Commvault Azure SQL Protection

August 2025

Table of Contents

OBJECTIVES

This solution guide aims to achieve the following objectives:

- Introduce Commvault's Azure SQL backup solution and its key features.
- Provide a detailed overview of the architecture, configuration, and processes involved in backing up and restoring Azure SQL databases using Commvault.

OVERVIEW

Azure SQL is a managed cloud database service from Microsoft that provides a highly available, scalable, and secure platform for your applications. You can choose from a variety of pricing options to fit your budget, and Azure SQL automatically handles infrastructure management tasks such as upgrades, patching, backups, and monitoring.

This document focuses on the following two Platform as a Service (PaaS) Azure SQL offerings from Microsoft Azure:

- Azure SQL Database (DB) – a fully managed single database or elastic pool service that powers cloud native workloads.
- Azure SQL Managed Instance (MI) – a near full SQL Server engine delivered “as a service,” ideal for lift and shift migrations and app modernization.

Throughout the document, paragraphs that apply to only one offering are labelled [DB only] or [MI only] so you can quickly zero in on relevant guidance. The term Azure SQL, when used as is, encompasses both these offerings.

Commvault provides robust Azure SQL backup and restore capabilities, encompassing entire Instances, and full Azure subscriptions. Flexible scheduling options and retention policies automate your backups. De-duplicated and compressed air-gapped backups make sure that customer is always ransomware ready.

AUDIENCE

This whitepaper is intended for:

- IT administrators and database professionals who are responsible for data availability and integrity of Azure SQL databases.
- Business decision-makers interested in understanding the benefits of Commvault's Azure SQL backup solution.

SOLUTION OVERVIEW

Commvault offers **two complementary backup approaches**. Administrators decide which to enable; Commvault then executes jobs according to the configured plan.

1. **Native Backup sync** (*Early adopter feature – write to products@commvault.com to get access)* – Commvault **catalogues Azure-native transaction-log, and Long-Term Retention (LTR) backups** and presents them in from the Commvault Command Center UI. From there, operators can launch point-in-time restores without ever opening the Azure portal. Azure native backups data never leaves the customer’s Azure account; Commvault stores only metadata pointers required to browse and restore it.

2. **Export Backups**

   - [DB-only] **Sqlpackage / DacFx export** writes a compressed **.bacpac** to an **SSD staging disk** on the Gateway, then streams that file into any Commvault storage target. **This is done for all databases with size < 2TB.** Starting with version 11.40, the Commvault Cloud software skips backing up Azure SQL databases with sizes larger than 2TB. This limitation arises because very large databases have a high rate of failure due to Azure's limitations, especially when the data export operation to the Azure cloud exceeds 20 hours. Customers are advised to rely on native Azure SQL backups to achieve their business continuity needs for databases sizes > 2TB.
   - [MI-only] **COPY_ONLY native backup** instructs SQL Server to produce a **.bak** file. The database engine **first stages the file in customer-owned Azure Blob storage**, after which Commvault copies it into the chosen backup library. *System-Managed Keys are **not** supported for this workflow; see Section 8.*

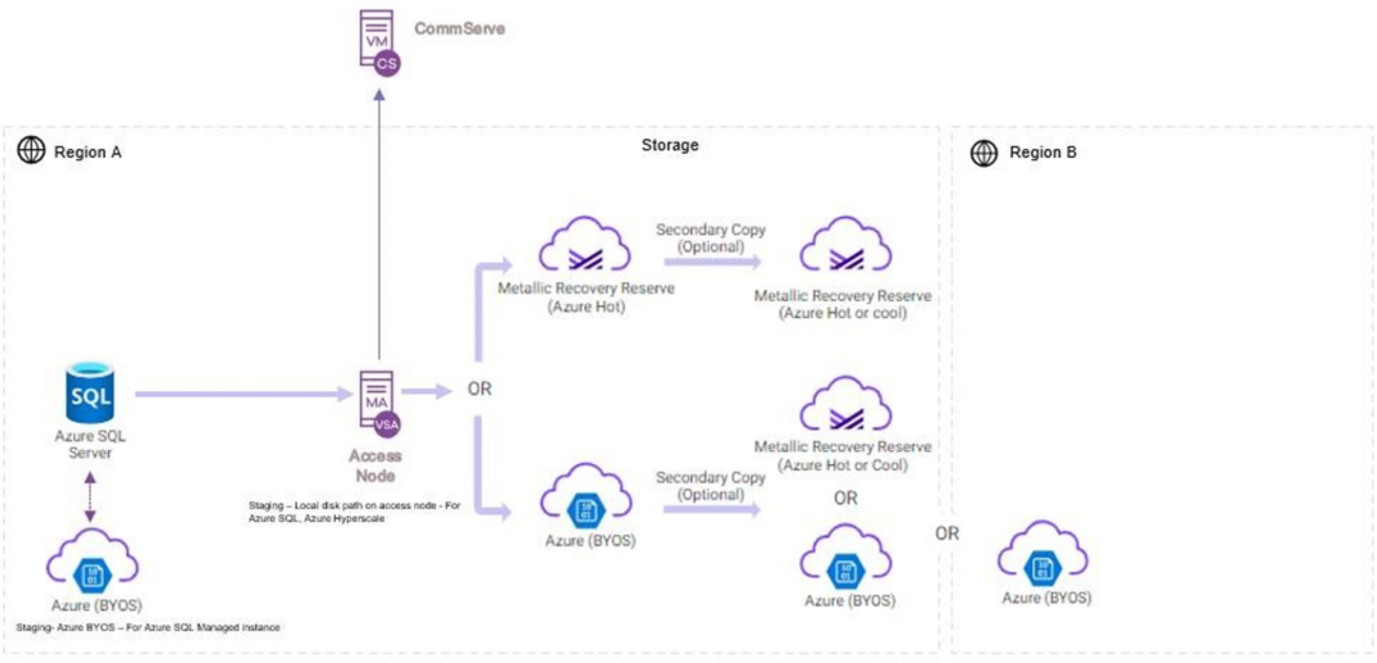
Both methods can operate side-by-side.

Suggested Method Selection

Commvault does **not** auto-switch between native and export paths. The table below summarizes **recommended choices** based on size and operational goals; you can override per instance in the Command Center UI.

Database Size or Requirement	**Recommended Path – Azure SQL DB**	**Recommended Path – Azure SQL MI**
≤ 2TB, need cross-region copy, compliance copy	Export Backups (.bacpac)	Export Backup (COPY_ONLY (.bak))
> 2TB	Native Backup sync	Export Backup (COPY_ONLY (.bak)) validated to 15TB
Frequent PITR within region	Native Backup sync	Native Backup sync
Hyperscale tier < 2TB	Export Backups (.bacpac)	NA
Hyperscale tier > 2TB	Native Backup sync	NA

SOLUTION ARCHITECTURE


VM
CS
CommServe
Region A
Storage
Region B
Secondary Copy (Optional)
Metallic Recovery Reserve (Azure Hot)
Metallic Recovery Reserve (Azure Hot or cool)
SQL
Azure SQL Server
MA
VSA
Access Node
OR
Metallic Recovery Reserve (Azure Hot or Cool)
Secondary Copy (Optional)
OR
OR
Staging – Local disk path on access node - For Azure SQL, Azure Hyperscale
Azure (BYOS)
Azure (BYOS)
Azure (BYOS)
Azure (BYOS)
Staging- Azure BYOS – For Azure SQL Managed Instance

CommServe – Responsible for configuration and scheduling.

MA(MediaAgent) – Manages deduplication database and communication to storage.

Access Node (Gateway) – Manages data transfer with Azure SQL

The Access Node (gateway) is a lightweight Windows **or Linux** VM deployed inside the customer subscription. It hosts the Commvault Media Agent plus database agents that drive export and restore operations.

- **Sizing guidance** – The built-in ARM template defaults to an **“Extra-Small”** size (2 vCPU / 8 GB RAM) which comfortably handles ~20 simultaneous export or restore streams. The Appendix contains a sizing calculator for larger estates.
- **Regional placement** – Deploy the Gateway in the **same Azure region** as protected databases to minimize latency and minimize export staging time.

Storage – Azure Blob (BYOS - Bring Your Own Storage) or Commvault Cloud Air Gap Protect – Stores de-duplicated and compressed backups and air gapped copies of backups.

Azure SQL Server – Azure SQL or Azure Managed Instance or Azure Hyperscale.

Staging – For staging backups during the backup and restore process.

Command Center (UI) – Starting in **Release 11.40**, Command Center surfaces Azure-native backups in their own view. Administrators can filter by database, timestamp, or backup type and launch restores directly from that list.

CONFIGURATION

PRE-REQUISITES

Network requirements

Verify that the following ports are open on the access node that has access to the Azure SQL:

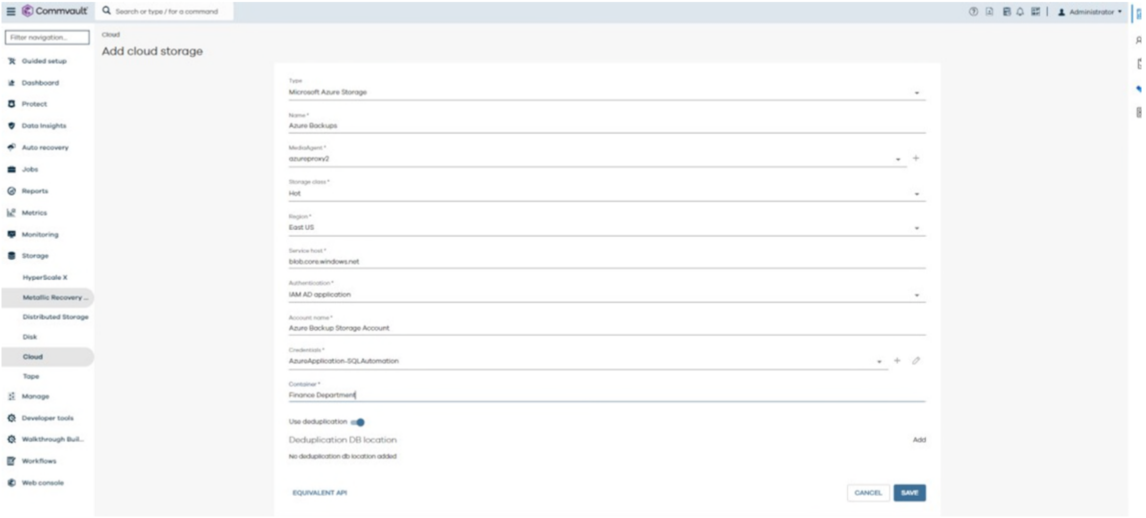
- 8443 to https://management.core.windows.net:8443
- 1433 to *.database.windows.net

The Azure SQL configuration workflow from the Commvault UI walks a customer through the following steps.

- Create a storage destination.
- Create a plan.
- Set up Azure SQL cloud account authentication.
- Discover SQL instances and databases.
- Associated discovered instances/databases to the plan.
- Run schedules backups or on demand backups and restores.

Create a storage destination

A storage destination streamlines data protection by offering a centralized repository for backups and archives across your entire cloud environment. It eliminates the need for managing multiple cloud storage accounts and simplifies operations through a unified web-based interface. Configuration is straightforward, with flexible options for data placement and encryption with compliance and security. Additionally, Commvault provides global deduplication and compression, minimizing storage requirements and optimizing costs.



Add cloud storage

Type
Microsoft Azure Storage

Name *
Azure Backups

MediaAgent *
azureproxy2

Storage class *
Hot

Region *
East US

Service host *
blob.core.windows.net

Authentication *
IAM AD application

Account name *
Azure Backup Storage Account

Credentials *
AzureApplication-SQLAutomation

Container *
Finance Department

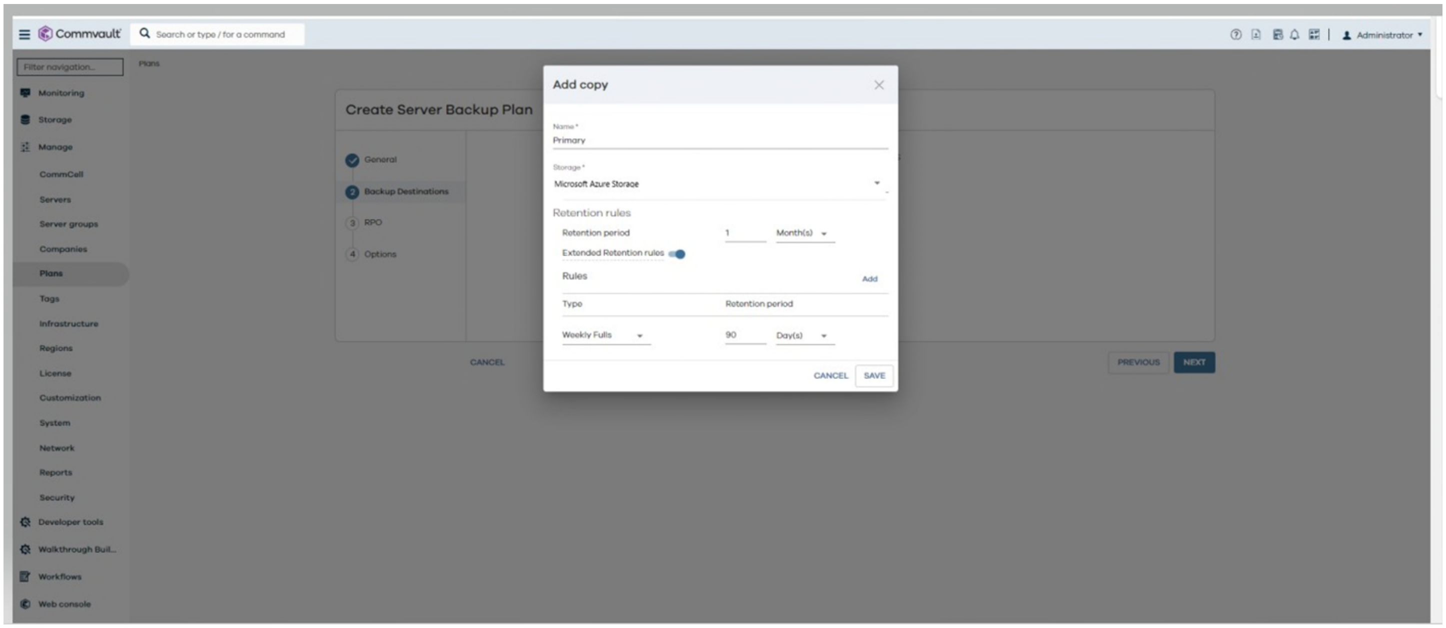
Use deduplication

Deduplication DB location

No deduplication db location added

Create a plan

A plan acts as a central orchestration tool, defining and automating backup workflows for a customer's entire IT environment, which eliminates manual configuration by encapsulating backup policies, schedules, and destinations within a single, reusable entity. This allows for easy deployment across diverse environments and provides consistent, reliable data protection. Additionally, plans also offer dynamic scheduling options.



Add copy

Name * Primary

Storage * Microsoft Azure Storage

Retention rules

Retention period 1 Month(s)

Extended Retention rules ●

Rules Add

Type	Retention period
Weekly Fulls	90 Day(s)

CANCEL SAVE

Set up Azure SQL cloud account authentication

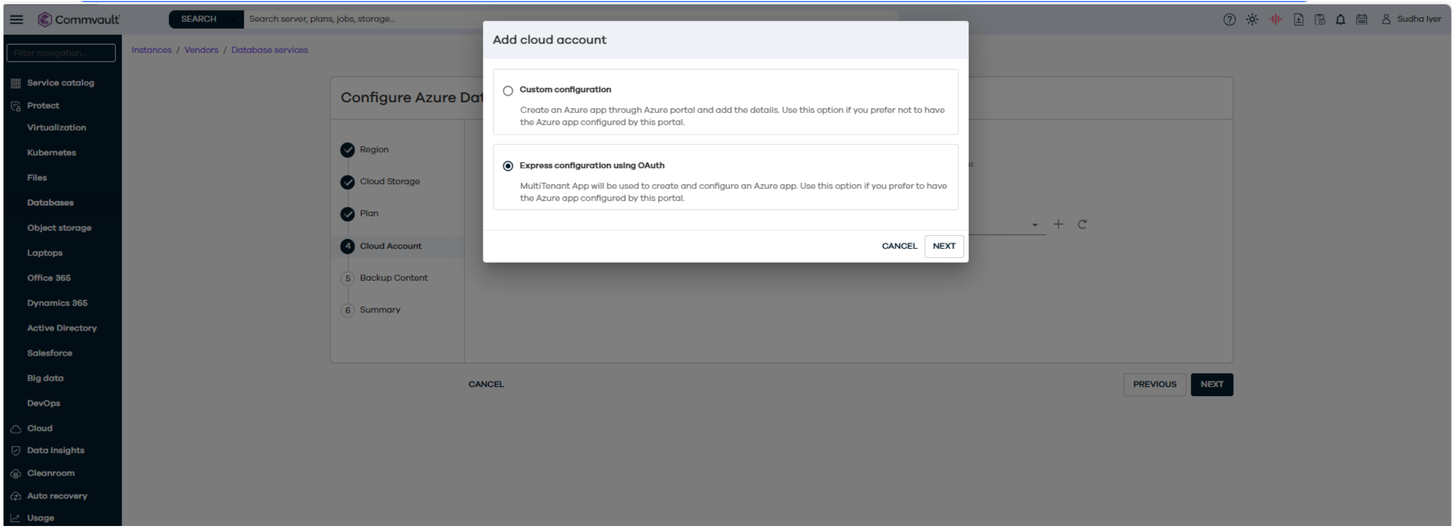
Commvault needs a connection to the customer's Azure subscription(s) to protect Microsoft Azure SQL. Commvault utilizes the Microsoft Azure SQL Database REST APIs that are secured by Azure Active Directory, which in turn provides identity and access management for the Azure cloud. This allows for a secure and controlled interaction between Commvault and a customer's Azure SQL environment.

Two steps of configuration are required in Commvault to protect Azure SQL. The first step of the configuration allows Commvault to automatically discover all Azure SQL instances in each subscription. The second step of the configuration sets up authentication to discover databases and back them up.

Commvault supplies **two onboarding flows**. Use the one that best matches your operating model.

1. Azure Express Configuration wizard [SaaS environments for now, coming soon in software]

   - Automatically **Creates or registers** an **Azure App Registration (service principal)** with the required roles (*Backup Contributor* on the subscription and *Storage Blob Data Contributor* on the staging account).
   - **Discovers** all Azure SQL DBs and MIs in the selected tenant and lists them for immediate plan assignment.
     The wizard is ideal for first-time setups or when you want to delegate all Azure artefact creation to Commvault. or pre-existing storage.



Add cloud account

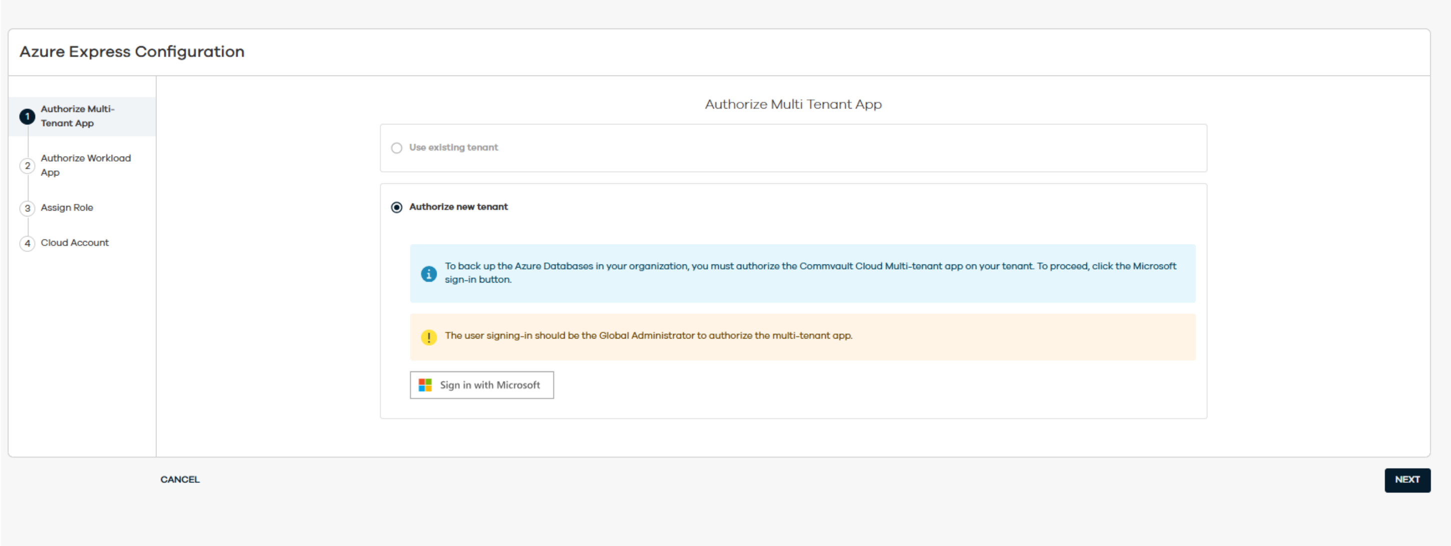
○ Custom configuration

Create an Azure app through Azure portal and add the details. Use this option if you prefer not to have the Azure app configured by this portal.

● Express configuration using OAuth

MultiTenant App will be used to create and configure an Azure app. Use this option if you prefer to have the Azure app configured by this portal.

CANCEL NEXT



Azure Express Configuration

Authorize Multi Tenant App

○ Use existing tenant

● Authorize new tenant

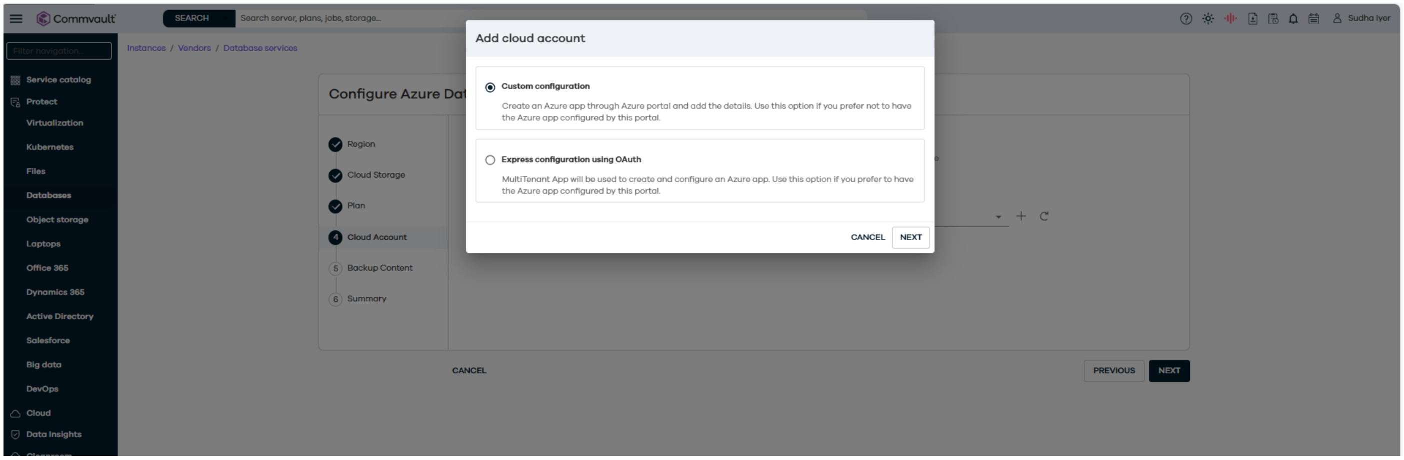
To back up the Azure Databases in your organization, you must authorize the Commvault Cloud Multi-tenant app on your tenant. To proceed, click the Microsoft sign-in button.

The user signing-in should be the Global Administrator to authorize the multi-tenant app.

Sign in with Microsoft

CANCEL NEXT

2. Custom Configuration (both SaaS and Software deployments)
   - Advanced users can use “Custom Configuration” to setup and supply their own service-principal credentials.



Add cloud account

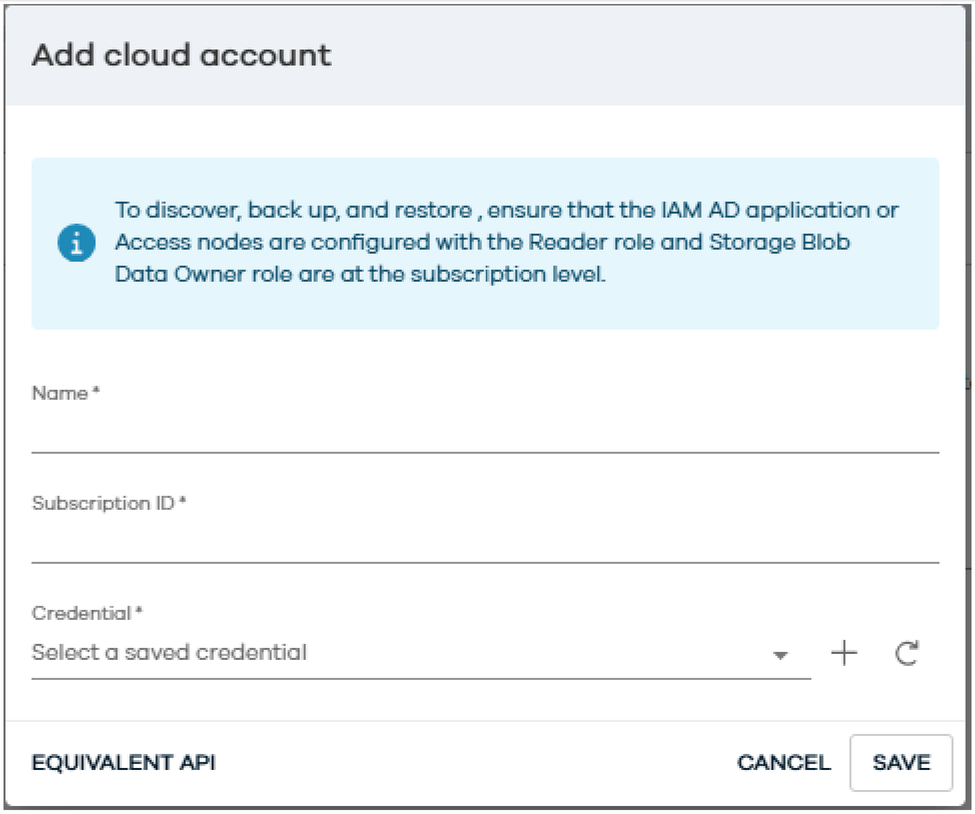
● Custom configuration

Create an Azure app through Azure portal and add the details. Use this option if you prefer not to have the Azure app configured by this portal.

○ Express configuration using OAuth

MultiTenant App will be used to create and configure an Azure app. Use this option if you prefer to have the Azure app configured by this portal.

CANCEL NEXT



Add cloud account

To discover, back up, and restore , ensure that the IAM AD application or Access nodes are configured with the Reader role and Storage Blob Data Owner role are at the subscription level.

Name *

Subscription ID *

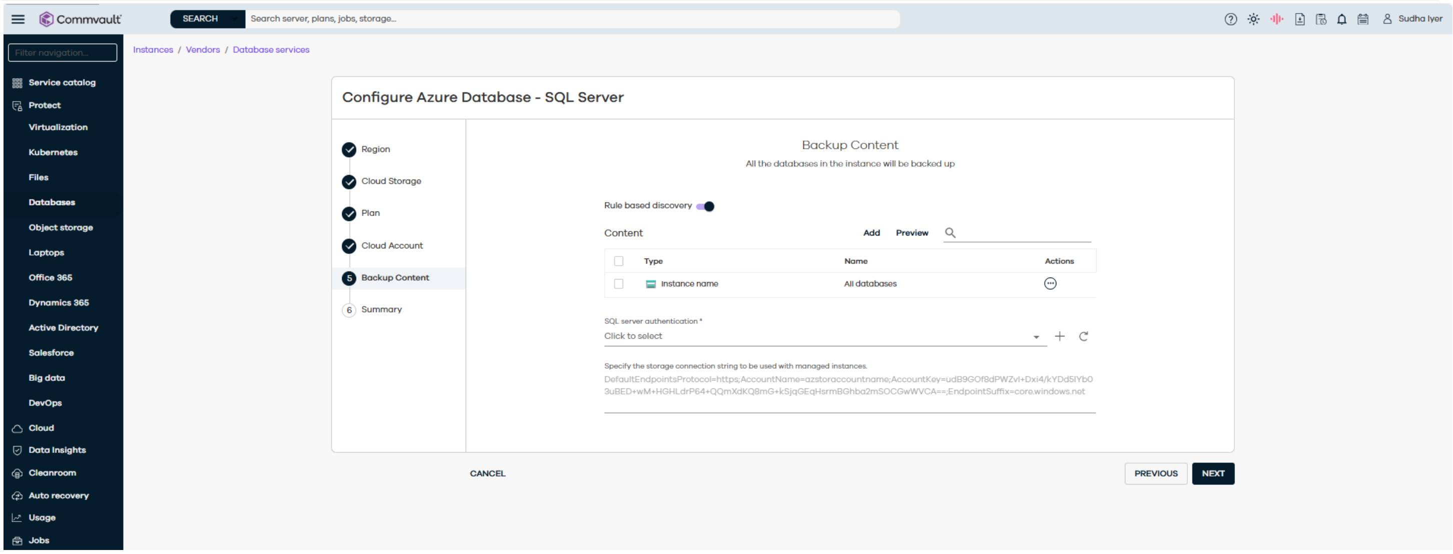
Credential *

Select a saved credential

Configure instances for backup

Database instances can be discovered using rules or tags. Instances are automatically discovered every 24 hours. Immediate manual discovery based on rules and tags is also supported.

- An instance group is created to represent rules. Instance groups can be used to run manual discovery or to modify rules after creation.
- Discovery runs once in 24 hours – no way to change duration or frequency as of now
  - Can run an on-demand discovery job
- Databases are discovered per subscription, i.e., per cloud account
  - The same instance if discovered through two cloud accounts will be configured twice



Configure Azure Database - SQL Server

Backup Content

All the databases in the instance will be backed up

Rule based discovery ●

Content

□	Type	Name	Actions
□	Instance name	All databases	⋯

SQL server authentication *

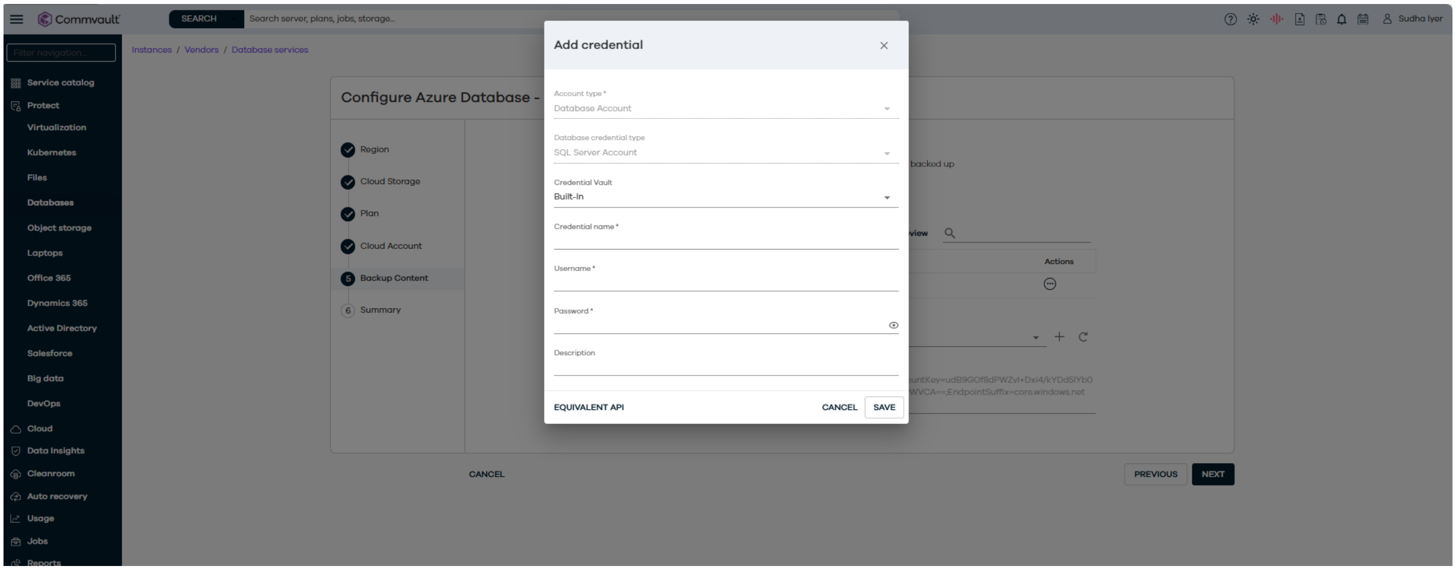
Click to select

Specify the storage connection string to be used with managed instances.

DefaultEndpointsProtocol=https;AccountName=azstoraccountname;AccountKey=udB9GOf8dPWZvl+Dxi4/kYDd5IYb03uBED+wM+HGHLdrP64+QQmXdKQ8mG+kSjqGEqHsrmBGhba2mSOCGwWVCA==;EndpointSuffix=core.windows.net

Setup database authentication

AD authentication or SQL server authentication can be used for database backups. Credentials can be created once and securely stored in a Commvault credential vault that allows for easy re-use for configuration and easy password rotation.



Add credential

Account type *
Database Account

Database credential type
SQL Server Account

Credential Vault
Built-In

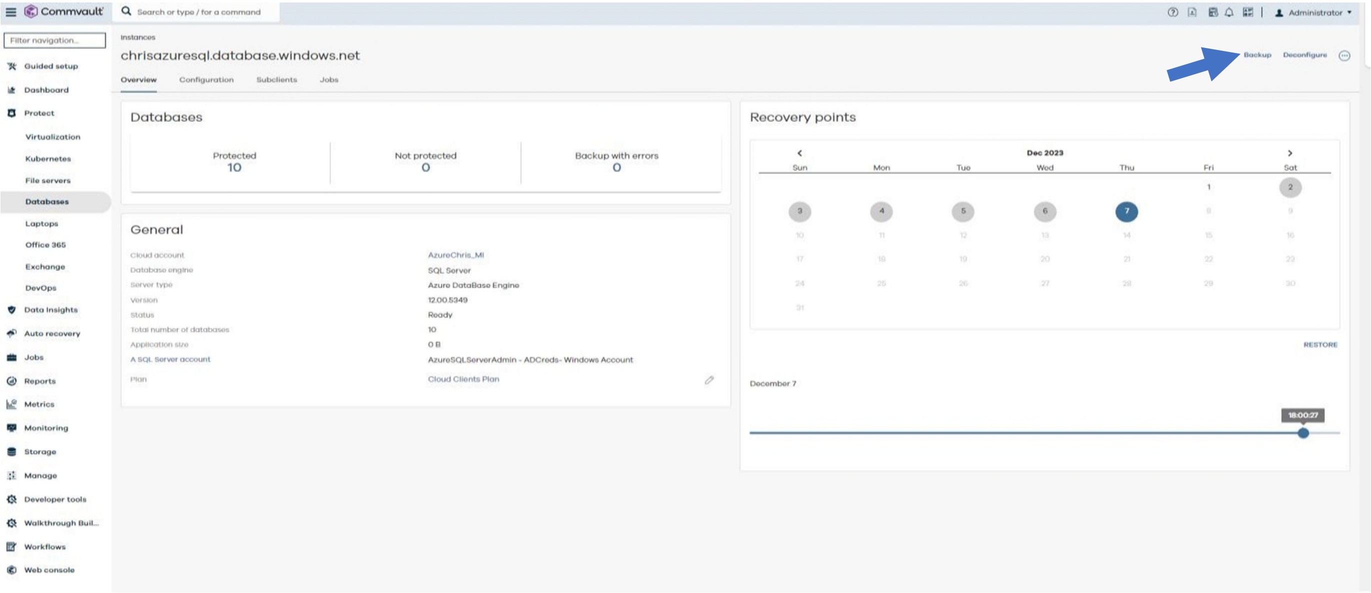
Credential name *

Username *

Password *

Description

RUN AND ON-DEMAND OR SCHEDULED BACKUPS



Instances

chrisazuresql.database.windows.net

Databases

Protected	Not protected	Backup with errors
10	0	0

General

Cloud account	AzureChris_MI
Database engine	SQL Server
Server type	Azure DataBase Engine
Version	12.00.5349
Status	Ready
Total number of databases	10
Application size	0 B
A SQL Server account	AzureSQLServerAdmin - ADCreds- Windows Account
Plan	Cloud Clients Plan

Recovery points

Dec 2023

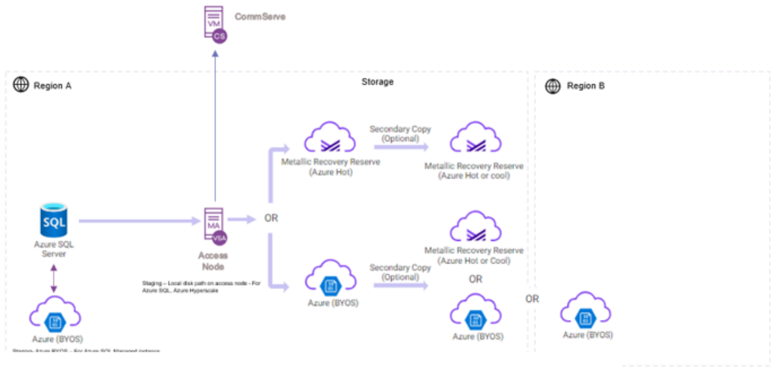
Sun	Mon	Tue	Wed	Thu	Fri	Sat
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

December 7

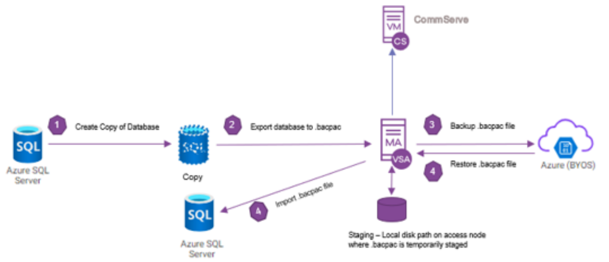
18:00:27

AZURE SQL BACKUP AND RESTORE PROCESS

Azure SQL (export)


CommServe
VM
CS
Region A
Storage
Region B
Secondary Copy (Optional)
Metallic Recovery Reserve (Azure Hot)
Metallic Recovery Reserve (Azure Hot or cool)
SQL
Azure SQL Server
MA
VSA
Access Node
OR
Staging – Local disk path on access node - For Azure SQL, Azure Hyperscale
Metallic Recovery Reserve (Azure Hot or Cool)
Secondary Copy (Optional)
OR
OR
Azure (BYOS)
Azure (BYOS)
Azure (BYOS)
Azure (BYOS)

Azure SQL Backup and Restore Flow



Backup Workflow

1. The Commvault SQL agent on the access node uses Azure SQL REST APIs to first create a transactionally consistent copy pf the Azure SQL database under the same server as the original database.
2. An export operation is then run on the copied database, and the resulting .bacpac file is copied to a preconfigured staging location on the access node. The database copy is deleted immediately after an export.
3. The access node then writes the .bacpac in Commvault's proprietary, compressed, and de-duped format to the storage destination.

Restore Workflow

4. A restore operation restores the .bacpac file to the staging location on the access node.
5. The access node then imports the .bacpac file to the destination server.

Backup Process:

1. The CommServe starts backup based on the schedule defined in the plan or a user can start an ondemand backup.

2. The CommServe sends the backup request to the Commvault SQL agent on the access node.
3. The Commvault SQL agent on the access node uses Azure SQL REST APIs to first create a transactionally consistent copy of the Azure SQL database under the same server as the original database. This is the method recommended by Microsoft.

   https://learn.microsoft.com/en-us/azure/azure-sql/database/database-copy?view=azuresql&tabs=azure-powershell

   - The copy has the same access controls as the source database. For example, if the source is accessed with a private end point, then the copy will continue to be accessed with a private end point.
   - Copy is named with prefix cv_copy_xxxx. The copy database should be excluded from any locking policies that may be in place since this copy is deleted as soon as the export process completes.
   - You can choose the pricing tier to be used for the copy database by configuring the Azure SQL purchasing model in the subclient properties window. By default, for databases residing inside an elastic pool, the copy is created inside the pool. However, when you configure a pricing tier, the copy is created with the chosen pricing tier, which may be outside the elastic pool. This allows you to control the resources assigned to the database copy, which in turn can help you reduce Azure costs.
4. An export operation is then run on the copied database and the resulting .bacpac file is copied to a preconfigured staging location on the access node.
   - Commvault uses the DacFx API to export the database.
   - Please check this link to see limitations and restrictions with DacFX
   - Sizing of the staging location – Access nodes processing import/export requests need to store the

     BACPAC file as well as temporary files generated by the Data-Tier Application Framework (DacFX). The disk space required varies significantly among databases with the same size and can require disk space up to three times the size of the database. As a result, some requests may fail with the error “There is not enough space on the disk.” The workaround is to increase the staging space on the access node.
   - TEMP/TMP system variables – If by chance you receive a failing with Out of Disk space message, it's advisable to configure the %TEMP% folder of the system to reside on a distinct data disk. By doing so, you can confirm sufficient space for the export process to execute smoothly, avoiding potential disk space complications.

   To configure the system's %TEMP% folder:

   i. Open the System Properties window by pressing the Windows key + Pause/Break or right-clicking on This PC and selecting Properties.

   ii. Select the link labeled Advanced system settings on the left-hand side.

   iii. In the ensuing System Properties window, navigate to the bottom and select Environment Variables.

iv. Under the section labeled System Variables, locate the **TEMP and TMP variables**, then select Edit associated **with each one.**

v. Modify the values of both variables to point to a pathway on the separate data disk you have established. For instance, if your data disk is designated as D:, set the values as D:\Temp.

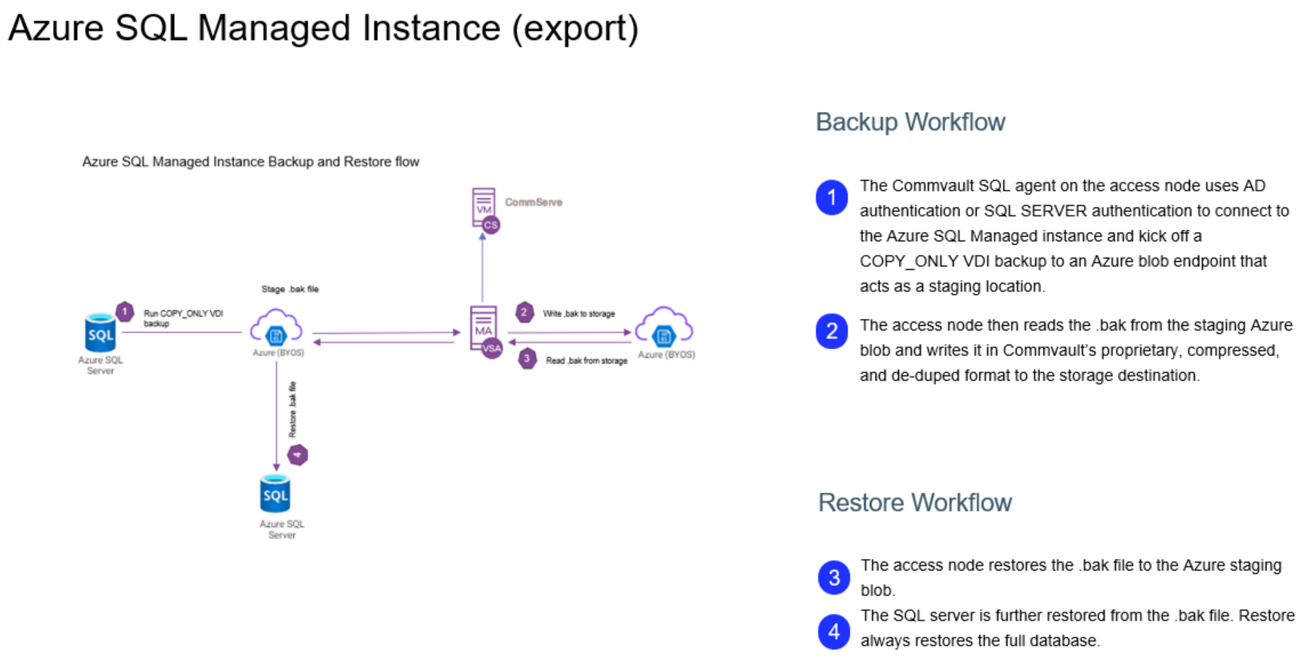
vi. Confirm the changes by selecting OK and closing all open windows.

5. The database copy is deleted immediately after an export operation.
6. The access node then writes the .bacpac in Commvault proprietary format to the storage destination.
7. Backup ends after the .bacpac is successfully copied.
8. The backup is always a full backup of the database.

Restore Process:

1. The user kicks of a manual in-place or out-of-place restore of a database to another SQL server.
2. A restore operation restores the .bacpac file to the staging location on the access node.
3. The access node then imports the .bacpac file to the destination server.
4. A restore always restores the full database.
5. The default Pricing Tier Standard S0 will be applied by default to the restored database but this can be changed by using the Advanced restore options.

AZURE SQL MANAGED INSTANCE BACKUP AND RESTORE PROCESS



Azure SQL Managed Instance (export)

Backup Workflow

1. The Commvault SQL agent on the access node uses AD authentication or SQL SERVER authentication to connect to the Azure SQL Managed instance and kick off a COPY_ONLY VDI backup to an Azure blob endpoint that acts as a staging location.
2. The access node then reads the .bak from the staging Azure blob and writes it in Commvault's proprietary, compressed, and de-duped format to the storage destination.

Restore Workflow

3. The access node restores the .bak file to the Azure staging blob.
4. The SQL server is further restored from the .bak file. Restore always restores the full database.

Backup Process:

1. The CommServe starts backup based on the schedule defined in the plan or a user can kick off an ondemand backup.
2. The CommServe sends the backup request to the Commvault SQL agent on the access node.
3. The Commvault SQL agent on the access node uses AD authentication or SQL SRVER authentication to connect to the Azure SQL Managed instance and kick off a COPY_ONLY VDI backup to an azure blob end point that acts as a staging location.
4. Copy-only backups are independent of the sequence of conventional SQL Server backups. This means they don't affect the existing backup chain.
5. The access node then reads the .bak from the staging azure blob and writes it in Commvault proprietary, compressed, and de-duped format to the storage destination.
6. Backup ends after the .bak is successfully written.
7. The backup is always a full backup of the database.

Restore Process:

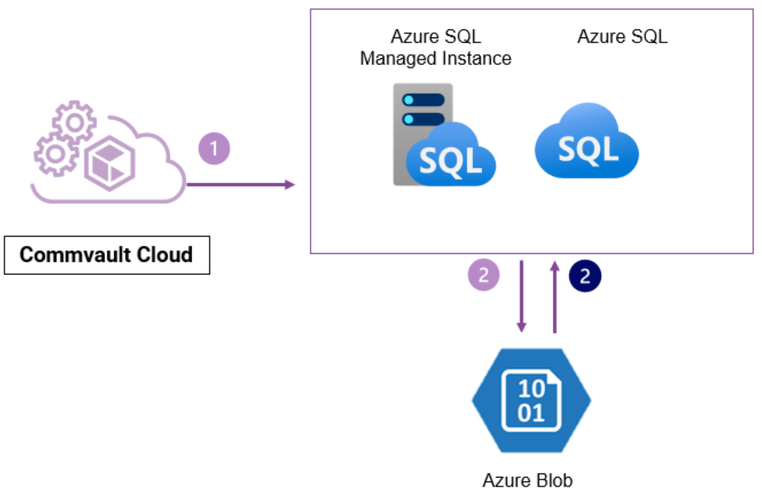
1. The user kicks of a manual in-place or out-of-place restore of a database to another SQL server.
2. The access node restores the .bak file to the Azure staging blob.
3. The SQL server is further restored from the .bak file. Restore always restores the full database.

RESTORE FROM NATIVE BACKUPS FOR AZURE SQL AND AZURE SQL MANAGED INSTANCE

(11.40 Early adopter feature – contact products@commvault.com to request access to this feature.)

Log backups and Long-Term backups for Azure SQL and Azure SQL Managed Instance

How it Works – Architecture



Backup:

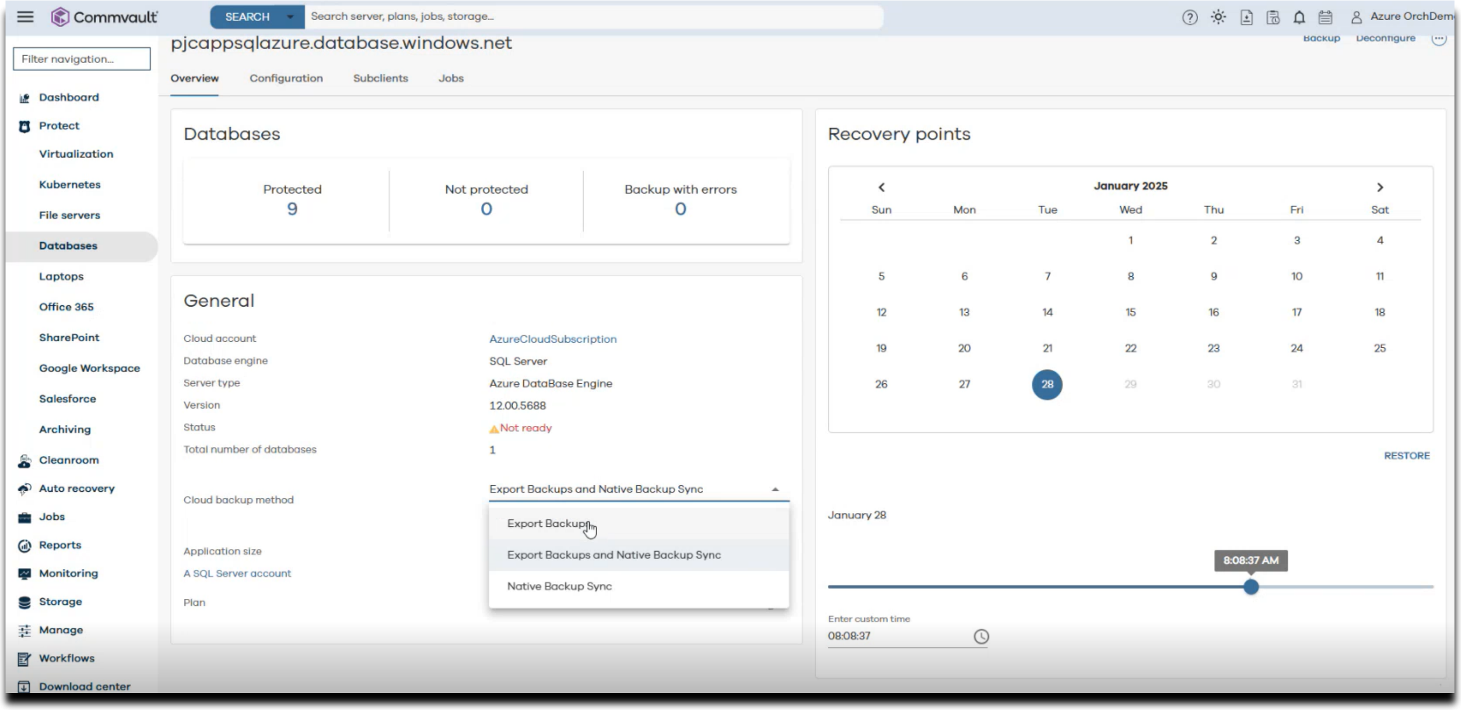
1. Commvault queries for automated log and LITR backups every 24 hours. Also queries for these backups on browse
2. Azure SQL and Managed Instance automated backups are executed automatically by Azure

Restore:

1. Restore job is initiated from the control plane and connects to the access nodes.
2. Commvault initiates the restore using Azure APIs and the point in time. Azure APIs restore the required logs for the recovery (up to the PIT specified by the user).

Configuration:

1. The backup method is set at the instance level **post configuration** if rule-based discovery is configured.
2. The backup method can be set during instance configuration itself if manual discovery is used instead of rule-based auto-discovery.



pjcappsqlazure.database.windows.net

Databases

Protected	Not protected	Backup with errors
9	0	0

General

Cloud account	AzureCloudSubscription
Database engine	SQL Server
Server type	Azure DataBase Engine
Version	12.00.5688
Status	⚠Not ready
Total number of databases	1
Cloud backup method	Export Backups and Native Backup Sync
Application size	
A SQL Server account	
Plan	

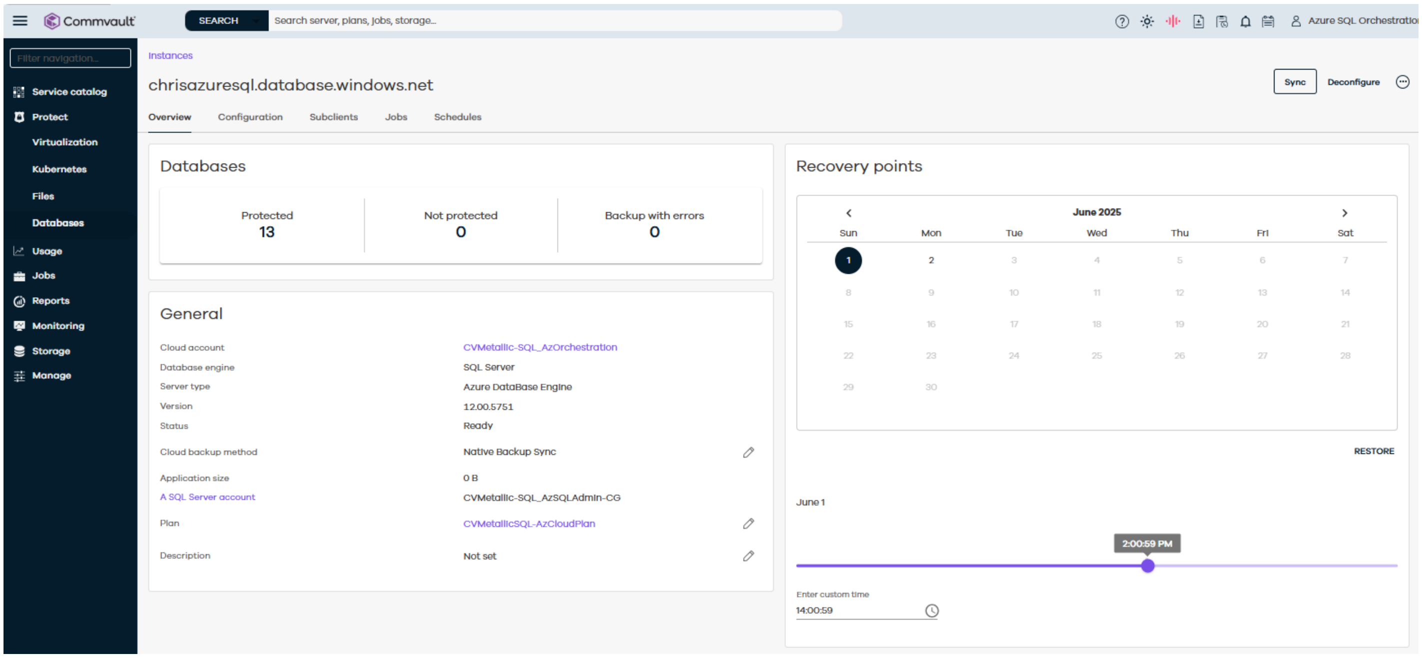
Cloud backup method options: Export Backups; Export Backups and Native Backup Sync; Native Backup Sync

Enter custom time: 08:08:37

Backup Process:

Azure SQL and Managed Instance automated backups are executed automatically by Azure.

Commvault syncs native backups every 24 hours automatically or while browsing in order to do a restore. On-demand sync is also available.



Instances

chrisazuresql.database.windows.net

Databases

Protected	Not protected	Backup with errors
13	0	0

General

Cloud account: CVMetallic-SQL_AzOrchestration
Database engine: SQL Server
Server type: Azure DataBase Engine
Version: 12.00.5751
Status: Ready
Cloud backup method: Native Backup Sync
Application size: 0 B
A SQL Server account: CVMetallic-SQL_AzSQLAdmin-CG
Plan: CVMetallicSQL-AzCloudPlan
Description: Not set

Recovery points

June 2025

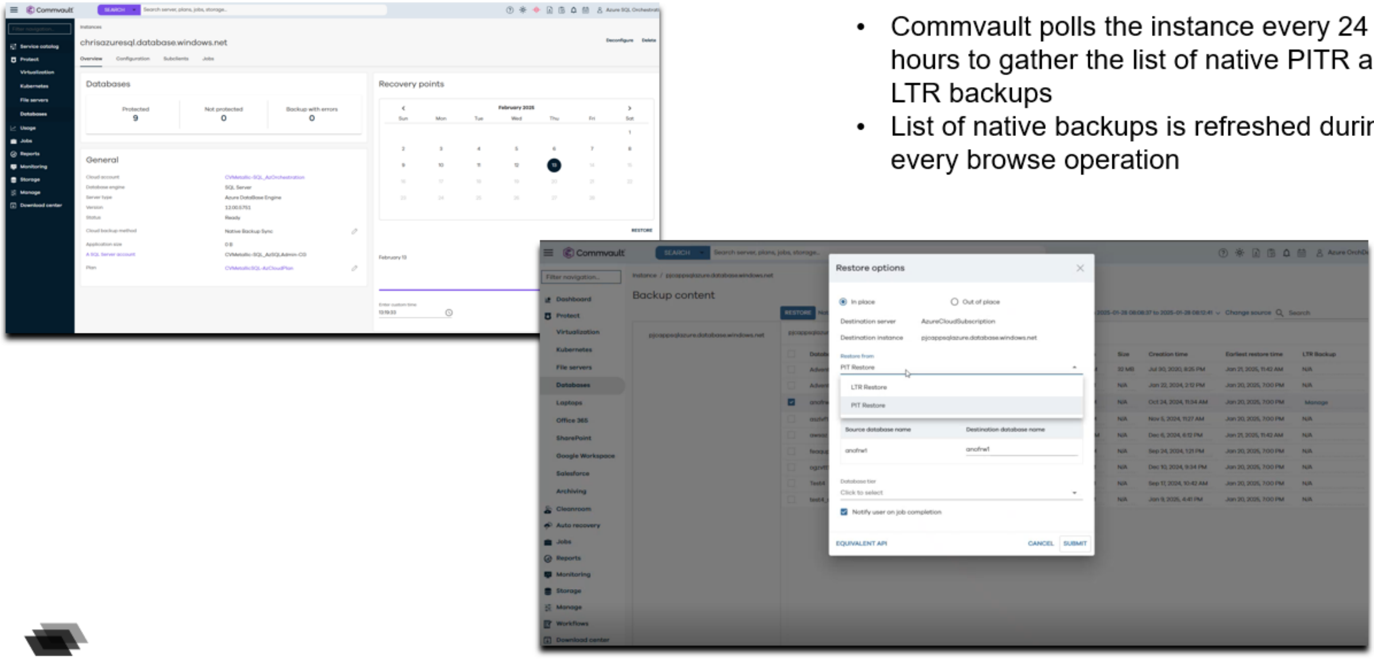
June 1

2:00:59 PM

Enter custom time: 14:00:59

Restore Process:

1. Commvault kicks of the restore of the Azure native backup



- Commvault polls the instance every 24 hours to gather the list of native PITR and LTR backups
- List of native backups is refreshed during every browse operation

Restore options

● In place ○ Out of place

Destination server: AzureCloudSubscription
Destination instance: pjcappsqlazure.database.windows.net

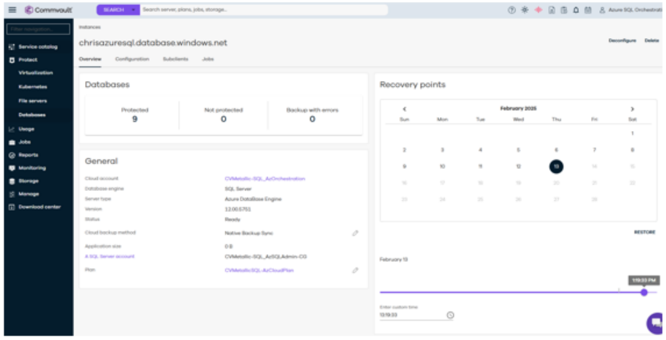
Restore from: PIT Restore

Source database name	Destination database name
anofnw1	anofnw1

Database tier: Click to select

☑ Notify user on job completion

Restore



Instances

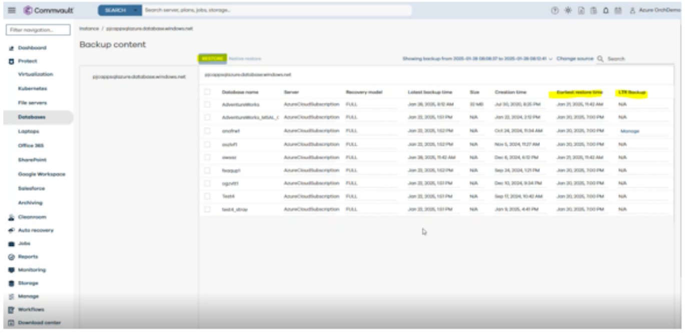
chrisazuresql.database.windows.net

Databases

Protected	Not protected	Backup with errors
9	0	0

General

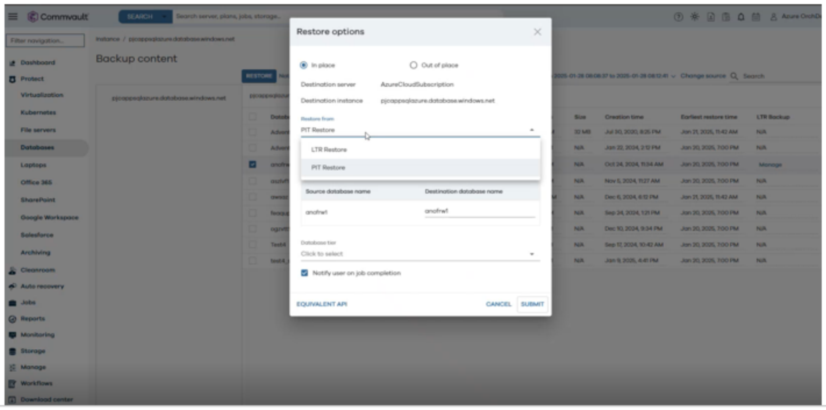
Cloud account	CVMetallic-SQL_AzOrchestration
Database engine	SQL Server
Server type	Azure DataBase Engine
Version	12.00.5751
Status	Ready
Cloud backup method	Native Backup Sync
Application size	0 B
A SQL Server account	CVMetallic-SQL_AzSQLAdmin-CG
Plan	CVMetallicSQL-AzCloudPlan



Backup content

Showing backup from 2025-01-28 08:08:37 to 2025-01-28 08:12:41

Database name	Server	Recovery model	Latest backup time	Size	Creation time	Earliest restore time	LTR Backup
AdventureWorks	AzureCloudSubscription	FULL	Jan 28, 2025, 8:10 AM	32 MB	Jul 30, 2020, 8:25 PM	Jan 21, 2025, 11:42 AM	N/A
AdventureWorks_MSAL_(	AzureCloudSubscription	FULL	Jan 22, 2025, 1:51 PM	N/A	Jan 22, 2024, 2:12 PM	Jan 20, 2025, 7:00 PM	N/A
anofrwl	AzureCloudSubscription	FULL	Jan 22, 2025, 1:52 PM	N/A	Oct 24, 2024, 11:34 AM	Jan 20, 2025, 7:00 PM	Manage
aszlvf1	AzureCloudSubscription	FULL	Jan 22, 2025, 1:52 PM	N/A	Nov 5, 2024, 11:27 AM	Jan 20, 2025, 7:00 PM	N/A
dwssz	AzureCloudSubscription	FULL	Jan 28, 2025, 11:42 AM	N/A	Dec 6, 2024, 6:12 PM	Jan 21, 2025, 11:42 AM	N/A
feaqup1	AzureCloudSubscription	FULL	Jan 22, 2025, 1:52 PM	N/A	Sep 24, 2024, 1:21 PM	Jan 20, 2025, 7:00 PM	N/A
ogzvtt1	AzureCloudSubscription	FULL	Jan 22, 2025, 1:51 PM	N/A	Dec 10, 2024, 9:34 PM	Jan 20, 2025, 7:00 PM	N/A
Test4	AzureCloudSubscription	FULL	Jan 22, 2025, 1:51 PM	N/A	Sep 17, 2024, 10:42 AM	Jan 20, 2025, 7:00 PM	N/A
test4_stray	AzureCloudSubscription	FULL	Jan 22, 2025, 1:51 PM	N/A	Jan 9, 2025, 4:41 PM	Jan 20, 2025, 7:00 PM	N/A



Restore options

● In place ○ Out of place

Destination server: AzureCloudSubscription

Destination instance: pjcappsqlazure.database.windows.net

Restore from: PIT Restore

Source database name	Destination database name
anofrwl	anofrwl

Database tier: Click to select

☑ Notify user on job completion

AZURE SQL BACKUP METRICS

The purpose of the following tests is to guide the user to the optimal database tier for their Azure SQL databases such that backups can meet the required SLAs without impacting the performance of the production database.

TEST LAB ENVIRONMENT

1. Database instance, access node and Media agent were deployed in the same Azure region.
2. Network speeds used for testing were at least 10Gbps between all Commvault infrastructure components and the DB instance.
3. Backups were stored on destination storage that was Azure Blob storage (hot tier) that was configured in the same region as the production Azure SQL Database.
4. Access Node and Mediagent were installed on the same Azure VM.
5. Access Node sizing – Linux Access Node, 8vCPUs, 32GB RAM
6. Access node had staging space = 20% of total DB workload set aside.

CAVEATS

- The results shown below demonstrate the performance of Commvault within a set environment to show what results are achievable within similar environments.
- Due to the variations in all environments, results will be varied compared to the results achieved within this paper. While some environments can and will achieve better performance, other environments may not match the performance metrics captured in this paper.
- This paper is not meant to show the maximum or minimum performance of the solution, but a snapshot of performance based on conditions used in this testing. This paper, including any results or statements herein, does not guarantee or warrant performance.

TEST PLAN AND RESULTS

Database size	vCores for PaaS SQL Instance Database tiers (General purpose, Gen5, serverless, V5)	Total Backup Time (hh:mm:ss)	Cost of Copy ($)
50GB	2	0:59:45	$0.50
50GB	4	0:32:26	$0.55
50GB	8	0:32:09	$1.08
50GB	16	0:32:13	$2.17
100GB	2	1:02:36	$0.53
100GB	4	1:01:43	$1.04
100GB	8	1:01:47	$2.08
100GB	16	1:01:27	$4.13
500GB	2	7:35:17	$3.83
500GB	4	7:26:22	$7.50
500GB	16	7:23:51	$29.85
1TB	4	14:20:00	$14.46

INFERENCE FROM TEST RESULTS

- Comparable Copy, export and write to media times are seen when vCores are increased beyond 4 vCores.

PERFORMANCE CONSIDERATIONS FOR AZURE SQL BACKUPS

- The copy is created with the same pricing/sizing tier as the production database.
- Performance data collected from customer environments has shown that having the production database run on a minimum of 4 cores helps meet the backup SLA while keeping costs down for database sizes of up to 500GB.
- For the best performance Azure SQL DB, access node, media agent and destination storage should be in the same region to avoid network latency.
- Copy and export operations are managed by Azure and hence there is not much that we can control in terms of performance for those operations from the Commvault end.
- Time for export also depends on the actual data being exported.
- Performance of write to destination storage/media is already tweaked to perform optimally for most use cases. For further fine tuning refer to:

  https://documentation.commvault.com/2023e/expert/improving_throughput_to_storage_media.html
- Please check this link to see limitations and restrictions with DacFX.
- Exporting a database using DacFX causes throttling by the Azure SQL Database service. You can view the DTU stats for the database on the Azure portal. If the database has reached its resource limits, upgrade the service tier to add more resources.
- Exporting large tables without clustered indexes can be very slow or even cause failure. This behavior occurs because the table can't be split up and exported in parallel. Instead, it must be exported in a single transaction, and that causes slow performance and potential failure during export, especially for large tables.
- Sizing of the staging location – Access nodes processing import/export requests need to store the BACPAC file as well as temporary files generated by the Data-Tier Application Framework (DacFX). The disk space required varies significantly among databases with the same size and can require disk space up to three times the size of the database. As a result, some requests may fail with the error “There is not enough space on the disk.” The workaround is to increase the staging space on the access node. It is recommended that this staging space be created on SSD disks for optimal performance.
- For more information - https://documentation.commvault.com/v11/essential/recommendations_for_improving_performance_of_azure_sql_backups_and_restores.html
- TEMP/TMP system variables - If by chance you receive a failing with Out of Disk space message, it's advisable to configure the %TEMP% folder of the system to reside on a distinct data disk. By doing so, you can confirm sufficient space for the export process to execute smoothly, avoiding potential disk space complications.

  To configure the system's %TEMP% folder:

  - Open the System Properties window by pressing the Windows key + Pause/Break or right-clicking on This PC and selecting Properties.
  - Select the link labeled Advanced system settings on the left-hand side.
  - In the ensuing System Properties window, navigate to the bottom and select Environment Variables.

  - Under the section labeled System Variables, locate the **TEMP and TMP variables**, then select Edit associated **with each.**
  - Modify the values of both variables to point to a pathway on the separate data disk you have established. For instance, if your data disk is designated as D:, set the values as D:\Temp.
  - Confirm the changes by selecting OK and closing all open windows.
- Restore operation restores the data to the lowest pricing tier by default. This may lead to slow restores. Use the UI options to select the pricing tier for the restored database such that required restore RTO is met.

AZURE SQL MANAGED INSTANCE BACKUP METRICS

Operation	Access Node Specs	Database size	Managed Instance Compute Specs	Total time to complete	Notes
Backup	2vCPU / 16GB RAM	1.3TB	General Purpose Standard-series (Gen 5) 8 vCores	2 hrs 16 mins 1 second	TDE Disabled
Restore	2vCPU / 16GB RAM	1.3TB	General Purpose Standard-series (Gen 5) 8 vCores	1 hr 55 mins 29 seconds	TDE Disabled
Backup	2vCPU / 16GB RAM	1.3TB	General Purpose Standard-series (Gen 5) 8 vCores	3 hr 2 mins 48 seconds	CMK TDE Enabled
Restore	2vCPU / 16GB RAM	1.3TB	General Purpose Standard-series (Gen 5) 8 vCores	3 hr 20 mins 14 seconds	CMK TDE Enabled

SECURITY & COMPLIANCE

Encryption Matrix

Encryption Type	Azure SQL DB	Azure SQL MI
System-Managed TDE Keys	**Supported**	✗ Not supported
Customer-Managed TDE Keys (CMK)	Supported	**Required for backups**

When an MI instance is converted from System-Managed Keys to CMK, older backups taken under SMK remain restorable. Commvault has validated this scenario.

Data-in-Flight Protection

- **Export-based backups (Azure SQL DB)** – Gateway uploads the .bacpac to Commvault storage over **HTTPS/TLS 1.2**.
- **COPY_ONLY backups (Azure SQL MI)** – SQL Server writes the .bak to customer Blob over HTTPS; the subsequent copy from Blob to Commvault storage also uses HTTPS.
- **Native-backup orchestration** – Azure exposes backups over secure REST; Commvault ingests only metadata.

SUMMARY

Commvault's Azure SQL protection safeguards your Azure SQL data with flexible retention policies, flexible scheduling policies and hybrid cloud mobility. It integrates with Azure SQL and allows for data availability and integrity for your critical databases.

APPENDIX

Access the Commvault Design tool to size for large Azure SQL estates.

https://cloud.commvault.com/commandcenter -> tools -> Commvault Solution Design Tool.

To learn more, visit **commvault.com**