Current document is for the Private Preview release of Azure Application Consistent Snapshot Tools (azacsnap).

Microsoft Azure Application Consistent Snapshot Tools for SAP HANA on Azure

Abstract

How to guide for the snapshot tools Microsoft provide to perform snapshots and disaster recovery fail-over of SAP HANA on Azure.

Table of Contents

Version

This document is for the Application Consistent Snapshot Tools for SAP HANA on Azure **version 5.0**.

Authors

Phil Jensen

Contributors

Juergen Thomas, Sachin Ghorpade, Amish Patel, Serge Muts, Vamsi Sangam.

Introduction

This document provides details on the application consistent snapshot tool used for the SAP HANA on Azure Large Instances and SAP HANA on Azure Virtual Machines with Azure NetApp Files.

This snapshot tool is tested and supported by Microsoft and serves the following purposes:

1. Create storage snapshots of SAP HANA.

2. Check status, backup and removal of storage snapshots.

For Azure Large Instances

1. Self Service Disaster Recovery failover when the DR scenario has been deployed.

This document is intended to provide "How To" details about these snapshot tools which are developed and supported by Microsoft for the specific purpose of performing application consistent storage snapshots for SAP HANA systems deployed on Azure Large Instances and Azure NetApp Files.

Disclaimer: *This guide is written based on successfully testing the set up in Microsoft's lab containing SAP HANA on Azure Large Instances and Azure NetApp Files. Customers are responsible for monitoring and testing to ensure the snapshot tools are performing as expected.*

Terms and Definitions

Glossary of terms and definitions used in this documentation:

- **SID** : A system identifier for HANA system.
- **Multipurpose DR** : A disaster recovery system which has more than one instances configured. One of the instances is intended to provide DR for the production SID, other instances are non-production HANA instances.
- **Normal DR** : A disaster recovery system which just has a primary SID configured with storage replication running. There is no other workload running on a “Normal DR” while production instance is up and running at primary site.
- **Single SID system** : A system which has only one HANA instance configured.
- **Multi SID system** : A system which has more than one HANA instances configured. Also known in SAP documentation as MCOS deployment.
- **HLI** : SAP HANA on Azure Large Instances Unit.
- **DR** : Disaster Recovery.
- **HSR** : HANA System Replication.

Overview

The self-installer provides a single file bundle for customers to install and configure the **Azure Application Consistent Snapshot tools for SAP HANA on Azure** with storage hosted on **Azure NetApp Files** or **Azure Lage Instance**.

The snapshot tools included in the installation file are described as follows:

Used at the Primary Site (Azure Large Instance and Azure NetApp Files)

- **azacsnap**: The primary tool to configure, test, and manage application consistent storage snapshots for SAP HANA.
  - This single tool extends the functionality and replaces the many of the older commands listed below which were available for Azure Large Instance systems *only*. The tools replaced with this command are:
    - `azure_hana_backup` The primary tool to execute database consistent storage snapshots for the SAP HANA data & shared, logbackups, or boot volumes.
    - `testHANAConnection`: This command tests the connection to the SAP HANA instance and is required to validate set up of the snapshot tools.
    - `testStorageSnapshotConnection`: This snapshot command has two main steps. First, it ensures that the SAP HANA on Azure Large Instance system that runs the snapshot tools can communicate with the underlying storage interface. Second, it creates a temporary storage snapshot for the SAP HANA on Azure Large Instance being tested. This command should be run for every HANA instance on a server to ensure the snapshot tools can communicate with the storage so they function as expected.
    - `removeTestStorageSnapshot`: Removed any test snapshots created with `testStorageSnapshotConnection`. This is automatically done when a `azacsnap -c test --test storage` command is run.
    - `azure_hana_snapshot_details`: Provides a list of basic details about the snapshots, per volume, that exist in your environment. This command can be run on the primary server or on a server in the disaster-recovery location. The command provides the following information broken down by each volume that contains snapshots:
      - Size of total snapshots in a volume.
      - Each snapshot in that volume includes the following details:
      - Snapshot name
      - Create time
      - Size of the snapshot
      - Frequency of the snapshot
    - `azure_hana_snapshot_delete`: This command deletes a storage snapshot or a set of snapshots. You can use either the SAP HANA Backup ID as found in HANA Studio or the storage snapshot name. The Backup ID is

only tied to the 'hana' snapshots which are created for the data & shared volumes. Otherwise, if the snapshot name is entered, it searches for all snapshots that match the entered snapshot name.

- `HANABackupCustomerDetails.txt`: the old line oriented configuration file has been replaced with a JSON configuration file which can be generated by using the `azacsnap -c configure --configuration new` command.

Used at the DR Site (Azure Large Instance only)

The following tools are for use at the Disaster Recovery Site with **Azure Large Instance** only when configured with storage replication to the DR Site. No such tools exist for **Azure NetApp Files** at this stage.

- **azure_hana_replication_status**: Provides basic details around the replication status from the production site to the disaster-recovery site. The snapshot command monitors to ensure that the replication is taking place, and it shows the size of the items that are being replicated. It also provides guidance if a replication is taking too long or if the link is down.
- **azure_hana_dr_failover:** This command forces a DR Failover into another paired region. This snapshot command **stops** storage replication from the primary site to the secondary site, and presents the latest available snapshot on the DR volumes along with recommended filesystem mountpoints for the DR volumes. This command must be run on the HANA Large Instance system **in the DR region** (i.e. the target fail-over system).
- **azure_hana_test_dr_failover:** This command performs a test failover to the paired DR site. Unlike the azure_hana_dr_failover command, this execution does not interrupt the storage replication from primary to secondary. Instead clones of the latest available snapshot are created at the DR site and recommended filesystem mountpoints of the cloned volumes are presented. This command must be run on the HANA Large Instance system **in the DR region** (i.e. the target fail-over system).

Getting the snapshot tools

It is recommended customers get the most recent version of the self-installation file (e.g. azacsnap_installer_v5.0.run or later) which contains the snapshot tools from Microsoft. Then follow the steps in the Technical Setup section of this guide to install.

The self-installation file is signed with Microsoft's public key to allow for GPG verification of the download.

Verifying the download

The installer, which is downloadable per above, has an associated PGP signature file with a .asc filename extension. This file can be used to verify the downloaded installer to ensure this is a Microsoft provided file. The Microsoft PGP Public Key used for signing Linux packages is available here (http://packages.microsoft.com/keys/microsoft.asc) and has been used to sign the signature file.

The Microsoft PGP Public Key can be imported to a user's local as follows:

```
> wget http://packages.microsoft.com/keys/microsoft.asc
> gpg --import microsoft.asc
```

The following commands trust the Microsoft PGP Public Key:

1. List the keys in the store.
2. Edit the Microsoft key.
3. Check the fingerprint with fpr
4. Sign the key to trust it.

```
> gpg --list-keys
----<snip>----
pub rsa2048 2015- 10 - 28 [SC]
BC528686B50D79E339D3721CEB3E94ADBE1229CF
uid [ unknown] Microsoft (Release signing) gpgsecurity@microsoft.com

> gpg --edit-key gpgsecurity@microsoft.com
gpg (GnuPG) 2.1.18; Copyright (C) 2017 Free Software Foundation, Inc.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
pub rsa2048/EB3E94ADBE1229CF
created: 2015- 10 - 28 expires: never usage: SC
trust: unknown validity: unknown
[ unknown] (1). Microsoft (Release signing) <gpgsecurity@microsoft.com>

gpg> fpr
pub rsa2048/EB3E94ADBE1229CF 2015- 10 - 28 Microsoft (Release signing)
<gpgsecurity@microsoft.com>
Primary key fingerprint: BC52 8686 B50D 79E3 39D3 721C EB3E 94AD BE12 29CF

gpg> sign
pub rsa2048/EB3E94ADBE1229CF
created: 2015- 10 - 28 expires: never usage: SC
trust: unknown validity: unknown
Primary key fingerprint: BC52 8686 B50D 79E3 39D3 721C EB3E 94AD BE12 29CF
Microsoft (Release signing) <gpgsecurity@microsoft.com>
Are you sure that you want to sign this key with your
key "XXX XXXX <xxxxxxx@xxxxxxxx.xxx>" (A1A1A1A1A1A1)
Really sign? (y/N) y

gpg> quit
Save changes? (y/N) y
```

The PGP signature file for the installer can be checked as follows:

```
> gpg --verify azacsnap_installer_v5.0.run.asc azazsnap_installer_v5.0.run
gpg: Signature made Sat 13 Apr 2019 07:51:46 AM STD
gpg: using RSA key EB3E94ADBE1229CF
gpg: Good signature from "Microsoft (Release signing)
<gpgsecurity@microsoft.com>" [full]
```

More details on using GPG can be found in the online manual:

- https://www.gnupg.org/gph/en/manual/book1.html

Supported Scenarios

The snapshot tools can be used in the following scenarios.

- Single SID
- Multiple SID
- HSR
- Scale-out
- MDC (Only single tenant supported)
- Single Container
- SUSE Operating System
- RHEL Operating System
- SKU TYPE I
- SKU TYPE II

ref: https://docs.microsoft.com/en-us/azure/virtual-machines/workloads/sap/hana-supported-scenario

Snapshot Support Matrix from SAP

The following matrix is provided as a guideline on which versions of SAP HANA are supported by SAP for Storage Snapshot Backups.

	1.0 SPS12	2.0 SPS0	2.0 SPS1	2.0 SPS2	2.0 SPS3	2.0 SPS4
Single Container Database	√	√	-	-	-	-
MDC Single Tenant	-	-	√	√	√	√
MDC Multiple Tenants	-	-	-	-	-	√

√ = supported by SAP for Storage Snapshots

Important things to remember

- SAP does not support snapshot on the MDC environment for the following releases. Though snapshot is supported with non-MDC setup for the following releases. The snapshot command does not work if you have following HANA releases with MDC setup.
  - HANA 2 SP
  - HANA 1 SP 12
  - HANA 1 SP
  - HANA 1 SP
- For HANA 2 SPS04 and later releases: for MDC environment with more than one tenant, a database consistent storage snapshot is supported by SAP and works with these snapshot tools.
- For HANA 2 SP1 and releases prior to SPS04: for MDC environment with more than one tenant, a database consistent storage snapshot is not supported by SAP. For single tenant it is supported by SAP and works with these snapshot tools.
- After setup of the snapshot tools, continuously monitor the storage space available and if required, delete the old snapshots on a regular basis to avoid storage fill out.
- Always use the latest snapshot tools. You can download the latest version from GitHub.
- Use the same version of the snapshot tools across the landscape.
- Test the snapshot tools and get comfortable with the parameters required and output of the snapshot command before using in the production system.
- Don't change the mount point name of the server provisioned by the Microsoft Operations. These snapshot tools rely on these standard mount point names to be available for a successful execution.
- When setting up the HANA user for backup (details below in this document), you need to setup the user for each HANA instance. Create an SAP HANA user account to access HANA instance under the SYSTEMDB (and not in the SID database) for MDC. In the single container environment, it can be setup under the tenant database.
- Customers must provide the SSH public key for storage access. This must be done once per node and for each user under which the snapshot command is executed.
- The number of snapshots per volume is limited to 250.
- If asked to modify the configuration file, always use the Linux text editor such as "vi" and not the Windows editors like notepad. Using Windows editor may corrupt the file format.
- The snapshot tools can now (v4.0+) run as a non-root user. Previously, they were only supported to run under root or sidadm user. The installer will setup and configure this automatically if the following pre-requisites are met as the root user.
  - Setup hdbuserstore for the SAP HANA user to communicate with SAP HANA.
  - Setup SSH private/public key pairs for the user to communicate with the storage sub- system.
- For DR: The snapshot tools must be tested on DR node before DR is setup.

- Monitor disk space regularly, automated log deletion is managed with the `--trim` option of the `azure_hana_backup tool` for SAP HANA 2 and later releases.
- **Risk of snapshots not being taken** - The snapshot tools only interact with the node of the SAP HANA system specified in the configuration file. If this node becomes unavailable there is no mechanism to automatically start communicating with another node.
  - For a **SAP HANA Scale-Out with Standby** scenario it is typical to install and configure the snapshot tools on the master node. But, if the master node becomes unavailable, the standby node will take over the master node role. In this case the implementation team should configure the snapshot tools on both nodes (Master and Stand-By) to avoid any missed snapshots. In the normal state the master node will take HANA snapshots initiated by crontab, but after master node failover those snapshots will have to be executed from another node such as the new master node (former standby). To achieve this the standby node would need the snapshot tool installed, storage communication enabled, hdbuserstore configured, HANABackupCustomerDetails.txt configured, and crontab commands staged in advance of the failover.
  - For a **SAP HANA HSR HA** scenario, it is recommended to install, configure, and schedule the snapshot tools on both (Primary and Secondary) nodes. Then, if the Primary node becomes unavailable, the Secondary node will take over with snapshots being taken on the Secondary. In the normal state the Primary node will take HANA snapshots initiated by crontab and the Secondary node would attempt to take snapshots but fail as the Primary is functioning correctly. But after Primary node failover those snapshots will be executed from the Secondary node. To achieve this the Secondary node needs the snapshot tool installed, storage communication enabled, hdbuserstore configured, HANABackupCustomerDetails.txt configured, and crontab enabled in advance of the failover.

Guidance provided in this document

The following guidance is provided in this document to illustrate the usage of the snapshot tools.

Taking Snapshot Backups

- What are the prerequisites for the storage snapshot
  - Enable communication with storage
  - Enable communication with SAP HANA
- How to take snapshots manually
- How to setup automatic snapshot backup
- How to monitor the snapshots
- How to delete a snapshot?
- How to restore a 'hana' snapshot
- How to restore a 'boot' snapshot
- What are key facts to know about the snapshots

Snapshots are tested for both single SID and multi SID.

Performing Disaster Recovery

- What are the prerequisites for DR setup
- How to setup a disaster recovery
- How to monitor the data replication from Primary to DR site
- How to perform a failover to DR site?

DR is tested for single SID failover on a multipurpose DR setup.

Technical Setup

To test and document these snapshot tools, the following set up was used. The information presented correspond to **version 5.0+** of the snapshot tools.

- Operating System: SLES 12 SP2+ for SAP
- HANA Large Instances: 2xS192 (four sockets, 2 TB); One for Primary site and second for DR site
  - Primary site HLI unit (sapprdhdb80) has 3 SIDs configured: H80, H81, and H82
  - DR site HLI unit (sapdrhdb80) has 2 SIDs configured: Q85, and H80 (replication from the Primary site)
- HANA Version: HANA 2.0 SP1+
- Server Names: sapprdhdb80 (Primary) and sapdrhdb80 (DR node).
  - Note `prd` and `dr` in their hostnames.

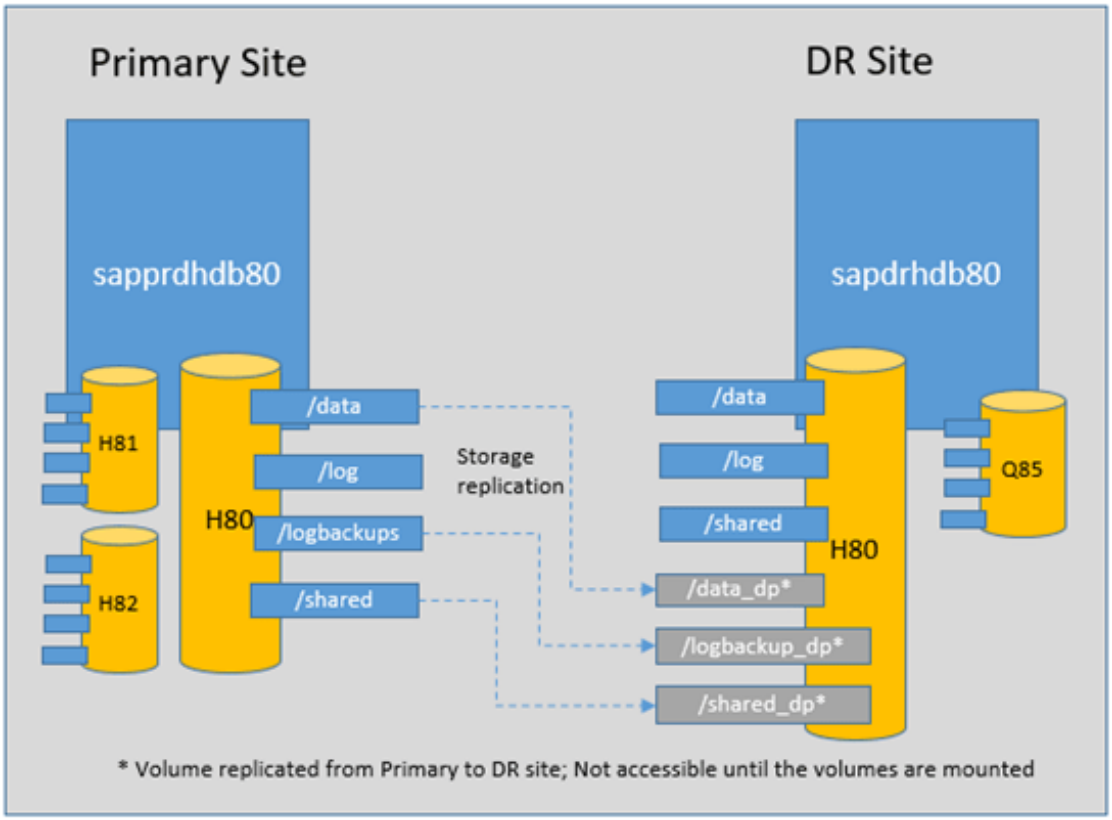

Figure 1 - Technical Setup

CAUTION You may have different screens for operating system or HANA depending on the version you are using. Also, based on your HANA version, the snapshot tools parameter may differ. Refer the snapshot command details before using them.

Pre-requisites for installation

Please follow the guidelines to setup and execute the snapshots and disaster recovery commands. It is recommended the following steps are completed as root before installing and using the snapshot tools.

1. **OS is patched** : Please refer for patching and SMT setup https://docs.microsoft.com/en-us/azure/virtual-machines/workloads/sap/hana-installation#setting-up-smt-server-for-suse-linux
2. **Time Synchronization is setup**. The customer will need to provide a NTP compatible time server, and configure the OS accordingly.
3. **HANA is installed** : Please refer for HANA installation instructions: https://blogs.msdn.microsoft.com/saponsqlserver/2017/11/21/sap-netweaver-installation- on-hana-database/.
    1. In this document, we installed HANA 2.0 SP1 with multi SID as following:
        1. **Primary site Azure Large Instance system** (sapprdhdb80) has three HANA instances configured with the SIDs: H80, H81, and H82. To install three instances, you need to run the hana installer (hdblcmgui) three times with "new instance" option, and then provide SID information each time for each instance. So, in this example, you run hdblcmgui, three times with H80, H81, and H82 SIDs.
        2. **DR site HLI Azure Large Instance system** (sapdrhdb80) has two HANA instances configured: Q85, and H80. To install two instances, you need to run the hana installer (hdblcmgui) two times with "new instance" option, and then provide SID information each time for each instance. So, in this example, you run hdblcmgui, 2 times with H80 and Q85 SIDs.
4. **Enable communication with storage** (refer separate section for more details) : Customer must setup SSH with a private/public key pair, and provide the public key for each node where the snapshot tools are planned to be executed to Microsoft Operations for setup on the storage back-end.
    1. **For Azure NetApp Files (refer separate section for details)**: Customer must generate the service principal authentication file.
    2. **For Azure Large Instance (refer separate section for details)**: Customer must setup SSH with a private/public key pair, and provide the public key for each node where the snapshot tools are planned to be executed to Microsoft Operations for setup on the storage back-end.

        Test this by running the command `ssh -l <Storage UserNameeg: clt1h80backup><Storage IP Address>` from one of the nodes to ensure the connectivity of the node to the storage. Type `exit` to logout of the storage prompt.

        Microsoft operations will provide the storage user and storage IP at the time of provisioning. Refer section HANABackupCustomerDetails.txtfor more details.

5. **Enable communication with SAP HANA** (refer separate section for more details) : Customer must setup an appropriate SAP HANA user with the required privileges to perform the snapshot.
    1. This can be tested from the command line as follows using the text in `grey`
        1. HANAv1

            ```
            hdbsql -n <HANA IP address> -i <HANA instance> -U <HANA user> "\s"
            ```

        2. HANAv2

            ```
            hdbsql -n <HANA IP address> -i <HANA instance> -d SYSTEMDB -U <HANA user> "\s"
            ```

        - The examples above are for non-SSL communication to SAP HANA.

Enable communication with storage

Azure NetApp Files

Create RBAC Service Principal

1. Within an Azure Cloud Shell session, make sure you're logged on at the subscription where you want to be associated with the service principal by default:

    ```
    Cloud Shell User@Azure> az account show
    ```

2. If this is not the correct subscription, use

    ```
    Cloud Shelluser@Azure> az account set -s <subscription name or id>
    ```

3. Create a service principal using Azure CLI per the following example

    ```
    Cloud Shell User@Azure> az ad sp create-for-rbac --sdk-auth
    ```

    1. This should generate an output like the following:

        ```
        {
          "clientId": "00aa000a-aaaa-0000-00a0-00aa000aaa0a",
          "clientSecret": "00aa000a-aaaa-0000-00a0-00aa000aaa0a",
          "subscriptionId": "00aa000a-aaaa-0000-00a0-00aa000aaa0a",
          "tenantId": "00aa000a-aaaa-0000-00a0-00aa000aaa0a",
          "activeDirectoryEndpointUrl": "https://login.microsoftonline.com",
          "resourceManagerEndpointUrl": "https://management.azure.com/",
          "activeDirectoryGraphResourceId": "https://graph.windows.net/",
          "sqlManagementEndpointUrl": "https://management.core.windows.net:8443/",
          "galleryEndpointUrl": "https://gallery.azure.com/",
          "managementEndpointUrl": "https://management.core.windows.net/"
        }
        ```

    This command will automatically assign RBAC contributor role to the service principal at subscription level, you can narrow down the scope to the specific resource group where your tests will create the resources.

4. Cut and Paste the output content into a file called `azureauth.json` stored on the same system as the `azacsnap` command and secure the file with appropriate system permissions.

Azure Large Instance

Communication with the storage back-end executes over an encrypted SSH channel. The following example steps are to provide guidance on setup of SSH for this communication.

1. Modify the `/etc/ssh/ssh_config` file

   Refer to the following output where the `MACs hmac-sha1` line has been added:

```
# RhostsRSAAuthentication no
# RSAAuthentication yes
# PasswordAuthentication yes
# HostbasedAuthentication no
# GSSAPIAuthentication no
# GSSAPIDelegateCredentials no
# GSSAPIKeyExchange no
# GSSAPITrustDNS no
# BatchMode no
# CheckHostIP yes
# AddressFamily any
# ConnectTimeout 0
# StrictHostKeyChecking ask
# IdentityFile ~/.ssh/identity
# IdentityFile ~/.ssh/id_rsa
# IdentityFile ~/.ssh/id_dsa
# Port 22
Protocol 2
# Cipher 3des
# Ciphers aes128-ctr,aes192-ctr,aes256-ctr,arcfour256,arcfour128,aes128-
cbc,3des-
cbc
# MACs hmac-md5,hmac-sha1,umac-64@openssh.com,hmac-ripemd
MACs hmac-sha
# EscapeChar ~
# Tunnel no
# TunnelDevice any:any
# PermitLocalCommand no
# VisualHostKey no
# ProxyCommand ssh -q -W %h:%p gateway.example.com
```

2. Create a private/public key pair

   Using the following example command to generate the key pair, do not enter a password when generating a key.

```
> ssh-keygen -t rsa –b 5120 -C ""
```

3. Send the public key to Microsoft Operations

Send the output of the `cat /root/.ssh/id_rsa.pub` command (example below) to Microsoft Operations to enable the snapshot tools to communicate with the storage sub-system.

```
> cat /root/.ssh/id_rsa.pub
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABAQDoaRCgwn1Ll31NyDZy0UsOCKcc9nu2qdAPHdCzleiTWISvPW
FzIFxz8iOaxpeTshH7GRonGs9HNtRkkz6mpK7pCGNJdxS4wJC9MZdXNt+JhuT23NajrTEnt1jXiVFH
bh3jD7LjJGMb4GNvqeiBExyBDA2pXdlednOaE4dtiZ1N03Bc/J4TNuNhhQbdsIWZsqKt9OPUuTfD
j0XvwUTLQbR4peGNfN1/cefcLxDlAgI+TmKdfgnLXIsSfbacXoTbqyBRwCi7p+bJnJD07zSc9YCZJa
wKGAIilSg7s6Bq/2lAPDN1TqwIF8wQhAg2C7yeZHyE/ckaw/eQYuJtN+RNBD
```

Enable communication with SAP HANA

The snapshot tools communicate with SAP HANA and need a user with appropriate permissions to initiate and release the database save-point. The following provides an example on setup of the SAP HANA v2 user and the hdbuserstore for communication to the SAP HANA database.

The following example commands setup a user (AZACSNAP) in the SYSTEMDB on SAP HANA 2. database, change the IP address, usernames and passwords as appropriate:

1. Connect to the SYSTEMDB to create the user

```
> hdbsql -n <IP_address_of_host>:30013 -i 00 -u SYSTEM -p <SYSTEM_USER_PASSWORD>

Welcome to the SAP HANA Database interactive terminal.

Type: \h for help with commands
\q to quit

hdbsql SYSTEMDB=>
```

2. Create the user

   This example creates the AZACSNAP user in the SYSTEMDB.

```
hdbsql SYSTEMDB=> CREATE USER AZACSNAP PASSWORD <AZACSNAP_PASSWORD_CHANGE_ME>
NO FORCE_FIRST_PASSWORD_CHANGE;
```

3. Grant the user permissions

   This example sets the permission for the AZACSNAP user to allow for performing a database consistent storage snapshot.

```
hdbsql SYSTEMDB=> GRANT BACKUP ADMIN, CATALOG READ, MONITORING TO AZACSNAP;
```

4. *OPTIONAL* - Prevent user's password from expiring

   CAUTION Recommend checking with corporate policy before making this change.

   This example disables the password expiration for the AZACSNAP user, without this change the user's password will expire preventing snapshots to be taken correctly.

```
hdbsql SYSTEMDB=> ALTER USER AZACSNAP DISABLE PASSWORD LIFETIME;
```

5. Setup the SAP HANA Secure User Store (change the password) This example uses the hdbuserstore command from the Linux shell to setup the SAP HANA Secure User store.

```
> hdbuserstore Set AZACSNAP <IP_address_of_host>:30013 AZACSNAP
<AZACSNAP_PASSWORD_CHANGE_ME>
```

6. Check the SAP HANA Secure User Store To check if the secure user store is setup correctly, use the hdbuserstore command to list the output which should be similar to the following. More details on using hdbuserstore are available on the SAP website.

```
> hdbuserstore List
DATA FILE : /home/azacsnap/.hdb/sapprdhdb80/SSFS_HDB.DAT
KEY FILE : /home/azacsnap/.hdb/sapprdhdb80/SSFS_HDB.KEY


KEY AZACSNAP
ENV : <IP_address_of_host>:
USER: AZACSNAP
```

Additional instructions for using the log trimmer (SAP HANA 2.0 and later)

If using the log trimmer, then the following example commands setup a user (AZACSNAP) in the TENANT database(s) on a SAP HANA 2.0 database system. Remember to change the IP address, usernames and passwords as appropriate:

1. Connect to the TENANT database to create the user, tenant specific details are `<IP_address_of_host>` and `<SYSTEM_USER_PASSWORD>`. Also, note the port (`30015`) required to communicate with the TENANT database.

```
> hdbsql -n <IP_address_of_host>:30015 - i 00 -u SYSTEM -p
<SYSTEM_USER_PASSWORD>

Welcome to the SAP HANA Database interactive terminal.

Type: \h for help with commands
\q to quit

hdbsql TENANTDB=>
```

2. Create the user

   This example creates the AZACSNAP user in the SYSTEMDB.

```
hdbsql TENANTDB=> CREATE USER AZACSNAP PASSWORD <AZACSNAP_PASSWORD_CHANGE_ME>
NO FORCE_FIRST_PASSWORD_CHANGE;
```

3. Grant the user permissions

   This example sets the permission for the AZACSNAP user to allow for performing a database consistent storage snapshot.

```
hdbsql TENANTDB=> GRANT BACKUP ADMIN, CATALOG READ, MONITORING TO AZACSNAP;
```

4. *OPTIONAL* - Prevent user's password from expiring

   CAUTION Recommend checking with corporate policy before making this change.

   This example disables the password expiration for the AZACSNAP user, without this change the user's password will expire preventing snapshots to be taken correctly.

   ```
   hdbsql TENANTDB=> ALTER USER AZACSNAP DISABLE PASSWORD LIFETIME;
   ```

Note repeat these steps for all the tenant databases. It's possible to get the connection details for all the tenants using the following SQL query against the SYSTEMDB.

```
SELECT HOST, SQL_PORT, DATABASE_NAME FROM SYS_DATABASES.M_SERVICES WHERE SQL_PORT
LIKE '3%'
```

See the following example query and output.

```
> hdbsql -jaxC -n 10.90.0.31:30013 -i 00 -u SYSTEM -p <SYSTEM_USER_PASSWORD> " SELECT
HOST,
SQL_PORT, DATABASE_NAME FROM SYS_DATABASES.M_SERVICES WHERE SQL_PORT LIKE '3%' "
sapprdhdb80,30013,SYSTEMDB
sapprdhdb80,30015,H81
sapprdhdb80,30041,H82
```

Using SSL for communication with SAP HANA

The `azacsnap` tool utilises SAP HANA's `hdbsql` command to communicate with SAP HANA. This includes the use of SSL options when encrypting communication with SAP HANA. `azacsnap` uses `hdbsql` command's SSL options as follows.

The following are always used when using the `azacsnap --ssl` option:

- `-e` - Enables TLS encryptionTLS/SSL encryption. The server chooses the highest available.
- `-ssltrustcert` - Specifies whether to validate the server's certificate.
- `-sslhostnameincert "*"` - Specifies the host name used to verify server's identity. By specifying `"*"` as the host name, then the server's host name is not validated.

SSL communication also requires Key Store and Trust Store files. While it is possible for these files to be stored in default locations on a Linux installation, to ensure the correct key material is being used for the various SAP HANA systems (i.e. in the cases where different key-store and trust-store files are used for each SAP HANA system) `azacsnap` expects the key-store and trust-store files to be stored in the `securityPath` location as specified in the `azacsnap` configuration file.

Key Store files - If using multiple SIDs with the same key material it's easier to create links into the securityPath location as defined in the `azacsnap` config file. Ensure these exist for every SID using SSL. - For openssl: - `ln $HOME/.ssl/key.pem <securityPath>/<SID>_keystore` - For commoncrypto: - `ln $SECUDIR/sapcli.pse <securityPath>/<SID>_keystore` - If using multiple SIDs with the different key material per SID, copy (or move and rename) the files into the securityPath location as defined in the SIDs `azacsnap` config file. - For openssl: - `mv`

`key.pem <securityPath>/<SID>_keystore` - For commoncrypto: - `mv sapcli.pse <securityPath>/<SID>_keystore`

When `azacsnap` calls `hdbsql` it will add `-sslkeystore=<securityPath>/<SID>_keystore` to the command line.

Trust Store files - If using multiple SIDs with the same key material create hard-links into the securityPath location as defined in the `azacsnap` config file. Ensure these exist for every SID using SSL. - For openssl: - `ln $HOME/.ssl/trust.pem <securityPath>/<SID>_truststore` - For commoncrypto: - `ln $SECUDIR/sapcli.pse <securityPath>/<SID>_truststore` - If using multiple SIDs with the different key material per SID, copy (or move and rename) the files into the securityPath location as defined in the SIDs `azacsnap` config file. - For openssl: - `mv trust.pem <securityPath>/<SID>_truststore` - For commoncrypto: - `mv sapcli.pse <securityPath>/<SID>_truststore`

Note The `<SID>` component of the file names must be the SAP HANA System Identifier all in upper case (e.g. `H80`, `PR1`, etc.)

When `azacsnap` calls `hdbsql` it will add `-ssltruststore=<securityPath>/<SID>_truststore` to the command line.

Therefore, running `azacsnap -c test --test hana --ssl openssl` wher the `SID` is `H80` in the config file, it would execute the `hdbsql`connections as follows:

```
hdbsql \
    -e \
    -ssltrustcert \
    -sslhostnameincert "*" \
    -sslprovider openssl \
    -sslkeystore ./security/H80_keystore \
    -ssltruststore ./security/H80_truststore
    "sql statement"
```

Note The `\` character is a command line line-wrap to improve clarity of the multiple parameters passed on the command line.

Installing the snapshot tools

The downloadable self-installer is designed to make the snapshot tools easy to setup and run with non-root user privileges (e.g. azacsnap). The installer will setup the user and put the snapshot tools into the users `$HOME/bin` subdirectory (default = `/home/azacsnap/bin`).

The self-installer tries to determine the correct settings and paths for all the files based on the configuration of the user performing the installation (e.g. root). If the previous setup steps (Enable communication with storage and SAP HANA) were run as root, then the installation will copy the private key and the hdbuserstore to the backup user's location. However, it is possible for the steps which enable communication with the storage back-end and SAP HANA to be manually done by a knowledgeable administrator after the installation.

Note For earlier SAP HANA on Azure Large Instance installations, the directory of pre-installed snapshot tools was `/hana/shared/<SID>/exe/linuxx86_64/hdb`.

With the pre-requisite steps completed, it's now possible to install the snapshot tools using the self- installer as follows:

1. Copy the downloaded self-installer to the target system.
2. Execute the self-installer as the `root` user, see the following example. If necessary, make the file executable using the `chmod +x *.run` command.

Running the self-installer command without any arguments will display help on using the installer to install the snapshot tools as follows:

```
# chmod +x azacsnap_installer_v5.0.run
# ./azacsnap_installer_v5.0.run
Usage: ./azacsnap_installer_v5.0.run [-v] -I [-u <HLI Snapshot Command user>]
./azacsnap_installer_v5.0.run [-v] -X [-d <directory>]
./azacsnap_installer_v5.0.run [-h]

Switches enclosed in [] are optional for each command line.
- h prints out this usage.
- v turns on verbose output.
- I starts the installation.
- u is the Linux user to install the scripts into, by default this is
'azacsnap'.
- X will only extract the snapshot commands.
- d is the target directory to extract into, by default this is
'./snapshot_cmds'.
Examples of a target directory are ./tmp or /usr/local/bin
```

Note The self-installer has an option to extract (-X) the snapshot tools from the bundle without performing any user creation and setup. This allows an experienced administrator to complete the setup steps manually, or to copy the commands to upgrade an existing installation.

Easy installation of snapshot tools (default)

The installer has been designed to quickly install the snapshot tools for SAP HANA on Azure. The manual steps are within each step.

By default, if the installer is run with only the -I option, it will do the following:

1. Create Snapshot user 'azacsnap', home directory, and set group membership.
2. Configure the azacsnap user's login `~/.profile`.
3. Search filesystem for directories to add to azacsnap's `$PATH`, these are typically the paths to the SAP HANA tools, such as `hdbsql` and `hdbuserstore`.
4. Search filesystem for directories to add to azacsnap's `$LD_LIBRARY_PATH`. Many commands require a library path to be set in order to execute correctly, this configures it for the installed user.
5. Copy the SSH keys for back-end storage for azacsnap from the "root" user (the user running the install). This assumes the "root" user has already configured connectivity to the storage
   - see section "Enable communication with storage".

6. Copy the SAP HANA connection secure user store for the target user, azacsnap. This assumes the "root" user has already configured the secure user store – see section "Enable communication with SAP HANA".
7. The snapshot tools are extracted into `/home/azacsnap/bin/`.
8. The commands in `/home/azacsnap/bin/` have their permissions set (ownership and executable bit, etc).

The following example shows the correct output of the installer when run with the default installation option.

```
> ./azacsnap_installer_v5.0.run -I
+------------------------------------------------------------+
| Azure Application Consistent Snapshot Tool Installer       |
+------------------------------------------------------------+
|-> Installer version '5.0'
|-> Create Snapshot user 'azacsnap', home directory, and set group membership.
|-> Configure azacsnap .profile
|-> Search filesystem for directories to add to azacsnap's $PATH
|-> Search filesystem for directories to add to azacsnap's $LD_LIBRARY_PATH
|-> Copying SSH keys for back-end storage for azacsnap.
|-> Copying HANA connection keystore for azacsnap.
|-> Extracting commands into /home/azacsnap/bin/.
|-> Making commands in /home/azacsnap/bin/ executable.
|-> Creating symlink for hdbsql command in /home/azacsnap/bin/.
+------------------------------------------------------------+
| Install complete! Follow the steps below to configure.     |
+------------------------------------------------------------+
+------------------------------------------------------------+
|  Install complete!  Follow the steps below to configure.   |
+------------------------------------------------------------+

1. Change into the snapshot user account.....
     su - azacsnap
2. Setup the HANA Secure User Store..... (command format below)
     hdbuserstore Set <ADMIN_USER> <HOSTNAME>:<PORT> <admin_user> <password>
3. Change to location of commands.....
     cd /home/azacsnap/bin/
4. Configure the customer details file.....
     azacsnap -c configure --configuration new
5. Test the connection to storage.....
     azacsnap -c test --test storage
6. Test the connection to HANA.....
   a. without SSL
     azacsnap -c test --test hana
   b. with SSL,  you will need to choose the correct SSL option
     azacsnap -c test --test hana --ssl=<commoncrypto|openssl>
7. Run your first snapshot backup..... (example below)
     azacsnap -c backup --volume=data --prefix=hana_test --frequency=15min --
retention=1
```

Uninstallation of the snapshot tools

If the snapshot tools have been installed using the default settings, uninstallation only requires removing the user the commands have been installed for (default = azacsnap).

```
> userdel -f -r azacsnap
```

Manual installation of the snapshot tools

In some cases, it is necessary to install the tools manually, but the recommendation is to use the installer's default option to ease this process.

Each line starting with a `#` character demonstrates the example commands following the character are run by the root user. The `\` at the end of a line is the standard line-continuation character for a shell command.

As the root superuser, a manual installation can be achieved as follows:

1. Get the "sapsys" group id, in this case the group id = 1010

   ```
   # grep sapsys /etc/group
   sapsys:x:1010:
   ```

2. Create Snapshot user 'azacsnap', home directory, and set group membership using the group id from step 1.

   ```
   # useradd -m -g 1010 -c "Azure SAP HANA Snapshots User" azacsnap
   ```

   Optionally add the user to the wheel group if needing to do privilege escalation as the user (e.g. `su - <sid>adm`). This step is not automatically done by the installer.

   ```
   # usermod -a -G wheel azacsnap
   ```

3. Make sure the user azacsnap's login `.profile` exists.

   ```
   # echo "" >> /home/azacsnap/.profile
   ```

4. Search filesystem for directories to add to azacsnap's $PATH, these are typically the paths to the SAP HANA tools, such as hdbsql and hdbuserstore.

   ```
   # HDBSQL_PATH=`find -L /hana/shared/[A-z0-9][A-z0-9][A-z0-9]/HDB*/exe
   /usr/sap/hdbclient \
       -name hdbsql -exec dirname {} + 2> /dev/null | sort | uniq | tr '\n' ':'`
   ```

5. Add the updated PATH to the user's profile

   ```
   # echo "export PATH=\"\$PATH:$HDBSQL_PATH\"" >> /home/azacsnap/.profile
   ```

6. Search filesystem for directories to add to azacsnap's $LD_LIBRARY_PATH.

   ```
   # NEW_LIB_PATH=`find -L /hana/shared/[A-z0-9][A-z0-9][A-z0-9]/HDB*/exe
   /usr/sap/hdbclient \
       -name "*.so" -exec dirname {} + 2> /dev/null | sort | uniq | tr '\n' ':'`
   ```

7. Add the updated library path to the user's profile

```
> echo "export LD_LIBRARY_PATH=\"\$LD_LIBRARY_PATH:$NEW_LIB_PATH\"" >>
/home/azacsnap/.profile
```

8. On Azure Large Instances
    1. Copy the SSH keys for back-end storage for azacsnap from the "root" user (the user running the install). This assumes the "root" user has already configured connectivity to the storage > see section "Enable communication with storage".

        ```
        > cp -pr ~/.ssh /home/azacsnap/.
        ```

    2. Set the user permissions correctly for the SSH files

        ```
        > chown -R azacsnap.sapsys /home/azacsnap/.ssh
        ```

9. On Azure NetApp Files
    1. Configure the user's `DOTNET_BUNDLE_EXTRACT_BASE_DIR` path per the .NET Core single-file extract guidance.
        1. SUSE Linux

            ```
            # echo "export DOTNET_BUNDLE_EXTRACT_BASE_DIR=~/.net" \
                >> /home/azacsnap/.profile
            # echo "[ -d $DOTNET_BUNDLE_EXTRACT_BASE_DIR] && \
                chmod 700 $DOTNET_BUNDLE_EXTRACT_BASE_DIR" \
                >> /home/azacsnap/.profile
            ```

        2. RHEL

            ```
            # echo "export DOTNET_BUNDLE_EXTRACT_BASE_DIR=~/.net" \
              >> /home/azacsnap/.bash_profile
            # echo "[ -d $DOTNET_BUNDLE_EXTRACT_BASE_DIR] && \
              chmod 700 $DOTNET_BUNDLE_EXTRACT_BASE_DIR"\
              >> /home/azacsnap/.bash_profile
            ```

10. Copy the SAP HANA connection secure user store for the target user, azacsnap. This assumes the "root" user has already configured the secure user store. > see section "Enable communication with SAP HANA".

    ```
    # cp -pr ~/.hdb /home/azacsnap/.
    ```

11. Set the user permissions correctly for the hdbuserstore files

    ```
    # chown -R azacsnap.sapsys /home/azacsnap/.hdb
    ```

12. Extract the snapshot tools into /home/azacsnap/bin/.

    ```
    # ./azacsnap_installer_v5.0.run -X -d /home/azacsnap/bin
    ```

13. Make the snapshot commands executable

    ```
    # chmod 700 /home/azacsnap/bin/*
    ```

14. Ensure the correct ownership permissions are set on the user's home directory

    ```
    # chown -R azacsnap.sapsys /home/azacsnap/*
    ```

Complete setup of snapshot tools

The installer provides steps to complete after the installation of the snapshot tools has been done. Follow these steps to complete the configuration of the snapshot tools and perform the first database consistent storage snapshot.

The following output shows the steps to complete after running the installer with the default installation options:

1. Change into the snapshot user account
    1. `su - azacsnap`
2. Setup the HANA Secure User Store
    1. `hdbuserstore Set <ADMIN_USER> <HOSTNAME>:<PORT> <admin_user> <password>`
3. Change to location of commands
    1. `cd /home/azacsnap/bin/`
4. Configure the customer details file
    1. `azacsnap -c configure –-configuration new`
5. Test the connection to storage…..
    1. `azacsnap -c test –-test storage`
6. Test the connection to HANA…..
    1. without SSL
        1. `azacsnap -c test –-test hana`
    2. with SSL, you will need to choose the correct SSL option
        1. `azacsnap -c test –-test hana --ssl=<commoncrypto|openssl>`
7. Run your first snapshot backup
    1. `azacsnap -c backup –-volume data--prefix=hana_test --retention=1`

Step 2 will be necessary if "Enable communication with SAP HANA" was not done before the installation.

CAUTION The test commands should execute correctly otherwise the snapshot commands may fail.

Configuration file for snapshot tools

From version 5.0 a configuration file can be created by running `azacsnap -c configure --configuration new`. The following example is for Azure Large Instance configuration:

```
$ azacsnap -c configure --configuration new
Building new config file
Add comment to config file (blank entry to exit adding comments):This is a new config
file for `azacsnap`
Add comment to config file (blank entry to exit adding comments):
Add database to config? (y/n) [n]: y
HANA SID (e.g. H80): H80
HANA Instance Number (e.g. 00): 00
HANA HDB User Store Key (e.g. `hdbuserstore List`): AZACSNAP
HANA Server's Address (hostname or IP address): testing01
Add ANF Storage to database section? (y/n) [n]:
```

```
Add HLI Storage to database section? (y/n) [n]: y
Add DATA Volume to HLI Storage section of Database section? (y/n) [n]: y
Storage User Name (e.g. clbackup25): clt1h80backup
Storage IP Address (e.g. 192.168.1.30): 172.18.18.11
Storage Volume Name (e.g. hana_data_h80_testing01_mnt00001_t250_vol):
hana_data_h80_testing01_mnt00001_t250_vol
Add DATA Volume to HLI Storage section of Database section? (y/n) [n]:
Add OTHER Volume to HLI Storage section of Database section? (y/n) [n]:
Add HLI Storage to database section? (y/n) [n]:
Add database to config? (y/n) [n]:
Editing configuration complete, writing output to 'azacsnap.json'
```

Azure Large Instance (HLI) storage values

When configuring Azure Large Instance storage the following values will be needed:

- **Storage User Name** = This is the user name used to establish the SSH connection to the Storage.
- **Storage IP Address** = The address of the Storage system.
- **Storage Volume Name** = the volume name to snapshot. These can be determined multiple ways, perhaps the simplest is to try the following shell command:

```
$  grep nfs /etc/fstab | cut -f2 -d"/" | sort | uniq
hana_data_p40_soldub41_mnt00001_t250_vol
hana_log_backups_p40_soldub41_t250_vol
hana_log_p40_soldub41_mnt00001_t250_vol
hana_shared_p40_soldub41_t250_vol
```

Azure NetApp Files (ANF) storage values

When configuring Azure NetApp Files storage the following values will be needed:

- **Service Principal Authentication filename** = this is the authfile.json file generated in the Cloud Shell whenc configuring communication with Azure NetApp Files storage.
- **Full ANF Storage Volume Resource ID** = the full Resource ID of the Volume being snapshot. This can be retrieved from the Azure Portal –> ANF –> Volume –> Settings/Properties –> Resource ID

Upgrading the snapshot tools for Azure Large Instance ONLY

This section is intended to provide a high-level workflow for customers to use as a base to aid them in upgrading from the prior generation snapshot tools being used on Azure Large Instance.

The snapshot tools continue to create compatible stored snapshots.

Gather Existing Installation details

All the tools and configuration files are installed and run from the same directory. Use the following commands to understand the current installation and record information for configuring the system after the upgrade.

To find where the existing installation is:

- check the crontab file (for the current schedule), e.g.:

```
# crontab -l
```

- search the filesystem for the configuration file in order to locate the commands, which are generally in the same location, e.g.:

```
# find / -name "HANABackupCustomerDetails.txt"
```

Use the content of `HANABackupCustomerDetails.txt` to complete the `azacsnap` configuration process.

From version 4.0

If the prior generation snapshot tools (version 4.0) have been installed, then follow either of the sub-sections based on whether the tools are being run as 'root' super-user or a standard user.

Currently running as standard non-root user (e.g. shoasnap)

The installer allows a user to upgrade an existing system by using the -X switch to extract the commands and then manually copy them into the target location.

In the following example, the installer has been copied into the `$HOME` directory for the user the commands are currently run as.

To perform an upgrade, the user should:

1. Backup the existing snapshot tools.
2. Extract the commands into a temporary directory using the -X and -d switches.
3. Copy the commands into the default location (e.g. `/home/azacsnap/bin/.`)
4. Reference the existing `HANABackupCustomerDetails.txt` file to create a new `azacsnap` configuration.
5. Test the tools and configuration by running the standard tests.

```
shoasnap@sapprdhdb80:~> mkdir -p ~/archive/snapshot_tools_backup

shoasnap@sapprdhdb80:~> cp ~/bin/* ~/archive/snapshot_tools_backup/.

shoasnap@sapprdhdb80:~> ./azacsnap_installer_v5.0.run -X -d tmp/
+------------------------------------------------------------+
|   Azure Application Consistent Snapshot Tool Installer     |
+------------------------------------------------------------+
|-> Installer version '5.0'
|-> Extracting commands into tmp/.

shoasnap@sapprdhdb80:~> cp tmp/* bin/.

shoasnap@sapprdhdb80:~> cat
~/archive/snapshot_tools_backup/HANABackupCustomerDetails.txt

shoasnap@sapprdhdb80:~> azacsnap -c configure --configuration new
```

```
shoasnap@sapprdhdb80:~> azacsnap -c test --test storage

shoasnap@sapprdhdb80:~> azacsnap -c test --test hana
```

Currently running as 'root' user

If the current tools are being run as the 'root' superuser account, it is recommended to follow the installation process as defined within this document to install as a non-root user.

If the system is already performing snapshots as the root user, then the pre-requisites (enabling communication with storage and SAP HANA) are assumed to be met.

Note If the tools are being run as root, it is possible they are installed in the `/hana/shared/<SID>/exe/linuxx86_64/hdb` directory as this was the original installation target directory.

From versions prior to and including v3.4.1

The versions prior to and including 3.4.1 of the snapshot tools did not have an installer, and the guidance was to install the snapshot tools into the same directory as other SAP HANA files and run them as the 'root' superuser.

Note For earlier SAP HANA on Azure Large Instance installations, the directory of pre-installed snapshot tools was `/hana/shared/<SID>/exe/linuxx86_64/hdb`.

If the administrator has followed the guidance in the section "Gather Existing Installation details", then the location of the tools and config file will already be recorded.

The general recommendation is to install the snapshot tools, using the "Microsoft Snapshot Tools for SAP HANA on Azure" guide allowing the tools to be run as a non-root user.

Before starting the installation, here are some useful tips:

- Use the existing configuration file (`HANABackupCustomerDetails.txt`), as reference.
- Use the existing crontab as an example for creating the schedule for the new user (if following the installation guide).
- Comment out the entries in the existing crontab before setting up the schedule for the newly installed user to avoid snapshot commands being run in parallel.

CAUTION > Ensure the old crontab file has been updated to comment out running the older commands. If upgrading from versions 3.4.1 or earlier, make sure all the snapshot tools commands in the crontab are updated to remove the `.pl` extension as from version 4.0 they are provided as binaries.

SAP HANA Configuration

There are some recommended changes to be applied to SAP HANA to ensure protection of the log backups and catalog. By default, the `basepath_logbackup` and `basepath_catalogbackup` will output their files to the `$(DIR_INSTANCE)/backup/log` directory, and it is unlikely this path is on a volume which `azacsnap` is configured to snapshot these files will not be protected with storage snapshots.

The following `hdbsql` command examples are intended to demonstrate setting the log and catalog paths to locations which are on storage volumes which can be snapshot by `azacsnap`. Be sure to check the values on the command line match the local SAP HANA configuration.

Configure log backup location

In this example, the change is being made to the `basepath_logbackup` parameter.

```
> hdbsql -jaxC -n <HANA_ip_address>:30013 -i 00 -u SYSTEM -p <SYSTEM_USER_PASSWORD>
"ALTER
SYSTEM ALTER CONFIGURATION ('global.ini', 'SYSTEM') SET ('persistence',
'basepath_logbackup') = '/hana/logbackups/H80' WITH RECONFIGURE"
```

Configure catalog backup location

In this example, the change is being made to the `basepath_catalogbackup` parameter. First, check to ensure the `basepath_catalogbackup` path exists on the filesystem, if not create the path with the same ownership as the directory.

```
> ls -ld /hana/logbackups/H80/catalog
drwxr-x--- 4 h80adm sapsys 4096 Jan 17 06:55 /hana/logbackups/H80/catalog
```

If the path needs to be created, the following example creates the path and sets the correct ownership and permissions. These commands will need to be run as root.

```
# mkdir /hana/logbackups/H80/catalog
# chown --reference=/hana/shared/H80/HDB00 /hana/logbackups/H80/catalog
# chmod --reference=/hana/shared/H80/HDB00 /hana/logbackups/H80/catalog
# ls -ld /hana/logbackups/H80/catalog
drwxr-x--- 4 h80adm sapsys 4096 Jan 17 06:55 /hana/logbackups/H80/catalog
```

The following example will change the SAP HANA setting.

```
> hdbsql -jaxC -n <HANA_ip_address>:30013 -i 00 -u SYSTEM -p <SYSTEM_USER_PASSWORD>
"ALTER
SYSTEM ALTER CONFIGURATION ('global.ini', 'SYSTEM') SET ('persistence',
'basepath_catalogbackup') = '/hana/logbackups/H80/catalog' WITH RECONFIGURE"
```

Check log and catalog backup locations

After making the changes above, confirm these are set correctly with the following command. In this example, the settings which have been set following the guidance above will display as SYSTEM settings.

This query also returns the DEFAULT settings for comparison.

```
> hdbsql -jaxC -n <HANA_ip_address> - i 00 -U AZACSNAP "select * from
sys.m_inifile_contents
where (key = 'basepath_databackup' or key ='basepath_datavolumes' or
key = 'basepath_logbackup' or key = 'basepath_logvolumes' or key =
'basepath_catalogbackup')"
global.ini,DEFAULT,,,persistence,basepath_catalogbackup,$(DIR_INSTANCE)/backup/log
global.ini,DEFAULT,,,persistence,basepath_databackup,$(DIR_INSTANCE)/backup/data
```

```
global.ini,DEFAULT,,,persistence,basepath_datavolumes,$(DIR_GLOBAL)/hdb/data
global.ini,DEFAULT,,,persistence,basepath_logbackup,$(DIR_INSTANCE)/backup/log
global.ini,DEFAULT,,,persistence,basepath_logvolumes,$(DIR_GLOBAL)/hdb/log
global.ini,SYSTEM,,,persistence,basepath_catalogbackup,/hana/logbackups/H80/catalog
global.ini,SYSTEM,,,persistence,basepath_datavolumes,/hana/data/H80
global.ini,SYSTEM,,,persistence,basepath_logbackup,/hana/logbackups/H80
global.ini,SYSTEM,,,persistence,basepath_logvolumes,/hana/log/H80
```

Configure log backup timeout

The default setting for SAP HANA to perform a log backup is 900 seconds (15 minutes). It's recommended to reduce this to 300 seconds (i.e. 5 minutes). Then it is possible to run very regular backups (e.g. every 10 minutes) by adding the log_backups volume into the OTHER volume section of the configuration file.

```
> hdbsql -jaxC -n <HANA_ip_address>:30013 -i 00 -u SYSTEM -p <SYSTEM_USER_PASSWORD>
"ALTER
SYSTEM ALTER CONFIGURATION ('global.ini', 'SYSTEM') SET ('persistence',
'log_backup_timeout_s') = '300' WITH RECONFIGURE"
```

Check log backup timeout

After making the change to the log backup timeout, check to ensure this has been set as follows. In this example, the settings which have just been set will display as the SYSTEM settings, but this query also returns the DEFAULT settings for comparison.

```
> hdbsql -jaxC -n <HANA_ip_address> - i 00 -U AZACSNAP "select * from
sys.m_inifile_contents
where key like '%log_backup_timeout%' "
global.ini,DEFAULT,,,persistence,log_backup_timeout_s,900
global.ini,SYSTEM,,,persistence,log_backup_timeout_s,300
```

Warning about diskspace

Customers are responsible for monitoring disk space usage. The `--trim` option can assist customers with management of disk space usage. Refer to the section on the `azacsnap -c backup` command for more details on this option.

CAUTION > SAP HANA can use up all the diskspace for the `/hana/logbackups/<SID>` filesystem which could halt the database. Prior to the `--trim` option there was no automated log management available with the snapshot tools and it was recommended a log deletion/trimming process be manually established by the customers.

Snapshot Tools Execution and Details

This section provides an overview of each of the files provided as part of the Microsoft Snapshot Tools for SAP HANA on Azure.

Logging of the commands is performed in the background, to provide a simpler user experience, but it is possible to still get the logging displayed in the console. This is done by using the `-v` (verbose) switch for the command being executed.

Config file - `azacsnap.json`

In the following screen, the `azacsnap.json` is configured with the one SIDs.

The parameter values must be set to the customer's specific SAP HANA environment. For **Azure Large Instance** system, this information is provided by Microsoft Service Management during the onboarding/handover call, and is made available in an Excel file which is provided during handover. Please open a service request if you need to be provided this information again.

** The following is an exmaple only, update all the values accordingly.

```
> cat azacsnap.json
{
  "version": "5.0 Preview",
  "logPath": "./logs",
  "securityPath": "./security",
  "comments": [],
  "database": [
    {
      "hana": {
        "serverAddress": "sapprdhdb80",
        "sid": "H80",
        "instanceNumber": "00",
        "hdbUserStoreName": "SCADMIN",
        "savePointAbortWaitSeconds": 600,
        "hliStorage": [
          {
            "dataVolume": [
              {
                "backupName": "clt1h80backup",
                "ipAddress": "172.18.18.11",
                "volume": "hana_data_h80_azsollabbl20a31_mnt00001_t210_vol"
              },
              {
                "backupName": "clt1h80backup",
                "ipAddress": "172.18.18.11",
                "volume": "hana_shared_h80_azsollabbl20a31_t210_vol"
              }
            ],
            "otherVolume": [
              {
                "backupName": "clt1h80backup",
                "ipAddress": "172.18.18.11",
```

```
                "volume": "hana_log_backups_h80_azsollabbl20a31_t210_vol"
              }
            ]
          }
        ],
        "anfStorage": []
      }
    }
  ]
}
```

Note For a DR scenario where backups are to be run at the DR site, then the HANA Server Name configured in the `HANABackupCustomerDetails.txt` file at the DR site should be the same as the production server name.

Customer is responsible to enable/disable the snapshot command on active/passive node accordingly.

- Single node: IP and Hostname of the node
- HSR with STONITH: IP and Hostname of the node
- Scale-out (N+N, N+M): Current master node IP and host name
- HSR without STONITH: IP and Hostname of the node
- Multi SID on Single node: Hostname and IP of the node hosting those SIDs

Note For Azure Large Instance your storage IP address must be in the same subnet as your server pool. For example, in this case, our server pool subnet is 172. 18. 18 .0/24 and our assigned storage IP is 172.18.18.11.

Check connectivity with SAP HANA - `azacsnap -c test --test hana`

This snapshot command checks the HANA connectivity for all the HANA instances in the configuration file. It uses the HDBuserstore to connect to the SYSTEMDB and fetches the SID information.

For SSL, this command takes the either of the following arguments:

- `--ssl=` forces an encrypted connection with the database and defines the encryption method used to communicate with SAP HANA, either `openssl` or `commoncrypto`. If defined, then this command expects to find two files in the same directory, these files must be named after the corresponding SID. Refer to Using SSL for communication with SAP HANA.

Output of the `azacsnap -c test --test hana` command

```
> azacsnap -c test --test hana
BEGIN : Test process started for 'hana'
BEGIN : SAP HANA tests
PASSED: Successful connectivity to HANA version 2.00.032.00.1533114046
END   : Test process complete for 'hana'
```

Check connectivity with storage - `azacsnap -c test --test storage`

The `azacsnap` command will take a snapshot for all the data volumes configured to verify that it has the correct access to the volumes for each SAP HANA instance. A temporary

snapshot is created and then removed for each data volume to verify snapshot access for each file system.

Output of the `azacsnap -c test --test storage` command

Note For Azure Large Instance, `azacsnap -c test --test storage` command extrapolates the storage generation and HLI SKU. Based on this information it then provides guidance on configuring 'boot' snapshots (see the line starting with `Action:` output).

```
SID1   : Generation 4
Storage: ams07-a700s-saphan-1-01v250-client25-nprod
HLI SKU: S96
Action : Please configure the 'boot' snapshots on ALL the servers.
```

Perform snapshot backup - `azazsnap -c backup`

This command performs the orchestration of a database consistent storage snapshot on the DATA volumes, and a storage snapshot (without any database consistency setup) on the OTHER volumes.

This command takes the following arguments:

- `--volume=` type of volume to snapshot, this parameter may contain `data` or `other`
  - `data` snapshots the volumes within the dataVolume stanza of the configuration file.
  - `other` snapshots the volumes within the otherVolume stanza of the configuration file.
    - By creating a separate config file with the boot volume as the otherVolume, it's possible for `boot` snapshots to be taken on an entirely different schedule (e.g. daily).
- `--prefix=` the customer snapshot prefix for the snapshot name. This parameter has two purposes. The first purpose is to provide a unique name for grouping of snapshots. The second purpose is for the snapshot command to determine the `--retention` number of storage snapshots that are kept for the specified `--prefix`.
  - **Only alpha numeric ("A-Z,a-z,0-9"), underscore ("_") and dash ("-") characters are allowed.**
- `--retention` the number of snapshots of the defined `--prefix` to be kept. Any additional snapshots are removed after a new snapshot is taken for this `--prefix`.
- `--trim` available for SAP HANA v2 and later, this option maintains the backup catalog and on disk catalog and log backups. The number of entries to keep in the backup catalog is determined by the `--retention` option above, and deletes older entries for the defined prefix (`--prefix`) from the backup catalog, and the related physical logs backup. It also deletes any logbackup entries which are older than the oldest non-logbackup entry. This helps to prevent the log backups from using up all available disk space.

  Note The following example command will keep 9 storage snapshots and ensure the backup catalog is continuously trimmed to match the 9 storage snapshots being retained.

  ```
  > azacsnap -c backup --volume data --prefix hana_TEST --retention 9 --trim
  ```

- `[--ssl=]` an optional parameter which defines the encryption method used to communicate with SAP HANA, either `openssl` or `commoncrypto`. If defined, then the azure_hana_backup command expects to find two files in the same directory, these files must be named after the corresponding SID. Refer to Using SSL for communication with SAP HANA.
- `[--frequency=]` Deprecated, do not use.

The following example takes a `hana` type snapshot with a prefix of `hana_TEST` and will keep `9` of them communicating with SAP HANA using SSL (`openssl`).

```
> azacsnap -c backup --volume data --prefix hana_TEST --retention 9 --trim --
ssl=openssl
```

Snapshot backups are very fast

The duration of a snapshot backup is independent of the volume size, with a 10TB volume being snapped within the same approximate time as a 10GB volume.

The primary factors impacting overall execution time are the number of volumes to be snapshot and any changes in the `--retention` parameter (where a reduction can increase the execution time as excess snapshots are removed).

In the example configuration above, which is for **Azure Large Instance**, snapshots for the two volumes took <5 seconds to complete. For **Azure NetApp Files**, snapshots for the two volumes would take about 60 seconds.

Note if the `--retention` is significantly reduced from the previous time `azacsnap` is run (e.g. from `--retention 50` to `--retention 5`), then the time taken will increase as `azacsnap` needs to remove the extra snapshots.

Example with `data` parameter

```
> azacsnap -c backup --volume data --prefix hana_TEST --retention 9 --trim
```

The command does not output to the console, but does write to a log file, a result file, and `/var/log/messages`.

The *log file* is made up of the command name + the -c option + the config filename. By default a log filename for a `-c backup` run with a default config filename `azacsnap-backup-azacsnap.log`.

The *result* file has the same base name as the log file, with `.result` as its suffix, for example `azacsnap-backup-azacsnap.result` which contains the following:

```
> cat logs/azacsnap-backup-azacsnap.result
Database # 1 (H80) : completed ok
```

The `/var/log/messages` file contains the same output as the `.result` file. See the following example (run as root):

```
# grep "azacsnap.*Database" /var/log/messages | tail -n10
Jul  2 05:22:07 server01 azacsnap[183868]: Database # 1 (H80) : completed ok
Jul  2 05:27:06 server01 azacsnap[4069]: Database # 1 (H80) : completed ok
```

```
Jul  2 05:32:07 server01 azacsnap[19769]: Database # 1 (H80) : completed ok
Jul  2 05:37:06 server01 azacsnap[35312]: Database # 1 (H80) : completed ok
Jul  2 05:42:06 server01 azacsnap[50877]: Database # 1 (H80) : completed ok
Jul  2 05:47:06 server01 azacsnap[66429]: Database # 1 (H80) : completed ok
Jul  2 05:52:06 server01 azacsnap[82964]: Database # 1 (H80) : completed ok
Jul  2 05:57:06 server01 azacsnap[98522]: Database # 1 (H80) : completed ok
Jul  2 05:59:13 server01 azacsnap[105519]: Database # 1 (H80) : completed ok
Jul  2 06:02:06 server01 azacsnap[114280]: Database # 1 (H80) : completed ok
```

Example with `other` parameter

```
> azacsnap -c backup --volume other --prefix logs_TEST --retention 9
```

The command does not output to the console, but does write to a log file only. It does *not* write to a result file or `/var/log/messages`.

The *log file* is made up of the command name + the -c option + the config filename. By default a log filename for a `-c backup` run with a default config filename `azacsnap-backup-azacsnap.log`.

Example with `other` parameter (to backup host OS)

Note The use of another configuration file (`--configfile bootVol.json`) which contains only the boot volumes.

```
> azacsnap -c backup --volume other --prefix boot_TEST --retention 9 --configfile
bootVol.json
```

The command does not output to the console, but does write to a log file only. It does *not* write to a result file or `/var/log/messages`.

The *log file* name in this example is `azacsnap-backup-bootVol.log`.

Note The log file name is made up of the (command name)-(the `-c` option)-(the config filename).

- HANA Large Instance Type: There are two valid values with `TYPEI` or `TYPEII` dependent on the HANA Large Instance Unit.
- Refer to the online documentation to confirm the available SKUs
  - https://docs.microsoft.com/en-us/azure/virtual-machines/workloads/sap/hana-available-skus

List snapshots - `azacsnap -c details`

This command lists the details of the snapshots (volume name, snapshot name, creation time, comments, snapshot size) stored on all the volumes in the configuration file. The console output can be pasted into a spreadsheet for further analysis.

Due to information stored at the back-end systems, the ouput of the command is slightly different when run on **Azure Large Instance** versus **Azure NetApp Files**.

Output of the `azacsnap -c details` command

The example below has been executed on an **Azure Large Instance** and the output has been trimmed for brevity. Also be aware some of the lines may have been wrapped to fit the output.

```
> azacsnap -c details
List snapshot details called with snapshotFilter ''
#, Volume, Snapshot, Create Time, HANA Backup ID, Snapshot Size
#1, hana_data_h31_azsollabbl20a31_mnt00001_t210_vol, hana_hourly.2020-06-
17T113043.1586971Z, "Wed Jun 17 11:31:14 2020", "HANA Backup ID: 1592393444174,
702.6MB
#2, hana_data_h31_azsollabbl20a31_mnt00001_t210_vol, shoasnap-hsr-ha.2020-07-
02_2200.5, "Thu Jul 02 22:01:37 2020", "Backup ID: 1593727205201, 342.3MB
#3, hana_data_h31_azsollabbl20a31_mnt00001_t210_vol, azacsnap-hsr-ha.2020-07-
02T220201.5332158Z, "Thu Jul 02 22:03:34 2020", "HANA Backup ID:
1593727322533|azacsnap version: 5.0 Preview (20200617.75879)", 3.27MB
#4, hana_data_h31_azsollabbl20a31_mnt00001_t210_vol, shoasnap-hsr-ha.2020-07-
02_2205.4, "Thu Jul 02 22:06:36 2020", "Backup ID: 1593727504776, 3.14MB
#5, hana_data_h31_azsollabbl20a31_mnt00001_t210_vol, azacsnap-hsr-ha.2020-07-
02T220702.3283669Z, "Thu Jul 02 22:08:37 2020", "HANA Backup ID:
1593727623339|azacsnap version: 5.0 Preview (20200617.75879)", 3.50MB
#6, hana_data_h31_azsollabbl20a31_mnt00001_t210_vol, shoasnap-hsr-ha.2020-07-
02_2210.3, "Thu Jul 02 22:11:37 2020", "Backup ID: 1593727805216, 2.85MB
#7, hana_data_h31_azsollabbl20a31_mnt00001_t210_vol, azacsnap-hsr-ha.2020-07-
02T221201.7013700Z, "Thu Jul 02 22:13:36 2020", "HANA Backup ID:
1593727922724|azacsnap version: 5.0 Preview (20200617.75879)", 3.34MB
#8, hana_data_h31_azsollabbl20a31_mnt00001_t210_vol, shoasnap-hsr-ha.2020-07-
02_2215.2, "Thu Jul 02 22:16:36 2020", "Backup ID: 1593728104772, 2.73MB
#9, hana_data_h31_azsollabbl20a31_mnt00001_t210_vol, azacsnap-hsr-ha.2020-07-
02T221702.2535255Z, "Thu Jul 02 22:18:35 2020", "HANA Backup ID:
1593728223274|azacsnap version: 5.0 Preview (20200617.75879)", 3.39MB
#10, hana_data_h31_azsollabbl20a31_mnt00001_t210_vol, shoasnap-hsr-ha.2020-07-
02_2220.1, "Thu Jul 02 22:21:37 2020", "Backup ID: 1593728405346, 3.29MB
#11, hana_data_h31_azsollabbl20a31_mnt00001_t210_vol, azacsnap-hsr-ha.2020-07-
02T222201.4988618Z, "Thu Jul 02 22:23:36 2020", "HANA Backup ID:
1593728522505|azacsnap version: 5.0 Preview (20200617.75879)", 3.68MB
#12, hana_data_h31_azsollabbl20a31_mnt00001_t210_vol, shoasnap-hsr-ha.2020-07-
02_2225.0, "Thu Jul 02 22:26:37 2020", "Backup ID: 1593728705321, 2.80MB
#13, hana_data_h31_azsollabbl20a31_mnt00001_t210_vol, azacsnap-hsr-ha.2020-07-
02T222702.0521995Z, "Thu Jul 02 22:28:37 2020", "HANA Backup ID:
1593728823058|azacsnap version: 5.0 Preview (20200617.75879)", 1.04MB
, hana_data_h31_azsollabbl20a31_mnt00001_t210_vol, , , Size used by Snapshots, 1.05GB
#1, hana_shared_h31_azsollabbl20a31_t210_vol, hana_hourly.2020-06-17T113043.1586971Z,
"Wed Jun 17 11:31:12 2020", "HANA Backup ID: 1592393444174, 2.55GB
#2, hana_shared_h31_azsollabbl20a31_t210_vol, shoasnap-hsr-ha.2020-07-02_2200.5, "Thu
Jul 02 22:01:37 2020", "Backup ID: 1593727205201, 4.30MB
#3, hana_shared_h31_azsollabbl20a31_t210_vol, azacsnap-hsr-ha.2020-07-
02T220201.5332158Z, "Thu Jul 02 22:03:36 2020", "HANA Backup ID:
1593727322533|azacsnap version: 5.0 Preview (20200617.75879)", 8.04MB
#4, hana_shared_h31_azsollabbl20a31_t210_vol, shoasnap-hsr-ha.2020-07-02_2205.4, "Thu
Jul 02 22:06:37 2020", "Backup ID: 1593727504776, 8.12MB
#5, hana_shared_h31_azsollabbl20a31_t210_vol, azacsnap-hsr-ha.2020-07-
02T220702.3283669Z, "Thu Jul 02 22:08:35 2020", "HANA Backup ID:
```

```
1593727623339|azacsnap version: 5.0 Preview (20200617.75879)", 4.28MB
#6, hana_shared_h31_azsollabbl20a31_t210_vol, shoasnap-hsr-ha.2020-07-02_2210.3, "Thu
Jul 02 22:11:37 2020", "Backup ID: 1593727805216, 4.33MB
#7, hana_shared_h31_azsollabbl20a31_t210_vol, azacsnap-hsr-ha.2020-07-
02T221201.7013700Z, "Thu Jul 02 22:13:34 2020", "HANA Backup ID:
1593727922724|azacsnap version: 5.0 Preview (20200617.75879)", 4.31MB
#8, hana_shared_h31_azsollabbl20a31_t210_vol, shoasnap-hsr-ha.2020-07-02_2215.2, "Thu
Jul 02 22:16:37 2020", "Backup ID: 1593728104772, 4.30MB
#9, hana_shared_h31_azsollabbl20a31_t210_vol, azacsnap-hsr-ha.2020-07-
02T221702.2535255Z, "Thu Jul 02 22:18:37 2020", "HANA Backup ID:
1593728223274|azacsnap version: 5.0 Preview (20200617.75879)", 7.84MB
#10, hana_shared_h31_azsollabbl20a31_t210_vol, shoasnap-hsr-ha.2020-07-02_2220.1,
"Thu Jul 02 22:21:37 2020", "Backup ID: 1593728405346, 8.09MB
#11, hana_shared_h31_azsollabbl20a31_t210_vol, azacsnap-hsr-ha.2020-07-
02T222201.4988618Z, "Thu Jul 02 22:23:34 2020", "HANA Backup ID:
1593728522505|azacsnap version: 5.0 Preview (20200617.75879)", 4.34MB
#12, hana_shared_h31_azsollabbl20a31_t210_vol, shoasnap-hsr-ha.2020-07-02_2225.0,
"Thu Jul 02 22:26:37 2020", "Backup ID: 1593728705321, 4.31MB
#13, hana_shared_h31_azsollabbl20a31_t210_vol, azacsnap-hsr-ha.2020-07-
02T222702.0521995Z, "Thu Jul 02 22:28:35 2020", "HANA Backup ID:
1593728823058|azacsnap version: 5.0 Preview (20200617.75879)", 4.31MB
, hana_shared_h31_azsollabbl20a31_t210_vol, , , Size used by Snapshots, 2.62GB
#1, hana_log_backups_h31_azsollabbl20a31_t210_vol, azacsnap_vesangam_other_test.2020-
06-17T113215.4462696Z, "Wed Jun 17 11:32:43 2020", "HANA Log-Backups, 156KB
#2, hana_log_backups_h31_azsollabbl20a31_t210_vol,
azacsnap_vesangam_other_test2.2020-06-17T114205.1041364Z, "Wed Jun 17 11:42:35 2020",
"HANA Log-Backups, 1.34MB
, hana_log_backups_h31_azsollabbl20a31_t210_vol, , , Size used by Snapshots, 1.49MB
```

Note This example shows output for snapshots run using the previous version (v4.3) as well as snapshots taken with the latest version (5.0).

Delete a snapshot - `azacsnap -c delete`

This command deletes snapshots from the volumes.

Snapshots created less than 10 minutes prior to running this command should not be deleted due to the potential for interference with snapshot replication.

This command takes the following arguments:

- `--delete=` type of delete operation ('sync', 'storage', 'hana').
  - `sync` type deletion tries to link the SAP HANA backup ID to the volume snapshot name in order to remove both the SAP HANA backup ID from the catalog and the snapshot from the storage volume.
  - `storage` type deletion removes the snapshot from each of the volumes in the configuration file. There is no interaction with SAP HANA.
  - `hana` type deletion deletes the SAP HANA backup ID entry from the backup catalog.
- `--snapshot=` (optional) The snapshot name to be permanently removed.
- `--hanasid=` (optional) SAP HANA database SID containing the backup ID to be deleted.

- `--hanabackupid=` (optional) SAP HANA backup ID to be deleted from SAP HANA backup catalog.
- `--force` (optional) **use with caution** this will force deletion without prompting for confirmation.

Delete a snapshot using `sync` option`

```
azacsnap -c delete --delete sync --hanasid H80 --hanabackupid 157979797979
```

**Note* Checks for any entries in the backup catalog for the SAP HANA backup ID 157979797979, gets the storage snapshot name and deletes both the entry in the backup catalog and the snapshot from all of the volumes containing the named snapshot.

```
azacsnap -c delete --delete sync --snapshot hana_hourly.2020-01-22_2358
```

**Note* Checks for any entries in the backup catalog for the snapshot named hana_hourly.2020-01-22_2358, gets the SAP HANA backup ID and deletes both the entry in the backup catalog and the snapshot from any of the volumes containing the named snapshot.

Delete a snapshot using `hana` option`

```
azacsnap -c delete --delete hana --hanasid H80 --hanabackupid 157979797979
```

**Note* Deletes the SAP HANA backup ID 157979797979 from the backup catalog for SID H80.

Delete a snapshot using `storage` option`

```
azacsnap -c delete --delete storage --snapshot hana_hourly.2020-01-22_2358
```

**Note* Deletes the snapshot from any volumes containing snapshot named hana_hourly.2020-01-22_2358.

Output using the `--delete storage` option

Note the user is asked to confirm the deletion.

```
> azacsnap -c delete --delete storage --snapshot azacsnap-hsr-ha.2020-07-
02T221702.2535255Z
Processing delete request for snapshot 'azacsnap-hsr-ha.2020-07-02T221702.2535255Z'.
Are you sure you want to permanently delete the snapshot 'azacsnap-hsr-ha.2020-07-
02T221702.2535255Z' from all storage volumes.? (y/n) [n]: y
Snapshot deletion completed
```

It is possible to avoid user confirmation, by using the optional `--force` parameter as follows:

```
> azacsnap -c delete --delete storage --snapshot azacsnap-hsr-ha.2020-07-
02T222201.4988618Z --force
Processing delete request for snapshot 'azacsnap-hsr-ha.2020-07-02T222201.4988618Z'.
Snapshot deletion completed
```

Get DR replication status - azure_hana_replication_status

This command checks the storage replication status from the primary site to DR location and *must* be executed on the **DR site server**. This snapshot command does not require any input to be provided, rather it reads the `HANABackupCustomerDetails.txt file`.

```
> ./azure_hana_replication_status
```

Output of the `azure_hana_replication_status` command

The following example has broken replication status and, in this scenario, activating DR would likely result in incomplete data at the DR site.

CAUTION Note the warning in the following example about replication broken off for two of the volumes.

This example has successful replication between the Primary site and the DR site, so these systems are ready to support a DR scenario.

Perform a test DR failover - azure_hana_test_dr_failover

This snapshot command is like the "full" DR Failover snapshot command, but rather than breaking the replication between the primary site and the disaster recovery site, a clone volume is created out of the disaster recovery volumes which allows the restoration of the most recent snapshot in the DR site. Those cloned volumes are then usable by the customer to test Disaster Recovery without having to execute a complete failover of their HANA environment which breaks the replication agreement between the primary site and the disaster recovery site. When the test snapshot command is executed it requires the SID and a contact email for operations to manage the deletion of the clones after 4 weeks.

Each execution of the Test DR command creates a new clone that must be deleted by Microsoft Operations when the test is concluded. Multiple different restore points can be tested in this way, each with their own restoration point. The clone is designated by the time-stamp at when the snapshot command was executed and represents the most recent data and logbackups snapshot available when run.

CAUTION > Clone volumes created will be automatically deleted after 4 weeks.

Output of the `azure_hana_test_dr_failover` command (for Single-Node scenario)

CAUTION The "Displaying Mount Points by Volume" output is different for the various scenarios.

Perform full DR failover - azure_hana_dr_failover

This snapshot command **stops** storage replication from the primary site to the secondary site, restores the latest snapshot on the DR volumes, and provides the mountpoints for the DR volumes.

This snapshot command MUST be executed on the DR server ONLY!

You perform a failover to DR site, by executing a snapshot command azure_hana_dr_failover. This snapshot command requires a SID to be added as a parameter. This is the SID of the HANA instance, which needs to be recovered at the DR site.

CAUTION Only run this command you are planning to perform the DR exercise or a test. This command breaks the replication. You must reach out to the Microsoft Operations to set up the replication back. Also, once the replication is re-setup, all the data at DR storage for this SID get initialized.

At the high level, here are the steps for executing a DR failover:

- You must shut down the HANA instance at **primary** site. This is only needed if you are truly doing the failover to DR site so you don't have data inconsistencies.
- Shutdown the HANA instance on the DR node for the production SID.
- Execute the snapshot command azure_hana_dr_failover on the DR node with the SID to be recovered
  - The snapshot command breaks the storage replication link from the Primary to the DR site
  - The snapshot command restores the `/hana/data` and `/hana/logbackups` volume only, `/hana/shared` volume is NOT recovered, but rather it uses the existing `/hana/shared` for SID at the DR location.
  - Mount the `/hana/data` and `/hana/logbackups` volumes – ensure they're added to the `/etc/fstab` file
- Restore the HANA SYSTEMDB snapshot. Please note, HANA studio only shows you the latest HANA snapshot available under the storage snapshot restored as part of the snapshot command `azure_hana_dr_failover` execution.
- Recover the tenant database
- Start the HANA instance on the DR site for the Production SID (Example: H80 in this case)
- Perform the testing

Run `azacsnap -c backup` at the DR site

For a DR scenario where backups are to be run at the DR site, then the HANA Server Name configured in the `azacsnap` configuration file at the DR site should be the same as the production server name.

CAUTION Running the `azacsnap -c backup` can create storage snapshots at the DR site, these are not automatically replicated to another site. Work with Microsoft Operations to better understand returning any files or data back to the original production site.

Here are the detailed steps for the failover.

Step1: Get the volume details of the DR node by executing the command "df –h". This is so you can reference after the failover

```
# df -h
Filesystem Size Used Avail Use% Mounted on
devtmpfs 378G 8.0K 378G 1% /dev
tmpfs 569G 0 569G 0%
/dev/shm
tmpfs 378G 18M 378G 1% /run
tmpfs 378G 0 378G 0%
/sys/fs/cgroup
/dev/mapper/3600a09803830444562 2b4b584c575a66-part2 47G 20G 28G 42% /
/dev/mapper/3600a09803830444562 2b4b584c575a66-part1 979M 57M 856M 7% /boot
172.18.20.241:/hana_log_h80_mnt00003_t020_vol 512G 2.1G 510G 1%
/hana/log/H80/mnt00003
172.18.20.241:/hana_log_h80_mnt00001_t020_vol 512G 5.5G 507G 2%
/hana/log/H80/mnt00001
```

```
172.18.20.241:/hana_data_h80_mnt00003_t020_vol 1.2T 332M 1.2T 1%
/hana/data/H80/mnt00003
172.18.20.241:/hana_log_h80_mnt00002_t020_vol 512G 2.1G 510G 1%
/hana/log/H80/mnt00002
172.18.20.241:/hana_data_h80_mnt00002_t020_vol 1.2T 300M 1.2T 1%
/hana/data/H80/mnt00002
172.18.20.241:/hana_data_h80_mnt00001_t020_vol 1.2T 6.4G 1.2T 1%
/hana/data/H80/mnt00001
172.18.20.241:/hana_shared_h80_t020_vol/usr_sap_node1 2.7T 11G 2.7T 1% /usr/sap/H80
tmpfs 76G 0 76G 0% /run/user/0
172.18.20.241:/hana_shared_h80_t020_vol 2.7T 11G 2.7T 1% /hana/shared
172.18.20.241:/hana_data_h80_mnt00001_t020_xdp 1.2T 6.4G 1.2T 1%
/hana/data/H80/mnt00001
172.18.20.241:/hana_data_h80_mnt00002_t020_xdp 1.2T 300M 1.2T 1%
/hana/data/H80/mnt00002
172.18.20.241:/hana_data_h80_mnt00003_t020_xdp 1.2T 332M 1.2T 1%
/hana/data/H80/mnt00003
172.18.20.241:/hana_log_backups_h80_t020_xdp 512G 15G 498G 3%
/hana/logbackups/H80_T250
```

Step2: Shut down the HANA instance on the primary site (The instance which is getting failed over)

Step3: Shut down the HANA instances on the DR site (The instance of the primary SID being failed over)

Step4: Execute the snapshot command

```
> ./azure_hana_dr_failover
```

Output of the DR failover command.

```
> ./azure_hana_dr_failover
Running per the following command line:
./azure_hana_dr_failover
********** Introduction ****************************************************
This script is designed for those customers who have previously installed the
Production HANA instance in the Disaster Recovery Location either as a
stand-alone instance or as part of a multi-purpose environment. This script
should only be run in the event of a declared disaster by Microsoft or as part
of required Disaster Recovery testing plans. A failback coordinated with
Microsoft Operations is required after this script has been executed.

WARNING: the failback process will not necessarily be a quick process and will
require multiple steps in coordination with Microsoft Operations so this script
should not be undertaken lightly. This script will restore only the most recent
snapshot for both the Data and Log Backups filesystems. Any other restore
points must be handled by Microsoft Operations. Please enter the HANA <SID>
you wish to restore. This script must be executed from the Disaster Recovery
location otherwise unintended consequences may occur.
Please enter (yes/no): yes
Please enter (yes/no): yes
Please enter either the HANA SID you wish to restore: h80
Please enter either the HANA SID you wish to restore: : h80
**************** Displaying Mount Points by Volume ****************************
```

```
10.230.251.43:/hana_data_h80_sapprdhdb80_mnt00001_t250_xdp /hana/data/H80/mnt00001
nfs rw,bg,hard,timeo=600,vers=4,rsize=1048576,wsize=1048576,intr,noatime,lock 0 0
10.230.251.43:/hana_log_backups_h80_sapprdhdb80_t250_xdp
/hana/logbackups/H80_SAPPRDHDB80 nfs
rw,bg,hard,timeo=600,vers=4,rsize=1048576,wsize=1048576,intr,noatime,lock 0 0
10.230.251.43:/hana_log_backups_h80_soldub42_t250_xdp /hana/logbackups/H80_SOLDUB42
nfs rw,bg,hard,timeo=600,vers=4,rsize=1048576,wsize=1048576,intr,noatime,lock 0 0
********************************************************************************
****************** HANA DR Recovery Steps **********************************
* Please complete the following steps to recover your HANA database:          *
* 1. Ensure ALL the target mount points exist to mount the snapshot clones.   *
*    e.g. mkdir /hana/logbackups/H99_SOURCE                                   *
* 2. Add Mount Point Details from 'Displaying Mount Points by Volume' as      *
*    output above into /etc/fstab of DR Server.                               *
* 3. Mount newly added filesystems.                                           *
* 4. Perform HANA Snapshot Recovery using HANA Studio.                        *
********************************************************************************
Command completed successfully.
Exiting with return code: 0
Log file created at ./snapshotLogs/FullDR.h80.20190409_0901.txt
Note The steps at the end of the console display need to be actioned to complete the
storage
preparation for a DR failover
```

Step5: Execute the command `umount` to unmount the necessary mountpoints.

```
# umount <Mount point>
```

Unmount the data and logbackup mountpoints. Please note, you may have multiple data mountpoint in the scale-out scenario.

Step6: Modify the file `/etc/fstab` to comment out the data and logbackups entries for the primary SID (In this example, SID=H80) and add the new mount point entries created from the Primary site DR volumes. The new mount point entries are provided in the snapshot command output.

- Comment out the existing mount points running on the DR site with the `#` character:

```
#172.18.20.241:/hana_data_h80_mnt00001_t020_vol /hana/data/H80/mnt00001 nfs
rw,hard,timeo=600,vers=4,rsize=1048576,wsize=1048576,intr,noatime,lock 0 0
#172.18.20.241:/hana_log_backups_h80_t020 /hana/logbackups/H80 nfs
rw,bg,hard,timeo=600,vers=4,rsize=1048576,wsize=1048576,intr,noatime,lock 0 0
```

- Add the following lines to `/etc/fstab` > this should be the same output from the snapshot command

```
172.18.20.241:/hana_data_h80_mnt00001_t020_dp /hana/data/H80/mnt00001 nfs
rw,bg,hard,timeo=600,vers=4,rsize=1048576,wsize=1048576,intr,noatime,lock 0 0
172.18.20.241:/hana_log_backups_h80_t020_dp /hana/logbackups/H80 nfs
rw,bg,hard,timeo=600,vers=4,rsize=1048576,wsize=1048576,intr,noatime,lock 0 0
```

Step7: Execute the command `mount –a` to mount all the mount points

```
# mount -a
#
```

Now, If you execute `df –h` you should see the `*_dp` volumes mounted.

```
# df -h
Filesystem Size Used Avail Use% Mounted
on
devtmpfs 378G 8.0K 378G 1% /dev
tmpfs 569G 0 569G 0%
/dev/shm
tmpfs 378G 18M 378G 1% /run
tmpfs 378G 0 378G 0%
/sys/fs/cgroup
/dev/mapper/3600a098038304445622b4b584c575a66-part2 47G 20G 28G 42% /
/dev/mapper/3600a098038304445622b4b584c575a66-part1 979M 57M 856M 7% /boot
172.18.20.241:/hana_log_h80_mnt00003_t020_vol 512G 2.1G 510G 1%
/hana/log/H80/mnt00003
172.18.20.241:/hana_log_h80_mnt00001_t020_vol 512G 5.5G 507G 2%
/hana/log/H80/mnt00001
172.18.20.241:/hana_data_h80_mnt00003_t020_vol 1.2T 332M 1.2T 1%
/hana/data/H80/mnt00003
172.18.20.241:/hana_log_h80_mnt00002_t020_vol 512G 2.1G 510G 1%
/hana/log/H80/mnt00002
172.18.20.241:/hana_data_h80_mnt00002_t020_vol 1.2T 300M 1.2T 1%
/hana/data/H80/mnt00002
172.18.20.241:/hana_data_h80_mnt00001_t020_vol 1.2T 6.4G 1.2T 1%
/hana/data/H80/mnt00001
172.18.20.241:/hana_shared_h80_t020_vol/usr_sap_node1 2.7T 11G 2.7T 1% /usr/sap/H80
tmpfs 76G 0 76G 0% /run/user/0
172.18.20.241:/hana_shared_h80_t020_vol 2.7T 11G 2.7T 1% /hana/shared
172.18.20.241:/hana_data_h80_mnt00001_t020_xdp 1.2T 6.4G 1.2T 1%
/hana/data/H80/mnt00001
172.18.20.241:/hana_data_h80_mnt00002_t020_xdp 1.2T 300M 1.2T 1%
/hana/data/H80/mnt00002
172.18.20.241:/hana_data_h80_mnt00003_t020_xdp 1.2T 332M 1.2T 1%
/hana/data/H80/mnt00003
172.18.20.241:/hana_log_backups_h80_t020_xdp 512G 15G 498G 3%
/hana/logbackups/H80_T250
```

Step 8 : Recover the SYSTEMDB. From the HANA Studio, right click SYSTEMDB instance and chose “Backup and Recovery”, and then “Recover System Database”

Note Refer to the guide to recover a database from a snapshot, specifically the SYSTEMDB.

Step 9: Recover the tenant database. From the HANA Studio, right click SYSTEMDB instance and chose “Backup and Recovery”, and then “Recover Tenant Database”.

Note Refer to the guide to recover a database from a snapshot, specifically the TENANT database(s).

Guidance for using these tools

How to take snapshots manually

Before executing a backup commands `azacsnap -c backup`, please run the following test options and check they get executed successfully. These tests ensure that snapshot tools can communicate with the installed SAP HANA database and the underlying storage system of the SAP HANA on **Azure Large Instance** or **Azure NetApp Files** system.

- `azacsnap -c test --test hana`
- `azacsnap -c test --test storage`

The to take a manual database snapshot backup run the following:

```
> azacsnap -c backup --volume data --prefix hana_TEST --retention=1
```

For details on the command options, please refer to section: Snapshot commands Execution and details.

How to setup automatic snapshot backup

It is common practice on Unix/Linux systems to use `cron` to automate running commands on a system. The standard practice for the snapshot tools is to setup the user's `crontab`.

An example of a `crontab` for the user `azacsnap` to automate snapshots is below.

CAUTION Output of `crontab -l` command wrapped at maximum of 80 characters per line. Ensure all the crontab entries a re on a single line, otherwise crontab will error when trying to save.

```
azacsnap@sapprdhdb80:~/bin> crontab -l
MAILTO=""
# =============== TEST snapshot schedule ===============
# Data Volume Snapshots - taken every hour.
@hourly (. /home/azacsnap/.profile ;
cd /home/azacsnap/bin ; azacsnap -c backup --volume data --prefix
hana_TEST --retention=9)
# Other Volume Snapshots - taken every 5 minutes, excluding the top of the hour when
hana snapshots taken
5,10,15,20,25,30,35,40,45,50,55 * * * * (. /home/azacsnap/.profile ;
cd /home/azacsnap/bin ; azacsnap -c backup --volume other --prefix logs_TEST
 --retention=9)
# Other Volume Snapshots - using an alternate config file to snapshot the boot volume
daily.
@daily (. /home/azacsnap/.profile ;
cd /home/azacsnap/bin ; azacsnap -c backup --volume other --prefix DailyBootVol
 --retention=7 --configfile boot-vol.json)
```

Explanation of the above crontab. - MAILTO="" : by having an empty value this prevents cron from automatically emailing the user when executing the crontab entry as it would likely end up in the local user's mail file. - Shorthand version of timing for crontab entries are fairly self-explanatory - @monthly = Run once a month, ie. "0 0 1 * *". - *@weekly = Run once a week, ie. "0 0 * 0*". - @daily = Run once a day, ie. "0 0 * * *". - *@hourly = Run once an*

*hour, ie."0 * * *"*. - The first 5 columns are used to designate times, refer to column examples below: - `0,15,30,45`: Every 15 minutes - `0-23`: Every hour - `*` : Every day - `*` : Every month - `*` : Every day of the week - Command line to execute included within backets"()" - . /home/azacsnap/.profile = pull in the user's .profile to setup their environment, including $PATH, etc. - cd /home/azacsnap/bin = change execution directory to the location "/home/azacsnap/bin" where config files are. - azacsnap -c ….. = the fulle azacsnap command to run, including all the options. -

Further explanation of cron and the format of the crontab file here: https://en.wikipedia.org/wiki/Cron

CAUTION Customers are responsible for monitoring the cron jobs to ensure snapshots are being generated successfully.

How to monitor the snapshots

The following should be monitored to ensure a healthy system:

1. Available disk space. Snapshots will slowly consume disk space as keeping older disk blocks are retained in the snapshot.
2. Successful execution of the snapshot tools
3. Consistency of the snapshots by restoring then to another system periodically.

To get the snapshot details, execute the snapshot command `azacsnap -c details`.

How to delete a snapshot

To delete a snapshot, please execute the command `azacsnap -c delete`. It's not possible to delete snapshots from the OS level. You must use the correct command (`azacsnap -c delete`) to delete the storage snapshots.

CAUTION Be extra vigilant while deleting a snapshot. Once deleted, it is **IMPOSSIBLE** to recover the deleted snapshots. However, to avoid filling up the storage space, use the `--retention` and `--trim` options to automatically clean up the old snapshots.

How to restore a 'hana' snapshot

In this version, there is **NO** restore snapshot command provided for the snapshot restore as a self- service (though snapshot restore can be performed as part of the self-service DR snapshot tools).

A snapshot can be copied back to the SAP HANA data area, but SAP HANA must not be running when a copy is made.

Alternately, you could contact the Microsoft operations team by opening a service request to restore a desired snapshot from the existing available snapshots. You can open a service request from Azure portal: https://portal.azure.com.

If you decide to perform the disaster recovery failover, running the `azure_hana_dr_failover` command at the DR site will automatically make available the most recent (`/hana/data` and

`/hana/logbackups`) snapshots to allow for a SAP HANA recovery. Use this command with caution as it breaks replication between production and DR sites.

How to setup snapshots for 'boot' only

In some cases customer's already have tools to protect SAP HANA and only want to configure 'boot' volume snapshots. In this case the task is simplified and the following steps should be taken.

1. Complete steps 1-4 of the pre-requisites for installation
2. Enable communication with storage.
3. Download the run the installer to install the snapshot tools
    1. Easy installation of snapshot tools (default)
    2. Manual installation of the snapshot tools
4. Complete setup of snapshot tools (steps 1, 3, 4, 5)
5. Create a new configuration file as follows. The boot volume details must be in the OtherVolume stanza (user entries in red):
6. Check the config file, refer to the following example:

```
> cat BootVolume.json
{
  "version": "5.0 Preview",
  "logPath": "./logs",
  "securityPath": "./security",
  "comments": [
    "Boot only config file."
  ],
  "database": [
    {
      "hana": {
        "serverAddress": "X",
        "sid": "X",
        "instanceNumber": "X",
        "hdbUserStoreName": "X",
        "savePointAbortWaitSeconds": 600,
        "hliStorage": [
          {
            "dataVolume": [],
            "otherVolume": [
              {
                "backupName": "shoasnap",
                "ipAddress": "10.1.1.10",
                "volume": "t210_sles_boot_azsollabbl20a31_vol"
              }
            ]
          }
        ],
        "anfStorage": []
      }
    }
```

```
  ]
}
```

7. Test a boot volume backup

```
> azacsnap -c backup --volume other --prefix TestBootVolume --retention 1 --
configfile BootVolume.json
```

8. Check it's listed, note the addition of the `--snapshotfilter` option to limit the snapshot list returned.

```
> azacsnap -c details --snapshotfilter TestBootVolume --configfile
BootVolume.json
List snapshot details called with snapshotFilter 'TestBootVolume'
#, Volume, Snapshot, Create Time, HANA Backup ID, Snapshot Size
#1, t210_sles_boot_azsollabbl20a31_vol, TestBootVolume.2020-07-
03T034651.7059085Z, "Fri Jul 03 03:48:24 2020", "otherVolume Backup|azacsnap
version: 5.0 Preview (20200617.75879)", 200KB
, t210_sles_boot_azsollabbl20a31_vol, , , Size used by Snapshots, 1.31GB
```

9. Now setup automatic snapshot backup.

Note Any setup for communication with SAP HANA is not required.

How to restore a 'boot' snapshot

A 'boot' snapshot can be recovered as follows:

1. The customer will need to shut down the server.
2. After the Server is shut down, the customer will need to open a service request which contains the Machine ID and Snapshot to restore. > Customers can open a service request from the Azure Portal: https://portal.azure.com.
3. Microsoft will restore the Operating System LUN using the specified Machine ID and Snapshot, and then boot the Server.
4. The customer will then need to confirm Server is booted and healthy.

No additional steps to be performed after the restore, but **the Server will be restored to the point when the Snapshot was taken.**

Key facts to know about snapshots

You must be familiar with the key attributes about the storage snapshots:

- **Location of snapshots** : Snapshots can be found in a virtual directory (`.snapshot`) within the volume. See the following examples for **Azure Large Instance**:
  - Database: `/hana/data/<SID>/mnt00001/.snapshot`
  - Shared: `/hana/shared/<SID>/.snapshot`
  - Logs: `/hana/logbackups/<SID>/.snapshot`

  - Boot: boot snapshots are **not visible** from OS level, but can be listed using `azacsnap -c details` > **Note** `.snapshot` is a read-only hidden *virtual* folder providing read-only access to the snapshots.
- **Max snapshot:** The hardware can sustain up to 250 snapshots per volume. The snapshot command will keep a maximum number of snapshots for the prefix based on the retention set on the command line, and will delete the oldest snapshot if it goes beyond the maximum number to retain.
- **Snapshot name:** The snapshot name includes the prefix label provided by the customer.
- **Size of the snapshot:** Depends upon the size/changes on the database level.
- **Log file location:** Log files generated by the snapshot commands are output into folders as defined in the JSON configuration file, which by default is a sub-folder under where the command is run (e.g. `./logs`)..

Disaster Recovery

1. What are the prerequisites for DR setup

The following pre-requisites must be met before you plan the disaster recovery failover.

- You have a DR node provisioned at the DR site. There are two options for DR. One is normal DR, and other is multipurpose DR (See the definition: Terms and Definitions).
- You have storage replication working. The Microsoft operations team performs the storage replication setup at the time of DR provisioning automatically. You can monitor the storage replication using the snapshot command `azure_hana_replication_status` at the DR site.
- You have setup and configured storage snapshots at the primary location.
- You have an HANA instance installed at the DR site for the primary with the same SID as the primary instance has.
- You read and understand the DR Failover procedure located at https://docs.microsoft.com/en-us/azure/virtual-machines/workloads/sap/hana-overview-high-availability-disaster-recovery#disaster-recovery-failover-procedure.
- You have setup and configured storage snapshots at the DR location.
- The `HANABackupCustomerDetail.txt` file has been updated with the DR location storage information on the DR server.
- You completed the steps at the DR site to:
  - Enable communication with storage.
  - Enable communication with SAP HANA.

2. How to setup a disaster recovery

Microsoft supports storage level replication for DR recovery. There are two ways to setup the DR.

One is **normal** and other is **multipurpose**. In the **normal** DR, you have a dedicated instance at the DR location for failover. In the **multipurpose** DR scenario, you have another QA or development HANA instance running on the HANA large instance unit at the DR site. But you also installed a pre-installed HANA instance that is dormant and has the same SID as the HANA instance you want to failover to that HANA Large Instance unit. Microsoft operations sets up the environment for you including the storage replication based on the input provided in the Service Request Form (SRF) at the time of onboarding.

Also, ensure that all the prerequisites are met for the DR setup.

3. How to monitor the data replication from Primary to DR site

Microsoft operations team already manage and monitor the DR link from Primary site to the DR site. You can monitor the data replication from your primary server to DR server using the snapshot command `azure_hana_replication_status`.

4. How to perform a failover to DR site

You perform a failover to DR site, by executing a snapshot command `azure_hana_dr_failover`.

CAUTION Use only when planning to perform a DR exercise. The `azure_hana_dr_failover` command breaks the storage replication. You must reach out to the Microsoft Operations to set up replication again. Once the replication is re-enabled, all the data at DR storage for this SID will get initialized. The command that performs the failover makes available the most recently replicated storage snapshot. If you need to restore back to an older snapshot, open a support request so operations can assist to provide an earlier snapshot restored in the DR site.

At a high level, here are the steps to follow for DR failover:

- You must shut down the HANA instance at **primary** site. This is only needed if you are truly doing the failover to DR site so you don't have data inconsistencies.
- Shutdown the HANA instance on the DR node for the production SID.
- Execute the snapshot command azure_hana_dr_failover on the DR node with the SID to be recovered o The snapshot command breaks the storage replication link from the Primary to the DR site o The snapshot command restores the /data and /logbackups volume only, /shared volume is NOT recovered, but rather it uses the existing /shared for SID at the DR location. o Mount the /data and /logbackups volume – ensure to add it to the fstab file
- Restore the HANA SYSTEMDB snapshot. Please note, HANA studio only shows you the latest HANA snapshot available under the storage snapshot restored as part of the snapshot command azure_hana_dr_failover execution.
- Recover the tenant database
- Start the HANA instance on the DR site for the Production SID (Example: H80 in this case)
- Perform the testing.

Troubleshooting

The following are common issues that you may encounter while running the commands. Please follow the resolution instructions mentioned to fix the issue. If you still encounter an issue, please open a Service Request from Azure portal and assign the request into the SAP HANA Large Instance queue for Microsoft Support to respond.

hdbuserstore location

The `hdbuserstore` program is typically found under `/usr/sap/<SID>/SYS/exe/hdb/`, but is typically added to the `azacsnap` user's `$PATH` if using the installer.

Failed test connection with SAP HANA

You run a test with `azacsnap -c test --test hana` and receive the following error:

Solution: Check the configuration file for each HANA instance to ensure the values are correct.

Try to run the command below to verify if the `hdbsql` command is in the path and it can connect to the SAP HANA Server. The following example shows the correct running of the command and its output.

```
> hdbsql -n 172.18.18.50 - i 00 -d SYSTEMDB -U SCADMIN "\s"
host           : 172.18.18.50:
sid            : H80
dbname         : SYSTEMDB
user           : SCADMIN
kernel version: 2.00.032.00.1533114046
SQLDBC version:        libSQLDBCHDB 2.03.112.1532025331
autocommit     : ON
locale         : en_US.UTF-8
input encoding: UTF8
```

In the following example, the `hdbsql` command isn't in the users `$PATH`.

In this example, the `hdbsql` command is temporarily added to the user's `$PATH`, and when run again shows the connection key hasn't been setup correctly with the `hdbuserstore Set` command (refer to SAP HANA documentation for details). To permanently add to the user's `$PATH`, update their `$HOME/.profile`:

```
# export PATH=$PATH:/hana/shared/H80/exe/linuxx86_64/hdb/
```

Failed test with storage

The command `azacsnap -c test --test storage` does not complete successfully.

```
azacsnap -c test --test storage
The authenticity of host '172.18.18.11 (172.18.18.11)' can't be established.
ECDSA key fingerprint is SHA256:QxamHRn3ZKbJAKnEimQpVVCknDSO9uB4c9Qd8komDec.
Are you sure you want to continue connecting (yes/no)?
```

Solution: The above error normally shows up when storage user has no access to the underlying storage. To check if you have access to storage using the storage user, please run the `ssh` command to validate communication with the storage platform.

```
> ssh <StorageBackupname>@<Storage IP address> "volume show -fields volume"
```

An example with expected output:

```
> ssh clt1h80backup@10.8.0.16 "volume show -fields volume"
vserver volume
------------------------------- ------------------------------
osa33-hana-c01v250-client25-nprod hana_data_h80_mnt00001_t020_vol
osa33-hana-c01v250-client25-nprod hana_data_h80_mnt00002_t020_vol
```

The authenticity of host '172.18.18.11 (172.18.18.11)' can't be established

```
> azacsnap -c test --test storage
The authenticity of host '10.3.0.18 (10.3.0.18)' can't be established.
ECDSA key fingerprint is SHA256:cONAr0lpafb7gY4l31AdWTzM3s9LnKDtpMdPA+cxT7Y.
Are you sure you want to continue connecting (yes/no)?
```

Solution: Do not select Yes. Please ensure that your storage IP address is correct. If there is still an issue, please confirm the storage IP address with Microsoft operations team.

Appendix - Changelog

The following lists changes made to the commands to provide new functionality or resolve defects.

Changes in v5.0

Published: Fri Jul 3 05:00:00 UTC 2020

MAJOR

- Support for an additional storage back-end (**Azure NetApp Files**) added.
- The following commands replaced with `azacsnap` and various options:
  - **azure_hana_backup:** = `azacsnap -c backup`
  - **azure_hana_snapshot_delete** = `azacsnap -c delete`
  - **azure_hana_snapshot_details** = `azacsnap -c details`
  - **removeTestStorageSnapshot** : automatically removed after `-c test --test storage`
  - **testHANAConnection** = `azacsnap -c test --test hana`
  - **testStorageSnapshotConnection:** = `azacsnap -c test --test storage`

Documentation :

- Updated entire documentation with the new `azacsnap` command line format.

Changes in v4.3

Published: Thu Apr 30 04:30:00 UTC 2020

Documentation :

- Added SQL command to prevent the AZACSNAP user's password from expiring (6332903).
- Clarify details of DR process and check for any missing points (6362513).

azure_hana_backup:

- Ensure `logMsg` receives a string for output (6332906).
- Record version number of `azure_hana_backup` in the SAP HANA Backup Catalog and the snapshot comment field for each run (5485633).

azure_hana_dr_failover:

- Provide correct storage IP address when "Displaying Mount Points by Volume". (6432149).

azure_hana_test_dr_failover:

- Provide correct storage IP address when "Displaying Mount Points by Volume". (6432149).

azure_hana_replication_status:

- Make replication status output parse-able for tools like `grep`. (6458458).

testStorageSnapshotConnection:

- When running the command, the output will provide guidance on configuring `boot` snapshots. (6339419).

Changes in v4.2

Published: Sun Sep 25 04:20:00 UTC 2019

Documentation :

- Added section to guide a manual installation, following the steps the installer automatically completes (5421385).

azure_hana_backup :

- Limit the `--trim` option to only remove backup catalog entries for the same `--prefix` as passed on the command line and remove any log entries older than the oldest non-log file backup entry in the backup catalog (5286457).
- Improve the `--trim` option to work with multiple tenants (4873256).
- Resolve logMsg output "uninitialized value $msgString in concatenation (.) or string at (eval 14) line 472" (4882230).

azure_hana_ snapshot_delete :

- No longer requires the `--sid` parameter to be in lower-case (5421378).

azure_hana_snapshot_installer_v4. 2.run:

- Updated to work with system with multiple instances installed (5421382).

Changes in v4.1

Published: Sun Jun 30 04:10:00 UTC 2019

ALL :

- Allow for Storage API change (4667060).

azure_hana_backup :

- Add backup catalog trimming (`--trim`) to help with disk space management (3788143).
- Allow snapshots at Disaster Recovery site where DR site hostname is different to production (4580377).
- Allow snapshot type "boot" even if an existing snapshot does not exist (4765161).

- Insert comment into the SAP HANA backup catalog containing the storage snapshot name (2719997).
- Prevent the new `*data_backups*` volumes from being included in snapshots. This volume is use for standard SAP HANA backups and not these snapshot tools (4667067).
- Remove `--frequency` from being a mandatory argument as this option is not currently used, will still parse this argument but won't impact on execution (4521966).

azure_hana_snapshot_details :

- Improve output to be more easily read by a human, or machine parsed, including pasting into a spreadsheet (4521979).

azure_hana_dr_failover :

- Add clone expiry date into the Clone comment field (4521977).

testStorageSnapshot & removeTestStorageSnapshot :

- Resolve how these commands create and delete the temporary snapshots for testing (4764966).

Changes in v4.0

Published: Wed Apr 1 0 04 : 00 :00 UTC 2019

ALL :

- Downloadable as a single self-extracting installer to ease system setup.
- Now provided as binary files to ensure they include all dependent libraries and can be more easily setup and supported.
- No longer have the .pl extension as they are provided as binaries.
- Added support for Generation 4 of SAP HANA on Azure Large Instance.
- Command line switches and parameters changed to meet the standard format expected of command line programs on Unix/Linux systems (e.g. –prefix). Refer to this documentation for guidance on using these commands.
- Various improvements in console output, especially reducing console output by default. Logging output to the console can be provided by passing the -v (verbose) switch to each command.
- hdbsql command needs to be part of the user's $PATH.

azure_hana_backup :

- Added support for SSL communication to SAP HANA. Use of this feature requires certificates to be named and stored in the location specified in this documentation.

azure_hana_dr_failover :

- Improved message to customers on steps to action after running this command.

azure_hana_test_dr_failover :

- Improved message to customers on steps to action after running this command.
- Changed to only allow a maximum of two clones, to prevent customers from unintentionally using up their free space.

Changes in v3.4.1

Published: Tue Oct 16 03 : 41 : 00 UTC 2018

azure_hana_backup : updated to v3.4.1) Following fix apply to these snapshot tools.

azure_hana_dr_failover : updated to v3.4.1)

azure_hana_replication_status : updated to v3.4.1)

azure_hana_snapshot_details : updated to v3.4.1)

azure_hana_test_dr_failover : updated to v3.4.1)

removeTestStorageSnapshot : updated to v3.4.1)

testStorageSnapshotConnection : updated to v3.4.1)

- Handle an environment where the HANA SID partially matches the Tenant ID (e.g. SID = H80 and Tenant ID = H800).

Changes in v3.4

ALL * :

- Addition of Common Log Format to start of each line written to the logs when this snapshot command is run, for better telemetry.
- Moved $version to just below opening header comment to avoid duplicating the version number in the snapshot command and ensure version# is consistent.

azure_hana_backup :

- Retry added to the runSSHDiagCmd function. Will try the to execute the command up to 4 times, with the wait in seconds between each attempt 7, 21, 63 seconds respectively.
- Now tests for HSR setup (new function runCheckHSRStatus), and if detects possibility of two nodes running as primary will exit without snapshot to avoid data snapshot taken on both nodes simultaneously.

azure_hana_test_dr_failover :

- Move the message (“This clone is kept for 4 weeks before it is automatically removed.”) to the end of the output so it becomes part of the steps for customer to follow. Was at the start of the output and scrolled off the screen.
- Modify clone creation to thin provision the clone.

- Prompt for a contact email when doing the Test DR Failover to provide a customer contact to get confirmation it is ok to delete the clone.
- Add the contact email and an expiry date into the clone comment field for automated clone deletion 4 weeks from date of creation (see above).

azure_hana_dr_failover :) Following changes apply to both snapshot tools.
azure_hana_test_dr_failover :)

- Add capability to run DR failover against Scale-Out nodes.
- Ensure the latest HANA data snapshot volume is presented for recovery.
- Add log_backups volumes from both nodes for DR recovery of HSR configuration.
- Simplify the log backups mount point for non-HSR.
- Reduce display mount point output to only show the recovery volumes.
- Simplify guidance on mounting volumes for recovery when snapshot tools complete.

Changes between release v3.3 and v3.4

Published: Wed Jul 11 03:31:03 UTC 2018

removeTestStorageSnapshot : updated to v3.3.1

- Fix for two variables declared twice in snapshot command with `my $<var> ($filename, $sshCmd)`, duplicate declarations removed.
- Fix missing `my $LOG_CRIT` declaration now added.

Published: Fri Jun 1 21:45:00 UTC 2018

azure_hana_backup : updated to v3.3.2

- Fix for hostnames which contain a hyphen (-), this character is not allowed in the volume name, so is converted to underscore (_).
- For HSR HANA installs, restrict the snapshot tools to only run against the volumes associated with the host they are being run from (i.e. where the volume name contains the same hostname).

Published: Wed May 30 01:43:37 UTC 2018

azure_hana_snapshot_delete : updated to v3.3.1

- Fix for time calculation to prevent a snapshot being deleted if it is less than 10 minutes old, all time calculations moved to UTC.

testStorageSnapshotConnection : updated to v3.3.1

- Snapshot command would sometimes fail the creation of the OS backups snapshot since a copy of that snapshot already existed with a given timestamp because of the placement of the calls for that snapshot command. Moved the OS backup calls to take place before creating snapshots for each of the SIDs.

Published: Mon May 28 07:34:45 UTC 2018

azure_hana_backup : updated to v3.3.1

- Fix for "`if ($numKeep le 0 or $numKeep gt 250)`" to change string comparison to the correct number comparison (e.g. "`if ($numKeep <= 0 or $numKeep > 250)`").

Published: Fri May 25 17:00:00 UTC 2018 **Snapshot Command Bundle v3.3 Released**