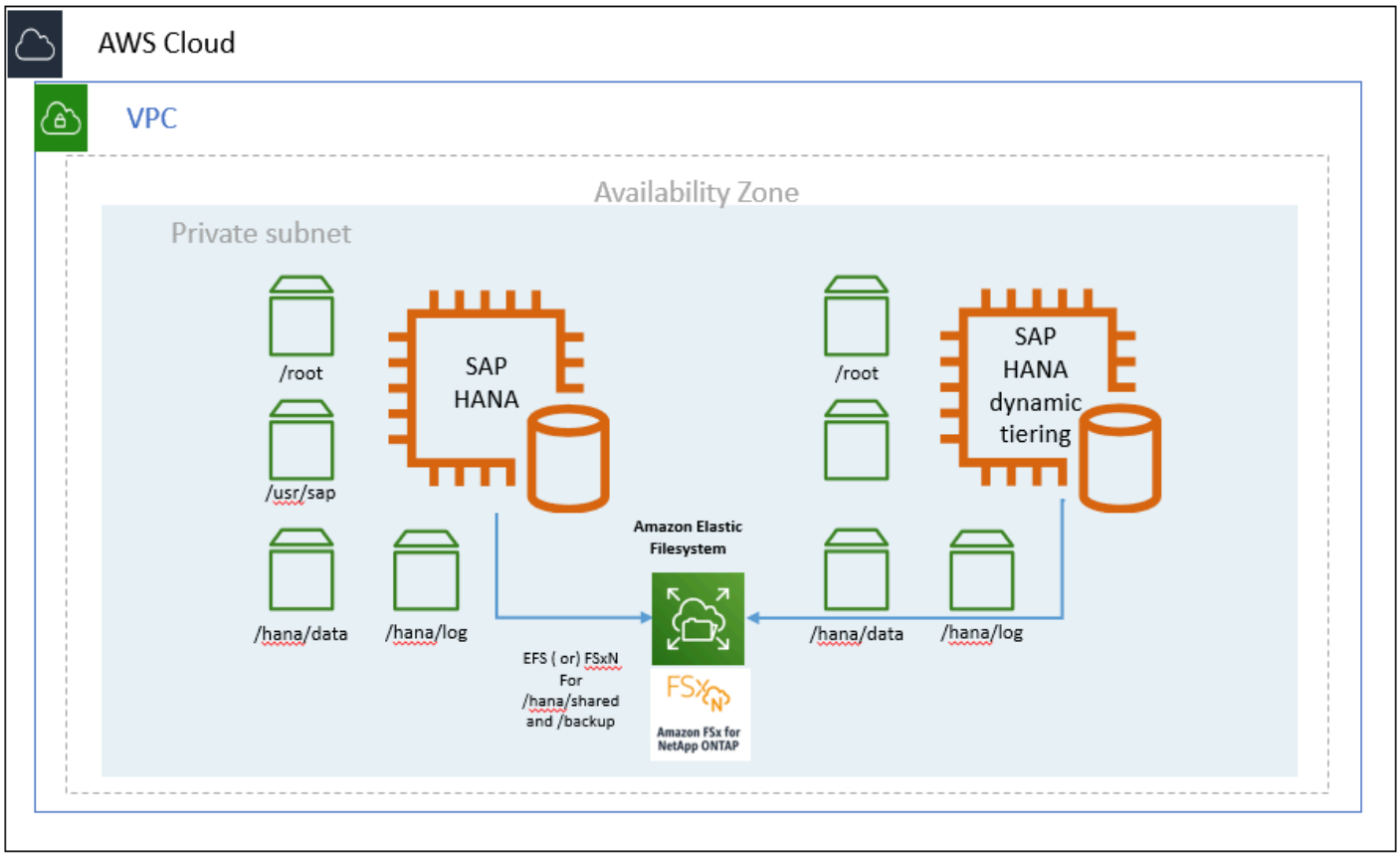


Figure 1: SAP HANA dynamic tiering on AWS (single-AZ)

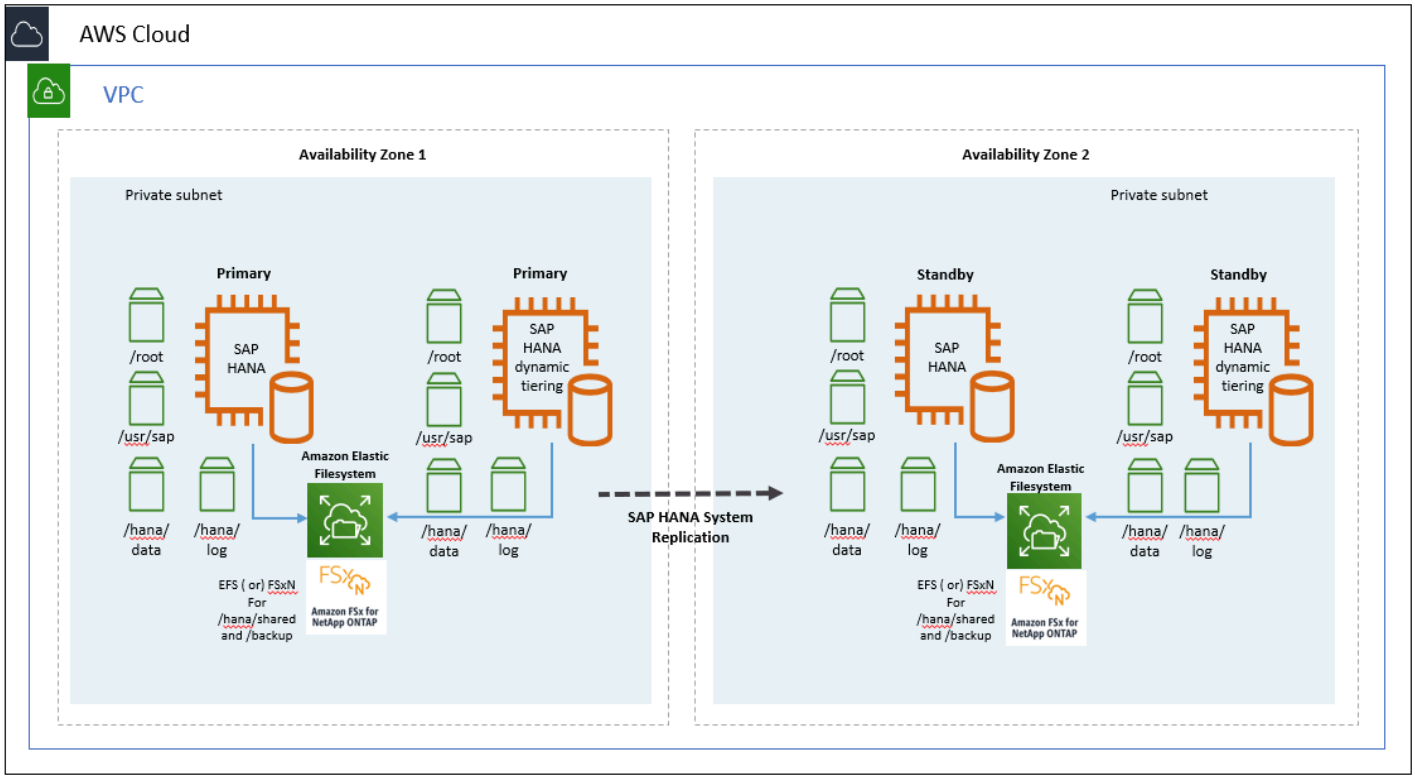


Figure 2: SAP HANA dynamic tiering on AWS (multi-AZ)

SAP HANA Extension Node

SAP HANA extension node is a special purpose SAP HANA worker node that is specifically set up and reserved for storing warm data. An important difference between SAP HANA dynamic tiering and SAP HANA extension node is that the extension node is a separate SAP HANA instance. It is not a separate *esserver*process like dynamic tiering. Because of this, the SAP HANA extension node offers the full feature set of the SAP HANA database. SAP HANA extension node allows you to store warm data for your SAP Business Warehouse (BW) or native SAP HANA use cases.

The total amount of data that can be stored on the SAP HANA extension node ranges from 1 to 2x of the total amount of memory of your extension node. For example, if your extension node had 2 TB of memory, you could potentially store up to 4 TB of warm data on your extension node.

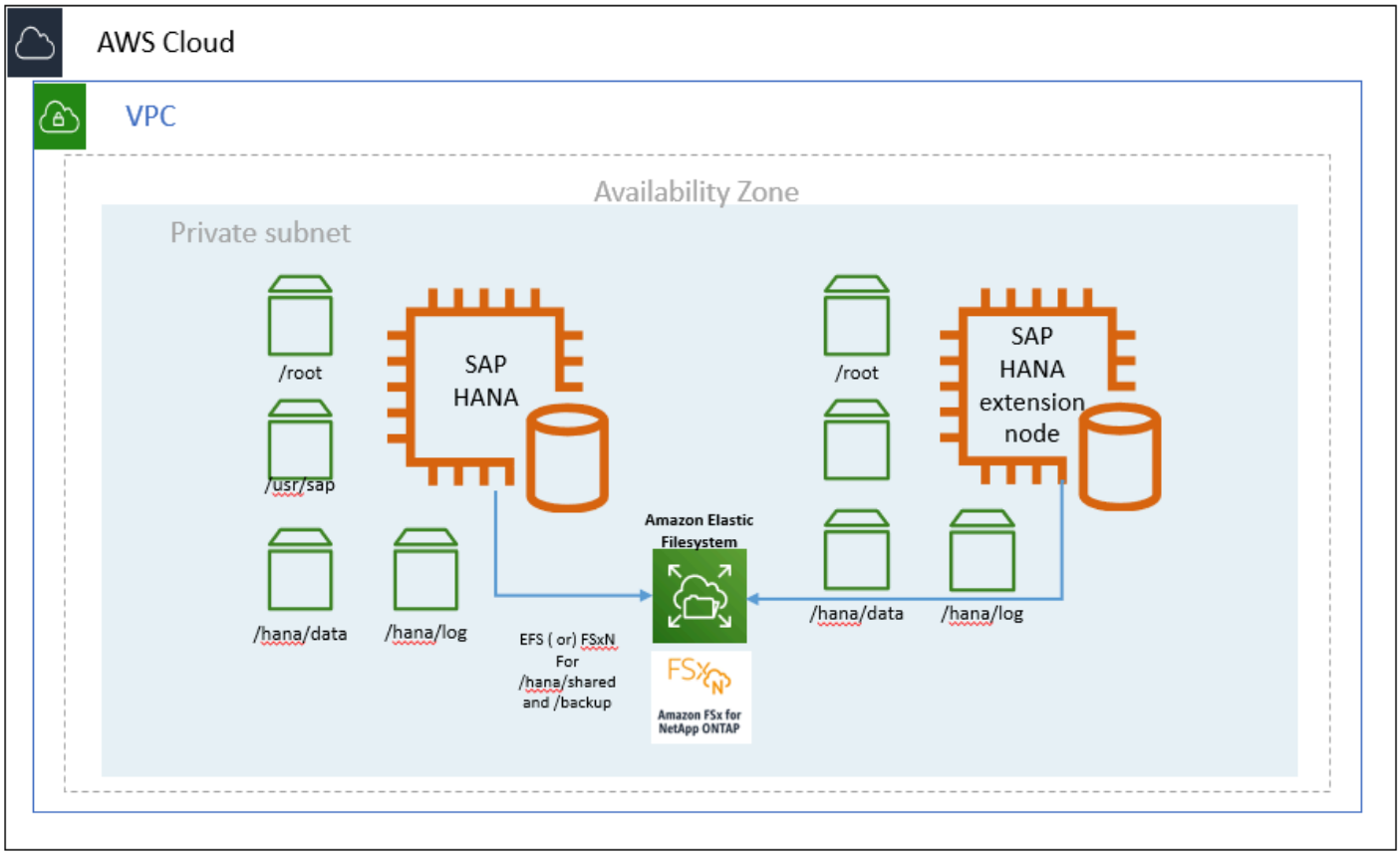


Figure 3: SAP HANA extension node on AWS

Data Aging

Data aging can be used for SAP products like SAP Business Suite on HANA (SoH) or SAP S/4HANA to move data from SAP HANA memory to the disk area. The disk area is additional disk space that is a part of the SAP HANA database. This helps free up more SAP HANA memory by storing older, less frequently accessed data in the disk area. When the data is read or updated, data aging uses the paged attribute property to selectively load the pages of a table into memory instead of loading the entire table into memory. This helps you conserve your memory space by only loading the required data (instead of the entire table) into memory. In addition, paged attributes are marked for a higher unload priority by SAP HANA and are paged out to disk first when SAP HANA needs to free up memory. To size your SAP HANA memory requirements for data aging, SAP recommends that you run the sizing report provided in the SAP Note 1872170 - ABAP on HANA sizing report (S/4HANA, Suite on HANA).

Cold Data Tiering Options

The following sections discuss cold data tiering options on AWS.

The Data Lifecycle Manager (DLM) tool, which is part of SAP HANA Data Warehousing Foundation, can be used to move data from SAP HANA memory to a cold storage location. For your native SAP HANA use case, you have two options.

DLM with SAP Data Hub

SAP Data Hub is a data orchestration and management solution running on Kubernetes. With this option, you can use the SAP Data Hub product to move data in and out of SAP HANA into your cold store location. On AWS, you are able to use native AWS services such as Amazon Simple Storage Service to store your cold data. Once your data is in Amazon S3, you can use Amazon S3 features such as S3 Intelligent-Tiering and Amazon S3 Lifecycle to optimize your costs. Once you have determined that you no longer need to access your cold data from SAP HANA, you can archive your data in Amazon S3 Glacier for long-term retention.

> **Note**
>
> SAP Data Hub is now released as managed service on SAP Cloud Platform with the name SAP Data Intelligence.

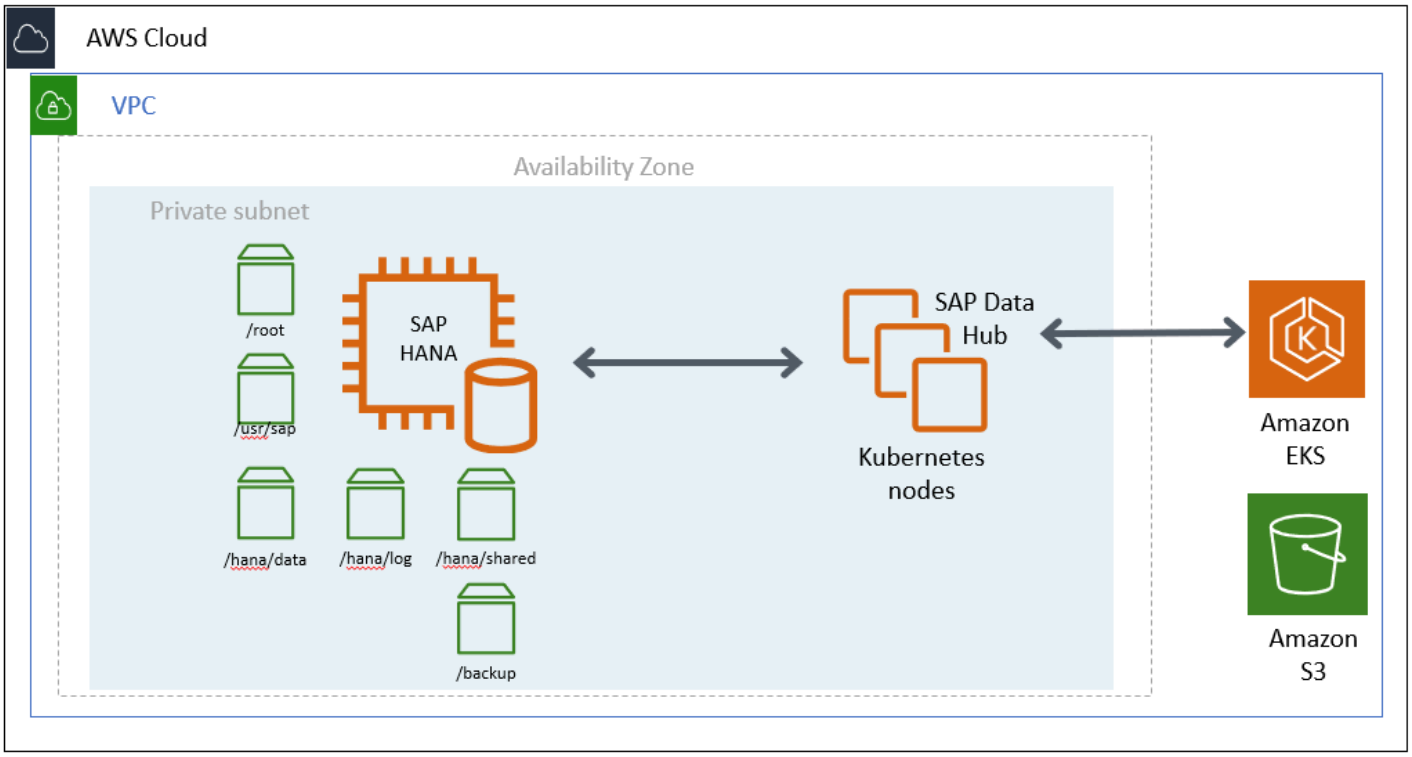


Figure 4: SAP Data Hub on Amazon EKS for cold tier

DLM with SAP HANA Spark Controller

SAP HANA Spark controller enables SAP HANA to access the data in Hadoop through an SQL interface. With this option, you can use the SAP HANA Spark Controller to allow SAP HANA to access cold data through the Spark SQL SDA adapter. On AWS, you can use an AWS native service like Amazon EMR for the Hadoop cold tier storage location. To use Amazon EMR with SAP HANA, see DLM on Amazon Elastic Map Reduce documentation from SAP. For more information about the Spark controller, see Using SAP HANA Spark Controller .

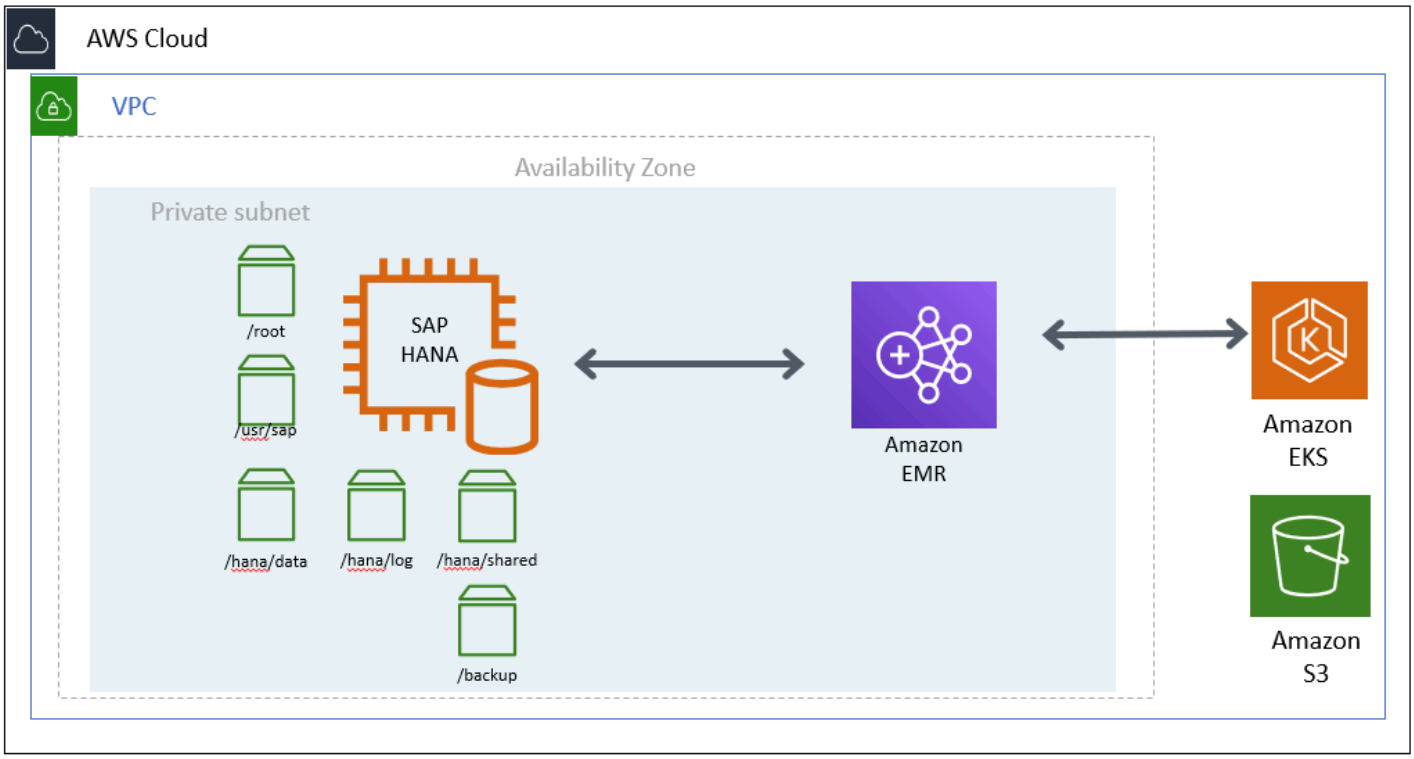


Figure 5: SAP HANA with Amazon EMR for cold tier

Cold Tier Options for SAP BW

For the SAP Business Warehouse (BW) on HANA or SAP BW/4 HANA use cases, you have additional options for cold tier storage.

SAP BW Near Line Storage (NLS) with SAP IQ

With this option, you can use SAP BW Near Line Storage (NLS) with SAP IQ or you can use Data Tiering Optimization (DTO) with SAP IQ to store your cold data. On AWS, you can run your SAP IQ server on Amazon Elastic Compute Cloud (Amazon EC2) instances for the cold tier storage.

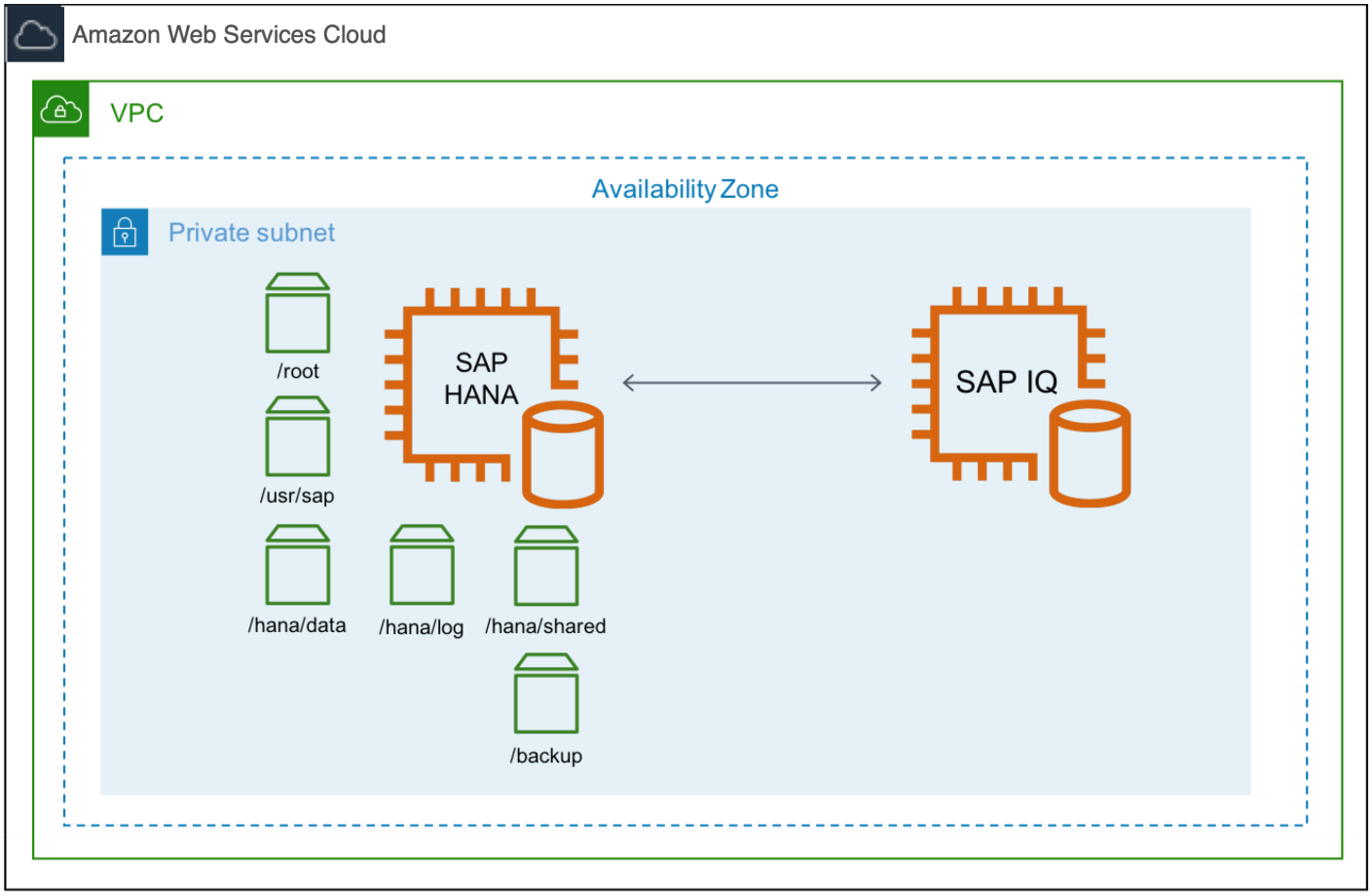


Figure 6: SAP BW NLS with SAP IQ for cold tier

SAP BW NLS with Hadoop

With this option, you can use SAP BW NLS with Apache Hadoop instead of SAP IQ, with this option you can persist your Hadoop data in Amazon S3 using a Hadoop third-party connector for Amazon S3. See Hadoop as a Near-Line Storage Solution documentation from SAP, SAP Note 2363218 – Hadoop NLS: Information, Recommendations and Limitations, and Cloud Data Access documentation for details.

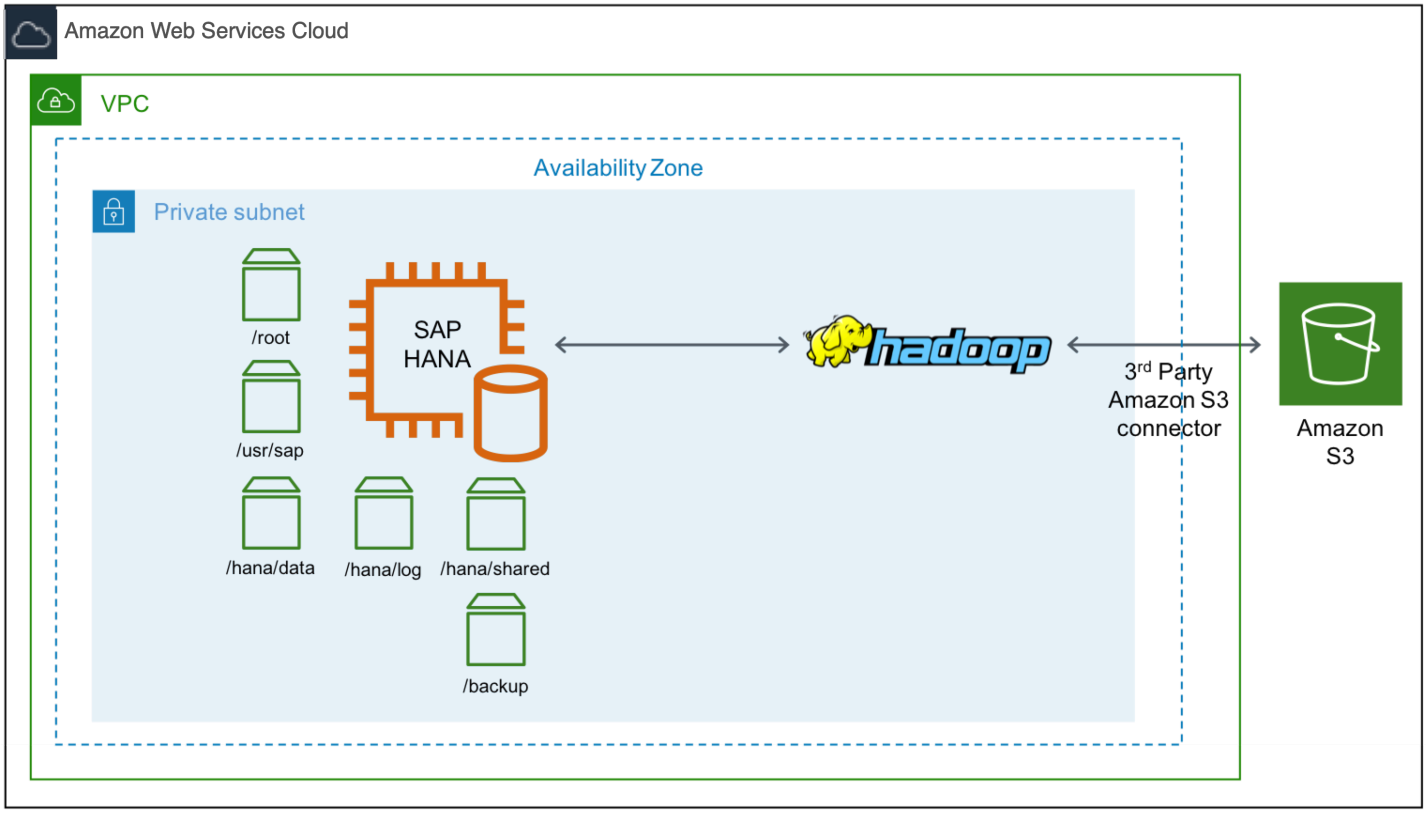


Figure 7: SAP BW NLS with Hadoop for cold tier

SAP BW/4HANA DTO with Data Hub

This option is similar to SAP Data Hub with SAP HANA. You can use DTO with SAP Data Hub to store your cold data in Amazon S3. This option only applies if you use SAP BW/4HANA.

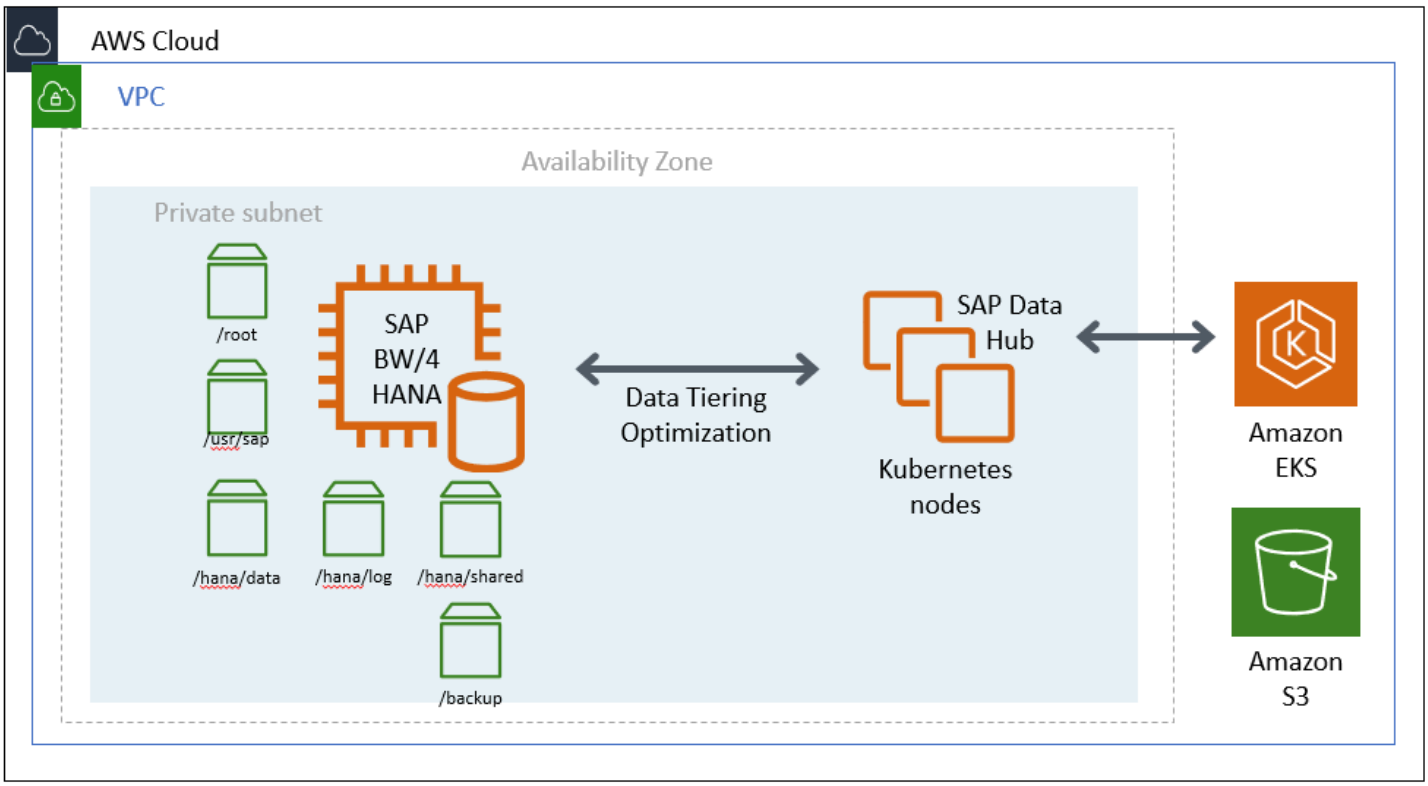


Figure 8: SAP Data Hub on Amazon EKS with BW4/HANA

Cold Tier Options for SAP S/4HANA or Suite on HANA

For S/4HANA or SOH, you can use SAP Information Life Cycle Management (ILM) for the cold data tiering. You have few options with ILM for cold tier. See ILM Store documentation from SAP for details.

SAP ILM with SAP IQ

With this option, you can use ILM with SAP IQ. Similar to the SAP BW NLS with SAP IQ scenario, you can run your SAP IQ server on AWS Amazon EC2 instances to store cold data.

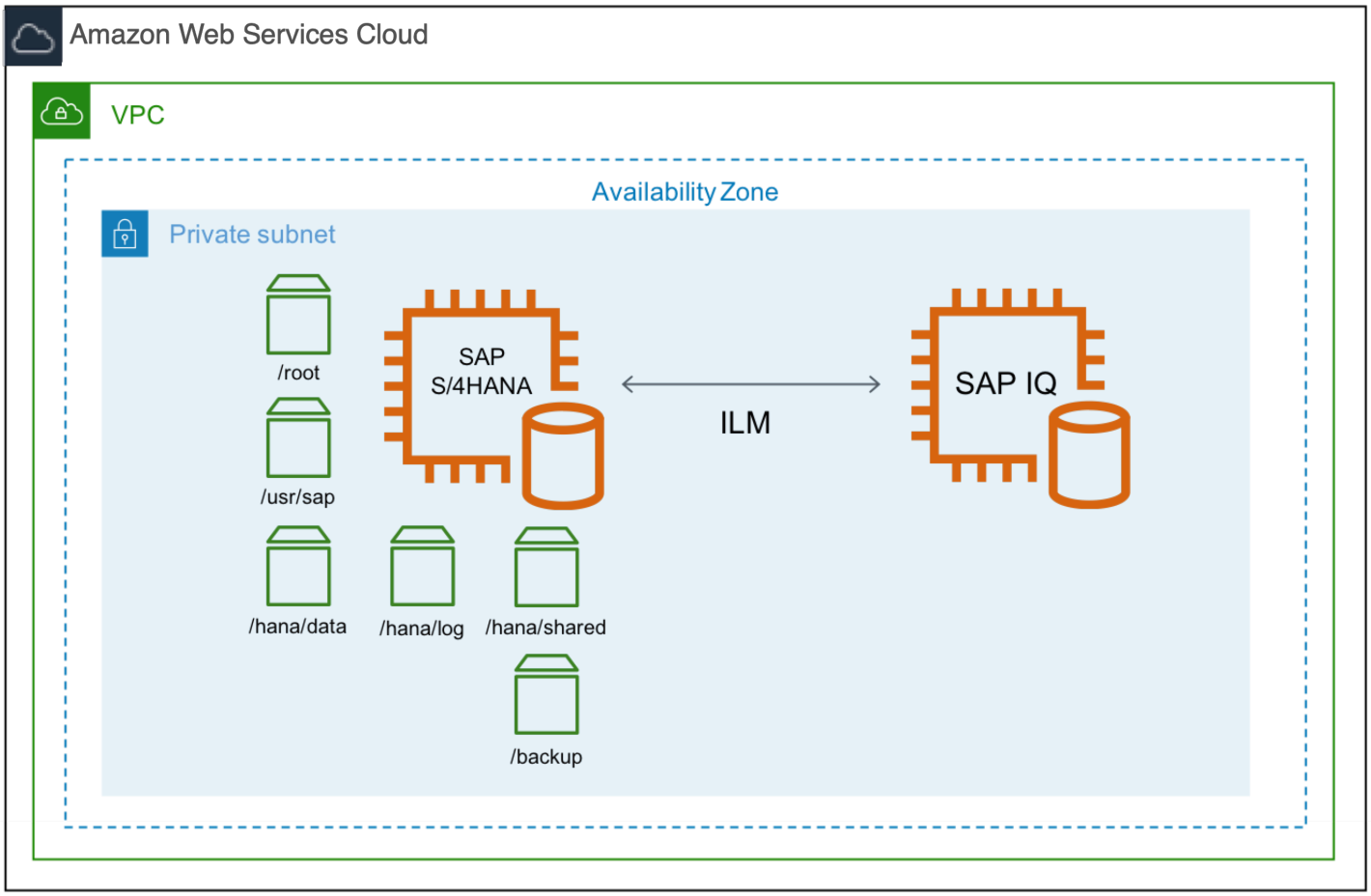


Figure 9: SAP ILM with SAP IQ for cold tier

SAP Archiving

With this option, you can use ILM or your standard data archiving process. You can use Amazon Elastic File System (Amazon EFS) and Amazon FSx for NetApp ONTAP to store your archive file in a highly available, scalable and durable manner. Amazon EFS and FSx for ONTAP can be mounted as your archive file system and you can archive your data from SAP to this file system through SAP transaction code SARA.

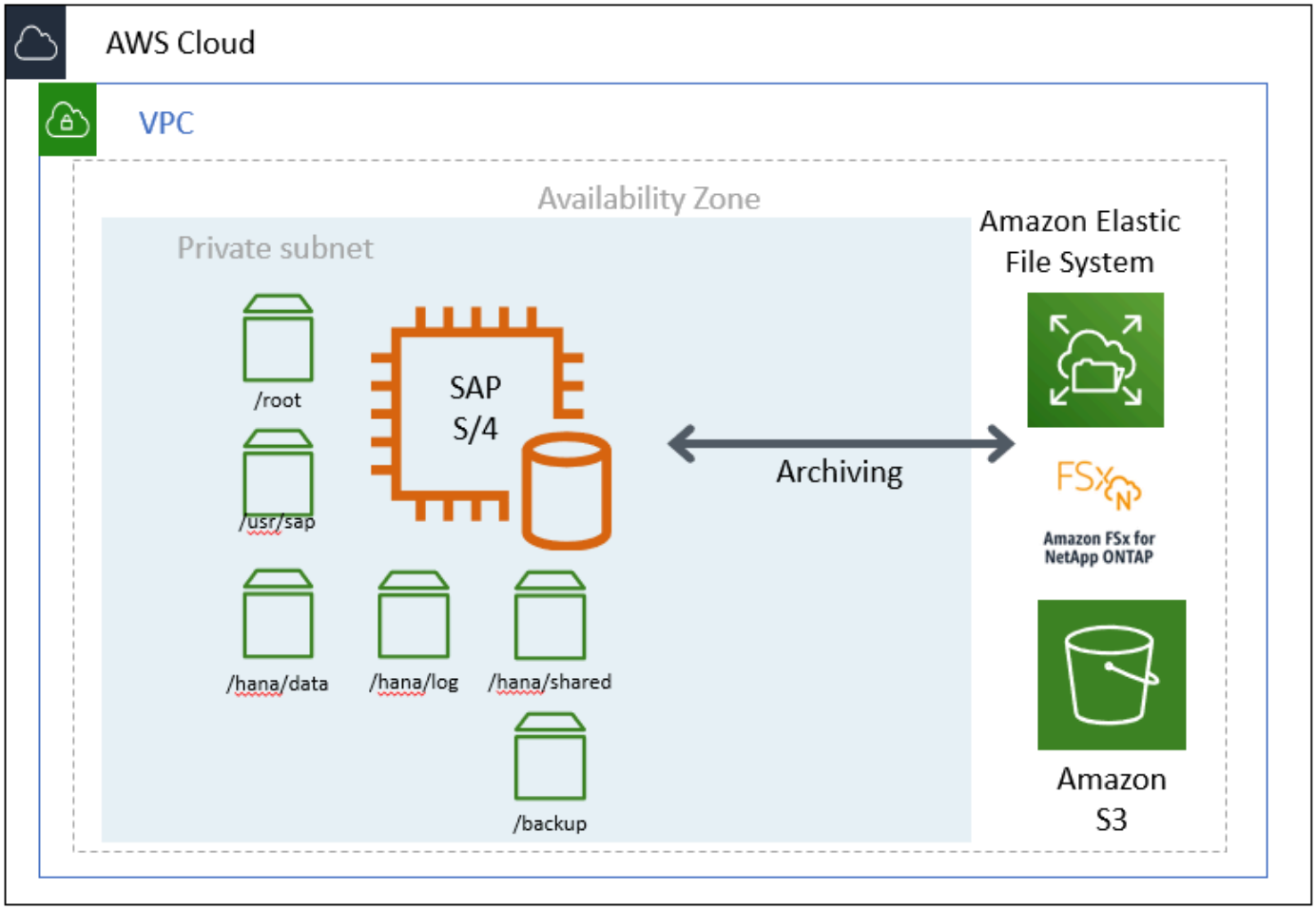


Figure 10: SAP archiving with Amazon EFS for cold tier

For archiving, another option is to use the Amazon Elastic Block Store (Amazon EBS) sc1 volume type as the underlying storage type for your archive file system. Amazon EBS sc1 volumes are inexpensive block storage and are designed for less frequently accessed workloads like data archiving. To increase durability and availability of your archived data, we recommend that you copy the data to Amazon S3 for backup and Amazon S3 Glacier for long term retention.

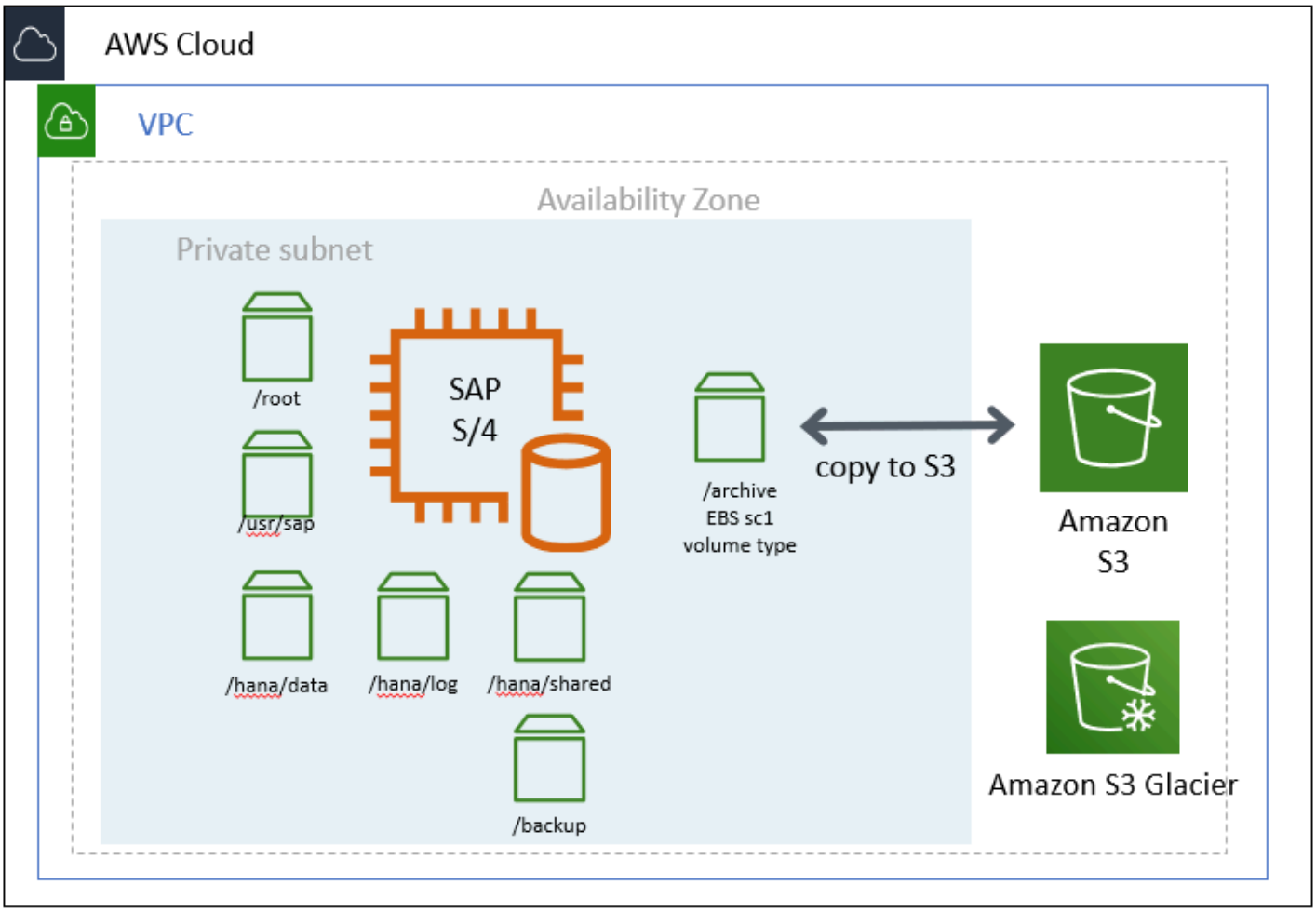


Figure 11: SAP archiving with Amazon EBS for cold tier

Document Revisions

Date	Change
December 2022	Updated document
July 2019	Initial publication

SAP on AWS High Availability with Overlay IP Address Routing

SAP specialists, Amazon Web Services

Last updated: June 2020

This guide is part of a content series that provides detailed information about hosting, configuring, and using SAP technologies in the Amazon Web Services Cloud. For the other guides in the series, ranging from overviews to advanced topics, see SAP on AWS Technical Documentation.

Overview

This guide provides SAP customers and partners instructions to set up a highly available SAP architecture that uses overlay IP addresses on Amazon Web Services. This guide includes two configuration approaches:

- AWS Transit Gateway serves as central hub to facilitate network connection to an overlay IP address.
- Elastic Load Balancing where a Network Load Balancer enables network access to an overlay IP address.

This guide is intended for users who have previous experience installing and operating highly available SAP environments and systems.

Prerequisites

Specialized Knowledge

Before you follow the configuration instructions in this guide, we recommend that you become familiar with the following AWS services. (If you are new to AWS, see Getting Started with AWS.)

- Amazon VPC
- AWS Transit Gateway
- Elastic Load Balancing

SAP on AWS High Availability Setup

SAP customers can fully realize the benefit of running mission-critical SAP workloads by building reliable, fault-tolerant, and highly available systems in the AWS Cloud depending on the operating system and database. AWS offers the use of multiple Availability Zones within an AWS Region to provide resiliency for the SAP applications.

As part of your SAP implementation, you create an Amazon Virtual Private Cloud (Amazon VPC) to logically isolate the network from other virtual networks in the AWS Cloud. Then, you use AWS network routing features to direct the traffic to any instance in the VPCs or between different subnets in a VPC. Amazon VPC setup includes assigning subnets to your SAP ASCS/ERS for NetWeaver and primary/secondary nodes for the SAP HANA database. Each of these configured subnets has a classless inter-domain routing (CIDR) IP assignment from the VPC which resides entirely within one Availability Zone. This CIDR IP assignment cannot span multiple zones or be reassigned to the secondary instance in a different AZ during a failover scenario.

For this reason, AWS allows you to configure Overlay IP (OIP) outside of your VPC CIDR block to access the active SAP instance. With IP overlay routing, you can allow the AWS network to use a non-overlapping RFC1918 private IP address that resides outside an VPC CIDR range and direct the SAP traffic to any instance setup across the Availability Zone within the VPC by changing the routing entry in AWS.

A SAP HANA database or SAP NetWeaver application that is protected by a cluster solution such as SUSE Linux Enterprise Server High Availability Extension (SLES HAE), RedHat Enterprise Linux HA Add-On(RHEL HA) or SIOS uses the overlay IP address assigned to ensure that the HA cluster is still accessible during the failover scenarios. Since the overlay IP address uses the IP address range outside the VPC CIDR range and Virtual Private Gateway connection, you can use AWS Transit Gateway as a central hub to facilitate the network connection to an overlay IP address from multiple locations including Amazon VPCs, other AWS Regions, and on-premises using AWS Direct Connect or AWS Client VPN.

If you do not have AWS Transit Gateway set up as a network transit hub or if AWS Transit Gateway is not available in your preferred AWS Region, you can use a Network Load Balancer to enable network access to an OIP.

Overlay IP Routing using AWS Transit Gateway

With Transit Gateway, you use route table rules which allow the overlay IP address to communicate to the SAP instance without having to configure any additional components, like a Network Load Balancer or Amazon Route 53. You can connect to the overlay IP from another VPC, another subnet (not sharing the same route table where overlay IP address is maintained), over a VPN connection, or via an AWS Direct Connect connection from a corporate network.

Note: If you do not use Amazon Route 53 or AWS Transit Gateway, see the Overlay IP Routing with Network Load Balancer section.

Architecture

AWS Transit Gateway acts as a hub that controls how traffic is routed among all the connected networks which act like spokes. Your Transit Gateway routes packets between source and destination attachments using Transit Gateway route tables. You can configure these route tables to propagate routes from the route tables for the attached VPCs and VPN connections. You can also add static routes to the Transit Gateway route tables. You can add the overlay IP address or address CIDR range as a static route in the transit gateway route table with a target as the VPC where the EC2 instances of SAP cluster are running. This way, all the network traffic directed towards overlay IP addresses is routed to this VPC. The following figure shows this scenario with connectivity from different VPC and corporate network.

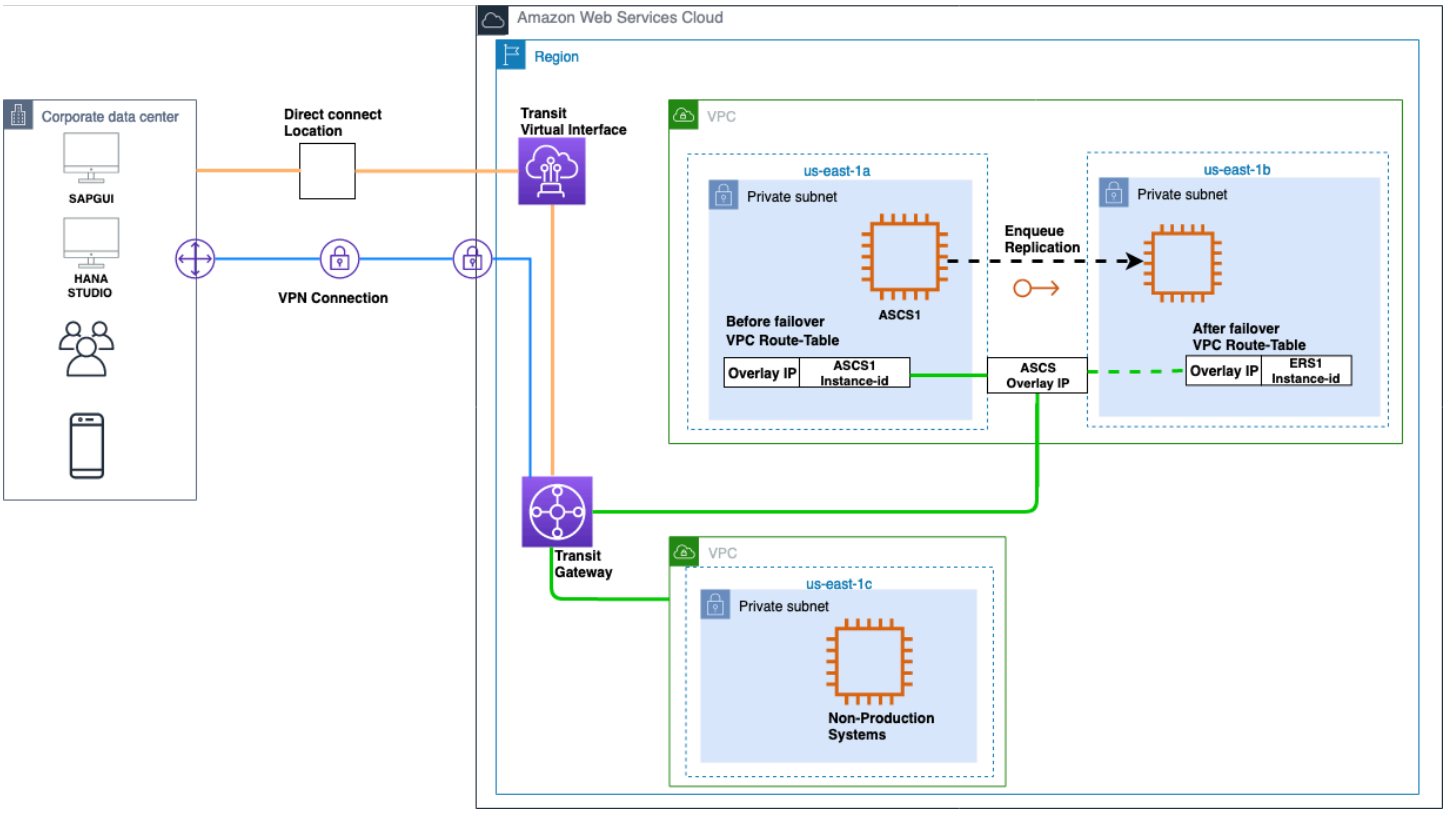


Figure 1: Overlay IP address setup with AWS Transit Gateway

Pricing for the AWS Transit Gateway:

AWS Transit Gateway pricing is based on the number of connections made to the Transit Gateway per hour and the amount of traffic that flows through AWS Transit Gateway. For more information, see AWS Transit Gateway Service Level Agreement.

Configuration Steps for AWS Transit Gateway

This section includes high-level steps necessary to understand overlay IP address configuration for this scenario. See the AWS Transit Gateway documentation for detailed steps regarding AWS Transit Gateway configuration.

Step 1. Set up the Transit Gateway architecture

1. Create a Transit Gateway in your AWS account in the AWS Region where the SAP instance is deployed. For detailed steps, see Getting Started with Transit Gateways.
2. Attach VPCs where SAP instances are deployed (and any other VPCs as required) to the Transit Gateway. For detailed steps, see Transit Gateway Attachments to a VPC.

Note: For attachment, select only the subnet where the SAP instances are running with cluster and overlay IP configured. In the following figure, the private subnet of the SAP instance is selected for the Transit Gateway attachment.

Figure 2: Attaching Transit Gateway to private subnet

3. Do one of the following, depending on your connection:

- **VPN connection**. Attach a VPN to this Transit Gateway. For detailed steps, see Transit Gateway VPN Attachments.

  When you create a site-to-site VPN connection, you specify the static routes for the overlay IP address. For detailed steps, see VPN routing options.

- **AWS Direct Connect**. Attach a Direct Connect Gateway to this Transit Gateway. First, associate a Direct Connect Gateway with the Transit Gateway. Then, create a transit virtual interface for your AWS Direct Connect connection to the Direct Connect gateway. Here, you can advertise prefixes from on-premises to AWS and from AWS to on-premises. For detailed steps, see Transit Gateway Attachments to a Direct Connect Gateway.

When you associate a Transit Gateway with a Direct Connect gateway, you specify the prefix lists to advertise the overlay IP address to the on-premises environment. For detailed steps, see Allowed prefixes interactions.

Note:AWS Direct Connect is recommended for business critical workloads. See Resilience in AWS Direct Connect to learn about resiliency at the network level.

Step 2. Configure routing for AWS and corporate networks

The following table lists the IP addresses used in the example configuration. Make sure to use your valid private IP addresses for your implementation.

Description	IP Range/IP Address
VPC CIDR of production SAP systems (with HA cluster running with Overlay IP)	10.0.0.0/16
VPC CIDR of non-production SAP systems (Instances in this VPC access the Production cluster overlay IP using AWS Transit Gateway)	192.168.1.0/24
Corporate network CIDR (Site-to-Site VPN is configured between corporate networks to AWS Transit Gateway)	192.168.2.0/24
Overlay IP address CIDR	172.16.1.0/26
Customer gateway IP address	34.216.94.150/32

Note

If you are using AWS Client VPN, you do not need to configure Transit Gateway. You can create additional entries in the routing table for overlay IP addresses. Route traffic to the subnets of the VPC of production SAP system where overlay IP addresses are configured.

When you create a Transit Gateway attachment to a VPC, the propagation route is created in the default Transit Gateway route table. In Figure 3, the first and second entry shows the propagated route created automatically for VPCs where SAP production and non-production systems are running through VPC attachment.

1. To route traffic from AWS Transit Gateway to the overlay IP address, create static routes in the **Transit Gateway route tables** to route overlay IP addresses to the VPC of production SAP system where the overlay IP addresses are configured. In Figure 3, the third entry shows that the static route created for the overlay IP range is attached. The target for this route is the SAP Production VPC.

	CIDR	Attachment	Resource Type	Route type	Route state
☐	10.0.0.0/16	tgw-attach-xxxxxxxxx | vpc-xxxxxxxx	VPC	propagated	active
☐	192.168.1.0/24	tgw-attach-yyyyyyyy | vpc-yyyyyyy	VPC	propagated	active
☐	172.16.1.0/26	tgw-attach-xxxxxxxxx | vpc-xxxxxxxx	VPC	static	active
☐	192.168.2.0/24	tgw-attach-xxxxxxxxx | vpn-xxxxxxxx(35.164.53.172)	VPN	static	active

Figure 3: Transit Gateway route table: Overlay IP static route with VPC of production SAP system target

2. To route the outgoing traffic from VPCs where SAP instances are running to private IP addresses of another VPC where SAP instances are running attached to same Transit Gateway, create entries in the **route tables associated with these VPC subnets**. The target of these routes is AWS Transit Gateway. In the following VPC of production SAP system route table example, the non-production SAP VPC (third entry) and corporate network (fourth entry) are routed to the Transit Gateway.

Destination	Target	Status	Propagated
10.0.0.0/16	local	active	No
0.0.0.0/0	nat-<resource-id>	active	No
192.168.1.0/24	tgw-<resource-id>	active	No
192.168.2.0/24	tgw-<resource-id>	active	No
172.16.1.0/26	eni-<resource-id>	active	No

Figure 4: VPC of production SAP system route table: VPC of production SAP system and corporate network routed to AWS Transit Gateway

3. In the VPC of the non-production SAP system, to route the outgoing traffic from the overlay IP address, create entries in the route tables with Transit Gateway as the target. In the following VPC of non-production SAP system route table example, the destination is the overlay IP range and the target is Transit Gateway.

Destination	Target	Status	Propagated
192.168.1.0/24	local	active	No
0.0.0.0/0	nat-<resource-id>	active	No
10.0.0.0/16	tgw-<resource-id>	active	No
192.168.2.0/24	tgw-<resource-id>	active	No
172.16.1.0/26	tgw-<resource-id>	active	No

Figure 5: VPC of non-production SAP system route table: Outgoing traffic from overlay IP address routed to Transit Gateway

4. Configure routing from corporate devices to Amazon VPC IP addresses.

Step 3. Disable the source/destination check

Each Amazon EC2 performs source/destination checks by default. This means that the instance must be the source or destination of any traffic it sends or receives. For cluster instances, source/destination check must be disabled on both Amazon EC2 instances which are supposed to receive traffic from the Overlay IP address. You can use the AWS CLI or the AWS Management Console to disable source/destination check. For details, see ec2 modify-instance-attribute.

Step 4. Test the configuration

Once the setup is complete, perform connectivity testing by making sure you can reach the SAP systems through overlay IP address. With this configuration, you can reach the overlay IP addresses from other VPCs and your corporate network just like any private IP address of the VPC. With the AWS Transit Gateway approach, no additional components are required for communication, such as Amazon Route 53 agent or Network Load Balancer.

Step 5. Update overlay IP address

Step 4: Once the network connectivity is tested successfully, update the overlay IP address of the production or non-production SAP system in the message server parameter of your SAP Graphical

User Interface (GUI) System Entry Properties along with other SAP connectivity properties for connection. You can use the corporate DNS or Amazon Route 53 to create a user friendly CNAME for the Overlay IP.

Overlay IP Routing with Network Load Balancer

If you do not use Amazon Route 53 or AWS Transit Gateway, you can use Network Load Balancer for accessing the overlay IP address externally. The Network Load Balancer functions at the fourth layer of the Open Systems Interconnection (OSI) model. It can handle millions of requests per second. After the load balancer receives a connection request, it selects a target from the Network Load Balancer target group to route network connection request to a destination address which can be an overlay IP address.

Architecture

The following figure shows the network access flow of ASCS or SAP HANA overlay IP from outside the VPC.

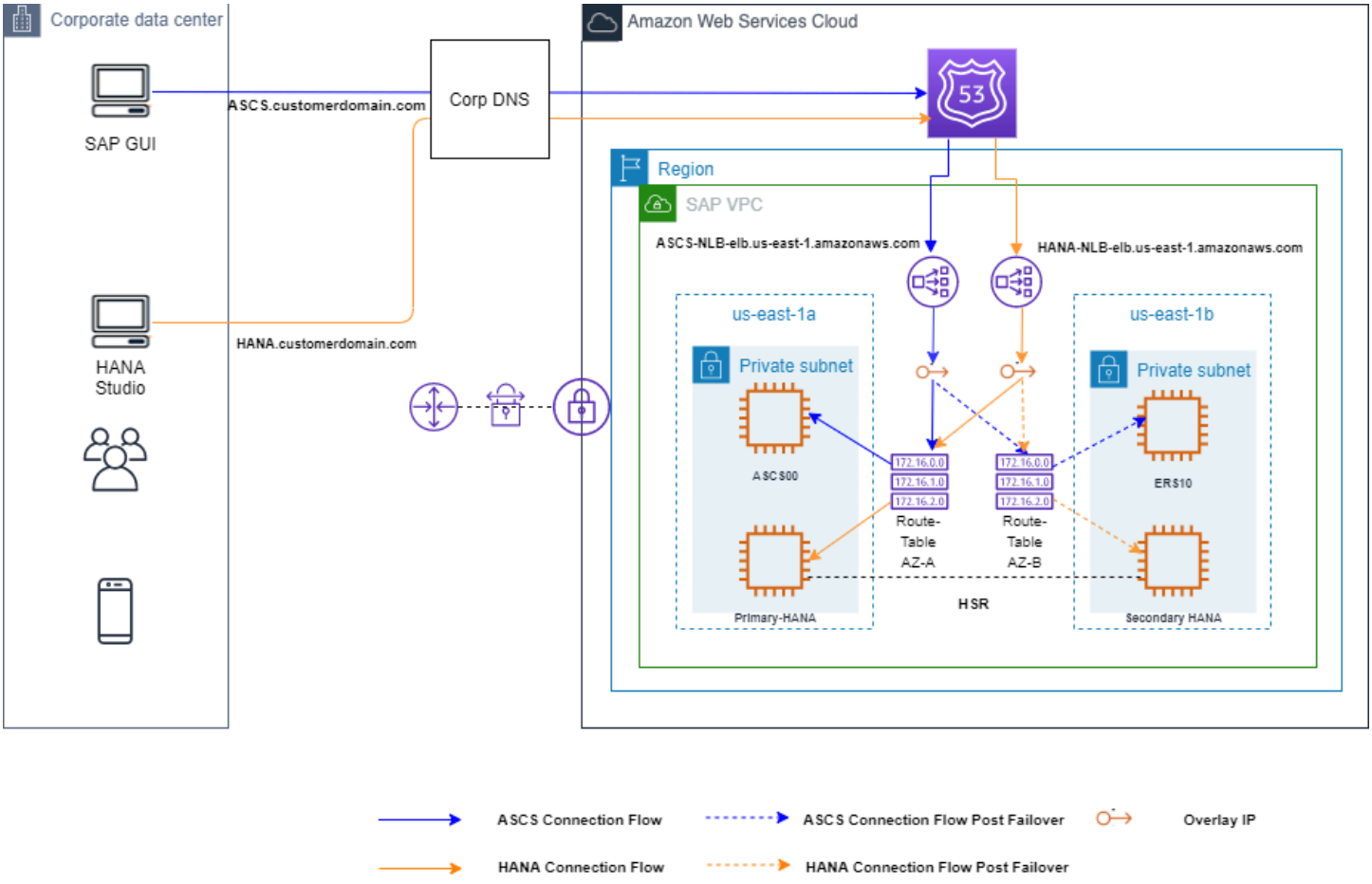


Figure 6: SAP High Availability with Overlay IP and Elastic Load Balancer

Pricing for Network Load Balancers:

With Network Load Balancers, you only pay for what you use. See Elastic Load Balancing pricing, for more information.

Configuration Steps for Network Load Balancer

Use the following instructions to set up the Network Load Balancer to access the overlay IP address. The following values are used for the example configuration.

Table 1: System Settings

System Setting	Value
Instance number for ASCS and SAP HANA	00
OIP for ASCS	192.168.0.20
OIP for HANA	192.168.1.99

Table 2: Listener Port Values

Listener Ports	Value
ASCS Message server port	36<instance number> (3600)
SAP HANA	SAP HANA Studio service connection (login required) SAP Note 1592925
SAPStartSrv/HTTP Port	5<instance number>13 (50013)
JDBC/SQL Port	3<instance number>15 (30015)

Step 1. Create the target group

1. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/
2. On the navigation pane, under **LOAD BALANCING**, choose **Target Groups**.

3. Choose Create target group.
4. For **Name**, type an easily identified target group name for the sap-ascs instance. (For example, type sap-ascs for your ASCS overlay IP address).
5. For **Target type**, select **IP**.
6. For **Protocol**, choose **TCP**.
7. For **Port**, type 36<ASCS instance number>. For example: 3600, where 00 is the instance number.
8. For **Health checks**, keep the default health check settings, or change settings based on your requirements.
9. Choose **Create**.
10. Repeat steps 1 to 9 to create target group for JDBC/SQL port 3<instance number>15 and SAP HANA HTTP port 5<instance number>13 to access your SAP HANA instance with the respective overlay IP address.
11. Choose the **Targets** tab, then choose **Edit**.
12. Choose **Add** to register your targets.
13. Choose the **Network** drop-down and select **Other private IP address**. Then, enter the ASCS overlay IP address and choose **Add to list**.
14. Repeat steps 11 to 13 to register JDBC/SQL and HTTP ports with the respective overlay IP address.

Step 2. Create the Network Load Balancer for ASCS

1. On the EC2 navigation pane, under **LOAD BALANCING**, choose **Load Balancers**.
2. Choose Create Load Balancer.
3. For Network Load Balancer, choose Create.
4. For **Name**, type a name for your load balancer. For example, sap-ha-nlb.
5. For **Scheme**, choose **internal**. An internal load balancer routes requests to targets using private IP addresses.
6. For **Listeners**, under Protocol, choose **TCP**. For **Port**, specify the ASCS port (36< SAP Instance number>. For example, use 3600 if your SAP instance number is 00.
7. For **Availability Zones**, select the VPC and subnets where the SAP instances with HA setup are deployed.
8. For **Tags**, choose **Add Tags** and for Key, type Name. For Value, type the name of the network load balancer, such as sap-ha-nlb.

9. Choose Next: Configure Security Settings.
10. Ignore the warning that appears and choose **Next: Configure Routing**. (In this scenario, the network load balancer is used as pass through without any SSL termination. For end-to-end encryption, use SNC from SAP GUI to SAP Instance.)
11. For **Target group**, choose **Existing target group** and select the **sap-ascs** target group created earlier.
12. Choose **Next: Register Targets**.
13. Choose **Next: Review**.
14. Choose **Create**.
15. Repeat the steps 1 to 14 to create another Network Load Balancer for SAP HANA setup with Network Load Balancer TCP protocol listener to JDBC/SQL port 3<instance number>15. Choose VPC and the subnets where the primary and secondary SAP HANA database is deployed and register the target JDBC/SQL target group.
16. Add an additional listener to the Network Load Balancer created in step 14 with SAP StartSrv/HTTP port 5<instance number>13 listener port and register the target StartSrv/HTTP port target group.

Step 3. Set up VPC routing table

This step enables the connection to your SAP instance.

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/
2. In the navigation pane, choose **Route Tables**, and select the Amazon VPC routing table where your SAP instance is deployed.
3. Choose **Actions**, **Edit routes**.
4. For **Destination**, specify your overlay IP address. For **Target**, specify the SAP instance Elastic Network Interface.
5. Choose Save routes.

This setup allows the static Network Load Balancer DNS to forward the traffic to your SAP instance network interface through the static overlay IP address. During failover scenarios, you can point to the elastic network interface of the active SAP instance using manual steps or automatically using cluster management software.

Step 4. Connect using SAP GUI

1. In the **Load Balancers** section of the EC2 console, make a note of the Network Load Balancer DNS name for the sap-ha-nlb.

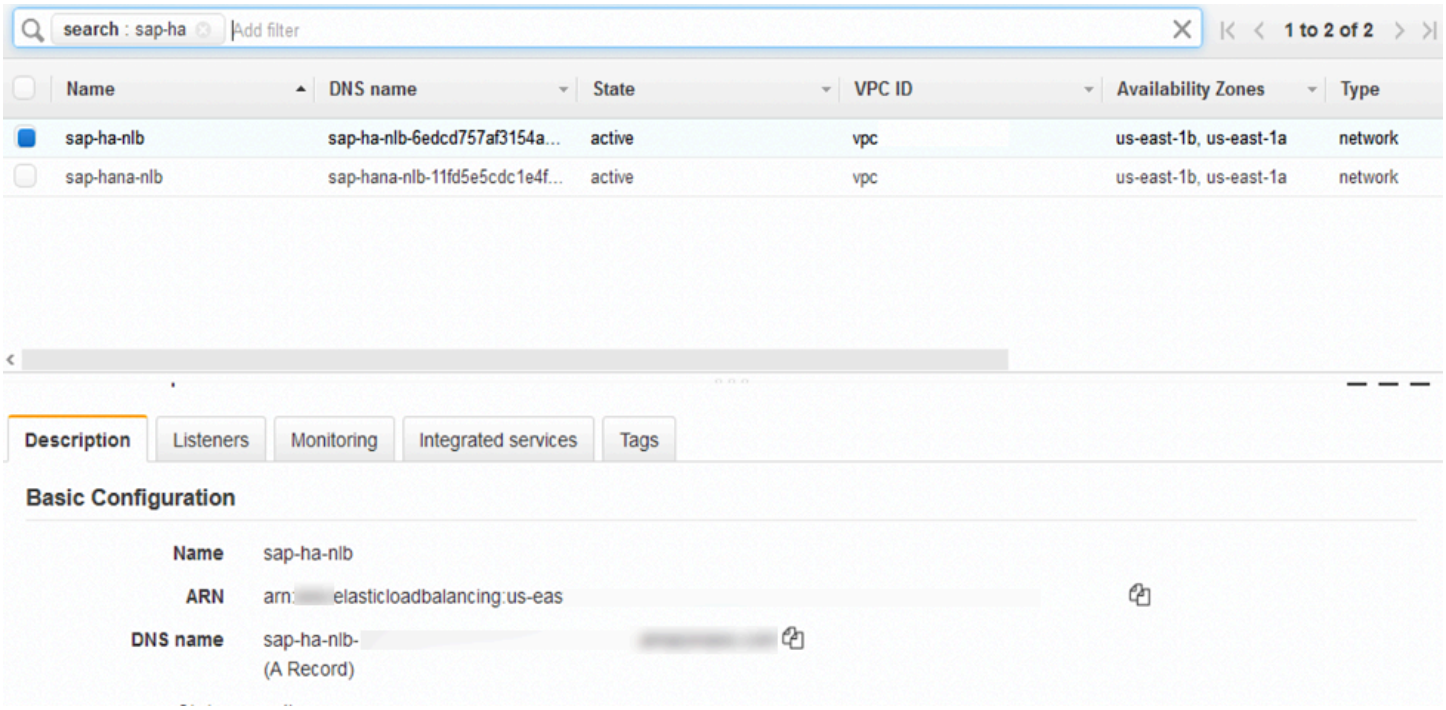


Figure 7: sap-ha-nlb DNS name

2. Start SAP Logon.
3. Choose **New**, then **Next**.
4. In the System Entry Properties box, for Connection Type, choose Group/Server Selection.
5. For **Message Server**, type the Network Load Balancer DNS name, and choose **OK**.

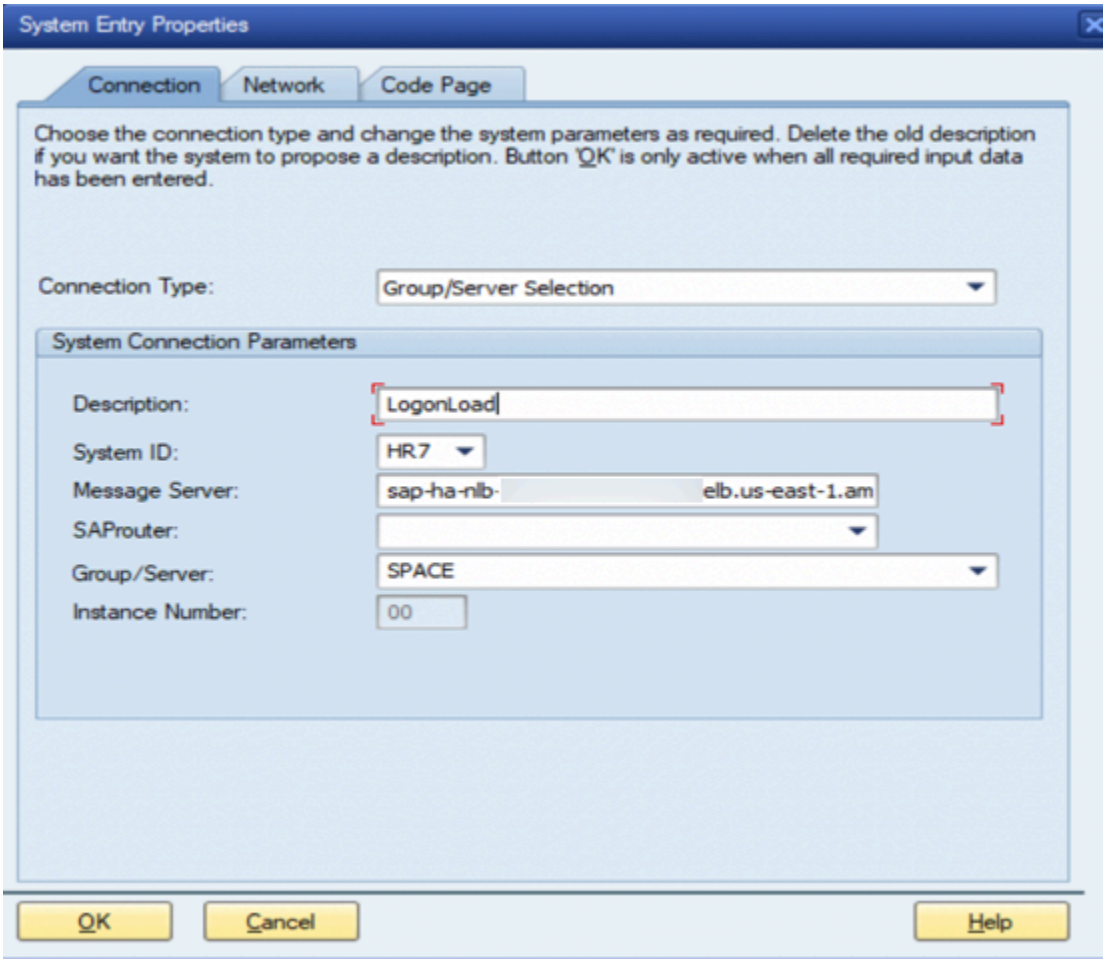


Figure 8: Configuring System Connection Parameters for SAP GUI

Step 5. Connect using SAP HANA Studio

1. In the **Load Balancers** section of the EC2 console, make a note of the Network Load Balancer DNS name for the JBDC/SQL and SAPStartSrv/HTTP ports.

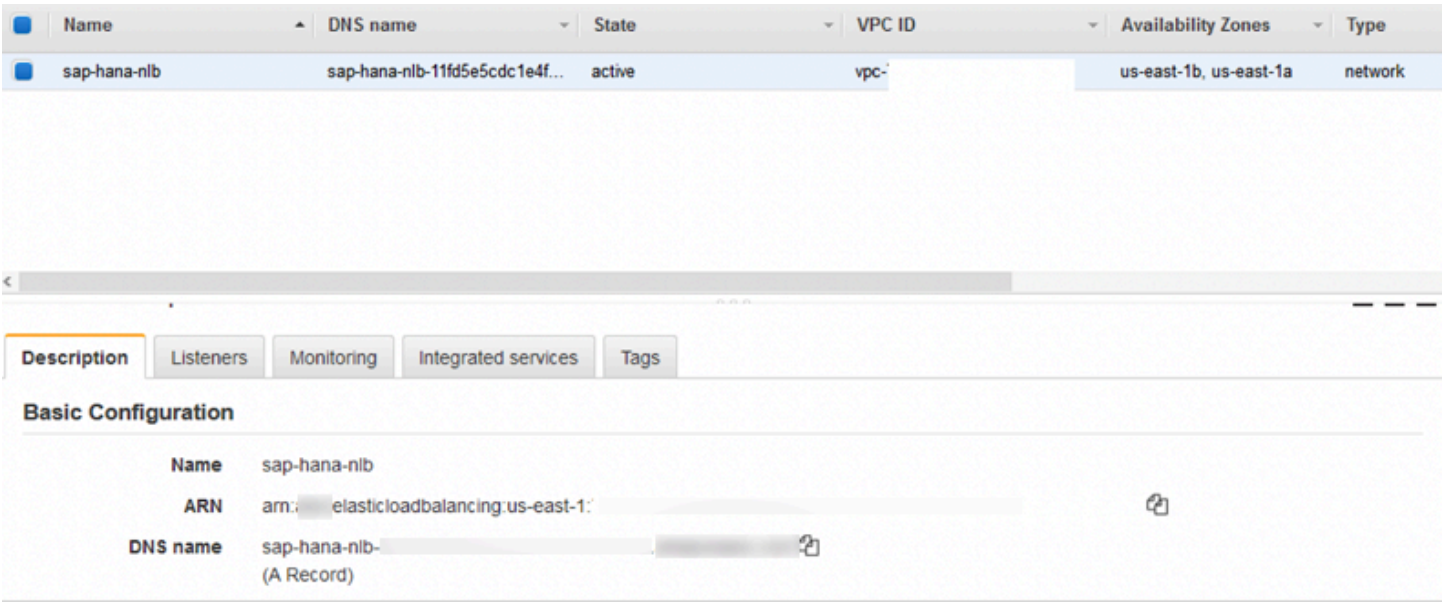


Figure 9: DNS name of ports

2. In the Host Name parameter of SAP HANA Studio, use the Network Load Balancer DNS name and provide additional credentials to connect to the SAP HANA system.

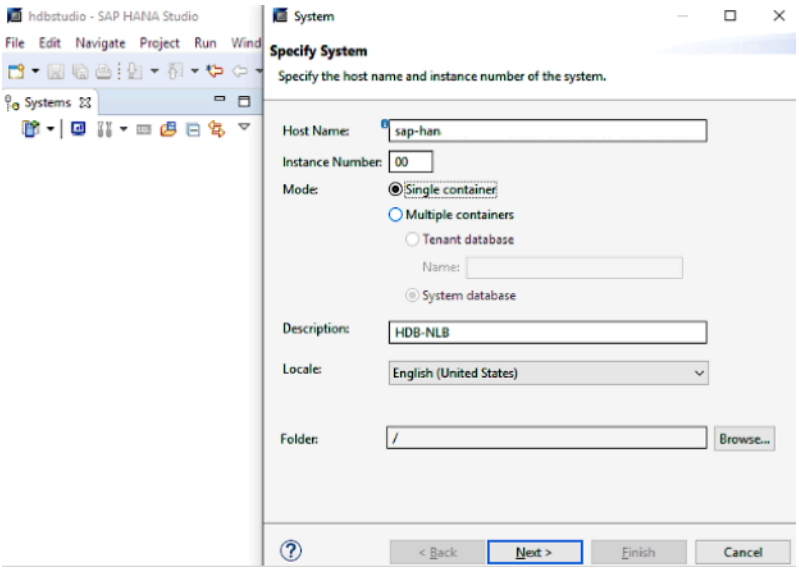


Figure 10: Updated Host Name in SAP HANA Studio

Additional Implementation Notes

- If other applications outside the VPC need to connect to the SAP system via the ASCS, create additional listeners with the ports on which these applications communicate.

- For customers using SAP Gateway Service (GW) and have designed HA for this service, create a target group for the GW service as well (33<instance-number>). Point the health check port for the GW target group to the message server port (36<instance-number>).
- You can use the corporate DNS or Amazon Route 53 Public Data Plane to create a user friendly CNAME for the Network Load Balancer DNS name. If you use an alias for connecting to the SAP GUI on-premises, the alias can be created as the CNAME for the Network Load Balancer DNS name. With this approach, there are no changes required on your SAP GUI configuration post migration to AWS. If other systems, such as SAP Landscape Management that requires a reverse lookup to function, are connecting to the highly available system, use A and PTR records instead of CNAME.

Document Revisions

Date	Change
June 2020	Minor updates to add additional links and to make a diagram clearer.
March 2020	Initial publication

SAP HANA on AWS: High Availability Configuration Guide for SLES and RHEL

SAP specialists, Amazon Web Services

First publication: March 25, 2021

This guide is part of a content series that provides detailed information about hosting, configuring, and using SAP technologies in the Amazon Web Services Cloud. For the other guides in the series, ranging from overviews to advanced topics, see the SAP on AWS Technical Documentation home page.

This guide provides guidance about how to set up AWS resources and configure a high availability cluster on SUSE Linux Enterprise Server (SLES) and Red Hat Enterprise Linux (RHEL) operating systems to deploy a highly available configuration of SAP HANA on Amazon Elastic Compute Cloud (Amazon EC2) instances in an existing virtual private cloud (VPC).

Automated deployment of SAP HANA on AWS with high availability

AWS Launch Wizard for SAP provides reference deployment for SAP HANA to fast-track your SAP HANA deployment on AWS. AWS Launch Wizard leverages AWS CloudFormation and scripts to quickly provision resources needed to deploy SAP HANA. It also encapsulates automated configuration of HANA System Replication (HSR) and SLES/RHEL high availability cluster with minimal manual intervention. For more information, see AWS Launch Wizard for SAP.

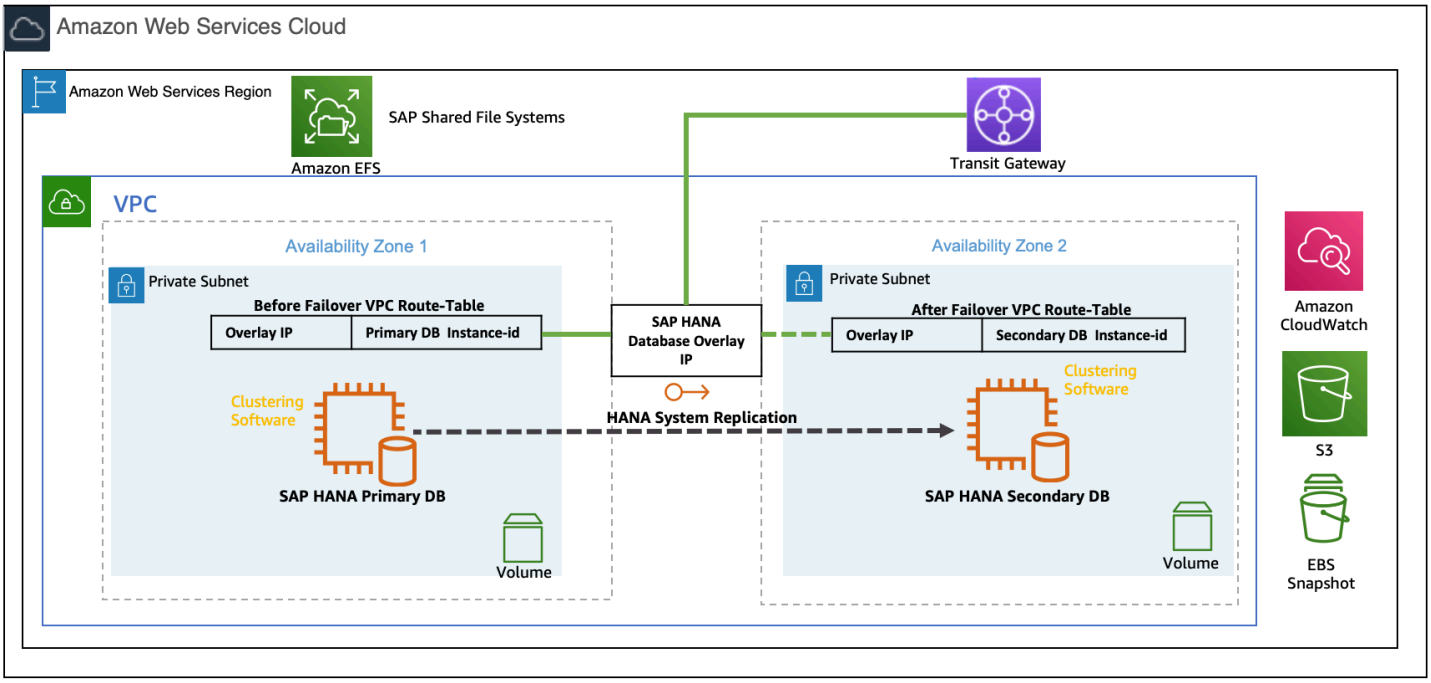
After you complete the deployment using Launch Wizard, you can follow the steps provided in these sections of the document to perform failover testing.

- the section called “Testing the cluster” (SLES)
- the section called “Testing the cluster” (RHEL)

Manual deployment of SAP HANA on AWS with high availability clusters

Architecture

This guide helps you configure high availability clusters on SLES or RHEL operating systems for your SAP HANA databases, deployed on Amazon EC2 instances in two different Availability Zones (AZs) within an AWS Region.



SAP HANA high availability cluster setup with Overlay IP

Operating systems

You can deploy your SAP workload on any of the following operating systems:

- SUSE Linux Enterprise Server (SLES) for SAP
- Red Hat Enterprise Linux for SAP with High Availability and Update Services (RHEL for SAP with HA and US)
- Red Hat Enterprise Linux for SAP Solutions (RHEL for SAP Solutions)

SLES for SAP and RHEL for SAP with HA and US are available in the AWS Marketplace with an hourly or an annual subscription model.

SUSE Linux Enterprise Server for SAP Applications (SLES for SAP)

SLES for SAP provides additional benefits, including Extended Service Pack Overlap Support (ESPOS), configuration and tuning packages for SAP applications, and High Availability Extensions (HAE). See the SUSE SLES for SAP product page.AWS strongly recommends using SLES for SAP instead of SLES for all your SAP workloads.

If you plan to use Bring Your Own Subscription (BYOS) images provided by SUSE, ensure that you have the registration code required to register your instance with SUSE to access repositories for software updates.

Red Hat Enterprise Linux (RHEL)

RHEL for SAP with HA and US provides access to Red Hat Pacemaker cluster software for High Availability, extended update support, and the libraries that are required to configure pacemaker cluster. For details, see the RHEL for SAP Offerings on AWS FAQ in the Red Hat knowledge base.

If you plan to use the BYOS model with RHEL, either through the Red Hat Cloud Access program or another means, ensure that you have access to a RHEL for SAP Solutions subscription. For details, see Overview of the Red Hat Enterprise Linux for SAP Solutions subscription in the Red Hat knowledge base.

The correct subscription is required to download the required packages for configuring the Pacemaker cluster.

SAP Notes

Refer to the SAP Note relevant to your choice of operating system.

SAP Note	Description
1656099	SAP Applications on AWS: Supported DB/OS and Amazon EC2 products
1984787	SUSE Linux Enterprise Server 12: Installation Notes
2578899	SUSE Linux Enterprise Server 15: Installation Notes
1275776	Linux: Preparing SLES for SAP environments
2002167	Red Hat Enterprise Linux 7.x: Installation and Upgrade
2772999	Red Hat Enterprise Linux 8.x: Installation and Configuration

AWS infrastructure, operating system setup and HANA installation

This guide mainly focuses on SAP HANA system replication setup and high availability cluster configuration steps on AWS. To set up AWS infrastructure, which is necessary to install primary and secondary SAP HANA databases, see the SAP HANA Environment Setup on AWS Guide and the following additional resources:

1. Amazon EC2 instances for SAP HANA
2. Storage recommendations for SAP HANA based on Amazon EC2 Instance Size
3. Deployment using AWS CLI and AWS Management Console
4. Operating system and storage configuration

After you have the AWS infrastructure ready, you will have to perform operating system configuration and installations of primary and secondary SAP HANA databases as per the architecture diagram in the previous section. SAP HANA installation steps are detailed in SAP Installation Guides and Setup Manuals available on the SAP Help Portal.

Hostname resolution

Ensure that both the systems are able to resolve the hostnames of both the cluster nodes. To fix any DNS issues, add the hostnames of both the cluster nodes to `/etc/hosts`.

```
  # cat /etc/hosts
10.0.0.1 prihana.example.com prihana
10.0.0.2 sechana.example.com sechana
```

Disable SAP HANA autostart

The high availability cluster will manage the start and stop operations for SAP HANA. Disable the autostart feature after installation is complete.

1. Login as the `sidadm` user on each cluster node and run the following command.

```
# HDB stop
# cdpro
```

2. Edit the SAP HANA profile file named `SID_HDB_instNum_hostname` and set the autostart property to **0**.
3. Save the profile file and start SAP HANA.

```
# HDB start
# cdpro
```

Configure SAP HANA System Replication (HSR)

Following are the high-level steps to setup HSR:

1. Enable HANA system replication for the database on the primary cluster node.
2. Register the secondary SAP HANA database node with the primary cluster node and start the secondary SAP HANA database.
3. Verify the state of replication.

The following values are used to configure HSR and high availability cluster in this example:

- Primary database host name – `prihana`
- Secondary database host name – `sechana`
- Database system identifier (DBSID) – `HDB`
- Instance number – `00`
- Site name of primary node – `PRI`
- Site name of secondary node – `SEC`

Enabling system replication in primary node

As `<sid>adm` user enables the system replication at the primary node:

```
hdbadm@prihana> hdbnsutil -sr_enable --name=PRI
```

Register the secondary node with the primary node

The SAP HANA database instance on the secondary cluster node must be stopped before registering the database instance for system replication.

After the database instance is stopped, you can register the instance using `hdbnsutil`. On the secondary node, the mode should be either "SYNC" or "SYNCMEM".

In the following example, the replication mode used is SYNC.

As a `<sid>adm` user, stop the secondary SAP HANA database, register the secondary node, and start the SAP HANA database:

```
hdbadm@sechana> HDB stop
hdbadm@sechana> hdbnsutil -sr_register --name=SEC \
--remoteHost=prihana --remoteInstance=00 \
--replicationMode=sync --operationMode=logreplay
hdbadm@sechana> HDB start
```

Verifying the state of system replication

You can use the `hdbnsutil` tool to check the system replication mode and site name:

```
hdbadm@prihana> hdbnsutil -sr_state
checking for active or inactive nameserver ...
System Replication State
~~~~~~~~~~~~~~~~~~~~~~~~
mode: primary site id: 1
site name: PRI
Host Mappings:
~~~~~~~~~~~~~~
done.
```

```
hdbadm@sechana> hdbnsutil -sr_state
checking for active or inactive nameserver ...
System Replication State
~~~~~~~~~~~~~~~~~~~~~~~~
mode: sync
site id: 2
site name: SEC
active primary site: 1
~~~~~~~~~~~~~~
```

You can view the replication state of the whole SAP HANA landscape using the following command as a `<sid>adm` user on the primary node:

```
hdbadm@prihana> HDBSettings.sh systemReplicationStatus.py --sapcontrol=1
...
site/2/SITE_NAME=SEC
site/2/SOURCE_SITE_ID=1
site/2/REPLICATION_MODE=SYNC
site/2/REPLICATION_STATUS=ACTIVE
site/1/REPLICATION_MODE=PRIMARY
site/1/SITE_NAME=PRI
local_site_id=1
...
```

Configuring system replication operation mode

When your SAP HANA database is connected as an SAPHanaSR target, you can find an entry in the `global.ini` which represents the operation mode.

To have your secondary site as a hot standby system, the operation mode configured must be '`logreplay`'.

For more details regarding all operation modes, see How To Perform System Replication for SAP HANA.

Ensure the operation_mode parameter is set to your desired operation mode in the `global.ini` configuration file on both the primary and secondary nodes.

The path for the `global.ini` is `/hana/shared/global/hdb/custom/config/`.

```
operation_mode = logreplay
```

Configuring the SAP HANA HA/DR provider hook

The following section is applicable if your SAP HANA database version is 2.0 and above. You can skip this section if your SAP HANA database is below version 2.0.

SAP HANA provides "hooks" that allows SAP HANA to send out notifications for certain events. A hook is used to improve the detection of when a takeover is required. Both SLES and RHEL provide such a hook in their respective resource packages which allows SAP HANA to report to the cluster

immediately if the secondary gets out of sync. These hooks must be configured on both nodes – primary and secondary. To integrate the HA/DR hook script with SAP HANA, you must stop the database and update the `global.ini` configuration file.

Implementing the Python hook SAPHanaSR in RHEL

As a `<sid>adm` user, stop the SAP HANA databases on both nodes, either with HDB or using `sapcontrol`, before proceeding further with changes, as seen in the following example.

```
hdbadm@prihana> sapcontrol -nr NN -function StopSystem
```

As a root user, copy the hook from the SAPHanaSR package into a read/writable directory on both nodes, as shown in the following example.

```
[root@prihana ~]# mkdir -p /hana/shared/myHooks
[root@prihana ~]# cp /usr/share/SAPHanaSR/srHook/SAPHanaSR.py /hana/shared/myHooks
[root@prihana ~]# chown -R hdbadm:sapsys /hana/shared/myHooks
```

Update the `global.ini` file **on each node** to enable use of the hook script by both SAP HANA instances. Ensure that you make a copy/backup of `global.ini` before updating the file.

See the following example for updating the global.ini at location (`/hana/shared/HDB/global/hdb/custom/config/global.ini`):

```
[ha_dr_provider_SAPHanaSR]
provider = SAPHanaSR
path = /hana/shared/myHooks
execution_order = 1

[trace]
ha_dr_saphanasr = info
```

The current version of the SAPHanaSR python hook uses the command `sudo` to allow the `<sid>adm` user to access the cluster attributes. To enable this, update the file `/etc/sudoers` as a root user with entries as shown in the following example:

```
# SAPHanaSR-ScaleUp entries for writing srHook cluster attribute
```

```
Cmnd_Alias SOK_SITEA   = /usr/sbin/crm_attribute -n hana_hdb_site_srHook_PRI -v SOK -t
 crm_config -s SAPHanaSR
Cmnd_Alias SFAIL_SITEA = /usr/sbin/crm_attribute -n hana_hdb_site_srHook_PRI -v SFAIL -
t crm_config -s SAPHanaSR
Cmnd_Alias SOK_SITEB   = /usr/sbin/crm_attribute -n hana_hdb_site_srHook_SEC -v SOK -t
 crm_config -s SAPHanaSR
Cmnd_Alias SFAIL_SITEB = /usr/sbin/crm_attribute -n hana_hdb_site_srHook_SEC -v SFAIL -
t crm_config -s SAPHanaSR
hdbadm ALL=(ALL) NOPASSWD: SOK_SITEA, SFAIL_SITEA, SOK_SITEB, SFAIL_SITEB
Defaults!SOK_SITEA, SFAIL_SITEA, SOK_SITEB, SFAIL_SITEB !requiretty
```

Note

hdb is the SAP HANA system ID used in the given example. You must replace hdb with *lowercase* SID of your SAP HANA installation. Replace the PRI and SEC references with your SAP HANA site names.

Implementing Python hook SAPHanaSR in SLES

Use the hook from the SAPHanaSR package. Optionally, you can copy it to your preferred directory; for example, /hana/share/myHooks. The hook must be available on all SAP HANA cluster nodes.

Stop the SAP HANA database, either with HDB or using sapcontrol, before proceeding further with changes, as shown in the following example.

```
hdbadm@prihana> sapcontrol -nr <instance_number> -function StopSystem
```

Update the global.ini file located at the /hana/shared/<SID>/global/hdb/custom/config/ directory on each node to enable the use of the hook script by both SAP HANA instances. Ensure that you make a copy/backup of global.inibefore updating the file.

```
[ha_dr_provider_SAPHanaSR]
provider = SAPHanaSR
path = /usr/share/SAPHanaSR
execution_order = 1

[trace]
```

```
ha_dr_saphanasr = info
```

The current version of the SAPHanaSR python hook uses the command sudo to allow the `<sid>adm` to access the cluster attributes. To enable this, edit and update the file `/etc/sudoers` as a root user with entries as shown in the following example:

```
# SAPHanaSR-ScaleUp entries for writing srHook cluster attribute
Cmnd_Alias SOK_SITEA = /usr/sbin/crm_attribute -n hana_hdb_site_srHook_PRI -v SOK -t
 crm_config -s SAPHanaSR
Cmnd_Alias SFAIL_SITEA = /usr/sbin/crm_attribute -n hana_hdb_site_srHook_PRI -v SFAIL -
t crm_config -s SAPHanaSR
Cmnd_Alias SOK_SITEB = /usr/sbin/crm_attribute -n hana_hdb_site_srHook_SEC -v SOK -t
 crm_config -s SAPHanaSR
Cmnd_Alias SFAIL_SITEB = /usr/sbin/crm_attribute -n hana_hdb_site_srHook_SEC -v SFAIL -
t crm_config -s SAPHanaSR
hdbadm ALL=(ALL) NOPASSWD: SOK_SITEA, SFAIL_SITEA, SOK_SITEB, SFAIL_SITEB
```

Note

hdb is the SAP HANA system ID used in the given example. You must replace hdb with *lowercase* `SID` of your SAP HANA installation. Replace the `PRI` and `SEC` references with your SAP HANA site names.

Cluster configuration prerequisites

Disable the source/destination check

Each EC2 instance performs source/destination checks by default. This means that the instance must be the source or destination of any traffic it sends or receives. For cluster instances, source/destination check must be disabled on both EC2 instances which are supposed to receive traffic from the Overlay IP address. You can use the AWS CLI or AWS Management Console to disable source/destination check. For details, see the ec2 modify-instance-attribute documentation.

Create a profile for AWS CLI

You need to create a profile for AWS CLI with the following command. This profile helps you run the cluster commands.

```
aws configure --profile cluster
```

The profile name must match the configuration of cluster resources, as seen in the following example.

```
primitive res_AWS_STONITH stonith:external/ec2 \
op start interval=0 timeout=180 \
op stop interval=0 timeout=180 \
op monitor interval=300 timeout=60 \
meta target-role=Started \
params tag=pacemaker profile=cluster pcmk_delay_max=45
```

AWS roles and policies

The SAP HANA database EC2 instances will run the SLES or RHEL cluster software and its agents. Because SLES and RHEL clustering software and its agents need to access AWS resources to perform failover activities, they need specific AWS IAM privileges.

Create a new IAM role and associate it to the two EC2 instances which are part of the cluster. Attach the following IAM policies to this IAM role.

Create the `STONITH` policy

Both instances of the cluster need the privilege to start and stop the other nodes within the cluster. Create a policy as shown in the following example and attach it to the IAM role which is assigned to both cluster instances.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Stmt1424870324000",
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeInstances",
                "ec2:DescribeTags"
            ],
            "Resource": "*"
```

```
        },
        {
            "Sid": "Stmt1424870324001",
            "Effect": "Allow",
            "Action": [
                "ec2:RebootInstances",
                "ec2:StartInstances",
                "ec2:StopInstances"
            ],
            "Resource": [
                "arn:aws:ec2:region-name:account-id:instance/instance-a",
                "arn:aws:ec2:region-name:account-id:instance/instance-b"

            ]
        }
    ]
},
{
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "EC2:DescribeInstances",
                "EC2:DescribeVolumes"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "cloudwatch:GetMetricStatistics",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "s3:GetObject",
            "Resource": "arn:aws:s3:::aws-sap-data-provider/config.properties"
        }
    ]
}
```

Replace region name, `account-id`, and instance identifier with the appropriate values.

Create an overlay IP agent policy

Amazon VPC setup includes assigning subnets to your primary/secondary nodes for the SAP HANA database. Each of these configured subnets has a classless inter-domain routing (CIDR) IP assignment from the VPC which resides entirely within one Availability Zone. This CIDR IP assignment cannot span multiple zones or be reassigned to the secondary instance in a different AZ during a failover scenario. For this reason,AWS enables you to configure Overlay IP (OIP) outside of your VPC CIDR block to access the active SAP instance. With IP overlay routing, you can allow the AWS network to use a non-overlapping RFC1918 private IP address that resides outside an VPC CIDR range and direct the SAP traffic to any instance setup across the Availability Zone within the VPC by changing the routing entry in AWS using SLES/RHEL Overlay IP agent.

For the SLES/RHEL Overlay IP agent to change a routing entry in AWS routing tables, create the following policy and attach to the IAM role which is assigned to both cluster instances:

```
        {
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": "ec2:ReplaceRoute",
            "Resource": "arn:aws:ec2:region-name:account-id:route-table/rtb-XYZ"
        },
        {
            "Sid": "VisualEditor1",
            "Effect": "Allow",
            "Action": "ec2:DescribeRouteTables",
            "Resource": "*"
        }
    ]
}
```

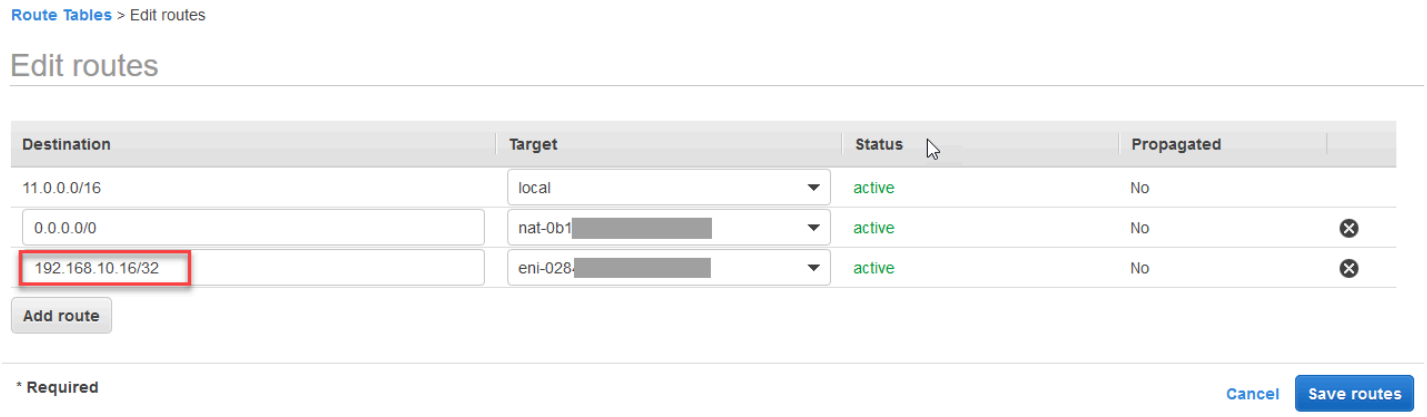
Replace region name, account-id, and route table identifiers with appropriate values.

Update routing tables

Add a routing entry to the routing tables which are assigned to the subnets of your primary and secondary EC2 instances. This IP address is the virtual IP (overlay IP) address of the SAP HANA

cluster which needs to be outside the CIDR range of the VPC. To modify or add a route to a route table using the console:

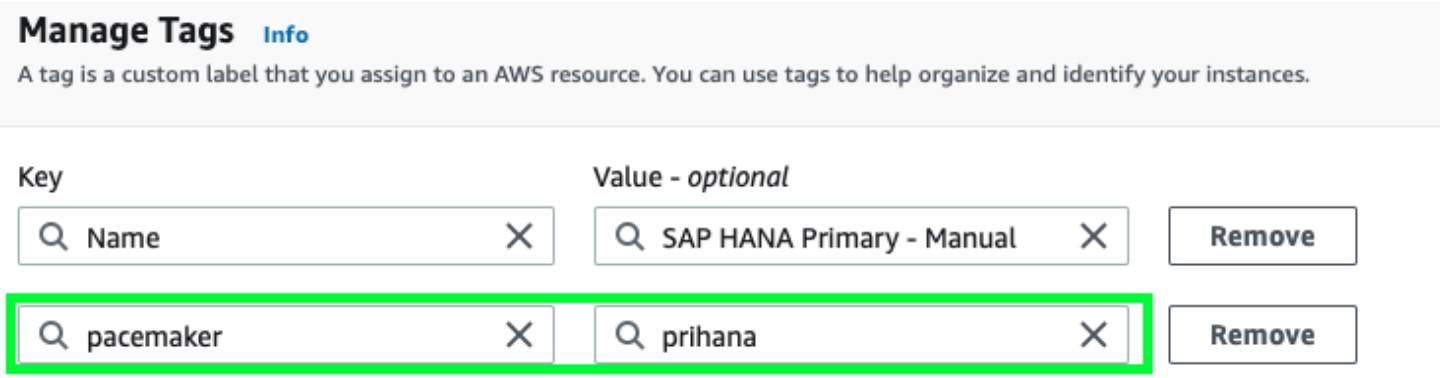
1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/ (signin required).
2. In the navigation pane, choose **Route Tables**, and select the route table.
3. Choose **Actions** > **Edit routes**.
4. Scroll to the end of the list and click **Add another route**.
5. Add the overlay IP address in the **Destination** section and select **Elastic Network Interface (ENI) name** for one of your existing instances.
6. Save your changes by clicking **Save routes**.



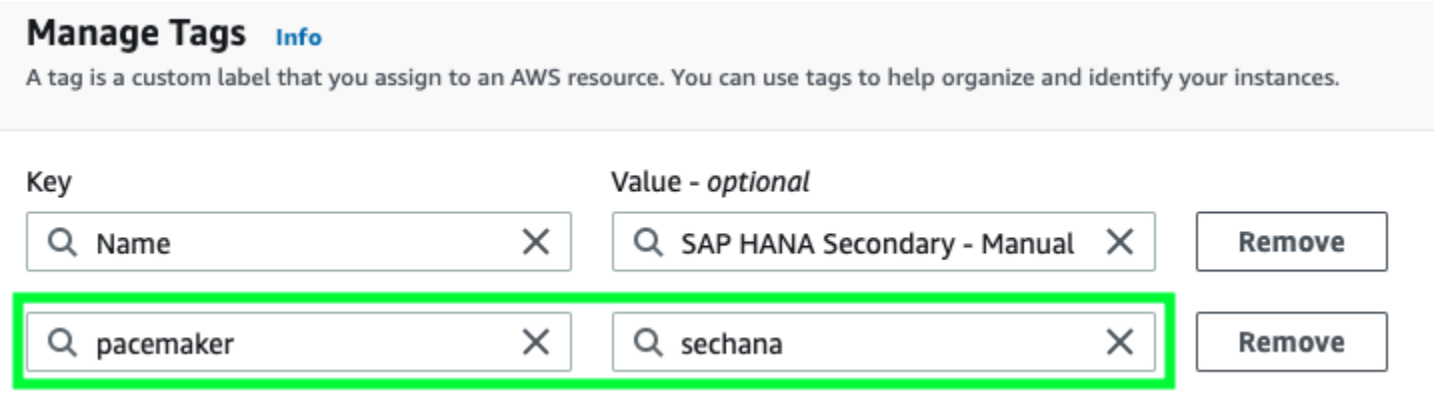
Overlay-IP address entry in route table

Tagging the EC2 instances (required only for SLES)

In SLES, AWS EC2 `STONITH` agents use AWS resource tags to identify the EC2 instances. Create a tag for the primary and secondary EC2 instances through the console or the AWS CLI. In the following example the user has chosen `pacemaker` and the hostname, which is shown in the command `uname -n`.



Tagging the primary database EC2 instance



Tagging the secondary database EC2 instance

HA cluster configuration on SLES

These instructions are applicable for SUSE Linux Enterprise Server for SAP Applications 12 and SUSE Linux Enterprise Server for SAP Applications 15.

Cluster installation

SLES for SAP Images sold by AWS through AWS Marketplace comes with pre-installed SUSE HAE packages. Ensure you have the latest version of the following packages. If needed, update them using the `zypper` command. If you are using BYOS images, ensure that the following packages are installed:

- `corosync`
- `crmsh`
- `fence-agents`
- `ha-cluster-bootstrap`

- `pacemaker`
- `patterns-ha-ha_sles`
- `resource-agents`
- `cluster-glue`

Cluster configuration

Topics

- System logging
- Corosync configuration
- Create encryption keys
- Create secondary IP addresses for a redundant cluster ring
- Review instance settings that conflict with cluster actions
- Create the Corosync configuration file
- Update the hacluster password
- Start the cluster

System logging

SUSE recommends using the rsyslogd daemon for logging in the SUSE cluster. Install the `rsyslog` package as a root user on all cluster nodes. `logd` is a subsystem to log additional information coming from the `STONITH` agent:

```
prihana:~ # zypper install rsyslog
prihana:~ # systemctl enable logd
prihana:~ # systemctl start logd
```

Corosync configuration

The cluster service (Pacemaker) should be in a stopped state when performing cluster configuration. Check the status and stop the Pacemaker service if it is running.

- This is the command to check the Pacemaker status:

```
prihana:~ # systemctl status pacemaker
```

- This is the command to stop Pacemaker:

```
prihana:~ # systemctl stop pacemaker
```

Create encryption keys

Run the following command to create a secret key which is used to encrypt all the cluster communication:

```
prihana:~ # corosync-keygen
```

A new key file called "`authkey`" is created at location `/etc/corosync/`. Copy this file to the same location on the second cluster node with the same permissions and ownership.

Create secondary IP addresses for a redundant cluster ring

For SUSE clusters, we recommend defining a redundant communication channel (a second ring) in corosync which the cluster nodes can use to communicate in case of disruptions.

To create a redundant communication channel, you must add a secondary IP address on both the nodes. These IPs are only used in cluster configurations. They provide the same fault tolerance as a secondary Elastic Network Interface (ENI). For more information, see Assign a secondary private IPv4 address.

Review instance settings that conflict with cluster actions

To ensure that restarts are predictable, we recommend disabling simplified automatic recovery and *not* configuring Amazon CloudWatch action based recovery for instances that are part of a pacemaker cluster. Use the following command to disable simplified automatic recovery.

```
aws ec2 modify-instance-maintenance-options --instance-id i-0abcdef1234567890 --auto-
recovery disabled
```

You must ensure that stop protection is disabled for Amazon EC2 instances that are part of a pacemaker cluster. Use the following command to disable stop protection.

```
aws ec2 modify-instance-attribute --instance-id i-1234567890abcdef0 --no-disable-api-
stop
```

Create the Corosync configuration file

All cluster nodes are required to have a local configuration file "`/etc/corosync/corosync.conf`", as shown in the following example.

```
prihana:/etc/corosync # cat corosync.conf
# Please read the corosync.conf.5 manual page
totem {
        version: 2
        token: 30000
        consensus: 36000
        token_retransmits_before_loss_const: 6
        crypto_cipher: none
        crypto_hash: none
        clear_node_high_bit: yes
        rrp_mode: passive

        interface {
                ringnumber: 0
                bindnetaddr: 11.0.1.132
                mcastport: 5405
                ttl: 1
        }
        transport: udpu
}
logging {
        fileline: off
        to_logfile: yes
        to_syslog: yes
        logfile: /var/log/cluster/corosync.log
        debug: off
        timestamp: on
        logger_subsys {
                subsys: QUORUM
                debug: off
        }
}
nodelist {
        node {
                ring0_addr: 11.0.1.132
                ring1_addr: 11.0.1.75
                nodeid: 1
        }
        node {
```

```
                ring0_addr: 11.0.2.139
                ring1_addr: 11.0.2.35
                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_node: 1
}
```

Replace the values for the following variables with those for your environment:

- `bindnetaddr` — IP address of the node where the file is being configured.
- `ring0_addr` — Primary IP address of cluster node 1.
- `ring1_addr` — Secondary IP address of cluster node 1.
- `ring0_addr` — Primary IP address of cluster node 2.
- `ring1_addr` — Secondary IP address of cluster node 2.

Also update the value of for `crypto_cipher` and `crypto_hash` as per your encryption requirements.

Update the `hacluster` password

Change the password of the user `haclustser` on both the nodes as shown in the following example:

```
prihana:~ # passwd hacluster
```

```
sechana:~ # passwd hacluster
```

Start the cluster

Start the cluster on both the primary and secondary nodes and check the status.

- This is the command to check the Pacemaker status:

```
prihana:~ # systemctl status pacemaker
```

- This is the command to start Pacemaker:

```
prihana:~ # systemctl start pacemaker
```

After the cluster service (Pacemaker) is started, check the cluster status with the `crm_mon` command as shown in the following example. You will see both nodes online and a full list of resources.

```
prihana:~ # crm_mon -r
Stack: corosync
Current DC: prihana (version 1.1.18+20180430.b12c320f5-3.24.1-b12c320f5) - partition
 with quorum
Last updated: Wed Nov 11 16:20:40 2020
Last change: Wed Nov 11 16:20:21 2020 by root via crm_attribute on sechana

2 nodes configured
0 resources configured

Online: [ prihana sechana ]

Full list of resources:

No resources
```

You can find the ring status and the associated IP address of the cluster with the `corosync-cfgtool` command as shown in the following example:

```
prihana:~ # corosync-cfgtool -s
Printing ring status.
Local node ID 1
RING ID 0
        id      = 11.0.1.132
        status  = ring 0 active with no faults
RING ID 1
        id      = 11.0.1.75
        status  = ring 1 active with no faults
```

Cluster resources

This section describes how to configure the bootstrap, STONITH, resources, and constraints using the `crm` command. You can use the command "`crm`" to add objects.

Cluster the bootstrap

Create a file called "`crm-bs.txt`" with the following cluster bootstrap options:

```
prihana:~ # cat crm-bs.txt
property $id="cib-bootstrap-options" \
   stonith-enabled="true" \
   stonith-action="off" \
   stonith-timeout="600s"
rsc_defaults $id="rsc-options" \
   resource-stickiness="1000" \
   migration-threshold="5000"
op_defaults $id="op-options" \
   timeout="600"
```

Setting the `stonith-action` parameter value to "off" forces the agents to shut down the instance during failover. This is desirable to avoid split brain scenarios.

Add the cluster bootstrap configuration to the cluster with the following command:

```
prihana:~ # crm configure load update crm-bs.txt
```

STONITH

Create a file called "`aws-stonith.txt`" with the following STONITH options:

```
prihana:~ # cat aws-stonith.txt
primitive res_AWS_STONITH stonith:external/ec2 \
        op start interval=0 timeout=180 \
        op stop interval=0 timeout=180 \
        op monitor interval=300 timeout=60 \
        meta target-role=Started \
        params tag=pacemaker profile=cluster pcmk_delay_max=45
```

Ensure the value parameter "`tag`" matches the tag key you created for your EC2 instance in the "Prerequisites" section. In this example, "`pacemaker`" is used for the parameter tag. The name of the profile "`cluster`" needs to be matched with the configured AWS profile.

Add the STONITH configuration file to the cluster with the following command:

```
prihana:~ # crm configure load update aws-stonith.txt
```

Overlay IP resource

Create a file called "aws-move-ip.txt" with the following cluster bootstrap options to move IP resources during failover:

```
prihana:~ # cat aws-move-ip.txt
primitive res_AWS_IP ocf:suse:aws-vpc-move-ip \
params ip=<overlay ip address> routing_table=<route table identifier 1>,
<route table identifier 2>  interface=eth0 profile=cluster \
op start interval=0 timeout=180 \
op stop interval=0 timeout=180 \
op monitor interval=60 timeout=60
```

Replace the value for parameters ip and routing_table with your overlay IP address and route table names.

Add the move IP configuration file to the cluster with the following command:

```
prihana:~ # crm configure load update aws-move-ip.txt
```

You can also use multiple Amazon VPC routing tables in the routing_table table parameter, as shown in the following example.

```
primitive res_AWS_IP ocf:suse:aws-vpc-move-ip \
   params ip=x.x.x.x \
   routing_table=rtb-xxxxxxxxx,rtb-yyyyyyyyy,rtb-zzzzzzzzz \
   interface=eth0 profile=cluster \
   op start interval=0 timeout=180 \
   op stop interval=0 timeout=180 \
   op monitor interval=60 timeout=60
```

SAPHanaTopology

Create a file called "crm-saphanatop.txt" with the following cluster bootstrap options for SAP HANA topology information:

```
prihana:~ # cat crm-saphanatop.txt
primitive rsc_SAPHanaTopology_HDB_HDB00 ocf:suse:SAPHanaTopology \
        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"
```

Update the value of parameters `SID` and `InstanceNumber` with your SAP HANA system information. In addition, update the SID and Instance number referred in the `rsc_SAPHanaTopology_<SID>HDB<Instance Number>` and `cln_SAPHanaTopology_<SID>_HDB<Instance Number>` configurations. Tune the timeout parameters (`start`, `stop`, and `monitor`) for your environment.

Add the SAP HANA topology configuration file to the cluster with the following command:

```
prihana:~ # crm configure load update crm-saphanatop.txt
```

SAPHana

Create a file called "`crm-saphana.txt`" with the following cluster bootstrap options for SAP HANA:

```
prihana:~ # cat crm-saphana.txt
primitive rsc_SAPHana_HDB_HDB00 ocf:suse:SAPHana \
        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="10" PREFER_SITE_TAKEOVER="true" \
        DUPLICATE_PRIMARY_TIMEOUT="7200" AUTOMATED_REGISTER="true"
ms msl_SAPHana_HDB_HDB00 rsc_SAPHana_HDB_HDB00 \
        meta clone-max="2" clone-node-max="1" interleave="true"
```

Update the value of parameters `SID` and `InstanceNumber` with your SAP HANA system information. In addition, update the SID and Instance number referred in the `rsc_SAPHana_<SID>HDB<Instance Number>` and `msl_SAPHana<SID>_HDB<Instance Number>` configuration.

> **Note**
>
> You can find the detailed information about all the parameters with the command "man ocf_suse_SAPHana"

Add the SAP HANA configuration file to the cluster with the following command:

```
prihana:~ # crm configure load update crm-saphana.txt
```

When using an SAP HANA version with systemd integration (SPS07 and later), you must run the following steps to prevent the nodes from being fenced when Amazon EC2 instances are intentionally stopped.

1. Verify if SAP HANA is integrated with systemd. If it is integrated, a systemd service name, such as SAP<SID>_XX.service is present. For example, for SID HDB and instance number 00, SAPHDB_00.service is the service name.

   Use the following command as root to find SAP systemd services.

   ```
   prihanadb:~ # systemctl list-units | grep SAP
     aws-dataprovider.service                                                    loaded
    active running   AWS Data Provider for SAP
     pacemaker.service                                                          loaded
    active running   pacemaker needs SAP instance service
     SAPHDB_00.service                                                          loaded
    active running   SAP Instance SAPHDB_00
     saphostagent.service                                                       loaded
    active running   SAP Host Agent
     SAP.slice                                                                  loaded
    active active    SAP Slice
   ```

2. Create a pacemaker service drop-in file.

   ```
   # mkdir -p /etc/systemd/system/pacemaker.service.d/
   cat <<_EOF > /etc/systemd/system/pacemaker.service.d/00-pacemaker.conf
   [Unit]
   Description=pacemaker needs SAP instance service
   Documentation=man:SAPHanaSR_basic_cluster(7)
   Wants=SAP<SID>_XX.service
   After=SAP<SID>_XX.service
   ```

```
_EOF
```

3. Enable the drop-in file by reloading `systemd`.

```
# systemctl daemon-reload
```

4. Verify that the change is active.

```
# systemctl show pacemaker.service | grep SAP<SID>_XX
```

For example, for SID HDB and instance number `00`, the following output is expected.

```
# systemctl show pacemaker.service | grep SAPHDB_00
Wants=SAPHDB_00.service resource-agents-deps.target dbus.service
After=system.slice network.target corosync.service resource-agents-deps.target
 basic.target rsyslog.service SAPHDB_00.service systemd-journald.socket
 sysinit.target time-sync.target dbus.service sbd.service
```

Constraints

Define two constraints, one for the Overlay IP address which helps with routing client traffic to active database host and the second one for the start order between the SAPHANA and SAPHANATopology resource agents.

Create a file called "`crm-cs.txt`" with following cluster bootstrap options for constraints:

```
prihana:~ # cat crm-cs.txt
colocation col_IP_Primary 2000: res_AWS_IP:Started msl_SAPHana_HDB_HDB00:Master
order ord_SAPHana 2000: cln_SAPHanaTopology_HDB_HDB00 msl_SAPHana_HDB_HDB00
```

Update the SID and `Instance` number referred in `cln_SAPHanaTopology_<SID>_HDB<Instance Number>` and `msl_SAPHana_<SID>_HDB<Instance Number>` configuration.

Add the constraints configuration file to the cluster with the following command:

```
prihana:~ # crm configure load update crm-cs.txt
```

Cluster status

After the cluster is configured, you should see two online nodes, and six resources. You can check it with the following command:

```
prihana:~ # crm status
Stack: corosync
Current DC: prihana (version 1.1.18+20180430.b12c320f5-3.24.1-b12c320f5) - partition
 with quorum
Last updated: Thu Nov 12 11:37:20 2020
Last change: Thu Nov 12 11:37:11 2020 by hacluster via crmd on sechana

2 nodes configured
6 resources configured

Online: [ prihana sechana ]

Full list of resources:

 res_AWS_STONITH        (stonith:external/ec2): Started prihana
 res_AWS_IP     (ocf::suse:aws-vpc-move-ip):    Started prihana
 Clone Set: cln_SAPHanaTopology_HDB_HDB00 [rsc_SAPHanaTopology_HDB_HDB00]
     Started: [ prihana sechana ]
 Master/Slave Set: msl_SAPHana_HDB_HDB00 [rsc_SAPHana_HDB_HDB00]
     Masters: [ prihana ]
     Slaves: [ sechana ]
```

You can check the status of the replication by executing the `crm_mon` command as shown in the following example. Ensure that the state of the replication in the secondary node is "SOK".

```
prihana:~ # crm_mon -A1
Stack: corosync
Current DC: prihana (version 1.1.18+20180430.b12c320f5-3.24.1-b12c320f5) - partition
 with quorum
Last updated: Thu Nov 12 11:38:25 2020
Last change: Thu Nov 12 11:37:33 2020 by root via crm_attribute on prihana

2 nodes configured
6 resources configured

Online: [ prihana sechana ]

Active resources:
```

```
 res_AWS_STONITH        (stonith:external/ec2): Started prihana
 res_AWS_IP     (ocf::suse:aws-vpc-move-ip):    Started prihana
 Clone Set: cln_SAPHanaTopology_HDB_HDB00 [rsc_SAPHanaTopology_HDB_HDB00]
     Started: [ prihana sechana ]
 Master/Slave Set: msl_SAPHana_HDB_HDB00 [rsc_SAPHana_HDB_HDB00]
     Masters: [ prihana ]
     Slaves: [ sechana ]

Node Attributes:
* Node prihana:
    + hana_hdb_clone_state              : PROMOTED
    + hana_hdb_op_mode                  : logreplay
    + hana_hdb_remoteHost               : sechana
    + hana_hdb_roles                    : 4:P:master1:master:worker:master
    + hana_hdb_site                     : PRI
    + hana_hdb_srmode                   : sync
    + hana_hdb_sync_state               : PRIM
    + hana_hdb_version                  : 2.00.030.00.1522209842
    + hana_hdb_vhost                    : prihana
    + lpa_hdb_lpt                       : 1605181053
    + master-rsc_SAPHana_HDB_HDB00      : 150
* Node sechana:
    + hana_hdb_clone_state              : DEMOTED
    + hana_hdb_op_mode                  : logreplay
    + hana_hdb_remoteHost               : prihana
    + hana_hdb_roles                    : 4:S:master1:master:worker:master
    + hana_hdb_site                     : SEC
    + hana_hdb_srmode                   : sync
    + hana_hdb_sync_state               : SOK
    + hana_hdb_version                  : 2.00.030.00.1522209842
    + hana_hdb_vhost                    : sechana
    + lpa_hdb_lpt                       : 30
    + master-rsc_SAPHana_HDB_HDB00      : 100
```

Testing the cluster

After the cluster setup is complete, perform the following tests to validate cluster setup. Run these tests in sequence.

- the section called “Stop the SAP HANA database on the primary node”

- the section called “Stop the SAP HANA database on the secondary node”
- the section called “Crash the primary SAP HANA database on node 1”
- the section called “Crash the primary database on node 2”
- the section called “Reboot SAP HANA on node 1”
- the section called “Reboot SAP HANA on node 2”
- the section called “Simulating a cluster network failure”

Stop the SAP HANA database on the primary node

Description — Stop the primary SAP HANA database during normal cluster operation.

Run node — Primary SAP HANA database node

Run steps:

- Stop the primary SAP HANA database gracefully as `<sid>adm`.

```
prihana:~ # su - hdbadm
hdbadm@prihana:/usr/sap/HDB/HDB00> HDB stop
hdbdaemon will wait maximal 300 seconds for NewDB services finishing.
Stopping instance using: /usr/sap/HDB/SYS/exe/hdb/sapcontrol -prot
NI_HTTP -nr 00 -function Stop 400

12.11.2020 11:39:19
Stop
OK
Waiting for stopped instance using: /usr/sap/HDB/SYS/exe/hdb/sapcontrol
-prot NI_HTTP -nr 00 -function WaitforStopped 600 2

12.11.2020 11:39:51
WaitforStopped
OK
hdbdaemon is stopped.
```

Expected result:

- The cluster detects stopped primary SAP HANA database (on node 1) and promotes secondary SAP HANA database (on node 2) to take over as primary.

```
prihana:~ # crm status
Stack: corosync
Current DC: prihana (version 1.1.18+20180430.b12c320f5-3.24.1-b12c320f5) -
partition with quorum
Last updated: Thu Nov 12 11:41:31 2020
Last change: Thu Nov 12 11:41:30 2020 by root via crm_attribute on sechana

2 nodes configured
6 resources configured

Online: [ prihana sechana ]

Full list of resources:

 res_AWS_STONITH        (stonith:external/ec2): Started prihana
 res_AWS_IP     (ocf::suse:aws-vpc-move-ip):    Started sechana
 Clone Set: cln_SAPHanaTopology_HDB_HDB00 [rsc_SAPHanaTopology_HDB_HDB00]
     Started: [ prihana sechana ]
 Master/Slave Set: msl_SAPHana_HDB_HDB00 [rsc_SAPHana_HDB_HDB00]
     Masters: [ sechana ]
     Slaves: [ prihana ]

Failed Actions:
* rsc_SAPHana_HDB_HDB00_monitor_60000 on prihana 'master (failed)' (9):
call=30, status=complete, exitreason='',
    last-rc-change='Thu Nov 12 11:40:42 2020', queued=0ms, exec=0ms
```

- The overlay IP address is migrated to the new primary (on node 2).

```
sechana:~ # ip addr show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
 qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
```

```
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9000 qdisc mq state UP group default
 qlen 1000
    link/ether 0e:ef:dd:3c:bf:1b brd ff:ff:ff:ff:ff:ff
    inet 11.0.2.139/24 brd 11.0.2.255 scope global eth0
       valid_lft forever preferred_lft forever
    inet 11.0.2.35/32 scope global eth0:1
       valid_lft forever preferred_lft forever
    inet 192.168.10.16/32 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::cef:ddff:fe3c:bf1b/64 scope link
       valid_lft forever preferred_lft forever
```

- With the AUTOMATIC_REGISTER parameter set to "`true`", the cluster restarts the failed SAP HANA database and automatically registers it against the new primary.

Recovery procedure:

- Clean up the cluster "`failed actions`" on node 1 as root.

```
prihana:~ # crm resource cleanup rsc_SAPHana_HDB_HDB00 prihana
Cleaned up rsc_SAPHana_HDB_HDB00:0 on prihana
Cleaned up rsc_SAPHana_HDB_HDB00:1 on prihana
Waiting for 1 replies from the CRMd. OK
```

- After you run the `crm` command to clean up the resource, "`failed actions`" messages should disappear from the cluster status.

```
prihana:~ # crm status
Stack: corosync
Current DC: prihana (version 1.1.18+20180430.b12c320f5-3.24.1-b12c320f5) - partition
 with quorum
Last updated: Thu Nov 12 11:44:05 2020
Last change: Thu Nov 12 11:43:39 2020 by hacluster via crmd on prihana

2 nodes configured
6 resources configured

Online: [ prihana sechana ]
```

```
Full list of resources:

 res_AWS_STONITH        (stonith:external/ec2): Started prihana
 res_AWS_IP     (ocf::suse:aws-vpc-move-ip):    Started sechana
 Clone Set: cln_SAPHanaTopology_HDB_HDB00 [rsc_SAPHanaTopology_HDB_HDB00]
     Started: [ prihana sechana ]
 Master/Slave Set: msl_SAPHana_HDB_HDB00 [rsc_SAPHana_HDB_HDB00]
     Masters: [ sechana ]
     Slaves: [ prihana ]
```

Stop the SAP HANA database on the secondary node

Description — Stop the primary SAP HANA database (on Node 2) during normal cluster operation.

Run node — Primary SAP HANA database node (on Node 2)

Run steps:

- Stop the SAP HANA database gracefully as `<sid>adm` on node 2.

```
sechana:~ # su - hdbadm
hdbadm@sechana:/usr/sap/HDB/HDB00> HDB stop
hdbdaemon will wait maximal 300 seconds for NewDB services finishing.
Stopping instance using: /usr/sap/HDB/SYS/exe/hdb/sapcontrol -prot
NI_HTTP -nr 00 -function Stop 400

12.11.2020 11:45:21
Stop
OK
Waiting for stopped instance using: /usr/sap/HDB/SYS/exe/hdb/sapcontrol
-prot NI_HTTP -nr 00 -function WaitforStopped 600 2


12.11.2020 11:45:53
WaitforStopped
OK
hdbdaemon is stopped.
```

Expected result:

- The cluster detects stopped primary SAP HANA database (on node 2) and promotes the secondary SAP HANA database (on node 1) to take over as primary.

```
sechana:~ # crm status
Stack: corosync
Current DC: prihana (version 1.1.18+20180430.b12c320f5-3.24.1-b12c320f5)
- partition with quorum
Last updated: Thu Nov 12 11:47:38 2020
Last change: Thu Nov 12 11:47:33 2020 by root via crm_attribute on prihana

2 nodes configured
6 resources configured

Online: [ prihana sechana ]

Full list of resources:

 res_AWS_STONITH        (stonith:external/ec2): Started prihana
 res_AWS_IP     (ocf::suse:aws-vpc-move-ip):    Started prihana
 Clone Set: cln_SAPHanaTopology_HDB_HDB00 [rsc_SAPHanaTopology_HDB_HDB00]
     Started: [ prihana sechana ]
 Master/Slave Set: msl_SAPHana_HDB_HDB00 [rsc_SAPHana_HDB_HDB00]
     Masters: [ prihana ]
     Slaves: [ sechana ]

Failed Actions:
* rsc_SAPHana_HDB_HDB00_monitor_60000 on sechana 'master (failed)' (9):
call=46, status=complete, exitreason='',
    last-rc-change='Thu Nov 12 11:46:45 2020', queued=0ms, exec=0ms
```

- The overlay IP address is migrated to the new primary (on node 1).

```
prihana:~ # ip addr show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
 qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9000 qdisc mq state UP group default
 qlen 1000
    link/ether 0a:38:1c:ce:b4:3d brd ff:ff:ff:ff:ff:ff
```

```
    inet 11.0.1.132/24 brd 11.0.1.255 scope global eth0
       valid_lft forever preferred_lft forever
    inet 11.0.1.75/32 scope global eth0:1
       valid_lft forever preferred_lft forever
    inet 192.168.10.16/32 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::838:1cff:fece:b43d/64 scope link
       valid_lft forever preferred_lft forever
```

- With the `AUTOMATIC_REGISTER` parameter set to "`true`", the cluster restarts the failed SAP HANA database and automatically registers it against the new primary.

Recovery procedure:

- After you run the `crm` command to clean up the resource, "`failed actions`" messages should disappear from the cluster status.

```
sechana:~ # crm resource cleanup rsc_SAPHana_HDB_HDB00 sechana
Cleaned up rsc_SAPHana_HDB_HDB00:0 on sechana
Cleaned up rsc_SAPHana_HDB_HDB00:1 on sechana
Waiting for 1 replies from the CRMd. OK
```

- After resource cleanup, the cluster "`failed actions`" are cleaned up.

```
sechana:~ # crm status
Stack: corosync
Current DC: prihana (version 1.1.18+20180430.b12c320f5-3.24.1-b12c320f5) - partition
 with quorum
Last updated: Thu Nov 12 11:50:05 2020
Last change: Thu Nov 12 11:49:39 2020 by root via crm_attribute on prihana

2 nodes configured
6 resources configured

Online: [ prihana sechana ]

Full list of resources:

 res_AWS_STONITH        (stonith:external/ec2): Started prihana
```

```
res_AWS_IP     (ocf::suse:aws-vpc-move-ip):    Started prihana
Clone Set: cln_SAPHanaTopology_HDB_HDB00 [rsc_SAPHanaTopology_HDB_HDB00]
    Started: [ prihana sechana ]
Master/Slave Set: msl_SAPHana_HDB_HDB00 [rsc_SAPHana_HDB_HDB00]
    Masters: [ prihana ]
    Slaves: [ sechana ]
```

Crash the primary SAP HANA database on node 1

Description: Simulate a complete breakdown of the primary database system.

Run node: Primary SAP HANA database node

Run steps:

- Stop the primary database system using the following command as <sid>adm.

```
prihana:~ # sudo su - hdbadm
hdbadm@prihana:/usr/sap/HDB/HDB00> HDB kill -9
hdbenv.sh: Hostname prihana defined in $SAP_RETRIEVAL_PATH=/usr/sap/
HDB/HDB00/prihana differs from host name defined on command line.
hdbenv.sh: Error: Instance not found for host -9
killing HDB processes:
kill -9 6011 /usr/sap/HDB/HDB00/prihana/trace/hdb.sapHDB_HDB00 -d -nw -f
/usr/sap/HDB/HDB00/prihana/daemon.ini pf=/usr/sap/HDB/SYS/profile/HDB_HDB00_prihana
kill -9 6027 hdbnameserver
kill -9 6137 hdbcompileserver
kill -9 6139 hdbpreprocessor
kill -9 6484 hdbindexserver -port 30003
kill -9 6494 hdbxsengine -port 30007
kill -9 7068 hdbwebdispatcher
kill orphan HDB processes:
kill -9 6027 [hdbnameserver] <defunct>
kill -9 6484 [hdbindexserver] <defunct>
```

Expected result:

- The cluster detects the stopped primary SAP HANA database (on node 1) and promotes the secondary SAP HANA database (on node 2) to take over as primary.

```
prihana:~ # crm status
```

```
Stack: corosync
Current DC: prihana (version 1.1.18+20180430.b12c320f5-3.24.1-b12c320f5) -
partition with quorum
Last updated: Thu Nov 12 11:53:21 2020
Last change: Thu Nov 12 11:53:19 2020 by root via crm_attribute on sechana

2 nodes configured
6 resources configured

Online: [ prihana sechana ]

Full list of resources:

 res_AWS_STONITH        (stonith:external/ec2): Started prihana
 res_AWS_IP     (ocf::suse:aws-vpc-move-ip):    Started sechana
 Clone Set: cln_SAPHanaTopology_HDB_HDB00 [rsc_SAPHanaTopology_HDB_HDB00]
     Started: [ prihana sechana ]
 Master/Slave Set: msl_SAPHana_HDB_HDB00 [rsc_SAPHana_HDB_HDB00]
     Masters: [ sechana ]
     Slaves: [ prihana ]

Failed Actions:
* rsc_SAPHana_HDB_HDB00_monitor_60000 on prihana 'master (failed)' (9): call=50,
status=complete, exitreason='',
    last-rc-change='Thu Nov 12 11:51:45 2020', queued=0ms, exec=0ms
```

- The overlay IP address is migrated to the new primary (on node 2).
- With the AUTOMATIC_REGISTER parameter set to "`true`", the cluster restarts the failed SAP HANA database and automatically registers it against the new primary.

Recovery procedure:

- Clean up the cluster "`failed actions`" on node 1 as root.

```
prihana:~ # crm resource cleanup rsc_SAPHana_HDB_HDB00 prihana
Cleaned up rsc_SAPHana_HDB_HDB00:0 on prihana
Cleaned up rsc_SAPHana_HDB_HDB00:1 on prihana
Waiting for 1 replies from the CRMd. OK
```

- After resource cleanup, the cluster "`failed actions`" are cleaned up.

Crash the primary database on node 2

Description — Simulate a complete breakdown of the primary database system.

Run node — Primary SAP HANA database node (on node 2).

Run steps:

- Stop the primary database (on node 2) system using the following command as `<sid>adm`.

```
sechana:~ # su - hdbadm
hdbadm@sechana:/usr/sap/HDB/HDB00> HDB kill -9
hdbenv.sh: Hostname sechana defined in $SAP_RETRIEVAL_PATH=/usr/sap/
HDB/HDB00/sechana differs from host name defined on command line.
hdbenv.sh: Error: Instance not found for host -9
killing HDB processes:
kill -9 30751 /usr/sap/HDB/HDB00/sechana/trace/hdb.sapHDB_HDB00 -d
-nw -f /usr/sap/HDB/HDB00/sechana/daemon.ini pf=/usr/sap/HDB/SYS/profile/
HDB_HDB00_sechana
kill -9 30899 hdbnameserver
kill -9 31166 hdbcompileserver
kill -9 31168 hdbpreprocessor
kill -9 31209 hdbindexserver -port 30003
kill -9 31211 hdbxsengine -port 30007
kill -9 31721 hdbwebdispatcher
kill orphan HDB processes:
kill -9 30899 [hdbnameserver] <defunct>
kill -9 31209 [hdbindexserver] <defunct>
```

Expected result:

- The cluster detects stopped primary SAP HANA database (on node 2) and promotes the secondary SAP HANA database (on node 1) to take over as primary.

```
sechana:~ # crm status
Stack: corosync
Current DC: prihana (version 1.1.18+20180430.b12c320f5-3.24.1-b12c320f5)
- partition with quorum
Last updated: Thu Nov 12 12:04:01 2020
```

```
Last change: Thu Nov 12 12:03:53 2020 by root via crm_attribute on prihana

2 nodes configured
6 resources configured

Online: [ prihana sechana ]

Full list of resources:

 res_AWS_STONITH        (stonith:external/ec2): Started prihana
 res_AWS_IP     (ocf::suse:aws-vpc-move-ip):    Started prihana
 Clone Set: cln_SAPHanaTopology_HDB_HDB00 [rsc_SAPHanaTopology_HDB_HDB00]
     Started: [ prihana sechana ]
 Master/Slave Set: msl_SAPHana_HDB_HDB00 [rsc_SAPHana_HDB_HDB00]
     Masters: [ prihana ]
     Slaves: [ sechana ]

Failed Actions:
* rsc_SAPHana_HDB_HDB00_monitor_60000 on sechana 'master (failed)' (9):
call=66, status=complete, exitreason='',
    last-rc-change='Thu Nov 12 11:58:53 2020', queued=0ms, exec=0ms
```

- The overlay IP address is migrated to the new primary (on node 1).
- With the AUTOMATIC_REGISTER parameter set to "`true`", the cluster restarts the failed SAP HANA database and automatically registers it against the new primary.

Recovery procedure:

- Clean up the cluster "`failed actions`" on node 2 as root.

```
sechana:~ # crm resource cleanup rsc_SAPHana_HDB_HDB00 sechana
Cleaned up rsc_SAPHana_HDB_HDB00:0 on sechana
Cleaned up rsc_SAPHana_HDB_HDB00:1 on sechana
Waiting for 1 replies from the CRMd. OK
```

- After resource cleanup, the cluster "`failed actions`" are cleaned up.

Reboot SAP HANA on node 1

Description: Simulate a crash of the primary site node running the primary SAP HANA database.

Run node: Primary SAP HANA database node

Run steps:

- Crash the primary database system using the following command as root:

```
prihana:~ # crm status
Stack: corosync
Current DC: prihana (version 1.1.18+20180430.b12c320f5-3.24.1-b12c320f5) - partition
 with quorum
Last updated: Thu Nov 12 12:09:44 2020
Last change: Thu Nov 12 12:09:11 2020 by root via crm_attribute on prihana

2 nodes configured
6 resources configured

Online: [ prihana sechana ]

Full list of resources:

 res_AWS_STONITH        (stonith:external/ec2): Started prihana
 res_AWS_IP     (ocf::suse:aws-vpc-move-ip):    Started prihana
 Clone Set: cln_SAPHanaTopology_HDB_HDB00 [rsc_SAPHanaTopology_HDB_HDB00]
     Started: [ prihana sechana ]
 Master/Slave Set: msl_SAPHana_HDB_HDB00 [rsc_SAPHana_HDB_HDB00]
     Masters: [ prihana ]
     Slaves: [ sechana ]

prihana:~ # echo 'b' > /proc/sysrq-trigger
```

> **Note**
>
> To simulate a system crash, you must first ensure that `/proc/sys/kernel/sysrq` is set to 1.

Expected result:

- The cluster detects failed node (node 1), declares it "UNCLEAN" and sets the secondary node (node 2) to status "`partition WITHOUT quorum`".
- The cluster fences node 1 and promotes the secondary SAP HANA database (on node 2) to take over as primary.

```
sechana:~ # crm status
Stack: corosync
Current DC: prihana (version 1.1.18+20180430.b12c320f5-3.24.1-b12c320f5) - partition
 with quorum
Last updated: Thu Nov 12 12:15:51 2020
Last change: Thu Nov 12 12:15:31 2020 by root via crm_attribute on sechana

2 nodes configured
6 resources configured

Online: [ sechana ]
OFFLINE: [ prihana ]

Full list of resources:

 res_AWS_STONITH        (stonith:external/ec2): Started sechana
 res_AWS_IP     (ocf::suse:aws-vpc-move-ip):    Started sechana
 Clone Set: cln_SAPHanaTopology_HDB_HDB00 [rsc_SAPHanaTopology_HDB_HDB00]
     Started: [ sechana ]
     Stopped: [ prihana ]
 Master/Slave Set: msl_SAPHana_HDB_HDB00 [rsc_SAPHana_HDB_HDB00]
     Masters: [ sechana ]
     Stopped: [ prihana ]
```

- The overlay IP address is migrated to the new primary (on node 2).
- With the AUTOMATIC_REGISTER parameter set to "true", the cluster restarts the failed SAP HANA database and automatically registers it against the new primary.

Recovery procedure:

- Start node 1 (EC2 instance) with the AWS Management Console or AWS CLI tools and start Pacemaker (if it's not enabled by default).

Reboot SAP HANA on node 2

Description — Simulate a crash of the primary site node (on node 2) running the primary SAP HANA database.

Run node — Primary SAP HANA database node (on node 2)

Run steps:

- Crash the primary database system (on node 2) using the following command as root:

```
sechana:~ # crm status
Stack: corosync
Current DC: sechana (version 1.1.18+20180430.b12c320f5-3.24.1-b12c320f5) - partition
 with quorum
Last updated: Thu Nov 12 12:16:57 2020
Last change: Thu Nov 12 12:16:41 2020 by root via crm_attribute on sechana

2 nodes configured
6 resources configured

Online: [ prihana sechana ]

Full list of resources:

 res_AWS_STONITH        (stonith:external/ec2): Started prihana
 res_AWS_IP     (ocf::suse:aws-vpc-move-ip):    Started sechana
 Clone Set: cln_SAPHanaTopology_HDB_HDB00 [rsc_SAPHanaTopology_HDB_HDB00]
     Started: [ prihana sechana ]
 Master/Slave Set: msl_SAPHana_HDB_HDB00 [rsc_SAPHana_HDB_HDB00]
     Masters: [ sechana ]
     Slaves: [ prihana ]

sechana:~ # echo 'b' > /proc/sysrq-trigger
```

> **Note**
>
> To simulate a system crash, you must first ensure that `/proc/sys/kernel/sysrq` is set to 1.

Expected result:

- The cluster detects failed node (node 2), declares it "UNCLEAN", and sets the secondary node (node 1) to status "`partition WITHOUT quorum`".
- The cluster fences node 2 and promotes the secondary SAP HANA database (on node 1) to take over as primary.

```
prihana:~ # crm status
Stack: corosync
Current DC: prihana (version 1.1.18+20180430.b12c320f5-3.24.1-b12c320f5) - partition
 with quorum
Last updated: Thu Nov 12 12:28:51 2020
Last change: Thu Nov 12 12:28:31 2020 by root via crm_attribute on prihana

2 nodes configured
6 resources configured

Online: [ prihana ]
OFFLINE: [ sechana ]

Full list of resources:

 res_AWS_STONITH        (stonith:external/ec2): Started prihana
 res_AWS_IP     (ocf::suse:aws-vpc-move-ip):    Started prihana
 Clone Set: cln_SAPHanaTopology_HDB_HDB00 [rsc_SAPHanaTopology_HDB_HDB00]
     Started: [ prihana ]
     Stopped: [ sechana ]
 Master/Slave Set: msl_SAPHana_HDB_HDB00 [rsc_SAPHana_HDB_HDB00]
     Masters: [ prihana ]
     Stopped: [ sechana ]
```

- The overlay IP address is migrated to the new primary (on node 1).
- With the AUTOMATIC_REGISTER parameter set to "`true`", the cluster restarts the failed SAP HANA database and automatically registers it against the new primary.

Recovery procedure:

- Start node 2 (EC2 instance) with AWS Management Console or AWS CLI tools and start Pacemaker (if it's not enabled by default).

Simulating a cluster network failure

Description — Simulate a network failure to test the cluster behavior in case of a split brain.

Run node — Can be run on any node. In this test case, this is done on node B.

Run steps:

- Drop all the traffic coming from and going to the subnet of the secondary node using the following command. This ensures that traffic is stopped on both the primary and secondary ring.

```
iptables -A INPUT -s <<Subnet_CIDR>> -j DROP; iptables
-A OUTPUT -d <<Subnet_CIDR>> -j DROP

sechana:~ # crm status
Stack: corosync
Current DC: prihana (version 1.1.18+20180430.b12c320f5-3.24.1-b12c320f5)
- partition with quorum
Last updated: Fri Jan 22 02:16:28 2021
Last change: Fri Jan 22 02:16:27 2021 by root via crm_attribute on sechana
2 nodes configured
6 resources configured
Online: [ prihana sechana ]
Full list of resources:
 res_AWS_STONITH        (stonith:external/ec2): Started prihana
 res_AWS_IP     (ocf::suse:aws-vpc-move-ip):    Started sechana
 Clone Set: cln_SAPHanaTopology_HDB_HDB00 [rsc_SAPHanaTopology_HDB_HDB00]
     Started: [ prihana sechana ]
 Master/Slave Set: msl_SAPHana_HDB_HDB00 [rsc_SAPHana_HDB_HDB00]
     Masters: [ prihana ]
     Slaves: [ sechana ]
sechana:~ # iptables -A INPUT -s 11.0.1.132 -j DROP; iptables -A OUTPUT -d 11.0.1.132
 -j DROP
```

Expected result:

- The cluster detects network failure and fence node 1. It promotes the secondary SAP HANA database (on node 2) to take over as primary without going to a split brain situation.

```
sechana:~ # crm status
Stack: corosync
Current DC: prihana (version 1.1.18+20180430.b12c320f5-3.24.1-b12c320f5)
- partition with quorum
Last updated: Fri Jan 22 17:08:09 2021
Last change: Fri Jan 22 17:07:46 2021 by root via crm_attribute on sechana

2 nodes configured
6 resources configured
```

```
Online: [ prihana sechana ]

Full list of resources:

 res_AWS_STONITH        (stonith:external/ec2): Started prihana
 res_AWS_IP     (ocf::suse:aws-vpc-move-ip):    Started sechana
 Clone Set: cln_SAPHanaTopology_HDB_HDB00 [rsc_SAPHanaTopology_HDB_HDB00]
     rsc_SAPHanaTopology_HDB_HDB00      (ocf::suse:SAPHanaTopology):
Started prihana (Monitoring)
     Started: [ sechana ]
 Master/Slave Set: msl_SAPHana_HDB_HDB00 [rsc_SAPHana_HDB_HDB00]
     Masters: [ sechana ]
     Stopped: [ prihana ]

Failed Actions:
* rsc_SAPHanaTopology_HDB_HDB00_monitor_10000 on prihana 'unknown error'
(1): call=317, status=Timed Out, exitreason='',
    last-rc-change='Fri Jan 22 16:58:19 2021', queued=0ms, exec=300001ms
* rsc_SAPHana_HDB_HDB00_start_0 on prihana 'unknown error' (1): call=28, status=Timed
 Out,
exitreason='',
    last-rc-change='Fri Jan 22 02:40:38 2021', queued=0ms, exec=3600001ms
```

Recovery procedure:

- Clean up the cluster "`failed actions`".

Administration and troubleshooting

Monitor the status of the cluster

You can check the status of the cluster with the following commands:

```
prihana:~ # crm status
Stack: corosync
Current DC: prihana (version 1.1.18+20180430.b12c320f5-3.24.1-b12c320f5) - partition
 with quorum
Last updated: Thu Nov 12 12:35:56 2020
Last change: Thu Nov 12 12:34:57 2020 by root via crm_attribute on prihana

2 nodes configured
6 resources configured
```

```
Online: [ prihana sechana ]

Full list of resources:

 res_AWS_STONITH       (stonith:external/ec2): Started prihana
 res_AWS_IP     (ocf::suse:aws-vpc-move-ip):    Started prihana
 Clone Set: cln_SAPHanaTopology_HDB_HDB00 [rsc_SAPHanaTopology_HDB_HDB00]
     Started: [ prihana sechana ]
 Master/Slave Set: msl_SAPHana_HDB_HDB00 [rsc_SAPHana_HDB_HDB00]
     Masters: [ prihana ]
     Slaves: [ sechana ]
```

```
prihana:~ # crm_mon -1
Stack: corosync
Current DC: prihana (version 1.1.18+20180430.b12c320f5-3.24.1-b12c320f5) - partition
 with quorum
Last updated: Thu Nov 12 12:36:24 2020
Last change: Thu Nov 12 12:36:01 2020 by root via crm_attribute on prihana

2 nodes configured
6 resources configured

Online: [ prihana sechana ]

Active resources:

 res_AWS_STONITH       (stonith:external/ec2): Started prihana
 res_AWS_IP     (ocf::suse:aws-vpc-move-ip):    Started prihana
 Clone Set: cln_SAPHanaTopology_HDB_HDB00 [rsc_SAPHanaTopology_HDB_HDB00]
     Started: [ prihana sechana ]
 Master/Slave Set: msl_SAPHana_HDB_HDB00 [rsc_SAPHana_HDB_HDB00]
     Masters: [ prihana ]
     Slaves: [ sechana ]
```

Check the status of replication with the following command:

```
prihana:~ # crm_mon -A1
Stack: corosync
Current DC: prihana (version 1.1.18+20180430.b12c320f5-3.24.1-b12c320f5) - partition
 with quorum
Last updated: Thu Nov 12 12:37:28 2020
Last change: Thu Nov 12 12:37:04 2020 by root via crm_attribute on prihana
```

```
2 nodes configured
6 resources configured

Online: [ prihana sechana ]

Active resources:

 res_AWS_STONITH        (stonith:external/ec2): Started prihana
 res_AWS_IP     (ocf::suse:aws-vpc-move-ip):    Started prihana
 Clone Set: cln_SAPHanaTopology_HDB_HDB00 [rsc_SAPHanaTopology_HDB_HDB00]
     Started: [ prihana sechana ]
 Master/Slave Set: msl_SAPHana_HDB_HDB00 [rsc_SAPHana_HDB_HDB00]
     Masters: [ prihana ]
     Slaves: [ sechana ]

Node Attributes:
* Node prihana:
    + hana_hdb_clone_state              : PROMOTED
    + hana_hdb_op_mode                  : logreplay
    + hana_hdb_remoteHost               : sechana
    + hana_hdb_roles                    : 4:P:master1:master:worker:master
    + hana_hdb_site                     : PRI
    + hana_hdb_srmode                   : sync
    + hana_hdb_sync_state               : PRIM
    + hana_hdb_version                  : 2.00.030.00.1522209842
    + hana_hdb_vhost                    : prihana
    + lpa_hdb_lpt                       : 1605184624
    + master-rsc_SAPHana_HDB_HDB00      : 150
* Node sechana:
    + hana_hdb_clone_state              : DEMOTED
    + hana_hdb_op_mode                  : logreplay
    + hana_hdb_remoteHost               : prihana
    + hana_hdb_roles                    : 4:S:master1:master:worker:master
    + hana_hdb_site                     : SEC
    + hana_hdb_srmode                   : sync
    + hana_hdb_sync_state               : SOK
    + hana_hdb_version                  : 2.00.030.00.1522209842
    + hana_hdb_vhost                    : sechana
    + lpa_hdb_lpt                       : 30
    + master-rsc_SAPHana_HDB_HDB00      : 100
```

Cluster administration

To manually migrate the cluster resources from one node to another, run the following command:

```
prihana:~ # crm resource move rsc_SAPHana_HDB_HDB00 force
INFO: Move constraint created for rsc_SAPHana_HDB_HDB00 to sechana
```

Check the status of the migration using the command "`crm_mon -r`".

```
prihana:~ # crm_mon -r
Stack: corosync
Current DC: prihana (version 1.1.18+20180430.b12c320f5-3.24.1-b12c320f5) - partition
 with quorum
Last updated: Thu Nov 12 12:39:00 2020
Last change: Thu Nov 12 12:38:47 2020 by root via crm_attribute on prihana

2 nodes configured
6 resources configured

Online: [ prihana sechana ]

Full list of resources:

res_AWS_STONITH (stonith:external/ec2): Started prihana
res_AWS_IP      (ocf::suse:aws-vpc-move-ip):    Started sechana
 Clone Set: cln_SAPHanaTopology_HDB_HDB00 [rsc_SAPHanaTopology_HDB_HDB00]
     Started: [ prihana sechana ]
 Master/Slave Set: msl_SAPHana_HDB_HDB00 [rsc_SAPHana_HDB_HDB00]
     rsc_SAPHana_HDB_HDB00      (ocf::suse:SAPHana):    Promoting sechana
     Slaves: [ prihana ]
```

After the resource is migrated, you can check the status of the cluster. Clean up the failed actions as shown in next section.

```
prihana:~ # crm status
Stack: corosync
Current DC: prihana (version 1.1.18+20180430.b12c320f5-3.24.1-b12c320f5) - partition
 with quorum
Last updated: Thu Nov 12 12:41:07 2020
Last change: Thu Nov 12 12:40:44 2020 by root via crm_attribute on prihana

2 nodes configured
```

```
6 resources configured

Online: [ prihana sechana ]

Full list of resources:

 res_AWS_STONITH        (stonith:external/ec2): Started prihana
 res_AWS_IP     (ocf::suse:aws-vpc-move-ip):    Started sechana
 Clone Set: cln_SAPHanaTopology_HDB_HDB00 [rsc_SAPHanaTopology_HDB_HDB00]
     Started: [ prihana sechana ]
 Master/Slave Set: msl_SAPHana_HDB_HDB00 [rsc_SAPHana_HDB_HDB00]
     Masters: [ sechana ]
     Slaves: [ prihana ]

Failed Actions:
* rsc_SAPHana_HDB_HDB00_monitor_61000 on prihana 'not running' (7): call=35,
status=complete, exitreason='',
    last-rc-change='Thu Nov 12 12:39:49 2020', queued=0ms, exec=0ms
```

Resource cleanup activities

- You can run the command "`crm resource cleanup rsc_SAPHana_<SID>_HDB<Instance Number> <hostname>`" to clean up any failed actions, as shown in the following example:

```
prihana:~ # crm status
Stack: corosync
Current DC: prihana (version 1.1.18+20180430.b12c320f5-3.24.1-b12c320f5) - partition
 with quorum
Last updated: Thu Nov 12 12:41:07 2020
Last change: Thu Nov 12 12:40:44 2020 by root via crm_attribute on prihana

2 nodes configured
6 resources configured

Online: [ prihana sechana ]

Full list of resources:

 res_AWS_STONITH        (stonith:external/ec2): Started prihana
 res_AWS_IP     (ocf::suse:aws-vpc-move-ip):    Started sechana
 Clone Set: cln_SAPHanaTopology_HDB_HDB00 [rsc_SAPHanaTopology_HDB_HDB00]
     Started: [ prihana sechana ]
 Master/Slave Set: msl_SAPHana_HDB_HDB00 [rsc_SAPHana_HDB_HDB00]
```

```
     Masters: [ sechana ]
     Slaves: [ prihana ]

Failed Actions:
* rsc_SAPHana_HDB_HDB00_monitor_61000 on prihana 'not running' (7): call=35,
  status=complete, exitreason='',
    last-rc-change='Thu Nov 12 12:39:49 2020', queued=0ms, exec=0ms

prihana:~ # crm resource cleanup rsc_SAPHana_HDB_HDB00 prihana
Cleaned up rsc_SAPHana_HDB_HDB00:0 on prihana
Cleaned up rsc_SAPHana_HDB_HDB00:1 on prihana
Waiting for 1 replies from the CRMd. OK
prihana:~ #
```

- When you manually migrate resources from one node to another, there will be constraints in the `crm` configuration. You can find the constraints with the command "`crm configure show`" as shown in the following example:

```
prihana:~ # crm configure show
node 1: prihana \
        attributes lpa_hdb_lpt=30 hana_hdb_vhost=prihana hana_hdb_site=PRI
hana_hdb_srmode=sync hana_hdb_remoteHost=sechana hana_hdb_op_mode=logreplay
node 2: sechana \
        attributes lpa_hdb_lpt=1605184953 hana_hdb_vhost=sechana hana_hdb_site=SEC
hana_hdb_srmode=sync hana_hdb_remoteHost=prihana hana_hdb_op_mode=logreplay
primitive res_AWS_IP ocf:suse:aws-vpc-move-ip \
        params ip=192.168.10.16 routing_table=rtb-06ca3aca4c58bd17d interface=eth0
profile=cluster \
        op start interval=0 timeout=180 \
        op stop interval=0 timeout=180 \
        op monitor interval=60 timeout=60 \
        meta target-role=Started
primitive res_AWS_STONITH stonith:external/ec2 \
        op start interval=0 timeout=180 \
        op stop interval=0 timeout=180 \
        op monitor interval=120 timeout=60 \
        meta target-role=Started \
        params tag=pacemaker profile=cluster
primitive rsc_SAPHanaTopology_HDB_HDB00 ocf:suse:SAPHanaTopology \
        operations $id=rsc_sap2_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
primitive rsc_SAPHana_HDB_HDB00 ocf:suse:SAPHana \
        operations $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=true HANA_CALL_TIMEOUT=60
ms msl_SAPHana_HDB_HDB00 rsc_SAPHana_HDB_HDB00 \
        meta clone-max=2 clone-node-max=1 interleave=true
clone cln_SAPHanaTopology_HDB_HDB00 rsc_SAPHanaTopology_HDB_HDB00 \
        meta clone-node-max=1 interleave=true
location cli-prefer-rsc_SAPHana_HDB_HDB00 rsc_SAPHana_HDB_HDB00 role=Started inf:
 sechana
colocation col_IP_Primary 2000: res_AWS_IP:Started msl_SAPHana_HDB_HDB00:Master
order ord_SAPHana 2000: cln_SAPHanaTopology_HDB_HDB00 msl_SAPHana_HDB_HDB00
property SAPHanaSR: \
        hana_hdb_site_srHook_SEC=PRIM \
        hana_hdb_site_srHook_PRI=SOK
property cib-bootstrap-options: \
        stonith-enabled=true \
        stonith-action=off \
        stonith-timeout=600s \
        have-watchdog=false \
        dc-version="1.1.18+20180430.b12c320f5-3.24.1-b12c320f5" \
        cluster-infrastructure=corosync \
        last-lrm-refresh=1605184909
rsc_defaults rsc-options: \
        resource-stickiness=1000 \
        migration-threshold=5000
op_defaults op-options: \
        timeout=600
```

You must clean up these location constraints before you perform any further cluster actions with following command:

```
prihana:~ # crm resource clear rsc_SAPHana_HDB_HDB00
INFO: Removed migration constraints for rsc_SAPHana_HDB_HDB00
```

Checking the logs

Start your troubleshooting by checking logs at `/var/log/messages`. For additional details, you can check cluster and Pacemaker logs.

- **Cluster logs** — Cluster logs are updated in the `corosync.log` file located under `/var/log/cluster` folder.
- **Pacemaker logs** — Pacemaker logs are updated in the `pacemaker.log` file located in the `/var/log/pacemaker` folder.

Sample working configuration

The example of a working configuration:

```
prihana:~ # crm configure show
node 1: prihana \
        attributes lpa_hdb_lpt=30 hana_hdb_vhost=prihana hana_hdb_site=PRI
hana_hdb_srmode=sync hana_hdb_remoteHost=sechana hana_hdb_op_mode=logreplay
node 2: sechana \
        attributes lpa_hdb_lpt=1605185144 hana_hdb_vhost=sechana hana_hdb_site=SEC
hana_hdb_srmode=sync hana_hdb_remoteHost=prihana hana_hdb_op_mode=logreplay
primitive res_AWS_IP ocf:suse:aws-vpc-move-ip \
        params ip=192.168.10.16 routing_table=rtb-06ca3aca4c58bd17d interface=eth0
        profile=cluster \
        op start interval=0 timeout=180 \
        op stop interval=0 timeout=180 \
        op monitor interval=60 timeout=60 \
        meta target-role=Started
primitive res_AWS_STONITH stonith:external/ec2 \
        op start interval=0 timeout=180 \
        op stop interval=0 timeout=180 \
        op monitor interval=120 timeout=60 \
        meta target-role=Started \
        params tag=pacemaker profile=cluster
primitive rsc_SAPHanaTopology_HDB_HDB00 ocf:suse:SAPHanaTopology \
        operations $id=rsc_sap2_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
primitive rsc_SAPHana_HDB_HDB00 ocf:suse:SAPHana \
        operations $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=true
ms msl_SAPHana_HDB_HDB00 rsc_SAPHana_HDB_HDB00 \
        meta clone-max=2 clone-node-max=1 interleave=true
clone cln_SAPHanaTopology_HDB_HDB00 rsc_SAPHanaTopology_HDB_HDB00 \
        meta clone-node-max=1 interleave=true
colocation col_IP_Primary 2000: res_AWS_IP:Started msl_SAPHana_HDB_HDB00:Master
order ord_SAPHana 2000: cln_SAPHanaTopology_HDB_HDB00 msl_SAPHana_HDB_HDB00
property SAPHanaSR: \
        hana_hdb_site_srHook_SEC=PRIM \
        hana_hdb_site_srHook_PRI=SOK
property cib-bootstrap-options: \
        stonith-enabled=true \
        stonith-action=off \
        stonith-timeout=600s \
        have-watchdog=false \
        dc-version="1.1.18+20180430.b12c320f5-3.24.1-b12c320f5" \
        cluster-infrastructure=corosync \
        last-lrm-refresh=1605184909
rsc_defaults rsc-options: \
        resource-stickiness=1000 \
        migration-threshold=5000
op_defaults op-options: \
        timeout=600
```

Corosync configuration file:

```
prihana:~ # cat /etc/corosync/corosync.conf
# Please read the corosync.conf.5 manual page
totem {
        version: 2
        token: 30000
        consensus: 36000
        token_retransmits_before_loss_const: 6
        crypto_cipher: none
        crypto_hash: none
        clear_node_high_bit: yes
        rrp_mode: passive
```

```
        interface {
                ringnumber: 0
                bindnetaddr: 11.0.1.132
                mcastport: 5405
                ttl: 1
        }
        transport: udpu
}
logging {
        fileline: off
        to_logfile: yes
        to_syslog: yes
        logfile: /var/log/cluster/corosync.log
        debug: off
        timestamp: on
        logger_subsys {
                subsys: QUORUM
                debug: off
        }
}
nodelist {
        node {
                ring0_addr: 11.0.1.132
                ring1_addr: 11.0.1.75
                nodeid: 1
        }
        node {
                ring0_addr: 11.0.2.139
                ring1_addr: 11.0.2.35
                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_node: 1
}
```

HA cluster configuration on RHEL

The following instructions are applicable to Red Hat Enterprise Linux for SAP with version 7.x and 8.x. You will see different instructions (where applicable) in the following sections.

Operating system configuration

If you are using Red Hat 8.6 or later, the following services must be stopped and disabled on both the cluster nodes. This prevents the NetworkManager from removing the overlay IP address from the network interface.

```
systemctl disable nm-cloud-setup.timer
systemctl stop nm-cloud-setup.timer
systemctl disable nm-cloud-setup
systemctl stop nm-cloud-setup
```

Cluster installation

Prerequisite – The system must be subscribed to the required subscription; in this case, RHEL for SAP Solutions.

> **Note**
>
> If you are using a BYOS image, ensure your system is configured with RHEL for SAP and Pacemaker repositories to install the required packages.

```
yum install -y pcs pacemaker fence-agents-aws
yum install -y resource-agents
yum install -y resource-agents-sap-hana
```

Cluster configuration

Topics

- Update user hacluster password
- Start and enable the pcs services
- Authenticate pcs with user hacluster

- Review instance settings that conflict with cluster actions
- Set up the cluster
- Enable and start the cluster
- Increase corosync totem token timeout

Update user `hacluster` password

Change the password of the user `haclustser` on both the nodes, as shown in the following example:

```
[root@prihana ~]# passwd hacluster
[root@sechana ~]# passwd hacluster
```

Start and enable the `pcs` services

The following commands start and enable the `pcs` service on both the nodes:

```
[root@prihana ~]# systemctl start pcsd.service
[root@prihana ~]# systemctl enable pcsd.service
```

Authenticate pcs with user hacluster

The following command authenticates `pcs` to the `pcs` daemon on the nodes in the cluster. The user name for the `pcs` administration must be `hacluster` on both the nodes with the same password.

RHEL 7.x

```
[root@prihana ~]# pcs cluster auth prihana sechana
Username: hacluster
Password:
sechana: Authorized
prihana: Authorized
[root@prihana ~]#
```

RHEL 8.x

```
[root@<host1> ~]# pcs host auth prihana sechana
Username: hacluster
```

```
Password:
sechana: Authorized
prihana: Authorized
[root@<host1> ~]#
```

Review instance settings that conflict with cluster actions

To ensure that restarts are predictable, we recommend disabling simplified automatic recovery and *not* configuring Amazon CloudWatch action based recovery for instances that are part of a pacemaker cluster. Use the following command to disable simplified automatic recovery.

```
aws ec2 modify-instance-maintenance-options --instance-id i-0abcdef1234567890 --auto-
recovery disabled
```

You must ensure that stop protection is disabled for Amazon EC2 instances that are part of a pacemaker cluster. Use the following command to disable stop protection.

```
aws ec2 modify-instance-attribute --instance-id i-1234567890abcdef0 --no-disable-api-
stop
```

Set up the cluster

The following command configures the cluster configuration file and syncs the configuration on both the nodes.

```
pcs cluster setup –name rhelhanaha prihana sechana

[rooteprihana~]pcs cluster setup --name rhelhanaha prihana sechana
Destroying cluster on nodes: prihana, sechana...
sechana: Stopping Cluster (pacemaker)...
prihana: Stopping Cluster (pacemaker)...
sechana: Successfully destroyed cluster
prihana: Successfully destroyed cluster
Sending 'pacemaker_remote authkey' to iprihana', 'sechana' prihana:
successful distribution of the file 'pacemaker remote authkey'
sechana: successful distribution of the file 'pacemaker_remote authkey'
Sending cluster config files to the nodes...
prihana: Succeeded
sechana: Succeeded
Synchronizing pcsd certificates on nodes prihana, sechana... saphdbdbe2: Success
prihana: Success
```

```
Restarting pcsd on the nodes in order to reload the certificates... sechana: Success
prihana: Success
```

Enable and start the cluster

The following commands enable and start the cluster:

```
pcs cluster enable –all

root@prihana etc]# pcs cluster enable --all
prihana: Cluster Enabled
sechana: Cluster Enabled
```

```
pcs cluster start –all

[root@prihana etc]# pcs cluster start --all
prihana: Starting Cluster (corosync)...
sechana: Starting Cluster (corosync)...
sechana: Starting Cluster (pacemaker)...
prihana: Starting Cluster (pacemaker)...
[rooteprihana etc]# I
```

Increase corosync totem token timeout

RHEL 7.x

1. Edit the `/etc/corosync/corosync.conf` file in all the cluster nodes and increase or add the value for token, as shown in the following example.

   ```
   totem {
       version: 2
       secauth: off
       cluster_name: my-rhel-sap-cluster
       transport: udpu
       rrp_mode: passive
       token: 30000  <------ Value to be set
   }
   ```

2. Reload `corosync` by running the following command in only one cluster node to reload. This will not require any downtime.

```
# pcs cluster reload corosync
```

3. Run the following command to confirm that your changes are active.

```
# corosync-cmapctl | grep totem.token
Runtime.config.totem.token (u32) = 30000
```

RHEL 8.x

Run the following command to increase the `corosync` token timeout.

```
# pcs cluster config update totem token=29000
```

Cluster resources

This section describes how to create the cluster resources.

`STONITH`

The following command creates the `STONITH` resource. This is to protect your data from being corrupted by rogue nodes or concurrent access in an event of split brain or dual primary situations.

```
[root@prihana ~]# pcs stonith create <resource-name> fence_aws \
region=<aws-region> \
pcmk_host_map="<primary-hostname>:<primary-instance-id>;<secondary-
hostname>:<secondary-instance-id>" \
pcmk_delay_max=45 \
power_timeout=600 pcmk_reboot_timeout=600 \
pcmk_reboot_retries=4 \
op start timeout=600 \
op monitor interval=300 timeout=60
```

The default `pcmk` action is reboot. If you want to have the instance remain in a stopped state until it has been investigated and then manually started again, add `pcmk_reboot_action=off`. Any **High Memory** (u-*tb1.*) instances or metal instance running on a dedicated host won't support

reboot and will require `pcmk_reboot_action=off`. To do this, update the previously created `STONITH` resource as:

```
[root@prihana ~]# pcs stonith create <resource-name> fence_aws \
region=<aws-region> \
pcmk_host_map="<primary-hostname>:<primary-instanceid>;<secondary-hostname>:<secondary-
instance-i>" \
pcmk_delay_max=45 pcmk_reboot_action=off \
power_timeout=600 pcmk_reboot_timeout=600 \
pcmk_reboot_retries=4 \
op start timeout=600 \
op monitor interval=300 timeout=60
```

SAPHanaTopology

The `SAPHanaTopology` resource gathers the status and configuration of SAP HANA System Replication on each node. Configure the following attributes for `SAPHanaTopology`.

Run the following command to create the `SAPHANATopology` resource:

```
[root@prihana~]# pcs resource create SAPHanaTopology_HDB_00 SAPHanaTopology \
SID=HDB InstanceNumber=00 \
op start timeout=600 \
op stop timeout=300 \
op monitor interval=10 timeout=600 \
clone clone-max=2 clone-node-max=1 interleave=true
```

SAPHana

The `SAPHana` resource is responsible for starting, stopping, and relocating the SAP HANA database. This resource must be run as a primary/secondary cluster resource. To create this resource, run the following command:

RHEL 7.x

```
[root@prihana~]# pcs resource create SAPHana_HDB_00 SAPHana \
SID=HDB InstanceNumber=00 PREFER_SITE_TAKEOVER=true \
```

```
DUPLICATE_PRIMARY_TIMEOUT=7200 AUTOMATED_REGISTER=true \
op start timeout=3600 \
op stop timeout=3600 \
op monitor interval=61 role="Slave" timeout=700 \
op monitor interval=59 role="Master" timeout=700 \
op promote timeout=3600 \
op demote timeout=3600 \
master notify=true clone-max=2 clone-node-max=1 interleave=true
```

RHEL 8.x

```
[root@prihana~]# pcs resource create SAPHana_HDB_00 \
SAPHana SID=HDB InstanceNumber=00 PREFER_SITE_TAKEOVER=true \
DUPLICATE_PRIMARY_TIMEOUT=7200 AUTOMATED_REGISTER=true \
op start timeout=3600 \
op stop timeout=3600 \
op monitor interval=61 role="Slave" timeout=700 \
op monitor interval=59 role="Master" timeout=700 \
op promote timeout=3600 \
op demote timeout=3600 \
promotable meta notify=true clone-max=2 clone-node-max=1 interleave=true
```

Note

If the `AUTOMATED_REGISTER` parameter is set to true, the secondary instance will automatically register after startup, and start the replication.

Overlay IP

Add the Overlay IP (OIP) address to the primary node using the following command:

```
[root@prihana ~]# ip address add <overlay IP address>/32  dev eth0
```

To route the traffic to your primary SAP HANA database with Overlay IP, you must update the route table and map the Overlay IP address to the primary SAP HANA database `instance-id`.

```
[root@prihana ~]# aws ec2 create-route --route-table-id rtb-xxxxxxxx \
--destination-cidr-block <overlay IP address> --instance-id i-xxxxxxxx
```

```
pcs resource create hana-oip \
aws-vpc-move-ip ip=<overlay IP address> interface=eth0 routing_table=rtb-dbexxxxx
```

If you are using different route tables for subnet in each Availability Zone where you are deploying the SAP HANA instances, you need to update the OIP in the route table associated with both the subnets. To create the resource in such scenario, you can use the previous command and mention both the route table IDs separated by a comma. See the following example:

```
[root@prihana ~]# pcs resource create hana-oip aws-vpc-move-ip \
ip=<overlay IP address>  \
interface=eth0 \
routing_table=rtb-xxxxxxx,rtb-yyyyyyyy
```

Constraints

Define two constraints, one for the Overlay IP address which helps with routing client traffic to active database host and the second one for the start order between the SAPHANA and SAPHanaTopology resource agents.

Cluster defaults

The following command creates the default cluster migration-threshold and stickiness for the cluster resources.

RHEL 7.x

```
[root@prihana ~]# pcs resource defaults resource-stickiness=1000
[root@prihana ~]# pcs resource defaults migration-threshold=5000
```

RHEL 8.x

For RHEL 8.0 through 8.3:

```
[root@prihana ~]# pcs resource defaults resource-stickiness=1000
```

```
[root@prihana ~]# pcs resource defaults migration-threshold=5000
```

Starting with RHEL 8.4 (pcs-0.10.8-1.el8):

```
[root@prihana ~]# pcs resource defaults update resource-stickiness=1000
[root@prihana ~]# pcs resource defaults update migration-threshold=5000
```

Constraint: start SAPHanaTopology before SAPHana

The following command creates the constraint that mandates the start order of these resources.

RHEL 7.x

```
[root@prihana ~]# pcs constraint order SAPHanaTopology_HDB_00-clone \
then SAPHana_HDB_00-master symmetrical=false
```

RHEL 8.x

```
[root@prihana ~]# pcs constraint order SAPHanaTopology_HDB_00-clone \
then SAPHana_HDB_00-clone symmetrical=false
```

- **`symmetrical=false`** — This attribute defines that it is just the start order of resources and they don't need to be stopped in reverse order.
- **`interleave = true`** — This attribute allows parallel start of these resources on nodes. This allows the SAPHana resource to start on any node as soon as the SAPHanaTopology resource is running on any one node.

Both resources (SAPHana and SAPHanaTopology) have the attribute `interleave=true` that allows parallel start of these resources on nodes.

Constraint co-locate the `aws-vpc-move-ip` resource with the primary SAPHana resource

The following command co-locates the `aws-vpc-move-ip` resource with the SAPHana resource when promoted as primary.

RHEL 7.x

```
[root@prihana ~]# pcs constraint colocation add hana-oip with master SAPHana_HDB_00-
master 2000
```

RHEL 8.x

```
[root@prihana ~]# pcs constraint colocation add hana-oip with master SAPHana_HDB_00-
clone 2000
```

You can use the following command to check the final status of the cluster:

```
[root@prihana ~]# pcs status
Cluster name: rhelhanaha
Stack: corosync
Current DC: sechana (version 1.1.19-8.el7_6.5-c3c624ea3d) - partition with quorum
Last updated: Tue Nov 10 17:54:13 2020
Last change: Tue Nov 10 17:53:48 2020 by root via crm_attribute on prihana

2 nodes configured
6 resources configured

Online: [ prihana sechana ]

Full list of resources:

 clusterfence   (stonith:fence_aws):    Started prihana
 Clone Set: SAPHanaTopology_HDB_00-clone [SAPHanaTopology_HDB_00]
     Started: [ prihana sechana ]
 Master/Slave Set: SAPHana_HDB_00-master [SAPHana_HDB_00]
     Masters: [ prihana ]
     Slaves: [ sechana ]
 hana-oip       (ocf::heartbeat:aws-vpc-move-ip):       Started prihana

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

```
[root@prihana ~]#
```

This concludes the configuration of the SAP HANA cluster setup. You can proceed with testing.

Testing the cluster

After the cluster setup is complete, perform the tests shown below to validate cluster setup. Run these tests in sequence.

- the section called “Stop the SAP HANA database on the primary node”
- the section called “Stop the SAP HANA database on the secondary node”
- the section called “Crash the primary database on node 1”
- the section called “Crash the primary database on node 2”
- the section called “Reboot SAP HANA on node 1”
- the section called “Reboot SAP HANA on node 2”
- the section called “Simulating a cluster network failure”

Stop the SAP HANA database on the primary node

Description — Stop the primary SAP HANA database during normal cluster operation.

Run node — Primary SAP HANA database node

Run steps:

- Stop the primary SAP HANA database gracefully as `<sid>adm`

```
[root@prihana ~]# su - hdbadm
hdbadm@prihana:/usr/sap/HDB/HDB00> HDB stop
hdbdaemon will wait maximal 300 seconds for NewDB services finishing.
Stopping instance using: /usr/sap/HDB/SYS/exe/hdb/sapcontrol -prot
NI_HTTP -nr 00 -function Stop 400
```

```
12.11.2020 11:39:19
Stop
OK
Waiting for stopped instance using:
/usr/sap/HDB/SYS/exe/hdb/sapcontrol -prot NI_HTTP -nr 00 -function
WaitforStopped 600 2

12.11.2020 11:39:51
WaitforStopped
OK
hdbdaemon is stopped.
```

Expected result:

- The cluster detects stopped primary SAP HANA database (on node 1) and promotes the secondary SAP HANA database (on node 2) to take over as primary.

```
[root@prihana ~]# pcs status
Cluster name: rhelhanaha
Stack: corosync
Current DC: sechana (version 1.1.19-8.el7_6.5-c3c624ea3d) - partition with quorum
Last updated: Tue Nov 10 17:58:19 2020
Last change: Tue Nov 10 17:57:41 2020 by root via crm_attribute on sechana

2 nodes configured
6 resources configured

Online: [ prihana sechana ]

Full list of resources:

 clusterfence   (stonith:fence_aws):    Started prihana
 Clone Set: SAPHanaTopology_HDB_00-clone [SAPHanaTopology_HDB_00]
     Started: [ prihana sechana ]
 Master/Slave Set: SAPHana_HDB_00-master [SAPHana_HDB_00]
     Masters: [ sechana ]
     Slaves: [ prihana ]
 hana-oip       (ocf::heartbeat:aws-vpc-move-ip):       Started sechana

Failed Actions:
* SAPHana_HDB_00_monitor_59000 on prihana 'master (failed)' (9): call=31,
```

```
status=complete, exitreason='',
    last-rc-change='Tue Nov 10 17:56:52 2020', queued=0ms, exec=0ms


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

- The overlay IP address is migrated to the new primary (on node 2).

```
[root@sechana ~]# ip addr show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9000 qdisc mq state
UP group default qlen 1000
    link/ether 0e:ef:dd:3c:bf:1b brd ff:ff:ff:ff:ff:ff
    inet xx.xx.xx.xx/24 brd 11.0.2.255 scope global eth0
       valid_lft forever preferred_lft forever
    inet xx.xx.xx.xx/32 scope global eth0:1
       valid_lft forever preferred_lft forever
    inet 192.168.10.16/32 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::cef:ddff:fe3c:bf1b/64 scope link
       valid_lft forever preferred_lft forever
```

- Because AUTOMATED_REGISTER is set to true, the cluster restarts the failed SAP HANA database and registers it against the new primary. Validate the status of the primary SAP HANA database using the following command:

```
hdbadm@prihana:/usr/sap/HDB/HDB00> sapcontrol -nr 00 -function GetProcessList

10.11.2020 17:59:49
GetProcessList
OK
```

```
name, description, dispstatus, textstatus, starttime, elapsedtime, pid
hdbdaemon, HDB Daemon, GREEN, Running, 2020 11 10 17:58:47, 0:01:02, 25979
hdbcompileserver, HDB Compileserver, GREEN, Running, 2020 11 10 17:58:52, 0:00:57,
 26152
hdbindexserver, HDB Indexserver-HDB, GREEN, Running, 2020 11 10 17:58:53, 0:00:56,
 26201
hdbnameserver, HDB Nameserver, GREEN, Running, 2020 11 10 17:58:48, 0:01:01, 25997
hdbpreprocessor, HDB Preprocessor, GREEN, Running, 2020 11 10 17:58:52, 0:00:57,
 26155
hdbwebdispatcher, HDB Web Dispatcher, GREEN, Running, 2020 11 10 17:59:02, 0:00:47,
 27100
hdbxsengine, HDB XSEngine-HDB, GREEN, Running, 2020 11 10 17:58:53, 0:00:56, 26204
hdbadm@prihana:/usr/sap/HDB/HDB00>
```

Recovery procedure:

- Clean up the cluster "`failed actions`" on node 1 as root using the following command:

```
[root@prihana ~]# pcs resource cleanup SAPHana_HDB_00 --node prihana
```

- After you run the cleanup command, "`failed actions`" messages should disappear from the cluster status.

```
 [root@prihana ~]# pcs status
Cluster name: rhelhanaha
Stack: corosync
Current DC: sechana (version 1.1.19-8.el7_6.5-c3c624ea3d) -
partition with quorum
Last updated: Tue Nov 10 18:01:02 2020
Last change: Tue Nov 10 18:00:45 2020 by root via crm_attribute on sechana

2 nodes configured
6 resources configured

Online: [ prihana sechana ]

Full list of resources:

 clusterfence   (stonith:fence_aws):    Started prihana
 Clone Set: SAPHanaTopology_HDB_00-clone [SAPHanaTopology_HDB_00]
```

```
     Started: [ prihana sechana ]
 Master/Slave Set: SAPHana_HDB_00-master [SAPHana_HDB_00]
     Masters: [ sechana ]
     Slaves: [ prihana ]
 hana-oip       (ocf::heartbeat:aws-vpc-move-ip):       Started sechana

Daemon Status:
  corosync: active/enabled
  pacemaker: active/enabled
  pcsd: active/enabled
[root@prihana ~]#
```

Stop the SAP HANA database on the secondary node

Description — Stop the primary SAP HANA database (on Node 2) during normal cluster operation.

Run node — Primary SAP HANA database node (on Node 2)

Run steps:

- Stop the SAP HANA database gracefully as <sid>adm on node 2.

```
[root@sechana ~]# su - hdbadm
hdbadm@sechana:/usr/sap/HDB/HDB00> HDB stop
hdbdaemon will wait maximal 300 seconds for NewDB services finishing.
Stopping instance using: /usr/sap/HDB/SYS/exe/hdb/sapcontrol -prot NI_HTTP -nr
00 -function Stop 400

12.11.2020 11:45:21
Stop
OK
Waiting for stopped instance using: /usr/sap/HDB/SYS/exe/hdb/sapcontrol
-prot NI_HTTP -nr 00 -function WaitforStopped 600 2


12.11.2020 11:45:53
WaitforStopped
OK
hdbdaemon is stopped.
```

Expected result:

- The cluster detects the stopped primary SAP HANA database (on node 2) and promotes the secondary SAP HANA database (on node 1) to take over as primary.

```
[root@sechana ~]# pcs status
Cluster name: rhelhanaha
Stack: corosync
Current DC: sechana (version 1.1.19-8.el7_6.5-c3c624ea3d) - partition with quorum
Last updated: Tue Nov 10 18:04:01 2020
Last change: Tue Nov 10 18:04:00 2020 by root via crm_attribute on prihana

2 nodes configured
6 resources configured

Online: [ prihana sechana ]

Full list of resources:

 clusterfence   (stonith:fence_aws):    Started prihana
 Clone Set: SAPHanaTopology_HDB_00-clone [SAPHanaTopology_HDB_00]
     Started: [ prihana sechana ]
 Master/Slave Set: SAPHana_HDB_00-master [SAPHana_HDB_00]
     SAPHana_HDB_00     (ocf::heartbeat:SAPHana):       Promoting prihana
     Slaves: [ sechana ]
 hana-oip       (ocf::heartbeat:aws-vpc-move-ip):       Started prihana

Failed Actions:
* SAPHana_HDB_00_monitor_59000 on sechana 'master (failed)' (9): call=41,
status=complete, exitreason='',
    last-rc-change='Tue Nov 10 18:03:49 2020', queued=0ms, exec=0ms


Daemon Status:
  corosync: active/enabled
  pacemaker: active/enabled
  pcsd: active/enabled
[root@sechana ~]#
```

- The overlay IP address is migrated to the new primary (on node 1).

```
[root@prihana ~]# ip addr show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
 qlen 1000
```

```
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9000 qdisc mq state UP group default
 qlen 1000
    link/ether 0a:38:1c:ce:b4:3d brd ff:ff:ff:ff:ff:ff
    inet xx.xx.xx.xx/24 brd 11.0.1.255 scope global eth0
       valid_lft forever preferred_lft forever
    inet xx.xx.xx.xx/32 scope global eth0:1
       valid_lft forever preferred_lft forever
    inet 192.168.10.16/32 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::838:1cff:fece:b43d/64 scope link
       valid_lft forever preferred_lft forever
```

- With AUTOMATED_REGISTER set to `true`, the cluster restarts the failed SAP HANA database and registers it against the new primary.

  Check the status of the secondary using the following command:

```
hdbadm@sechana:/usr/sap/HDB/HDB00> sapcontrol -nr 00 -function GetProcessList

10.11.2020 18:08:47
GetProcessList
OK
name, description, dispstatus, textstatus, starttime, elapsedtime, pid
hdbdaemon, HDB Daemon, GREEN, Running, 2020 11 10 18:05:44, 0:03:03, 6601
hdbcompileserver, HDB Compileserver, GREEN, Running, 2020 11 10 18:05:48, 0:02:59,
 6725
hdbindexserver, HDB Indexserver-HDB, GREEN, Running, 2020 11 10 18:05:49, 0:02:58,
 6828
hdbnameserver, HDB Nameserver, GREEN, Running, 2020 11 10 18:05:44, 0:03:03, 6619
hdbpreprocessor, HDB Preprocessor, GREEN, Running, 2020 11 10 18:05:48, 0:02:59, 6730
hdbwebdispatcher, HDB Web Dispatcher, GREEN, Running, 2020 11 10 18:05:58, 0:02:49,
 7797
hdbxsengine, HDB XSEngine-HDB, GREEN, Running, 2020 11 10 18:05:49, 0:02:58, 6831
hdbadm@sechana:/usr/sap/HDB/HDB00>
```

Recovery procedure:

- Clean up the cluster "`failed actions`" on node 2 as root using the following command:

```
[root@sechana ~]# pcs resource cleanup SAPHana_HDB_00 --node sechana
```

- After resource cleanup, ensure the cluster "`failed actions`" are cleaned up.

```
root@sechana ~]# pcs status
Cluster name: rhelhanaha
Stack: corosync
Current DC: sechana (version 1.1.19-8.el7_6.5-c3c624ea3d) - partition with quorum
Last updated: Tue Nov 10 18:13:35 2020
Last change: Tue Nov 10 18:12:51 2020 by hacluster via crmd on sechana

2 nodes configured
6 resources configured

Online: [ prihana sechana ]

Full list of resources:

 clusterfence   (stonith:fence_aws):    Started prihana
 Clone Set: SAPHanaTopology_HDB_00-clone [SAPHanaTopology_HDB_00]
     Started: [ prihana sechana ]
 Master/Slave Set: SAPHana_HDB_00-master [SAPHana_HDB_00]
     Masters: [ prihana ]
     Slaves: [ sechana ]
 hana-oip       (ocf::heartbeat:aws-vpc-move-ip):       Started prihana

Daemon Status:
  corosync: active/enabled
  pacemaker: active/enabled
  pcsd: active/enabled
[root@sechana ~]#
```

Crash the primary database on node 1

Description — Simulate a complete breakdown of the primary database system.

Run node: Primary SAP HANA database node

Run steps:

- Crash the primary database system using the following command as <sid>adm:

```
[root@prihana ~]# sudo su - hdbadm
hdbadm@prihana:/usr/sap/HDB/HDB00> HDB kill -9
hdbenv.sh: Hostname prihana defined in $SAP_RETRIEVAL_PATH=/usr/sap/HDB/HDB00/
prihana differs from host name defined on command line.
hdbenv.sh: Error: Instance not found for host -9
killing HDB processes:
kill -9 6011 /usr/sap/HDB/HDB00/prihana/trace/hdb.sapHDB_HDB00 -d -nw -f
/usr/sap/HDB/HDB00/prihana/daemon.ini pf=/usr/sap/HDB/SYS/profile/HDB_HDB00_prihana
kill -9 6027 hdbnameserver
kill -9 6137 hdbcompileserver
kill -9 6139 hdbpreprocessor
kill -9 6484 hdbindexserver -port 30003
kill -9 6494 hdbxsengine -port 30007
kill -9 7068 hdbwebdispatcher
kill orphan HDB processes:
kill -9 6027 [hdbnameserver] <defunct>
kill -9 6484 [hdbindexserver] <defunct>
```

Expected result:

- The cluster detects the stopped primary SAP HANA database (on node 1) and promotes the secondary SAP HANA database (on node 2) to take over as primary.

```
[root@prihana ~]# pcs status
Cluster name: rhelhanaha
Stack: corosync
Current DC: sechana (version 1.1.19-8.el7_6.5-c3c624ea3d) - partition with quorum
Last updated: Tue Nov 10 17:58:19 2020
Last change: Tue Nov 10 17:57:41 2020 by root via crm_attribute on sechana
2 nodes configured
6 resources configured
Online: [ prihana sechana ]
Full list of resources:
 clusterfence   (stonith:fence_aws):    Started prihana
 Clone Set: SAPHanaTopology_HDB_00-clone [SAPHanaTopology_HDB_00]
     Started: [ prihana sechana ]
 Master/Slave Set: SAPHana_HDB_00-master [SAPHana_HDB_00]
     Masters: [ sechana ]
     Slaves: [ prihana ]
 hana-oip       (ocf::heartbeat:aws-vpc-move-ip):       Started sechana
```

```
Failed Actions:
* SAPHana_HDB_00_monitor_59000 on prihana 'master (failed)' (9): call=31,
status=complete, exitreason='',
    last-rc-change='Tue Nov 10 17:56:52 2020', queued=0ms, exec=0ms
Daemon Status:
  corosync: active/enabled
  pacemaker: active/enabled
  pcsd: active/enabled
[root@prihana ~]#
```

- The overlay IP address is migrated to the new primary (on node 2).
- Because AUTOMATED_REGISTER is set to `true`, the cluster restarts the failed SAP HANA database and registers it against the new primary.

Recovery procedure:

- Clean up the cluster "`failed actions`" on node 1 as root.

```
root@prihana ~]# pcs resource cleanup SAPHana_HDB_00 --node prihana
```

- After resource cleanup, ensure the cluster "`failed actions`" are cleaned up.

Crash the primary database on node 2

Description — Simulate a complete breakdown of the primary database system.

Run node — The primary SAP HANA database node (on node 2).

Run steps:

- Crash the primary database (on node 2) system using the following command as `<sid>adm`.

```
[root@sechana ~]# su - hdbadm
hdbadm@sechana:/usr/sap/HDB/HDB00> HDB kill -9
hdbenv.sh: Hostname sechana defined in $SAP_RETRIEVAL_PATH=/usr/sap/
HDB/HDB00/sechana differs from host name defined on command line.
hdbenv.sh: Error: Instance not found for host -9
killing HDB processes:
kill -9 30751 /usr/sap/HDB/HDB00/sechana/trace/hdb.sapHDB_HDB00 -d -nw -f
```

```
/usr/sap/HDB/HDB00/sechana/daemon.ini pf=/usr/sap/HDB/SYS/profile/HDB_HDB00_sechana
kill -9 30899 hdbnameserver
kill -9 31166 hdbcompileserver
kill -9 31168 hdbpreprocessor
kill -9 31209 hdbindexserver -port 30003
kill -9 31211 hdbxsengine -port 30007
kill -9 31721 hdbwebdispatcher
kill orphan HDB processes:
kill -9 30899 [hdbnameserver] <defunct>
kill -9 31209 [hdbindexserver] <defunct>
```

Expected result:

- The cluster detects the stopped primary SAP HANA database (on node 2) and promotes the secondary SAP HANA database (on node 1) to take over as primary.

```
[root@sechana ~]# pcs status
Cluster name: rhelhanaha
Stack: corosync
Current DC: prihana (version 1.1.19-8.el7_6.5-c3c624ea3d) - partition with quorum
Last updated: Tue Nov 10 18:13:35 2020
Last change: Tue Nov 10 18:12:51 2020 by hacluster via crmd on sechana

2 nodes configured
6 resources configured

Online: [ prihana sechana ]

Full list of resources:

 clusterfence   (stonith:fence_aws):    Started prihana
 Clone Set: SAPHanaTopology_HDB_00-clone [SAPHanaTopology_HDB_00]
     Started: [ prihana sechana ]
 Master/Slave Set: SAPHana_HDB_00-master [SAPHana_HDB_00]
     Masters: [ prihana ]
     Slaves: [ sechana ]
 hana-oip       (ocf::heartbeat:aws-vpc-move-ip):       Started prihana

Failed Actions:
* SAPHana_HDB_00_monitor_59000 on sechana 'master (failed)' (9): call=41,
status=complete, exitreason='',
    last-rc-change='Tue Nov 10 18:03:49 2020', queued=0ms, exec=0ms
```

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

- The overlay IP address is migrated to the new primary (on node 1).
- Because AUTOMATED_REGISTER is set to true, the cluster restarts the failed SAP HANA database and registers it against the new primary.

Recovery procedure:

- Clean up the cluster "`failed actions`" on node 2 as root.

```
[root@prihana ~]# pcs resource cleanup SAPHana_HDB_00
--node sechana
```

- After resource cleanup, ensure the cluster "`failed actions`" are cleaned up.

Reboot SAP HANA on node 1

Description — Simulate a crash of the primary node running the primary SAP HANA database.

Run node: Primary SAP HANA database node

Run steps:

- Crash the primary database system using the following command as root:

```
[root@prihana ~]# pcs status
Cluster name: rhelhanaha
Stack: corosync
Current DC: sechana (version 1.1.19-8.el7_6.5-c3c624ea3d) - partition with quorum
Last updated: Tue Nov 10 17:54:13 2020
Last change: Tue Nov 10 17:53:48 2020 by root via crm_attribute on prihana
2 nodes configured
6 resources configured
Online: [ prihana sechana ]
```

```
Full list of resources:
 clusterfence   (stonith:fence_aws):    Started prihana
 Clone Set: SAPHanaTopology_HDB_00-clone [SAPHanaTopology_HDB_00]
     Started: [ prihana sechana ]
 Master/Slave Set: SAPHana_HDB_00-master [SAPHana_HDB_00]
     Masters: [ prihana ]
     Slaves: [ sechana ]
 hana-oip       (ocf::heartbeat:aws-vpc-move-ip):       Started prihana
Daemon Status:
  corosync: active/enabled
  pacemaker: active/enabled
  pcsd: active/enabled
[root@prihana ~]# echo 'b' > /proc/sysrq-trigger
```

Note

To simulate a system crash, you must first ensure that `/proc/sys/kernel/sysrq` is set to 1.

Expected result:

- The cluster detects the failed node (node 1), declares it "UNCLEAN", and sets the secondary node (node 2) to status "`partition WITHOUT quorum`".
- The cluster fences node 1, promotes the secondary SAP HANA database, and registers it against the new primary when the EC2 instance is back up. Node 1 is currently in a stopped state because it is being rebooted.

```
[root@sechana ~]# pcs status
Cluster name: rhelhanaha
Stack: corosync
Current DC: sechana (version 1.1.19-8.el7_6.5-c3c624ea3d) - partition with quorum
Last updated: Tue Nov 10 18:17:24 2020
Last change: Tue Nov 10 18:17:06 2020 by root via crm_attribute on sechana

2 nodes configured
6 resources configured

Online: [ prihana sechana ]
```

```
Full list of resources:

 clusterfence   (stonith:fence_aws):    Started sechana
 Clone Set: SAPHanaTopology_HDB_00-clone [SAPHanaTopology_HDB_00]
     Started: [ prihana sechana ]
 Master/Slave Set: SAPHana_HDB_00-master [SAPHana_HDB_00]
     Masters: [ sechana ]
     OFFLINE: [ prihana ]
 hana-oip       (ocf::heartbeat:aws-vpc-move-ip):       Started sechana

Daemon Status:
  corosync: active/enabled
  pacemaker: active/enabled
  pcsd: active/enabled
[root@sechana ~]#
```

- The overlay IP address is migrated to the new primary (on node 2).
- Because AUTOMATIC_REGISTER = `true`, the cluster restarts the failed HANA database and registers it against the new primary when the EC2 instance is back up.

Recovery procedure:

- Start node 1 (EC2 Instance) using AWS Management Console or AWS CLI tools.

Reboot SAP HANA on node 2

Description — Simulate a crash of the primary node (on node 2) running the primary SAP HANA database.

Run node — Primary SAP HANA database node (on node 2)

Run steps:

- Crash the node running primary SAP HANA (on node 2) using the following command as root:

```
shutdown
```

```
[root@sechana ~]# pcs status
Cluster name: rhelhanaha
Stack: corosync
```

```
Current DC: sechana (version 1.1.19-8.el7_6.5-c3c624ea3d) - partition with quorum
Last updated: Tue Nov 10 17:54:13 2020
Last change: Tue Nov 10 17:53:48 2020 by root via crm_attribute on prihana
2 nodes configured
6 resources configured
Online: [ prihana sechana ]
Full list of resources:
 clusterfence   (stonith:fence_aws):    Started prihana
 Clone Set: SAPHanaTopology_HDB_00-clone [SAPHanaTopology_HDB_00]
     Started: [ prihana sechana ]
 Master/Slave Set: SAPHana_HDB_00-master [SAPHana_HDB_00]
     Masters: [ sechana ]
     Slaves: [ prihana ]
 hana-oip       (ocf::heartbeat:aws-vpc-move-ip):       Started sechana
Daemon Status:
  corosync: active/enabled
  pacemaker: active/enabled
  pcsd: active/enabled
[root@sechana ~]# echo 'b' > /proc/sysrq-trigger
```

Note

To simulate a system crash, you must first ensure that `/proc/sys/kernel/sysrq` is set to 1.

Expected result:

- The cluster detects the failed node (node 2), declares it "UNCLEAN", and sets the secondary node (node 1) to status "`partition WITHOUT quorum`".
- The cluster fences node 2 and promotes the secondary SAP HANA database (on node 1) to take over as primary.

```
[root@prihana ~]# pcs status
Cluster name: rhelhanaha
Stack: corosync
Current DC: prihana (version 1.1.19-8.el7_6.5-c3c624ea3d) - partition with quorum
Last updated: Tue Nov 10 18:22:00 2020
Last change: Tue Nov 10 18:21:49 2020 by root via crm_attribute on prihana
```

```
2 nodes configured
6 resources configured

Online: [ prihana ]
OFFLINE: [ sechana ]

Full list of resources:

 clusterfence   (stonith:fence_aws):    Started prihana
 Clone Set: SAPHanaTopology_HDB_00-clone [SAPHanaTopology_HDB_00]
     Started: [ prihana ]
     Stopped: [ sechana ]
 Master/Slave Set: SAPHana_HDB_00-master [SAPHana_HDB_00]
     Masters: [ prihana ]
     Stopped: [ sechana ]
 hana-oip       (ocf::heartbeat:aws-vpc-move-ip):       Started prihana

Daemon Status:
  corosync: active/enabled
  pacemaker: active/enabled
  pcsd: active/enabled
[root@prihana ~]#
```

- The overlay IP address is migrated to the new primary (on node 2).
- Because AUTOMATED_REGISTER is set to `true`, the cluster restarts the failed SAP HANA database and registers it against the new primary when the EC2 instance is back up.

Recovery procedure:

- Start node 2 (EC2 instance) using AWS Management Console or AWS CLI tools.

Simulating a cluster network failure

Description —To simulate a network failure to test the cluster behavior in case of a split brain.

Run node: Can be run on any node. In this test case, this is done on node B.

Run steps:

- Drop all the traffic coming from and going to node A with the following command:

```
iptables -A INPUT -s <<Primary IP address of Node A>> -j DROP;
iptables -A OUTPUT -d <<Primary IP address of Node A>> -j DROP
```

```
[root@sechana ~]# pcs status
Cluster name: rhelhanaha
Stack: corosync
Current DC: prihana(version 1.1.19-8.el7_6.5-c3c624ea3d) - partition with quorum
Last updated: Fri Jan 22 14:45:24 2021
Last change: Fri Jan 22 14:45:11 2021 by hacluster via crmd on  sechana
2 nodes configured
6 resources configured
Online: [ prihana sechana ]
Full list of resources:
 clusterfence   (stonith:fence_aws):    Started prihana
 Clone Set: SAPHanaTopology_DRL_00-clone [SAPHanaTopology_DRL_00]
     Started: [ prihana sechana ]
 Master/Slave Set: SAPHana_DRL_00-master [SAPHana_DRL_00]
     Masters: [ prihana]
     Slaves: [ sechana ]
 hana-oip       (ocf::heartbeat:aws-vpc-move-ip):       Started prihana
Daemon Status:
  corosync: active/enabled
  pacemaker: active/enabled
  pcsd: active/enabled
[root@ sechana ~]#sechana:~ # iptables -A INPUT -s xxx.xxx.xxx.xxx -j DROP;
iptables -A OUTPUT -d xxx.xxx.xxx.xxx -j DROP
```

Expected result:

- The cluster detects network failure and fences node 1. The cluster promotes the secondary SAP HANA database (on node 2) to take over as primary without going to a split brain situation.

```
[root@sechana ~]# pcs status
Cluster name: rhelhanaha
Stack: corosync
Current DC: sechana (version 1.1.19-8.el7_6.5-c3c624ea3d) - partition with quorum
Last updated: Fri Jan 22 15:11:43 2021
Last change: Fri Jan 22 15:10:48 2021 by root via crm_attribute on sechana
2 nodes configured
6 resources configured
```

```
Online: [ sechana ]
OFFLINE: [ prihana]
Full list of resources:
 clusterfence   (stonith:fence_aws):    Started sechana
 Clone Set: SAPHanaTopology_DRL_00-clone [SAPHanaTopology_DRL_00]
     Started: [ sechana ]
     Stopped: [ prihana]
 Master/Slave Set: SAPHana_DRL_00-master [SAPHana_DRL_00]
     Masters: [ sechana ]
     Stopped: [ prihana]
 hana-oip       (ocf::heartbeat:aws-vpc-move-ip):       Started sechana
Failed Actions:
* clusterfence_monitor_60000 on sechana 'unknown error' (1): call=-1,
status=Timed Out, exitreason='',
    last-rc-change='Fri Jan 22 14:59:14 2021', queued=0ms, exec=0ms
Daemon Status:
  corosync: active/enabled
  pacemaker: active/enabled
  pcsd: active/enabled
[root@sechana ~]#
```

Recovery procedure:

- Clean up the cluster "`failed actions`".

Administration and troubleshooting

Monitor the status of cluster

You can check the status of the cluster with the following command as root user:

```
root@prihana ~]# pcs status
Cluster name: rhelhanaha
Stack: corosync
Current DC: sechana (version 1.1.19-8.el7_6.5-c3c624ea3d) - partition with quorum
Last updated: Thu Nov 12 09:44:08 2020
Last change: Thu Nov 12 09:43:20 2020 by root via crm_attribute on sechana

2 nodes configured
6 resources configured

Online: [ prihana sechana ]
```

```
Full list of resources:

 clusterfence   (stonith:fence_aws):    Started prihana
 Clone Set: SAPHanaTopology_HDB_00-clone [SAPHanaTopology_HDB_00]
     Started: [ prihana sechana ]
 Master/Slave Set: SAPHana_HDB_00-master [SAPHana_HDB_00]
     Masters: [ prihana ]
     Slaves: [ sechana ]
 hana-oip       (ocf::heartbeat:aws-vpc-move-ip):       Started prihana

Daemon Status:
  corosync: active/enabled
  pacemaker: active/enabled
  pcsd: active/enabled
[root@prihana ~]#
```

You can check the SAP HANA replication status with the following command as a <sid>adm user:

```
hdbadm@prihana:/usr/sap/HDB/HDB00> python
/usr/sap/HDB/HDB00/exe/python_support/systemReplicationStatus.py
| Database | Host       | Port  | Service Name | Volume ID | Site
ID | Site Name  | Secondary  | Secondary | Secondary | Secondary
| Secondary     | Replication | Replication | Replication    |
|          |            |       |              |           |
|            | Host       | Port      | Site ID   | Site Name    |
Active Status | Mode        | Status      | Status Details |
| -------- | ---------- | ----- | ------------ | --------- | -------
| ---------- | ---------- | --------- | --------- | ------------
| ------------- | ----------- | ----------- | -------------- |
| SYSTEMDB | prihana | 30001 | nameserver   |         1 |       1 |
HDBPrimary | sechana |     30001 |         2 | HDBSecondary | YES
| SYNCMEM     | ACTIVE      |           |
| HDB      | prihana | 30007 | xsengine     |         2 |       1 |
HDBPrimary | sechana |     30007 |         2 | HDBSecondary | YES
| SYNCMEM     | ACTIVE      |           |
| HDB      | prihana | 30003 | indexserver  |         3 |       1 |
HDBPrimary | sechana |     30003 |         2 | HDBSecondary | YES
| SYNCMEM     | ACTIVE      |           |

status system replication site "2": ACTIVE
overall system replication status: ACTIVE
```

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

mode: PRIMARY
site id: 1
site name: HDBPrimary
hdbadm@prihana:/usr/sap/HDB/HDB00>
```

Cluster administration

You can manually migrate cluster resources from one node to another with the following command as root user:

```
root@prihana ~]# pcs resource move SAPHana_HDB_00-master
Warning: Creating location constraint cli-ban-SAPHana_HDB_00-
master-on-prihana with a score of -INFINITY for resource
SAPHana_HDB_00-master on node prihana.
This will prevent SAPHana_HDB_00-master from running on prihana
until the constraint is removed. This will be the case even if
prihana is the last node in the cluster.
```

You can check the status of the cluster again to verify the status of resource migration.

```
[root@prihana ~]#pcs status
Thu Nov 12 10:45:14 2020

Cluster name: rhelhanaha
Stack: corosync
Current DC: sechana (version 1.1.19-8.el7_6.5-c3c624ea3d) - partition with quorum
Last updated: Thu Nov 12 10:45:14 2020
Last change: Thu Nov 12 10:45:06 2020 by root via crm_attribute on sechana

2 nodes configured
6 resources configured

Online: [ prihana sechana ]

Full list of resources:

 clusterfence   (stonith:fence_aws):    Started prihana
 Clone Set: SAPHanaTopology_HDB_00-clone [SAPHanaTopology_HDB_00]
     Started: [ prihana sechana ]
 Master/Slave Set: SAPHana_HDB_00-master [SAPHana_HDB_00]
```

```
     Masters: [ sechana ]
     Stopped: [ prihana ]
 hana-oip       (ocf::heartbeat:aws-vpc-move-ip):       Started sechana

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

Clean up the failed actions as shown in next section. With each `pcs` resource move command invocation, the cluster creates location constraints to cause the resource to move. These constraints must be removed to allow automated failover in the future. To remove the constraints created by the move, run the following command:

```
root@prihana ~]# pcs resource clear SAPHana_HDB_00-master
```

Check the status of the cluster:

```
root@prihana ~]# pcs status
Cluster name: rhelhanaha
Stack: corosync
Current DC: sechana (version 1.1.19-8.el7_6.5-c3c624ea3d) - partition with quorum
Last updated: Thu Nov 12 10:49:44 2020
Last change: Thu Nov 12 10:49:12 2020 by root via crm_attribute on sechana

2 nodes configured
6 resources configured

Online: [ prihana sechana ]

Full list of resources:

 clusterfence   (stonith:fence_aws):    Started prihana
 Clone Set: SAPHanaTopology_HDB_00-clone [SAPHanaTopology_HDB_00]
     Started: [ prihana sechana ]
 Master/Slave Set: SAPHana_HDB_00-master [SAPHana_HDB_00]
     Masters: [ sechana ]
     Slaves: [ prihana ]
 hana-oip       (ocf::heartbeat:aws-vpc-move-ip):       Started sechana

Daemon Status:
  corosync: active/enabled
```

```
  pacemaker: active/enabled
  pcsd: active/enabled
```

Resource cleanup activities

Clean up the failed actions using following command:

```
[root@prihana ~]# pcs resource cleanup SAPHana_HDB_00 --node prihana
```

```
[root@prihana ~]#pcs status
Thu Nov 12 10:45:14 2020

Cluster name: rhelhanaha
Stack: corosync
Current DC: sechana (version 1.1.19-8.el7_6.5-c3c624ea3d) - partition with quorum
Last updated: Thu Nov 12 10:45:14 2020
Last change: Thu Nov 12 10:45:06 2020 by root via crm_attribute on sechana

2 nodes configured
6 resources configured

Online: [ prihana sechana ]

Full list of resources:

 clusterfence   (stonith:fence_aws):    Started prihana
 Clone Set: SAPHanaTopology_HDB_00-clone [SAPHanaTopology_HDB_00]
     Started: [ prihana sechana ]
 Master/Slave Set: SAPHana_HDB_00-master [SAPHana_HDB_00]
     Masters: [ sechana ]
     Stopped: [ prihana ]
 hana-oip       (ocf::heartbeat:aws-vpc-move-ip):       Started sechana

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

Manual migration of resources from one node to another node (as shown in the preceding section) will create constraints in pcs configuration “`pcs config show`”.

```
root@prihana ~]# pcs config show
```

```
Cluster Name: rhelhanaha
Corosync Nodes:
 prihana sechana
Pacemaker Nodes:
 prihana sechana

Resources:
 Clone: SAPHanaTopology_HDB_00-clone
  Meta Attrs: clone-max=2 clone-node-max=1 interleave=true
  Resource: SAPHanaTopology_HDB_00 (class=ocf provider=heartbeat type=SAPHanaTopology)
   Attributes: InstanceNumber=00 SID=HDB
   Operations: methods interval=0s timeout=5 (SAPHanaTopology_HDB_00-methods-
interval-0s)
               monitor interval=10 timeout=600 (SAPHanaTopology_HDB_00-monitor-
interval-10)
               reload interval=0s timeout=5 (SAPHanaTopology_HDB_00-reload-interval-0s)
               start interval=0s timeout=600 (SAPHanaTopology_HDB_00-start-interval-0s)
               stop interval=0s timeout=300 (SAPHanaTopology_HDB_00-stop-interval-0s)
 Master: SAPHana_HDB_00-master
  Meta Attrs: clone-max=2 clone-node-max=1 interleave=true notify=true
  Resource: SAPHana_HDB_00 (class=ocf provider=heartbeat type=SAPHana)
   Attributes: AUTOMATED_REGISTER=true DUPLICATE_PRIMARY_TIMEOUT=7200
               InstanceNumber=00 PREFER_SITE_TAKEOVER=true SID=HDB
   Operations: demote interval=0s timeout=3600 (SAPHana_HDB_00-demote-interval-0s)
               methods interval=0s timeout=5 (SAPHana_HDB_00-methods-interval-0s)
               monitor interval=61 role=Slave timeout=700 (SAPHana_HDB_00-monitor-
interval-61)
               monitor interval=59 role=Master timeout=700 (SAPHana_HDB_00-monitor-
interval-59)
               promote interval=0s timeout=3600 (SAPHana_HDB_00-promote-interval-0s)
               reload interval=0s timeout=5 (SAPHana_HDB_00-reload-interval-0s)
               start interval=0s timeout=3600 (SAPHana_HDB_00-start-interval-0s)
               stop interval=0s timeout=3600 (SAPHana_HDB_00-stop-interval-0s)
 Resource: hana-oip (class=ocf provider=heartbeat type=aws-vpc-move-ip)
  Attributes: interface=eth0 ip=192.168.1.99 routing_table=rtb-0027679b7a9eff404
  Operations: monitor interval=60s timeout=30s (hana-oip-monitor-interval-60s)
              start interval=0s timeout=180s (hana-oip-start-interval-0s)
              stop interval=0s timeout=180s (hana-oip-stop-interval-0s)

Stonith Devices:
 Resource: clusterfence (class=stonith type=fence_aws)
  Attributes: pcmk_host_map=prihana:i-01b7ceb0d8799eccf;sechana:i-05b924af2f83ffe0b
  pcmk_reboot_retries=4 pcmk_reboot_timeout=480 power_timeout=240 region=us-east-1
  Operations: monitor interval=60s (clusterfence-monitor-interval-60s)
```

```
Fencing Levels:

Location Constraints:
Ordering Constraints:
  start SAPHanaTopology_HDB_00-clone then start SAPHana_HDB_00-master
  (kind:Mandatory) (non-symmetrical)
Colocation Constraints:
  hana-oip with SAPHana_HDB_00-master (score:2000) (rsc-role:Started)
  (with-rsc-role:Master)
Ticket Constraints:

Alerts:
 No alerts defined

Resources Defaults:
 resource-stickiness: 1000
 migration-threshold: 5000
Operations Defaults:
 No defaults set

Cluster Properties:
 cluster-infrastructure: corosync
 cluster-name: rhelhanaha
 dc-version: 1.1.19-8.el7_6.5-c3c624ea3d
 have-watchdog: false
 last-lrm-refresh: 1605053571
Node Attributes:
 prihana: hana_hdb_op_mode=logreplay hana_hdb_remoteHost=sechana
  hana_hdb_site=HDBPrimary hana_hdb_srmode=syncmem hana_hdb_vhost=prihana
  lpa_hdb_lpt=1605196167
 sechana: hana_hdb_op_mode=logreplay hana_hdb_remoteHost=prihana
  hana_hdb_site=HDBSecondary hana_hdb_srmode=syncmem hana_hdb_vhost=sechana
  lpa_hdb_lpt=30

Quorum:
  Options:
```

These location constraints need to be cleaned up before you perform any further cluster actions with the following command:

```
[root@prihana ~]# pcs constraint list --full
Location Constraints:
Ordering Constraints:
```

```
  start SAPHanaTopology_HDB_00-clone then start SAPHana_HDB_00-master
(kind:Mandatory) (non-symmetrical) (id:order-SAPHanaTopology_HDB_00-
clone-SAPHana_HDB_00-master-mandatory)
Colocation Constraints:
  hana-oip with SAPHana_HDB_00-master (score:2000) (rsc-role:Started)
(with-rsc-role:Master) (id:colocation-hana-oip-SAPHana_HDB_00-master-2000)
Ticket Constraints:
[root@prihana ~]#


root@prihana ~]# pcs constraint remove colocation-hana-oip-SAPHana_HDB_00-master-2000

[root@prihana ~]# pcs constraint list --full
Location Constraints:
Ordering Constraints:
  start SAPHanaTopology_HDB_00-clone then start SAPHana_HDB_00-master
(kind:Mandatory) (non-symmetrical) (id:order-SAPHanaTopology_HDB_00-clone-
SAPHana_HDB_00-master-mandatory)
Colocation Constraints:
Ticket Constraints:
[root@prihana ~]#
```

Checking the logs

Start your troubleshooting by checking logs at `/var/log/messages`. You can find additional information in cluster and Pacemaker logs.

- **Cluster logs** — Cluster logs are updated in the `corosync.log` file located at `var/log/cluster/corosync.log`.
- **Pacemaker logs** — Pacemaker logs are updated in the pacemaker.log file located at `/var/log/pacemaker`.

Sample working configuration

```
[root@sechana ~]# pcs config
Cluster Name: rhelhanaha
Corosync Nodes:
 prihana sechana
Pacemaker Nodes:
 prihana sechana
```

```
Resources:
 Clone: SAPHanaTopology_HDB_00-clone
  Meta Attrs: clone-max=2 clone-node-max=1 interleave=true
  Resource: SAPHanaTopology_HDB_00 (class=ocf provider=heartbeat type=SAPHanaTopology)
   Attributes: InstanceNumber=00 SID=HDB
   Operations: methods interval=0s timeout=5 (SAPHanaTopology_HDB_00-methods-
interval-0s)
               monitor interval=60 timeout=60 (SAPHanaTopology_HDB_00-monitor-
interval-60)
               start interval=0s timeout=180 (SAPHanaTopology_HDB_00-start-interval-0s)
               stop interval=0s timeout=60 (SAPHanaTopology_HDB_00-stop-interval-0s)
 Master: SAPHana_HDB_00-master
  Resource: SAPHana_HDB_00 (class=ocf provider=heartbeat type=SAPHana)
   Attributes: AUTOMATED_REGISTER=true DUPLICATE_PRIMARY_TIMEOUT=7200
 InstanceNumber=00
               PREFER_SITE_TAKEOVER=true SID=HDB
   Meta Attrs: clone-max=2 clone-node-max=1 interleave=true notify=true
   Operations: demote interval=0s timeout=320 (SAPHana_HDB_00-demote-interval-0s)
               methods interval=0s timeout=5 (SAPHana_HDB_00-methods-interval-0s)
               monitor interval=120 timeout=60 (SAPHana_HDB_00-monitor-interval-120)
               monitor interval=121 role=Slave timeout=700 (SAPHana_HDB_00-monitor-
interval-121)
               monitor interval=119 role=Master timeout=700 (SAPHana_HDB_00-monitor-
interval-119)
               promote interval=0s timeout=320 (SAPHana_HDB_00-promote-interval-0s)
               start interval=0s timeout=180 (SAPHana_HDB_00-start-interval-0s)
               stop interval=0s timeout=240 (SAPHana_HDB_00-stop-interval-0s)
 Resource: hana-oip (class=ocf provider=heartbeat type=aws-vpc-move-ip)
  Attributes: interface=eth0 ip=192.168.0.1 routing_table=rtb-dbe0eba1
  Operations: monitor interval=60 timeout=30 (hana-oip-monitor-interval-60)
              start interval=0s timeout=180 (hana-oip-start-interval-0s)
              stop interval=0s timeout=180 (hana-oip-stop-interval-0s)

Stonith Devices:
 Resource: clusterfence (class=stonith type=fence_aws)
  Attributes: pcmk_host_map=prihana:i-0df8622xxxxxxxxxxx;sechana:i-0b2e372xxxxxxxxxxx
pcmk_reboot_retries=4 pcmk_reboot_timeout=480 power_timeout=240 region=us-east-1
 pcmk_reboot_action=off
  Operations: monitor interval=60s (clusterfence-monitor-interval-60s)
Fencing Levels:

Location Constraints:
Ordering Constraints:
  start SAPHanaTopology_HDB_00-clone then start SAPHana_HDB_00-master (kind:Mandatory)
```

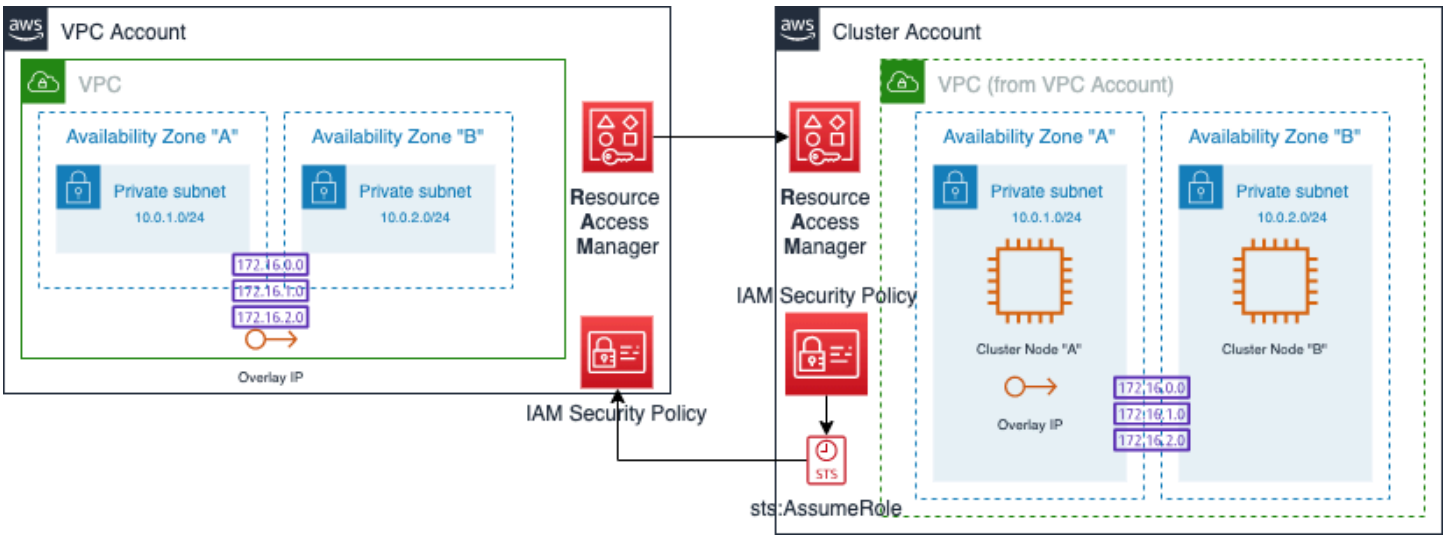
```
  (non-symmetrical)
Colocation Constraints:
  hana-oip with SAPHana_HDB_00-master (score:2000) (rsc-role:Started) (with-rsc-
role:Master)
Ticket Constraints:

Alerts:
 No alerts defined

Resources Defaults:
 No defaults set
Operations Defaults:
 No defaults set

Cluster Properties:
 cluster-infrastructure: corosync
 cluster-name: rhelhanaha
 dc-version: 1.1.19-8.el7_6.4-c3c624ea3d
 have-watchdog: false
 last-lrm-refresh: 1553719142
 maintenance-mode: false
Node Attributes:
 prihana: hana_hdb_op_mode=logreplay hana_hdb_remoteHost=sechana hana_hdb_
 site=SiteA hana_hdb_srmode=syncmem hana_hdb_vhost=prihana lpa_hdb_lpt=10
 sechana: hana_hdb_op_mode=logreplay hana_hdb_remoteHost=prihana hana_hdb_
 site=SiteB hana_hdb_srmode=syncmem hana_hdb_vhost=sechana lpa_hdb_lpt=1553719113
Cluster name: rhelhanaha
Stack: corosync
```

High availability cluster and shared Amazon VPC

Amazon VPC sharing enables you to share subnets with other AWS accounts within the same AWS Organization. Amazon EC2 instances can be deployed using the subnets of the shared Amazon VPC. For more information, see Amazon VPC sharing blog.

This guide assumes that an AWS Organization is already setup and that Amazon VPC subnets have been shared between AWS accounts using the AWS RAM. For more details, see Create a resource share.

The following image shows a sample architecture design.



Note

Further in this guide, we refer to the AWS account that owns the Amazon VPC as the **Amazon VPC account** and to the account using the Amazon VPC where the cluster nodes are going to be deployed as the **Cluster account**.

Overlay IP with shared Amazon VPC

Using the overlay IP agent with a shared Amazon VPC requires a different set of IAM permissions to be granted on both AWS accounts (sharing and consumer). The cluster resource agent `aws-vpc-move-ip` also uses a different configuration syntax.

Overlay IP address

Create an overlay IP address on the Amazon VPC routing table which will be used by the Amazon VPC subnets and will be accessible to the cluster. This must be created on the AWS account sharing the Amazon VPC.

IAM roles and policies

Amazon VPC account

Create an IAM role to delegate permissions to the Amazon EC2 instances that will be a part of the cluster. When creating the IAM role, select **Another AWS account** for the type of trusted entity and enter the AWS Account ID where the Amazon EC2 instances will be deployed.

Create the following IAM policy on the Amazon VPC account and attach it to the IAM role. Add or remove route table entries as needed.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": “ec2:ReplaceRoute”,
            "Resource": [
   "arn:aws:ec2:<AWS Region>:<VPC-Account-Number>:route-table/rtb-xxxxxxxxxxxxxxxx",
   "arn:aws:ec2:<AWS Region>:<VPC-Account-Number>:route-table/rtb-xxxxxxxxxxxxxxxx"
   ],
        },
        {
            "Effect": "Allow",
            "Action": “ec2:DescribeRouteTables”,
            "Resource": “*”
        }
    ]
}
```

Cluster account

Create a new IAM role and select **Amazon EC2** as the use case. Associate this IAM role to the two Amazon EC2 instances which are a part of the cluster. Attach the following IAM policies (AWS STS and STONITH) to the IAM role.

AWS STS policy

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": "sts:AssumeRole",
            "Resource": "arn:aws:iam::<VPC-Account-Number>:role/<Sharing-VPC-Account-
Cluster-Role>"
```

```
        }
    ]
}
```

Replace VPC-Account-Number with your AWS account number that owns the Amazon VPC. Replace Sharing-VPC-Account-Cluster-Role with the IAM role that was created in the AWS account owning the Amazon VPC.

STONITH policy

Both instances of the cluster require access to start and stop other nodes within the cluster. Create the following STONITH policy and attach it to the IAM role that is assigned to both of the cluster instances.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeInstances",
                "ec2:DescribeInstanceAttribute",
                "ec2:DescribeTags"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:ModifyInstanceAttribute",
                "ec2:RebootInstances",
                "ec2:StartInstances",
                "ec2:StopInstances"
            ],
            "Resource": [
                "arn:aws:ec2:<Region-name>:<account-id>:instance/<instance-id>",
                "arn:aws:ec2: <Region-name>:<account-id>:instance/<instance-id>"
            ]
        }
    ]
}
```

Replace Region-name, account-id, and instance-id with the appropriate values.

High availability cluster setup

The following are the minimum version requirements needed to support shared Amazon VPC:

- Red Hat 7.9 - resource-agents-4.1.1-61.10
- Red Hat 8.1 - resource-agents-4.1.1-33.10
- Red Hat 8.2 - resource-agents-4.1.1-44.12
- SLES 12 SP5 - resource-agents-4.3.018.a7fb5035-3.79.1.x86_64
- SLES 15 SP2 - resource-agents-4.4.0+git57.70549516-3.30.1.x86_64
- SLES 15 SP3 - resource-agents-4.8.0+git30.d0077df0-8.5.1

Setup on SLES

1. Add the overlay ip address to the primary node using the following command.

```
prihana:~ # ip address add xxx.xxx.xxx.xxx/32 dev eth0
```

2. Create a file named `aws-move-ip.txt` with the following cluster options.

```
prihana:~ # cat aws-move-ip.txt
primitive res_AWS_IP ocf:suse:aws-vpc-move-ip \
params ip=<overlay ip address> \
routing_table=<route table identifier 1>,<route table identifier 2> \
interface=eth0 \
profile=cluster \
lookup_type=NetworkInterfaceId \
routing_table_role="arn:aws:iam::<VPC-Account-Number>:role/<VPC-Account-Cluster-
Role>" \
```

```
op start interval=0 timeout=180 \
op stop interval=0 timeout=180 \
op monitor interval=60 timeout=60
```

3. Use the following command to add the overlay ip configuration file to the cluster.

```
prihana:~ # crm configure load update aws-move-ip.txt
```

Setup on RHEL

1. Add the overlay ip address to the primary node using the following command.

```
[root@prihana ~]# ip address add xxx.xxx.xxx.xxx/32 dev eth0
```

2. Configure the cluster resource agent according to the following example.

```
[root@prihana ~]# pcs resource create res_AWS_IP aws-vpc-move-ip \
ip=<overlay ip address> \
routing_table=<route table identifier 1>,<route table identifier 2> \
interface=eth0 \
profile=cluster \
lookup_type=NetworkInterfaceId \
routing_table_role="arn:aws:iam::<VPC-Account-Number>:role/<VPC-Account-Cluster-
Role>" \
op start interval=0 timeout=180 \
op stop interval=0 timeout=180 \
op monitor interval=60 timeout=60
```

Document history

Date	Change
March, 2022	Added the section on high availability cluster and shared Amazon VPC
March, 2021	Initial publication