Backup and Recovery Approaches Using AWS

June 2016

Notices

This document is provided for informational purposes only. It represents AWS's current product offerings and practices as of the date of issue of this document, which are subject to change without notice. Customers are responsible for making their own independent assessment of the information in this document and any use of AWS's products or services, each of which is provided "as is" without warranty of any kind, whether express or implied. This document does not create any warranties, representations, contractual commitments, conditions or assurances from AWS, its affiliates, suppliers or licensors. The responsibilities and liabilities of AWS to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.

Contents

Abstract

This paper is intended for enterprise solution architects, backup architects, and IT administrators who are responsible for protecting data in their corporate IT environments. It discusses production workloads and architectures that can be implemented using AWS to augment or replace a backup and recovery solution. These approaches offer lower costs, higher scalability, and more durability to meet Recovery Time Objective (RTO), Recovery Point Objective (RPO), and compliance requirements.

Introduction

As the growth of enterprise data accelerates, the task of protecting it becomes more challenging. Questions about the durability and scalability of backup methods are commonplace, including this one: How does the cloud help meet my backup and archival needs?

This paper covers a number of backup architectures (cloud-native applications, hybrid and on-premises environments) and associated AWS services that can be used to build scalable and reliable data-protection solutions.

Why Use AWS as a Data-Protection Platform?

Amazon Web Services (AWS) is a secure, high-performance, flexible, cost-effective, and easy-to-use cloud computing platform. AWS takes care of the undifferentiated heavy lifting and provides tools and resources you can use to build scalable backup and recovery solutions.

There are many advantages to using AWS as part of your data protection strategy:

- **Durability:** Amazon Simple Storage Service (Amazon S3) and Amazon Glacier are designed for 99.999999999% (11 nines) of durability for the objects stored in them. Both platforms offer reliable locations for backup data.

- **Security:** AWS provides a number of options for access control and encrypting data in transit and at rest.
- **Global infrastructure:** AWS services are available around the globe so you can back up and store data in the region that meets your compliance requirements.
- **Compliance:** AWS infrastructure is certified for compliance with standards such as Service Organization Controls (SOC), Statement on Standards for Attestation Engagements (SSAE) 16, International Organization for Standardization (ISO) 27001, Payment Card Industry Data Security Standard (PCI DSS), Health Insurance Portability and Accountability Act (HIPPA), SEC[1], and Federal Risk and Authorization Management Program (FedRAMP) so you can easily fit the backup solution into your existing compliance regimen.
- **Scalability:** With AWS, you don't have to worry about capacity. You can scale your consumption up or down as your needs change without administrative overhead.
- **Lower TCO:** The scale of AWS operations drives down service costs and helps lower the total cost of ownership (TCO) of the storage. AWS passes these cost savings on to customers in the form of price drops.
- **Pay-as-you-go pricing:** Purchase AWS services as you need them and only for the period you plan to use them. AWS pricing has no upfront fees, termination penalties, or long-term contracts.

AWS Storage Services for Data Protection

Amazon S3 and Amazon Glacier are ideal services for backup and archival. Both are durable, low-cost storage platforms. Both offer unlimited capacity and require no volume or media management as backup data sets grow. The pay-for-what-you-use model and low cost per GB/month make these services a good fit for data protection use cases.

[1] https://aws.amazon.com/about-aws/whats-new/2015/09/amazon-glacier-receives-third-party-compliance-assessment-for-sec-rule-17a-4f-from-cohasset-associates-inc/

Amazon S3

Amazon S3 provides highly secure, scalable object storage.

You can use Amazon S3 to store and retrieve any amount of data, at any time, from anywhere on the web. Amazon S3 stores data as objects within resources called *buckets*. AWS Storage Gateway and many third-party backup solutions can manage Amazon S3 objects on your behalf. You can store as many objects as you want in a bucket, and you can write, read, and delete objects in your bucket. Single objects can be up to 5 TB in size.

Amazon S3 offers a range of storage classes designed for different use cases. These include:

- **Amazon S3 Standard** for general-purpose storage of frequently accessed data.
- **Amazon S3 Standard - Infrequent Access** for long-lived, but less frequently accessed data.
- **Amazon Glacier** for long-term archive.

Amazon S3 also offers lifecycle policies you can configure to manage your data throughout its lifecycle. After a policy is set, your data will be migrated to the appropriate storage class without any changes to your application. For more information, see S3 Storage Classes.

Amazon Glacier

Amazon Glacier is an extremely low-cost, cloud archive storage service that provides secure and durable storage for data archiving and online backup. To keep costs low, Amazon Glacier is optimized for data that is infrequently accessed and for which retrieval times of several hours are acceptable. With Amazon Glacier, you can reliably store large or small amounts of data for as little as $0.007 per gigabyte per month, a significant savings compared to on-premises solutions. Amazon Glacier is well suited for storage of backup data with long or indefinite retention requirements and for long-term data archiving. For more information, see Amazon Glacier.

AWS Storage Gateway

AWS Storage Gateway connects an on-premises software appliance with cloud-based storage to provide seamless and highly secure integration between your on-premises IT environment and the AWS storage infrastructure. For more information, see AWS Storage Gateway.

AWS Transfer Services

In addition to third-party gateways and connectors, you can use AWS options like AWS Direct Connect, AWS Snowball, AWS Storage Gateway, and Amazon S3 Transfer Acceleration to quickly transfer your data. For more information, see Cloud Data Migration.

Designing a Backup and Recovery Solution

When you develop a comprehensive strategy for backing up and restoring data, you must first identify the failure or disaster situations that can occur and their potential business impact. In some industries, you must consider regulatory requirements for data security, privacy, and records retention.

You should implement backup processes that will offer the appropriate level of granularity to meet the RTO and RPO of the business, including:

- File-level recovery
- Volume-level recovery
- Application-level recovery (for example, databases)
- Image-level recovery

The following sections describe backup, recovery, and archive approaches based on the organization of your infrastructure. IT infrastructure can broadly be categorized as cloud native, on-premises, and hybrid.

Cloud-Native Infrastructure

This scenario describes a workload environment that exists entirely on AWS. As the following figure shows, it includes web servers, application servers, monitoring servers, databases, and Active Directory.

If you are running all of your services from AWS, you can leverage many built-in features to meet your data protection and recovery needs.

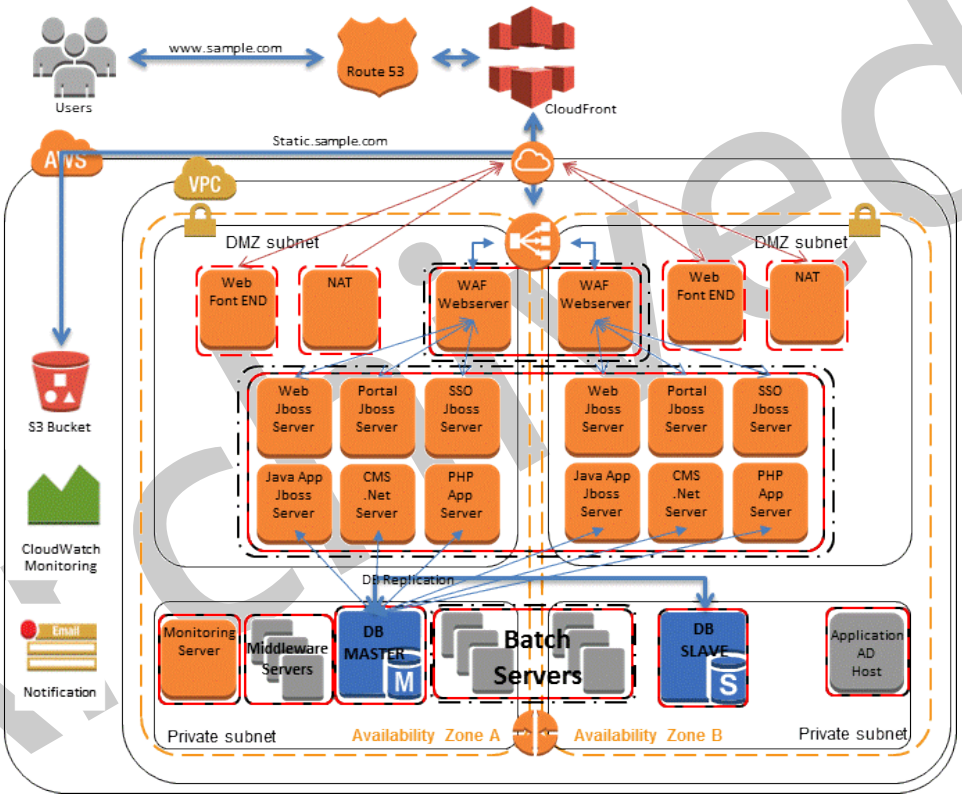

Figure 1: AWS Native Scenario

EBS Snapshot-Based Protection

When services are running in Amazon Elastic Compute Cloud[2] (Amazon EC2), compute instances can use Amazon Elastic Block Store (Amazon EBS) volumes to store and access primary data. You can use this block storage for structured data, such as databases, or unstructured data, such as files in a file system on the volume.

Amazon EBS provides the ability to create snapshots (backups) of any Amazon EBS volume. It takes a copy of the volume and places it in Amazon S3, where it is stored redundantly in multiple Availability Zones. The first snapshot is a full copy of the volume; ongoing snapshots store incremental block-level changes only.

This is a fast and reliable way to restore full volume data. If you only need a partial restore, you can attach the volume to the running instance under a different device name, mount it, and then use operating system copy commands to copy the data from the backup volume to the production volume.

Amazon EBS snapshots can also be copied between AWS regions using the Amazon EBS snapshot copy capability available in the console or from the command line, as described in the Amazon Elastic Cloud Compute User Guide.[3] You can use this feature to store your backup in another region without having to manage the underlying replication technology.

[2] http://aws.amazon.com/ec2/

[3] http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ebs-copy-snapshot.html

Creating EBS Snapshots

When you create a snapshot, you protect your data directly to durable disk-based storage. You can use the AWS Management Console, the command line interface (CLI), or the APIs to create the Amazon EBS snapshot.

In the Amazon EC2 console, on the **Elastic Block Store Volumes** page, choose **Create Snapshot** from the **Actions** menu. On the **Create Snapshot** dialog box, choose **Create** to create a snapshot that will be stored in Amazon S3.

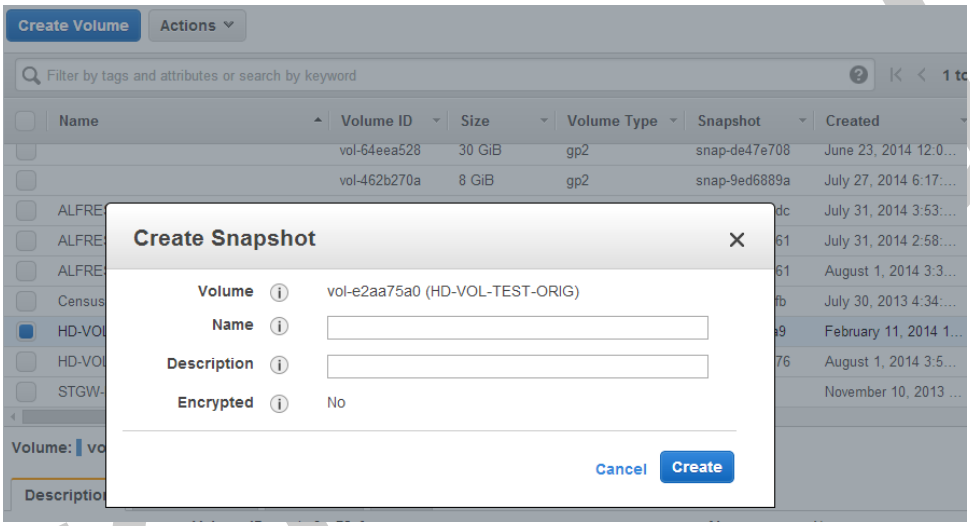

Figure 2: Using the EC2 Console to Create a Snapshot

To use the CLI command to create the snapshot, run the following command:

```
➢ aws ec2 create-snapshot
```

You can schedule and run the `aws ec2 create-snapshot` commands on a regular basis to back up the EBS data. The economical pricing of Amazon S3 makes it possible for you to retain many generations of data. And because snapshots are block-based, you consume space only for data that's changed after the initial snapshot was created.

Restoring from an EBS Snapshot

To restore data from a snapshot, you can use the AWS Management Console, the CLI, or the APIs to create a volume from an existing snapshot.

For example, follow these steps to restore a volume to an earlier point-in-time backup:

1. Use the following command to create a volume from the backup snapshot:

```
➢ aws ec2 create-volume --region us-west-1b --snapshot-
  id mysnapshot-id
```

2. On the Amazon EC2 instance, unmount the existing volume.

   In Linux, use `umount`. In Windows, use the Logical Volume Manager (LVM).

3. Use the following command to detach the existing volume from the instance:

```
➢ aws ec2 detach-volume --volume-id oldvolume-id –
  instance-id myec2instance-id
```

4. Use the following command to attach the volume that was created from the snapshot:

```
➢ aws ec2 attach-volume --volume-id newvolume-id
  --instance-id myec2instance-id --device /dev/sdf
```

5. Remount the volume on the running instance.

Creating Consistent or Hot Backups

When you perform a backup, it's best to have the system in a state where it is not performing any I/O. In the ideal case, the machine isn't accepting traffic, but this is increasingly rare as 24/7 IT operations become the norm.

For this reason, you must quiesce the file system or database in order to make a clean backup. The way in which you do this depends on your database or file system.

The process for a database is as follows:

- If possible, put the database into hot backup mode.
- Run the Amazon EBS snapshot commands.
- Take the database out of hot backup mode or, if using a read replica, terminate the read replica instance.

The process for a file system is similar, but depends on the capabilities of the operating system or file system. For example, XFS is a file system that can flush its data for a consistent backup. For more information, see xfs_freeze.[4]

If your file system does not support the ability to freeze, you should unmount it, issue the snapshot command, and then remount the file system. Alternatively, you can facilitate this process by using a logical volume manager that supports the freezing of I/O.

Because the snapshot process continues in the background and the creation of the snapshot is fast to execute and captures a point in time, the volumes you're backing up only need to be unmounted for a matter of seconds. Because the backup window is as small as possible, the outage time is predictable and can be scheduled.

[4] https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Storage_Administration_Guide/xfsfreeze.html

Performing Multivolume Backups

In some cases, you can stripe data across multiple Amazon EBS volumes by using a logical volume manager to increase potential throughput. When you use a logical volume manager (for example, mdadm or LVM), it is important to perform the backup from the volume manager layer rather than the underlying EBS volumes. This ensures all metadata is consistent and the subcomponent volumes are coherent.

There are a number of ways to accomplish this. For example, you can use the script created by alestic.com[5]. The memory buffers should be flushed to disk; the file system I/O to disk should be stopped; and a snapshot should be initiated simultaneously for all the volumes making up the RAID set. After the snapshot for the volumes is initiated (usually a second or two), the file system can continue its operations. The snapshots should be tagged so that you can manage them collectively during a restore.

You can also perform these backups from the logical volume manager or file-system level. In these cases, using a traditional backup agent enables the data to be backed up over the network. A number of agent-based backup solutions are available on the internet and in the AWS Marketplace.[6] Remember that agent-based backup software expects a consistent server name and IP address. As a result, using these tools with instances deployed in an Amazon virtual private cloud (VPC)[7] is the best way to ensure reliability.

An alternative approach is to create a replica of the primary system volumes that exist on a single large volume. This simplifies the backup process, because only one large volume must be backed up, and the backup does not take place on the primary system. However, you should first determine whether the single volume can perform sufficiently during the backup and whether the maximum volume size is appropriate for the application.

---

[5] https://github.com/alestic/ec2-consistent-snapshot

[6] https://aws.amazon.com/marketplace/

[7] http://aws.amazon.com/vpc/

Database Backup Approaches

AWS has many options for databases. You can run your own database on an EC2 instance or use one of the managed service database options provided by the Amazon Relational Database Service [8](Amazon RDS). If you are running your own database on an EC2 instance, you can back up data to files using native tools (for example, MySQL[9], Oracle[10], MSSQL[11], PostgreSQL[12]) or create a snapshot of the volumes containing the data using one of the methods described in "EBS Snapshot-Based Protection."

Using Database Replica Backups

For databases that are built on RAID sets of Amazon EBS volumes, you can remove the burden of backups on the primary database by creating a read replica of the database. This is an up-to-date copy of the database that runs on a separate Amazon EC2 instance. The replica database instance can be created using multiple disks similar to the source, or the data can be consolidated to a single EBS volume. You can then use one of the procedures described in "EBS Snapshot-Based Protection" to snapshot the EBS volumes. This approach is often used for large databases that are required to run 24/7. When that is the case, the backup window required is too long and the production database cannot be taken down for such long periods.

Using Amazon RDS for Backups

Amazon RDS includes features for automating database backups. Amazon RDS creates a storage volume snapshot of your database instance, backing up the entire DB instance, not just individual databases.

---

[8] https://aws.amazon.com/rds/

[9] http://dev.mysql.com/doc/refman/5.7/en/backup-and-recovery.html

[10] http://docs.oracle.com/cd/E11882_01/backup.112/e10642/rcmbckba.htm#BRADV8003

[11] http://msdn.microsoft.com/en-us/library/ms187510.aspx

[12] http://www.postgresql.org/docs/9.3/static/backup.html

Amazon RDS provides two different methods for backing up and restoring your DB instances:

- **Automated backups** enable point-in-time recovery of your DB instance. Automated backups are turned on by default when you create a new DB instance. Amazon RDS performs a full daily backup of your data during a window that you define when you create the DB instance. You can configure a retention period of up to 35 days for the automated backup. Amazon RDS uses these periodic data backups in conjunction with your transaction logs to enable you to restore your DB instance to any second during your retention period, up to the `LatestRestorableTime` (typically, the last five minutes). To find the latest restorable time for your DB instances, you can use the `DescribeDBInstances` API call or look on the **Description** tab for the database in the Amazon RDS console.

  When you initiate a point-in-time recovery, transaction logs are applied to the most appropriate daily backup in order to restore your DB instance to the time you requested.

- **DB snapshots** are user-initiated backups that enable you to back up your DB instance to a known state as frequently as you like, and then restore to that state at any time. You can use the Amazon RDS console or the `CreateDBSnapshot` API call to create DB snapshots. These snapshots have unlimited retention. They are kept until you use the console or the `DeleteDBSnapshot` API call to explicitly delete them.

When you restore a database to a point in time or from a DB snapshot, a new database instance with a new endpoint will be created. In this way, you can create multiple database instances from a specific DB snapshot or point in time.

You can use the AWS Management Console or a `DeleteDBInstance` call to delete the old database instance.

Using AMI to Back Up EC2 Instances

AWS stores system images in what are called Amazon Machine Images (AMIs). These images consist of the template for the root volume required to launch an instance. You can use the AWS Management Console or the `aws ec2 create-image` CLI command to back up the root volume as an AMI.

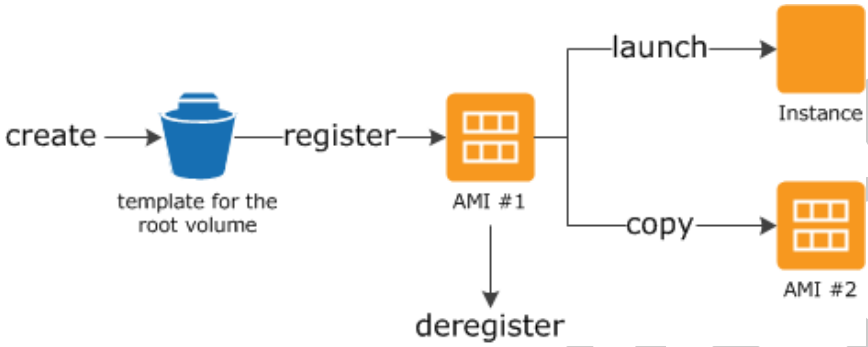

Figure 3: Using an AMI to Back Up and Launch an Instance

When you register an AMI, it is stored in your account using Amazon EBS snapshots. These snapshots reside in Amazon S3 and are highly durable.

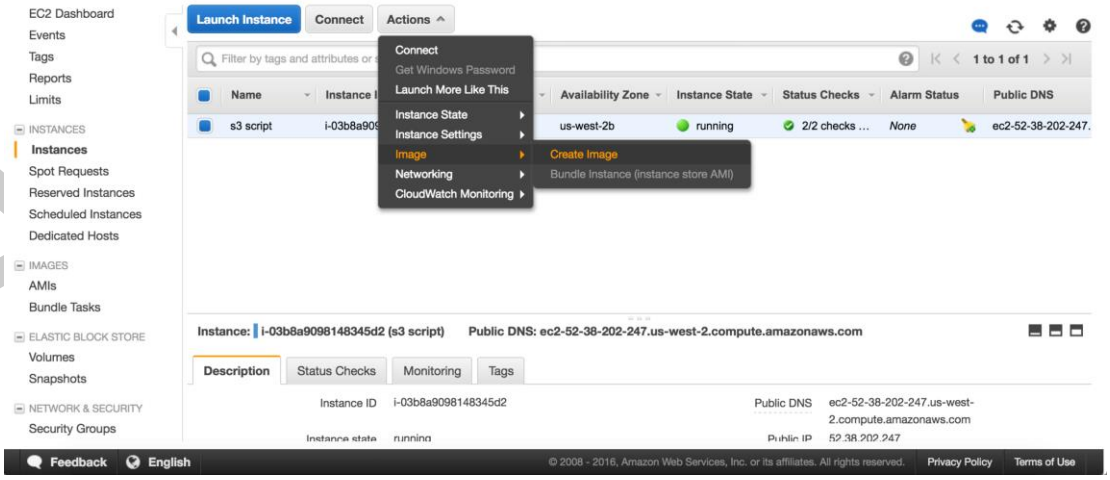

Figure 4: Using the EC2 Console to Create a Machine Image

After you have created an AMI of your Amazon EC2 instance, you can use the AMI to re-create the instance or launch more copies of the instance. You can also copy AMIs from one region to another for application migration or disaster recovery.

On-Premises to AWS Infrastructure

This scenario describes a workload environment with no components in the cloud. All resources, including web servers, application servers, monitoring servers, databases, Active Directory, and more are hosted either in the customer data center or through colocation.

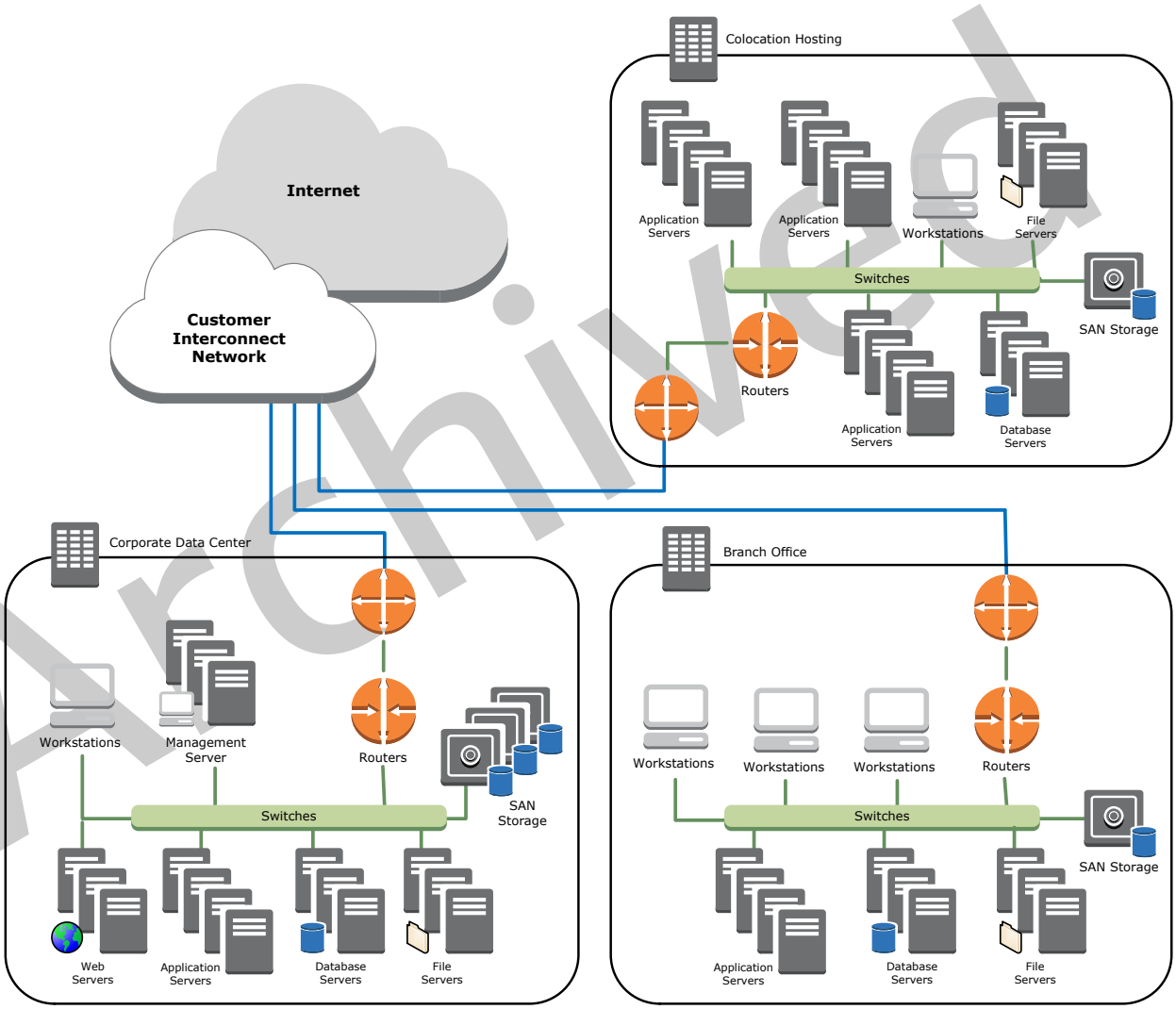

Figure 5: On-Premises Environment

By using AWS storage services in this scenario, you can focus on backup and archiving tasks. You don't have to worry about storage scaling or infrastructure capacity to accomplish the backup task.

Amazon S3 and Amazon Glacier are natively API-based and available through the Internet. This allows backup software vendors to directly integrate their applications with AWS storage solutions, as shown in the following figure.

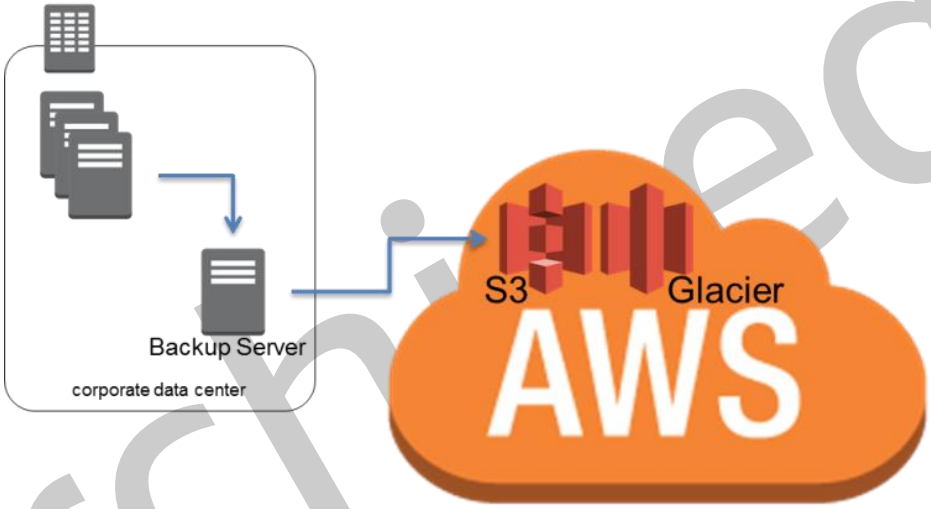

Figure 6: Backup Connector to Amazon S3 or Amazon Glacier

In this scenario, backup and archive software directly interfaces with AWS through the APIs. Because the backup software is AWS-aware, it will back up the data from the on-premises servers directly to Amazon S3 or Amazon Glacier.

If your existing backup software does not natively support the AWS cloud, you can use AWS storage gateway products. AWS Storage Gateway[13] is a virtual appliance that provides seamless and secure integration between your data center and the AWS storage infrastructure. The service allows you to securely store data

[13] http://aws.amazon.com/storagegateway/

in the AWS cloud for scalable and cost-effective storage. Storage Gateway supports industry-standard storage protocols that work with your existing applications while securely storing all of your data encrypted in Amazon S3 or Amazon Glacier.

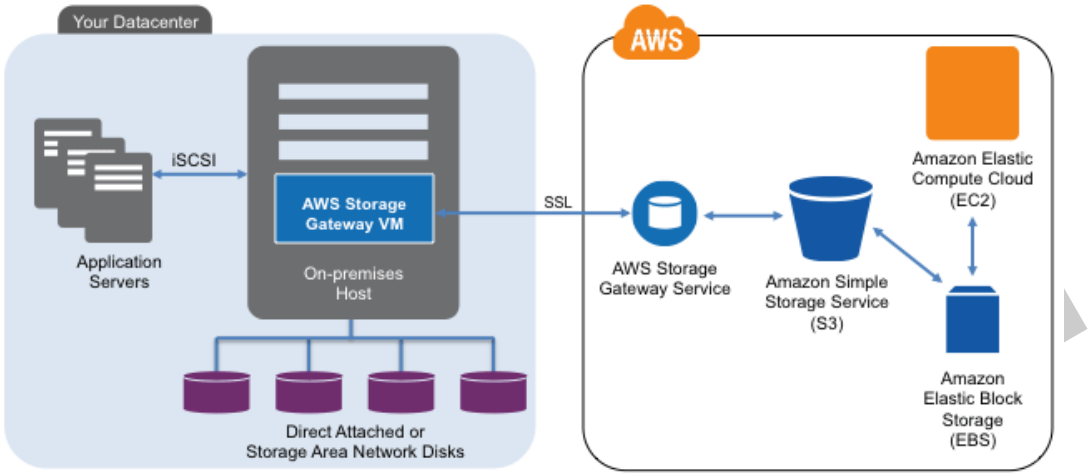

Figure 7: Connecting On-Premises to AWS Storage

AWS Storage Gateway supports the following configurations:

- **Volume gateways:** Volume gateways provide cloud-backed storage volumes that you can mount as Internet Small Computer System Interface (iSCSI) devices from your on-premises application servers. The gateway supports the following volume configurations:
  - **Gateway-cached volumes:** You can store your primary data in Amazon S3 and retain your frequently accessed data locally. Gateway-cached volumes provide substantial cost savings on primary storage, minimize the need to scale your storage on premises, and retain low-latency access to your frequently accessed data.
  - **Gateway-stored volumes:** In the event you need low-latency access to your entire data set, you can configure your on-premises data gateway to store your primary data locally, and asynchronously back up point-in-time snapshots of this data to Amazon S3. Gateway-stored volumes provide durable and inexpensive off-site backups that you can recover locally or from Amazon EC2.
- **Gateway-virtual tape library (gateway-VTL):** With gateway-VTL, you can have a limitless collection of virtual tapes. Each virtual tape can be stored

in a virtual tape library backed by Amazon S3 or a virtual tape shelf backed by Amazon Glacier. The virtual tape library exposes an industry-standard iSCSI interface, which provides your backup application with online access to the virtual tapes. When you no longer require immediate or frequent access to data contained on a virtual tape, you can use your backup application to move it from its virtual tape library to your virtual tape shelf to further reduce your storage costs.

These gateways act as plug-and-play devices providing standard iSCSI devices, which can be integrated into your backup or archive framework. You can use the iSCSI disk devices as storage pools for your backup software or the gateway-VTL to offload tape-based backup or archive directly to Amazon S3 or Amazon Glacier.

Using this method, your backup and archives are automatically offsite (for compliance purposes) and stored on durable media, eliminating the complexity and security risks of off-site tape management.

Hybrid Environments

The two infrastructure deployments discussed to this point, cloud-native and on-premises, can be combined into a hybrid scenario where the workload environment has on-premises and AWS infrastructure components. Resources, including web servers, application servers, monitoring servers, databases, Active Directory, and more are hosted either in the customer data center or AWS. Applications running in the AWS cloud are connected to applications running on-premises.

This is becoming a common scenario for enterprise workloads. Many enterprises have data centers of their own and use AWS to augment capacity. These customer data centers are often connected to the AWS network by high-capacity network links. For example, with AWS Direct Connect[14], you can establish private, dedicated connectivity from your premises to AWS. This provides the bandwidth

[14] http://aws.amazon.com/directconnect/

and consistent latency to upload data to the cloud for the purposes of data protection and consistent performance and latency for hybrid workloads.

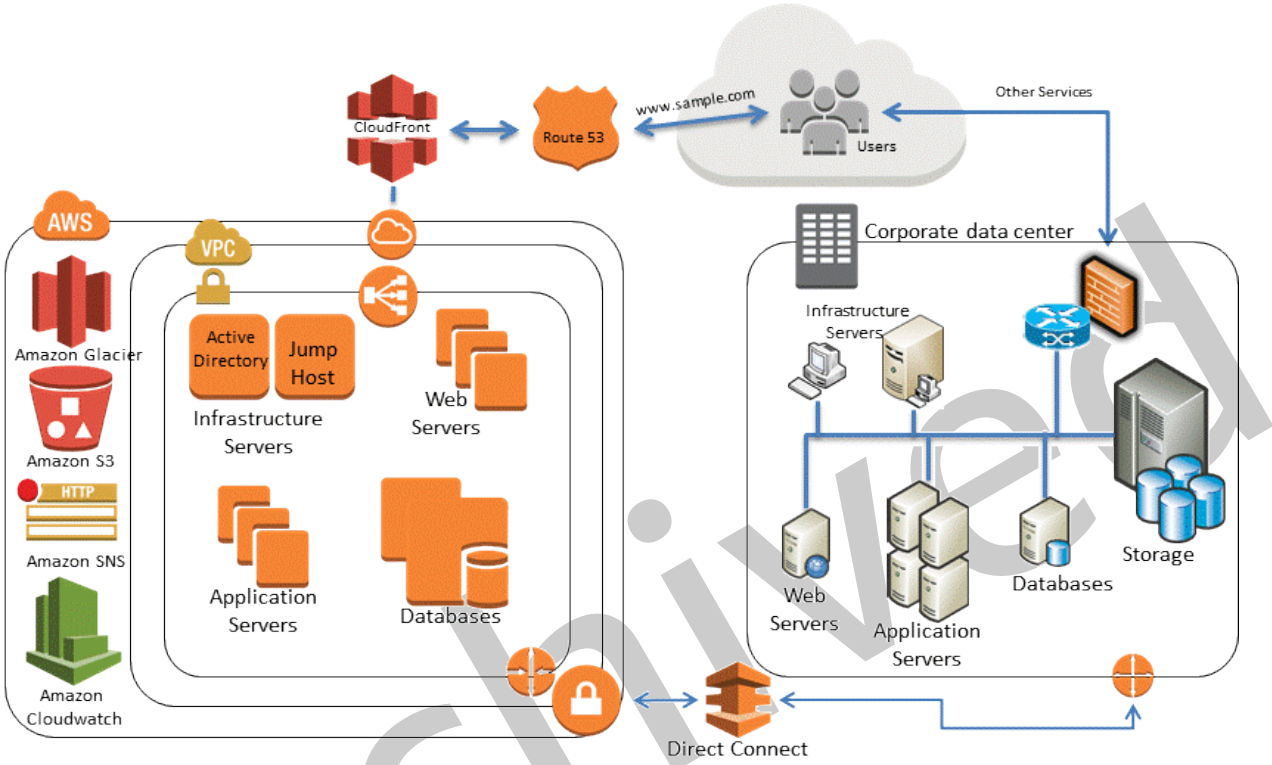

Figure 8: A Hybrid Infrastructure Scenario

Well-designed data protection solutions typically use a combination of the methods described in the cloud-native and on-premises solutions.

Backing Up AWS-Based Applications to Your Data Center

If you already have a framework that backs up data for your on-premises servers, then it is easy to extend it to your AWS resources over a VPN connection or through AWS Direct Connect. You can install the backup agent on the Amazon EC2 instances and back them up per your data-protection policies.

Migrating Backup Management to the Cloud for Availability

Depending on your backup architecture, you may have a master backup server and one or more media or storage servers located on-premises with the services it's protecting. In this case, you might want to move the master backup server to an Amazon EC2 instance to protect it from on-premises disasters and have a highly available backup infrastructure.

To manage the backup data flows, you might also want to create one or more media servers on Amazon EC2 instances. Media servers near the Amazon EC2 instances will save you money on internet transfer and, when backing up to S3 or Amazon Glacier, increase overall backup and recovery performance.

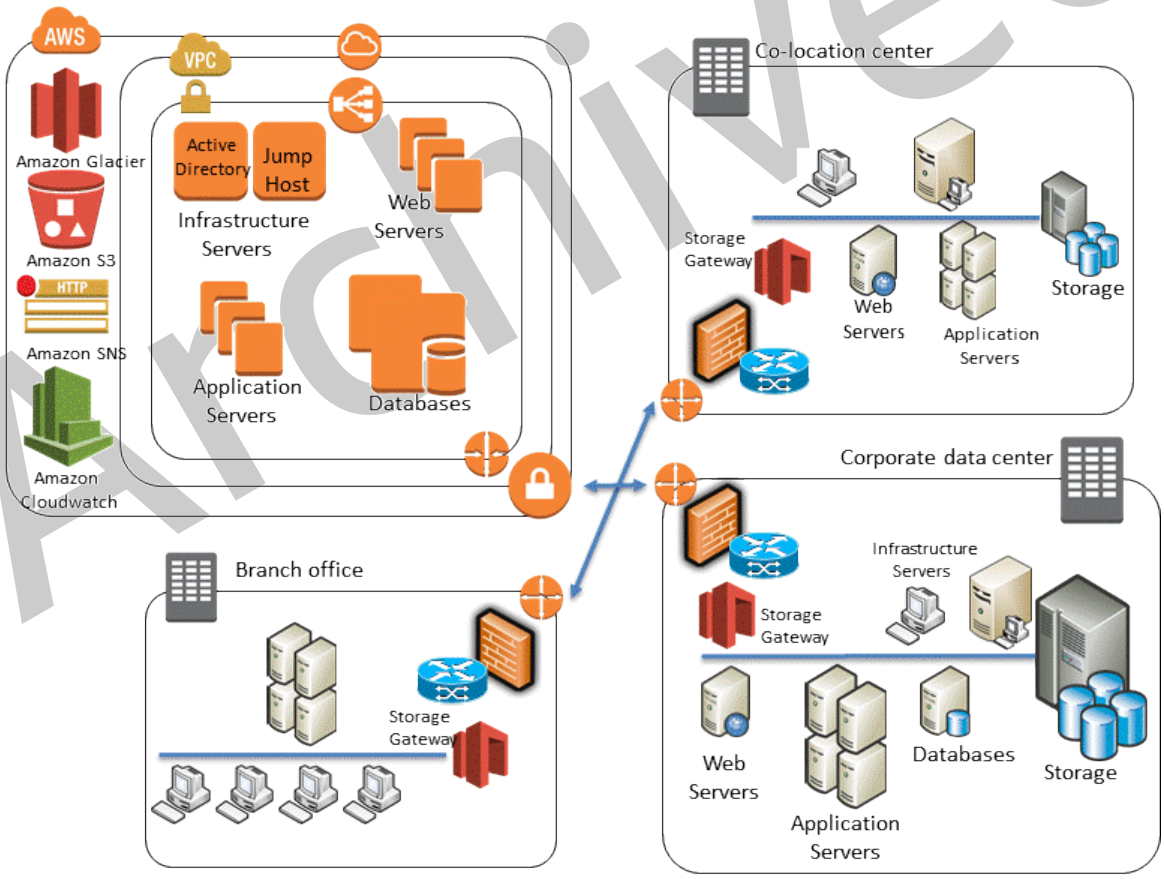

Figure 9: Using Gateways in the Hybrid Scenario

Example Hybrid Scenario

Assume that you are managing an environment where you are backing up Amazon EC2 instances, standalone servers, virtual machines, and databases. This environment has 1,000 servers, and you back up the operating system, file data, virtual machine images, and databases. There are 20 databases (a mixture of MySQL, Microsoft SQL Server, and Oracle) to back up.

Your backup software has agents that back up operating systems, virtual machine images, data volumes, SQL Server databases, and Oracle databases (using RMAN). For applications like MySQL that your backup software does not have an agent for, you might use the mysqldump client utility to create a database dump file to disk where standard backup agents can then protect the data.

To protect this environment, your third-party backup software most likely has a global catalog server or master server that controls the backup, archive, and restore activities as well as multiple media servers that are connected to disk-based storage, Linear Tape-Open (LTO) tape drives, and AWS storage services.

The simpliest way to augment your backup solution with AWS storage services is to take advantage of your backup vendor's support for Amazon S3 or Amazon Glacier. We suggest you work with your vendor to understand their integration and connector options. For a list of backup software vendors who work with AWS, see our partner directory[15].

If your exising backup software does not natively support cloud storage for backup or archive, you can use a storage gateway device, such as a bridge, between the backup software and Amazon S3 or Amazon Glacier.

There are many third-party gateway solutions. You can also use AWS Storage Gateway virtual appliances to bridge this gap because it uses generic techniques such as iSCSI-based volumes and virtual tape libraries (VTLs). This configuration requires a supported hypervisor (VMware or Microsoft Hyper-V) and local storage to host the appliance.

[15] http://www.aws-partner-directory.com/PartnerDirectory/PartnerSearch?type=ISV

Archiving Data with AWS

When you need to preserve data for compliance or corporate reasons, you archive it. Unlike backups, which are usually performed to keep a copy of the production data for a short duration to recover from data corruption or data loss, archiving maintains all copies of data until the retention policy expires.

A good archive meets the following criteria:

- Data durability for long-term integrity
- Data security
- Ease of recoverability
- Low cost

Immutable data stores can be another regulatory or compliance requirement.

Amazon Glacier provides archives at low cost, native encryption of data at rest, 11 nines of durability, and unlimited capacity.

Amazon S3 Standard - Infrequent Access is a good choice for use cases that require the quick retrieval of data. Amazon Glacier is a good choice for use cases where data is infrequently accessed and retrieval times of several hours are acceptable.

Objects can be tiered into Amazon Glacier either through lifecycle rules in S3 or the Amazon Glacier API. The Amazon Glacier Vault Lock feature allows you to easily deploy and enforce compliance controls for individual Amazon Glacier vaults with a vault lock policy. You can specify controls such as “write once, read many” (WORM) in a vault lock policy and lock the policy from future edits. For more information, see Amazon Glacier.

Securing Backup Data in AWS

Data security is a common concern. AWS takes security very seriously. It’s the foundation of every service we launch. Storage services like Amazon S3 provide strong capabilities for access control and encryption both at rest and in transit. All Amazon S3 and Amazon Glacier API endpoints support SSL encryption for

data in transit. Amazon Glacier encrypts all data at rest by default. With Amazon S3, customers can choose server-side encryption for objects at rest by letting AWS manage the encryption keys, providing their own keys when they upload an object, or using AWS Key Management Service (AWS KMS)[16] integration for the encryption keys. Alternatively, customers can always encrypt their data before uploading it to AWS. For more information, see Amazon Web Services: Overview of Security Processes.

Conclusion

Gartner has recognized AWS as a leader in public cloud storage services[17]. AWS is well positioned to help organizations move their workloads to cloud-based platforms, the next generation of backup. AWS provides cost-effective and scalable solutions to help organizations balance their requirements for backup and archiving. These services integrate well with technologies you are using today.

Contributors

The following individuals contributed to this paper:

- Pawan Agnihotri, Solutions Architect, Amazon Web Services
- Lee Kear, Solutions Architect, Amazon Web Services
- Peter Levett, Solutions Architect, Amazon Web Services

[16] http://docs.aws.amazon.com/AmazonS3/latest/dev/UsingKMSEncryption.html

[17] http://www.gartner.com/technology/reprints.do?id=1-1WWKTQ3&ct=140709&st=sb

Document Revisions

Updated May 2016