Backup and restore options for SQL Server on Amazon EC2

AWS Prescriptive Guidance

AWS Prescriptive Guidance: Backup and restore options for SQL Server on Amazon EC2

Table of Contents

Introduction **1**
Server-level backup options **2**
SQL Server backup using VSS-enabled EBS snapshots 2
SQL Server backup using AWS Backup 4
Database-level backup options **7**
SQL Server native backup and restore to Amazon S3 using AWS Storage Gateway 7
SQL Server native backup to EBS volumes 9
SQL Server native backup to Amazon FSx for Windows File Server 10
Backup option comparison **13**
Conclusion **15**
Resources **16**
AWS services 16
Amazon EBS snapshots 16
AWS Backup 16
AWS Storage Gateway 16
Amazon FSx 17
AWS Prescriptive Guidance 17
Microsoft documentation 17

Backup and restore options for SQL Server on Amazon EC2

Yogi Barot and Reghardt van Rooyen, Amazon Web Services

Customers have asked for the right solution to safeguard their data on Microsoft SQL Server on Amazon Elastic Compute Cloud (Amazon EC2) and meet their current requirements for Recovery Point Objective (RPO), the maximum acceptable amount of time since the last backup, and Recovery Time Objective (RTO), the maximum acceptable delay between the interruption of service and restoration of service. When you are running SQL Server on EC2 instances, you have multiple options for creating backups of the data and also restoring it.

Backup strategies for safeguarding data for SQL Server on Amazon EC2 include the following:

- Server-level backup using Windows Volume Shadow Copy Service (VSS)-enabled Amazon Elastic Block Store (Amazon EBS) snapshots or AWS Backup
- Database-level backup using native backup and restore

When you choose database-level native backup, you have the following storage options:

- An Amazon EBS volume
- An Amazon FSx for Windows File Server file system
- Amazon Simple Storage Service (Amazon S3), using AWS Storage Gateway

This guide compares these options, including the benefits and limitations of each. It will also compare the performance of each option for a sample 1 TB database.

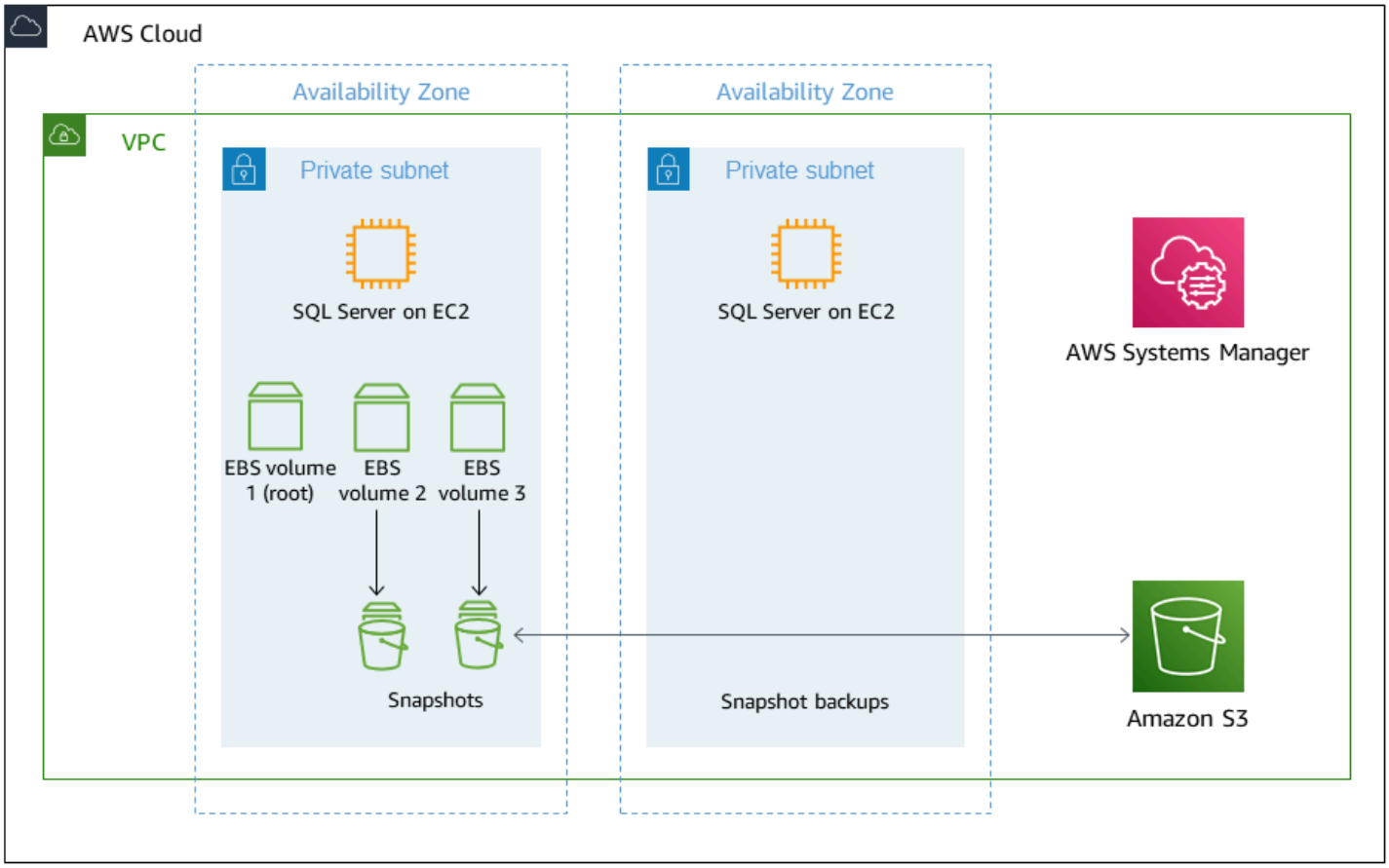
Server-level backup options

Server-level backup covers backing up your whole SQL Server EC2 instance, which includes all system and user databases on the EC2 instance. If you have multiple SQL Server instances on Amazon EC2, that is also included in the server-level backup.

SQL Server backup using VSS-enabled EBS snapshots

Windows Volume Shadow Copy Service (VSS) orchestrates operating system and application access to underlying inputs and outputs (I/O), enabling an application-consistent snapshot of underlying storage without any application down-time. SQL Server is VSS-aware, so you can use VSS agent to take EBS snapshots of SQL Server instances.

The following diagram shows the architecture of a backup and restore solution using VSS-enabled EBS snapshots.



This architecture uses the AWS Systems Manager Run Command to install the VSS agent on your SQL Server instances. You can also use the Run Command to invoke the entire workflow of flushing

operating system and application buffers to the disk, pausing I/O operations, taking a point-in-time snapshot of the EBS volumes, and then resuming I/O.

This Run Command creates automated snapshots of all EBS volumes attached to a target instance. You also have the option to exclude the root volume, because user database files are usually stored on other volumes. In case you stripe multiple EBS volumes to create a single file system for SQL Server files, Amazon EBS also supports crash-consistent multivolume snapshots using a single API command.

The process for taking application-consistent, VSS-enabled EBS snapshots is covered in the AWS documentation.

Using this method provides the following benefits:

- The first snapshot of a DB instance contains the data for the full DB instance. Subsequent snapshots of the same DB instance are incremental, which means that only the data that has changed after your most recent snapshot is saved.
- You can restore to a new SQL Server EC2 instance from a snapshot.
- If an instance is encrypted using Amazon EBS or if a database is encrypted in the instance using transparent data encryption (TDE), that instance or database is automatically restored with the same encryption.
- You can copy your automated cross-Region backups.
- When you restore an EBS volume from a snapshot, it becomes immediately available for applications to access it. This means that you can immediately bring SQL Server online after restoring one or more of its underlying EBS volumes from snapshots.
- By default, restored volumes fetch underlying blocks from Amazon S3 the first time an application tries to read them. This means that there can be a lag in performance after an EBS volume is restored from a snapshot. The volume eventually catches up with the nominal performance. However, you can avoid that lag by using fast snapshot-restore (FSR) snapshots. This feature enables you to get full performance immediately after restoring a volume.
- You can use life cycle management for EBS snapshots.

However, using automated snapshots has certain limitations:

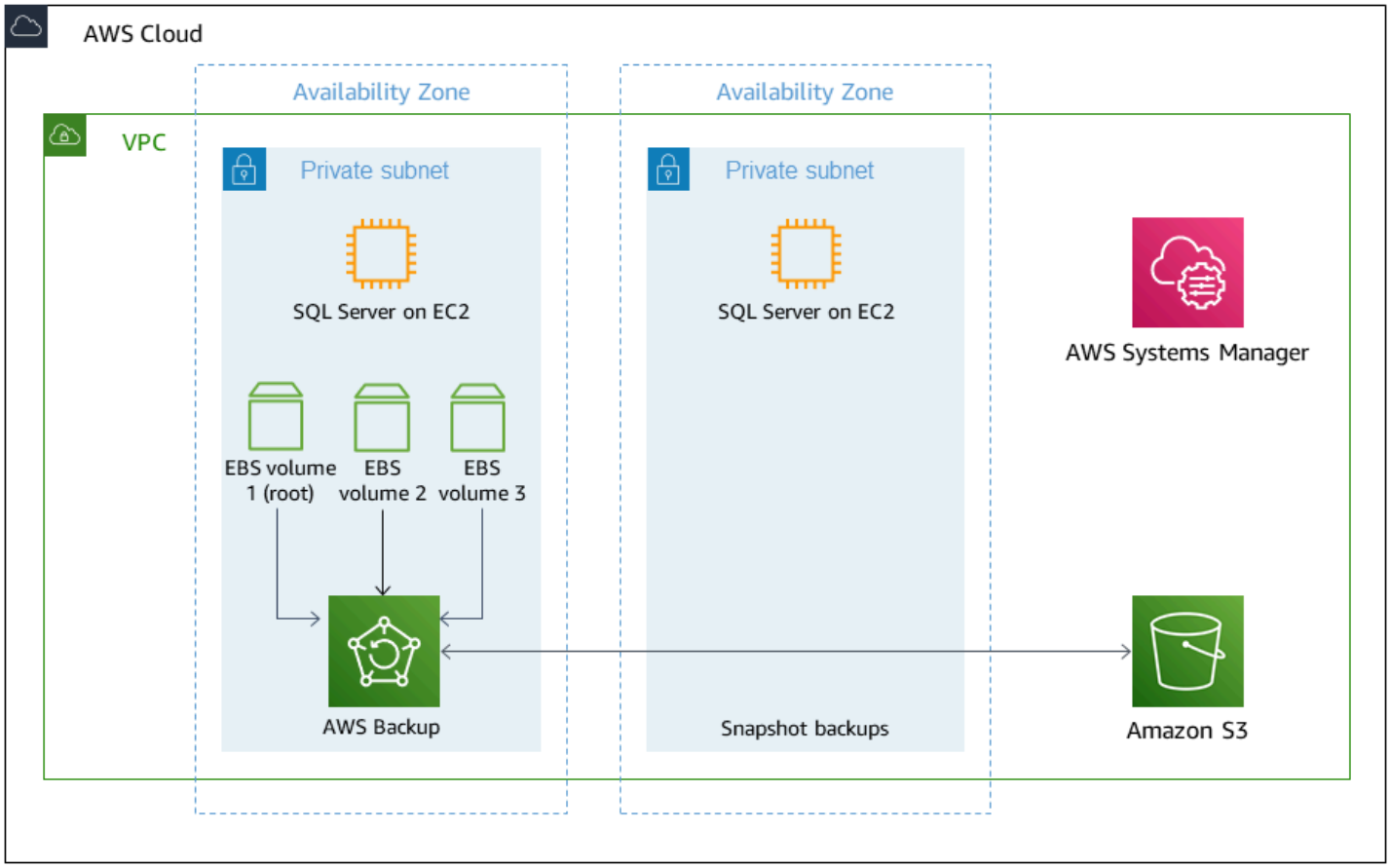
- You can't perform cross-Region point-in-time recovery (PITR) with an encrypted snapshot for a SQL Server instance.

- You can't create an encrypted snapshot of an unencrypted instance.
- You can't restore an individual database because the snapshot is taken at the EBS volume level.
- You can't restore the instance to itself.
- A snapshot of the DB instance must be encrypted using the same AWS Key Management System (AWS KMS) key as the DB instance.
- Storage I/O is suspended for a fraction of a second (approximately 10 milliseconds) during the snapshot backup process.

SQL Server backup using AWS Backup

Using AWS Backup, you can centralize and automate data protection across AWS services. AWS Backup offers a cost-effective, fully managed, policy-based service that further simplifies data protection at scale. AWS Backup also helps you support your regulatory compliance obligations and meet your business continuity goals. Together with AWS Organizations, AWS Backup enables you to centrally deploy data protection (backup) policies to configure, manage, and govern your backup activity across your organization's AWS accounts and resources.

The following diagram shows the architecture of a backup and restore solution for SQL Server on EC2 using AWS Backup.



This solution has the following benefits:

- You can automate backup scheduling, retention management, and lifecycle management.
- You can centralize your backup strategy across your organization, spanning multiple accounts and AWS Regions.
- You can centralize monitoring your backup activity and alerting across AWS services.
- You can implement cross-Region backups for disaster recovery planning.
- The solution supports cross-account backups.
- You can perform secure backups with secondary backup encryption.
- All backups support encryption using AWS KMS encryption keys.
- The solution works with TDE.
- You can restore to a specific recovery point from the AWS Backup console.
- You can back up an entire SQL Server instance, which includes all SQL Server databases.

Using AWS Backup has certain limitations:

- You can't perform native SQL backup and restore.
- You can't do cross-Region point-in-time recovery (PITR).
- You must enable the EC2 instance for Systems Manager access.
- You must configure the instance with VSS and enable VSS backups in the backup job.

For more information about using AWS Backup for SQL Server backups, see the blog post How to simplify Microsoft SQL Server backup using AWS Backup and VSS.

For information about using AWS Backup to do point-in-time recovery for Amazon RDS for SQL Server, see the blog post Point-in-time recovery and continuous backup for Amazon RDS with AWS Backup.

Database-level backup options

These approaches use native Microsoft SQL Server backup functionality. You can take backups of individual databases on the SQL Server instance and restore an individual database.

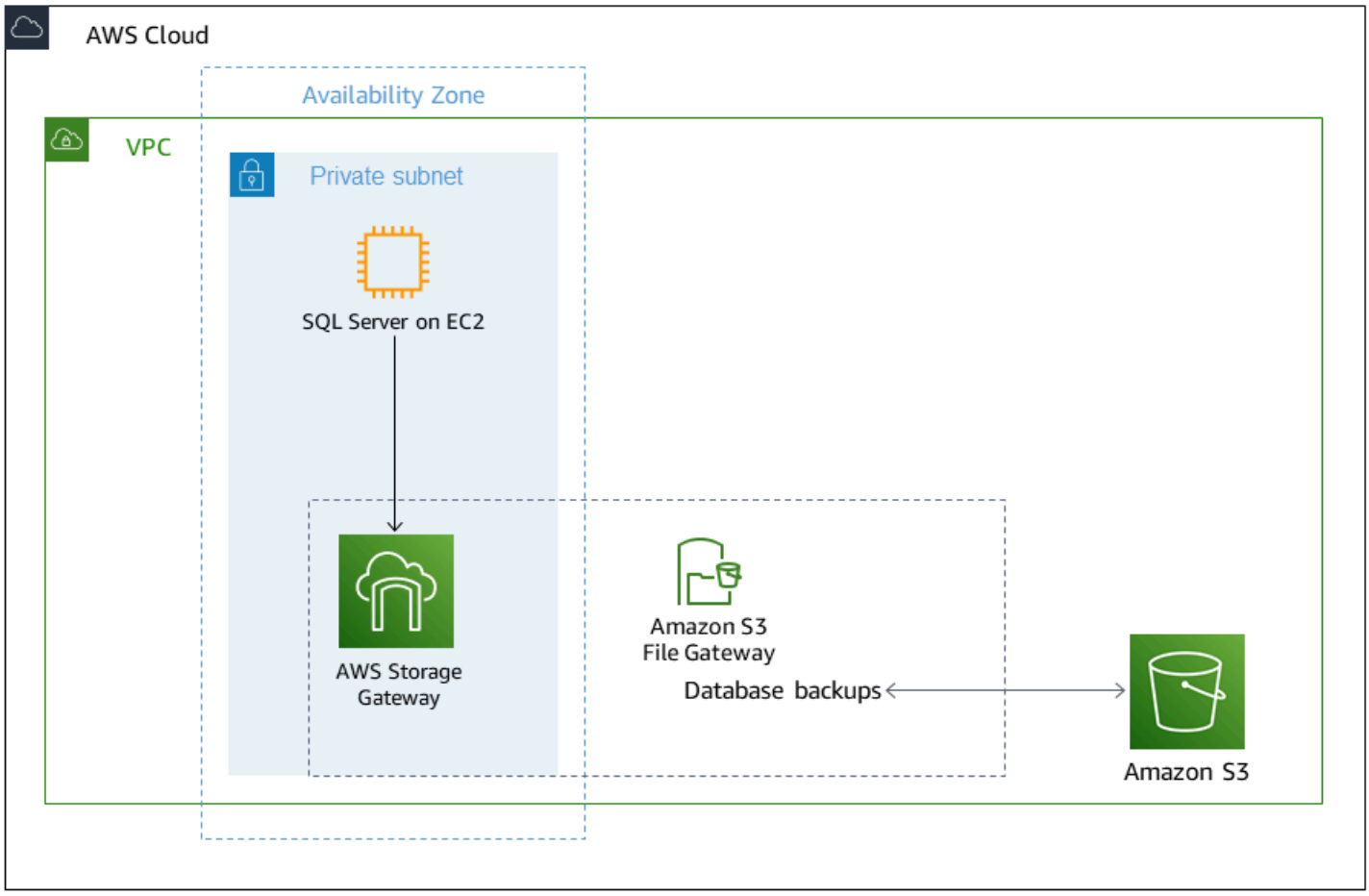
Each of these options for native SQL Server backup and restore also support the following:

- Compression and multiple-file backup
- Full, differential, and T-log backups
- Transparent data encryption (TDE)-encrypted databases

SQL Server native backup and restore to Amazon S3 using AWS Storage Gateway

SQL Server on EC2 supports native backup and restore for SQL Server databases. You can take a backup of your SQL Server database and then restore the backup file to an existing database or to a new SQL Server EC2 instance, Amazon RDS for SQL Server, or an on-premises server. You can use AWS Storage Gateway to back up your Microsoft SQL Server databases directly to Amazon S3, reducing your on-premises storage footprint and using Amazon S3 for durable, scalable, and cost-effective storage. Storage Gateway is a hybrid cloud storage service that provides on-premises applications with access to virtually unlimited cloud storage.

The following diagram shows the architecture of a native backup and restore solution using Storage Gateway and Amazon S3.



Native SQL Server backup with Storage Gateway provides the following benefits:

- You can map a storage gateway as a Server Message Block (SMB) file share on the EC2 instance and send the backup to Amazon S3.
- The backup occurs directly to the S3 bucket or through the Storage Gateway file cache.
- Multiple-file backups are supported.

Native backup using Storage Gateway has the following limitations:

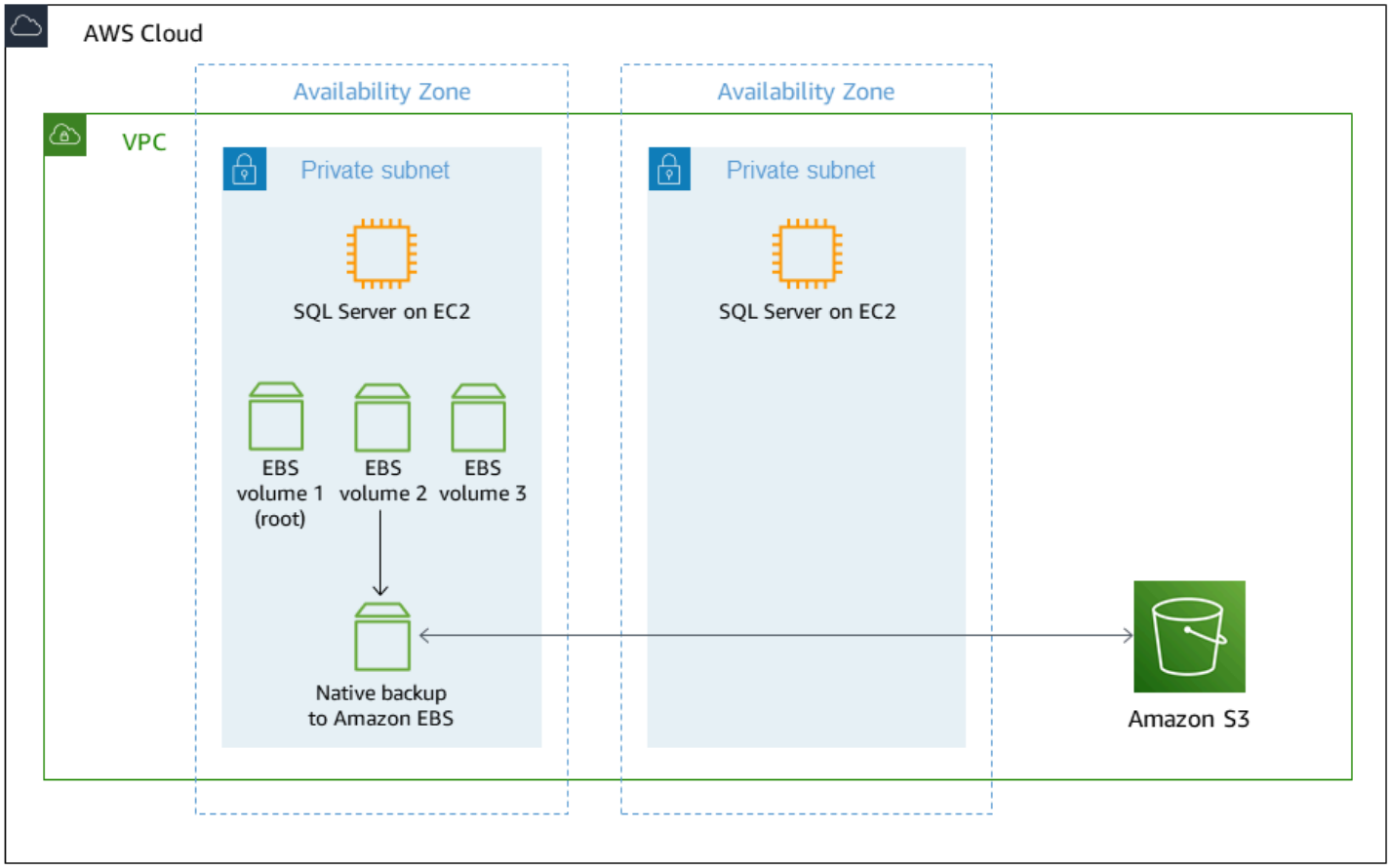
- You must set up backup and restore for each individual database.
- You must manage the Amazon S3 lifecycle policy for the backup files.

For more information about how to set up Storage Gateway, see Store SQL Server backups in Amazon S3 using AWS Storage Gateway.

SQL Server native backup to EBS volumes

You can take a native backup of your SQL Server database and store the file in an Amazon Elastic Block Store (Amazon EBS) volume. Amazon EBS is a highly performant block storage service. EBS volumes are elastic, which supports encryption, and they can be detached and attached to an EC2 instance. SQL Server on an EC2 instance can be backed up on same EBS volume type or on a different EBS volume type. One advantage of backing up to a different EBS volume is cost savings.

The following diagram shows the architecture of a native backup to an EBS volume.



SQL Server native backup to EBS volumes provides the following benefits:

- You can take backups of individual databases on a SQL Server EC2 instance and restore an individual database instead of having to restore the complete instance.
- Multiple-file backups are supported.
- You can schedule backup jobs by using SQL Server Agent and the SQL Server job engine.

- You can get performance benefits through your hardware choices. For example, you can use st1 storage volumes to achieve higher throughput.

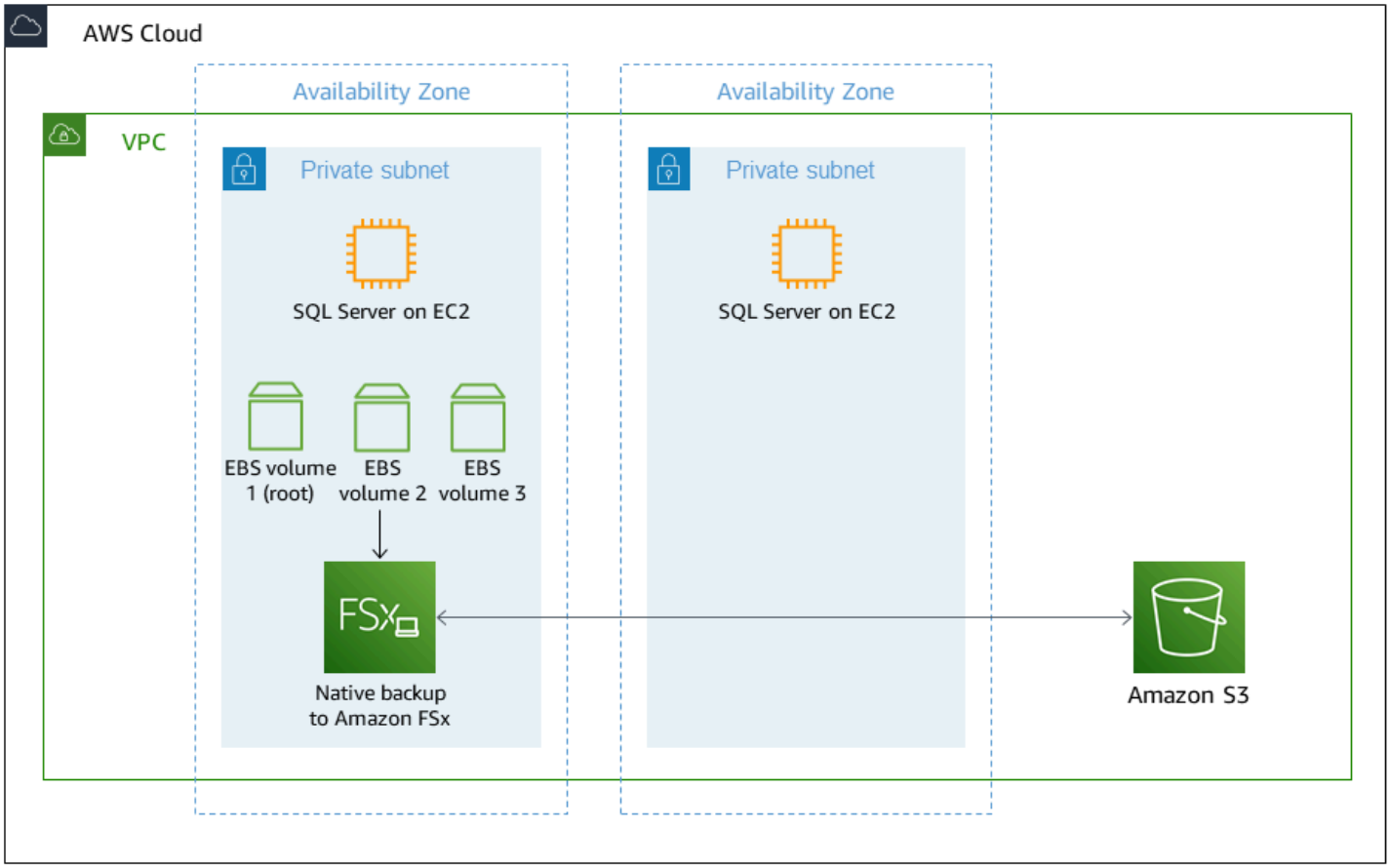
When using native backup to EBS volumes, consider the following points:

- You must manually move backups to Amazon S3 from the EBS volume.
- For large backups, you must manage disk space on EC2.
- On the EC2 instance, Amazon EBS throughput can be a bottleneck.
- Additional storage is required to store backups on Amazon EBS.

SQL Server native backup to Amazon FSx for Windows File Server

Amazon FSx for Windows File Server is a fully managed native Windows file system that offers up to 64 TB of storage designed to deliver fast, predictable, and consistent performance. AWS introduced native support for Multi-AZ file system deployments on FSx for Windows File Server. Native support makes it easier to deploy Windows file storage on AWS with high availability and redundancy across multiple Availability Zones. Along with this launch, AWS also introduced support for SMB Continuously Available (CA) file shares. You can use FSx for Windows File Server as backup storage for a SQL Server database.

The following diagram shows the architecture of a native SQL Server backup to FSx for Windows File Server.



This approach has the following benefits:

- You can back up your SQL Server database to an Amazon FSx file share.

- You can take backups of individual databases on a SQL Server instance and restore an individual database instead of having to restore the complete instance.
- Multiple-part backups are supported.
- You can schedule backup jobs by using SQL Server Agent and the job engine.
- The instances have higher network bandwidth, compared with Amazon EBS.

Native backup to FSx for Windows File Server has the following limitations:

- You must manually move backups to Amazon S3 from Amazon FSx by using AWS Backup or AWS DataSync.
- Large backups might require additional overhead for disk space management on Amazon FSx.
- EC2 instance network throughput can be a bottleneck.

- Additional storage is required to store backups on FSx for Window File Server.

Backup option comparison

We completed performance benchmark testing on the approaches covered in this guide. The tests were performed with an r5d.8xlarge instance and 1TB SQL Server database as the source. The source system was configured according to best practices, and the source database contained four data files (250 GB each) and one log file (50 GB) spread across separate GP3 volumes. The SQL Server native BACKUP command included writing to 10 backup files, using compression to optimize backup performance and reduce the amount of data sent across the network and written to the target. In all test cases, the source database or target backup option, storage performance was the bottleneck.

There is an almost endless variety of possible configurations for these types of test. We took care to optimize for performance, cost, scalability, and real-world use cases. The backup target options with details and performance are included in the following table.

Backup options	Run duration	Backup rate
Native backup to local EBS volume st1 HDD 2 TB st1 500 IOPS 500 MiB/s	00:30:46	554.7 Mbps
Native backup to FSx for Windows File Server HDD 10 TB at 512 Mbps throughput	00:20:58	814.0 Mbps
Native backup to Amazon S3 File Gateway m6i.4xlarge (16 vCPU, 64 GB) with 2 TB gp3	00:23:20	731.5 Mbps
VSS-enabled EBS snapshot	00:00:53	Not applicable (snapshot)
AWS Backup AMI	00:08:00	Not applicable (snapshot)

Based on our testing, native SQL Server database backups to FSx for Windows File Server is the fastest option, while backups to Storage Gateway and locally attached EBS volumes are more cost-efficient with slower performance. For server-level backups (AMI), AWS Backup should be used for optimal performance, cost, and manageability.

Conclusion

Understanding the possible solutions for backing up SQL Server on Amazon EC2 is key to safeguarding your data and insuring you meet your backup needs, as well as helping to ensure that you can recover from critical events.

In this guide, we discussed different ways to backup and restore your SQL Server instances and database. This can help you devise a backup and restore strategy that protects your data and meets your company's requirements.

For information about migrating SQL Server workloads to SQL Server on Amazon EC2, see Migrating Microsoft SQL Server Enterprise Workloads to Amazon EC2. You can also review the best practicesfor working with SQL Server on Amazon EC2.

Resources

AWS services

- AWS Backup
- AWS DataSync
- Amazon EBS
- Amazon FSx for Windows File Server
- AWS Organizations
- Amazon S3
- AWS Storage Gateway

Amazon EBS snapshots

- Create a VSS application-consistent snapshot
- Take Microsoft VSS-Enabled Snapshots Using Amazon EC2 Systems Manager (blog post)
- How incremental snapshots work
- Automate cross-account snapshot copies
- Amazon EBS fast snapshot restore
- Lifecycle Management for Amazon EBS Snapshots (blog post)

AWS Backup

- How to simplify Microsoft SQL Server backup using AWS Backup and VSS (blog post)
- Point-in-time recovery and continuous backup for Amazon RDS with AWS Backup (blog post)

AWS Storage Gateway

- Managing your storage lifecycle
- Store SQL Server backups in Amazon S3 using AWS Storage Gateway (blog post)

Amazon FSx

- Amazon FSx For Windows File Server Update – Multi-AZ, & New Enterprise-Ready Features (blog post)

AWS Prescriptive Guidance

- Migrating Microsoft SQL Server databases to the AWS Cloud (guide)
- Restoring from an Amazon EBS snapshot or an AMI (guide)

Microsoft documentation

- Windows Volume Shadow Copy Service (VSS)
- Back Up and Restore of SQL Server Databases