An Overview of AWS Cloud Data Migration Services

Published May 1, 2016

Updated June 13, 2021

Notices

Customers are responsible for making their own independent assessment of the information in this document. This document: (a) is for informational purposes only, (b) represents current AWS product offerings and practices, which are subject to change without notice, and (c) does not create any commitments or assurances from AWS and its affiliates, suppliers or licensors. AWS products or services are provided “as is” without warranties, representations, or conditions of any kind, whether express or implied. 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

One of the most challenging steps required to deploy an application infrastructure in the cloud is moving data into and out of the cloud. Amazon Web Services (AWS) provides multiple services for moving data, and each solution offers various levels of speed, security, cost, and performance. This whitepaper outlines the different AWS services that can help seamlessly transfer data to and from the AWS Cloud.

Introduction

As you plan your data migration strategy, you will need to determine the best approach to use based on the specifics of your environment. There are many different ways to *lift-and-shift* data to the cloud such as one-time large batches, constant device streams, intermittent updates, or even hybrid data storage combining the AWS Cloud and on-premises data stores. These methods can be used individually or together to help streamline the realities of cloud data migration projects.

Cloud Data Migration Challenges

When planning a data migration, you need to determine how much data is being moved and the bandwidth available for the transfer of data. This will determine how long the transfer will take. AWS offers several methods to transfer data into your account including the AWS Snow Family of storage devices, AWS Direct Connect, and AWS Site-to-Site VPN over your existing internet connection. The network bandwidth that is consumed for data migration will not be available for your organization's typical application traffic. In addition, your organization might be concerned with moving sensitive business information from your internal network to a secure AWS environment. Determining the security level for your organization helps you select the appropriate AWS services for your data migration.

Security and Data Sensitivity

When customers migrate data, ensuring the security of data both in transit and at-rest is critical. AWS takes security very seriously and builds security features into all data migration services. Every service uses AWS Identity and Access Management (IAM) to control programmatic and AWS Console access to resources. The following table lists these features.

Table 1 – AWS Services Security Features

AWS Service	Security Features
AWS Direct Connect	• Provides a dedicated physical connection with no data transfer over the Internet. • Integrates with AWS CloudTrail to capture API calls made by or on behalf of a customer account.
AWS Snow Family	• Integrates with the AWS Key Management Service (AWS KMS) to encrypt data-at-rest that is stored on AWS Snowcone, Snowball, or Snowmobile. • Uses an industry-standard Trusted Platform Module (TPM) that has a dedicated processor designed to detect any unauthorized modifications to the hardware, firmware, or software to physically secure the AWS Snowcone or Snowball device.

AWS Service	Security Features
AWS Transfer Family	• SFTP uses SSH while FTPS uses TLS to transfer data through a secure and encrypted channel. • AWS Transfer Family is PCI-DSS and GDPR compliant, and HIPAA eligible. The service is also SOC 1, 2, and 3 compliant. Learn more about services in scope grouped by compliance programs. • The service supports three modes of authentication: Service Managed, where you store user identities within the service, Microsoft Active Directory, and Custom (BYO), which enables you to integrate an identity provider of your choice. Service Managed authentication is supported for server endpoints that are enabled for SFTP only. • You can use Amazon CloudWatch to monitor your end users' activity and use AWS CloudTrail to access a record of all S3 API operations invoked by your server to service your end users' data requests.

AWS Service	Security Features
AWS DataSync	• All data transferred between the source and destination is encrypted via Transport Layer Security (TLS), which replaced Secure Sockets Layer (SSL). Data is never persisted in AWS DataSync itself. The service supports using default encryption for S3 buckets, Amazon EFS file system encryption of data at rest, and Amazon FSx For Windows File Server encryption at rest and in transit. • When copying data to or from your premises, there is no need to setup a VPN/tunnel or allow inbound connections. Your AWS DataSync agent can be configured to route through a firewall using standard network ports. • Your AWS DataSync agent connects to DataSync service endpoints within your chosen AWS Region. You can choose to have the agent connect to public internet facing endpoints, Federal Information Processing Standards (FIPS) validated endpoints, or endpoints within one of your VPCs.
AWS Storage Gateway	• Encrypts all data in transit to and from AWS by using SSL/TLS. • All data in AWS Storage Gateway is encrypted at rest using AES-256 while data transfers are encrypted with AES-128 GCM or AES-128 CCM. • Authentication between your gateway and iSCSI initiators can be secured by using Challenge-Handshake Authentication Protocol (CHAP).

AWS Service	Security Features
Amazon S3 Transfer Acceleration	• Access to Amazon S3 can be restricted by granting other AWS accounts and users permission to perform the resource operations by writing an access policy. • Encrypt data at-rest by performing server-side encryption using Amazon S3-Managed Keys (SSE-S3), AWS Key Management Service (KMS)-Managed Keys (SSE-KMS), or Customer Provided Keys (SSE-C). Or by performing client-side encryption using AWS KMS–Managed Customer Master Key (CMK) or Client-Side Master Key. • Data in transit can be secured by using SSL/TLS or client-side encryption. • Enable Multi-Factor Authentication (MFA) Delete for an Amazon S3 bucket.
AWS Kinesis Data Firehose	• Data in transit can be secured by using SSL/TLS. • If you send data to your delivery stream using PutRecord or PutRecordBatch, or if you send the data using AWS IoT, Amazon CloudWatch Logs, or CloudWatch Events, you can turn on server-side encryption by using the StartDeliveryStreamEncryption operation. • You can also enable SSE when you create the delivery stream.

Cloud Data Migration Tools

This section discusses managed and self-managed migration tools, with a brief description of how each solution works. You can select AWS managed or self-managed migration methods, and make your choice based on your specific use case.

Time and Performance

When you migrate data from your on-premises storage to AWS storage services you want to take the least amount of time to move data over your internet connection with minimal disruption to the existing systems.

To calculate the number of days required to migrate a given amount of data, you can use the following formula:

```
Number of Days = (Terabytes * 8 bites per Byte)/(CIRCUIT gigabits
per second * NETWORK_UTILIZATION percent * 3600 seconds per hour *
AVAILABLE_HOURS)
```

For example, if you have an GigabitEthernet connection (1 Gbps) to the Internet and 100 TB of data to move to AWS, theoretically, the minimum time it would take over the network connection at 80 percent utilization is approximately 28 days.

```
(100,000,000,000,000 Bytes * 8 bits per byte) /(1,000,000,000 bps *
80 percent * 3600 seconds per hour * 10 hours per day) = 27.77 days
```

If this amount of time is not practical for you, there are many ways to reduce migration time for large amounts of data. You can use AWS managed migration tools that automate data transfers and optimize your internet connection to the AWS Cloud. Alternatively, you may develop or purchase your own tools and create your own transfers processes that the utilize the native HTTP interfaces to Amazon Simple Storage Service (Amazon S3). For moving small amounts of data from your on-site location to the AWS Cloud, you may use ad-hoc methods that get the job done quickly with minimal use of automation methods discussed in the AWS migration tools section.

For the best results we suggest the following:

Table 2 – Recommended migration methods

Connection	&	Data Scale	Method	Duration
Less than 10 Mbps	&	Less than 100 GB	Self-managed	~ 3 days
Less than 10 Mbps	&	Between 100 GB – 1 TB	AWS-Managed	~ 30 days
Less than 10 Mbps	&	Greater than 1 TB	AWS Snow Family	~ weeks
Less than 1 Gbps	&	Between 100 GB – 1 TB	Self-managed	~ days

Connection	&	Data Scale	Method	Duration
Less than 1 Gbps	&	Greater than 1 TB	AWS- Managed / Snow Family	~ weeks

Choosing a Migration Method

There are several factors to consider when choosing the appropriate migration method and tool. As discussed in the previous section, time allocated to perform data transfers, the volume of data, and network speeds influence the decision between different data migration methods. You should also consider, for each data store, server, or application stack, the number of repetitive steps required to transfer data from source to target. Then, evaluate the variance of these steps as they are repeated. In other words, are there unique requirements per data store that require non-trivial changes to the data migration procedures? Then, evaluate the level of existing investments in custom tooling and automation in your organization. You will need to determine if it is more worthwhile to use existing self-managed tooling and automation or sunset them in favor of managed services and tools. You can use following decision tree as a framework to choose a suitable migration method and tool:

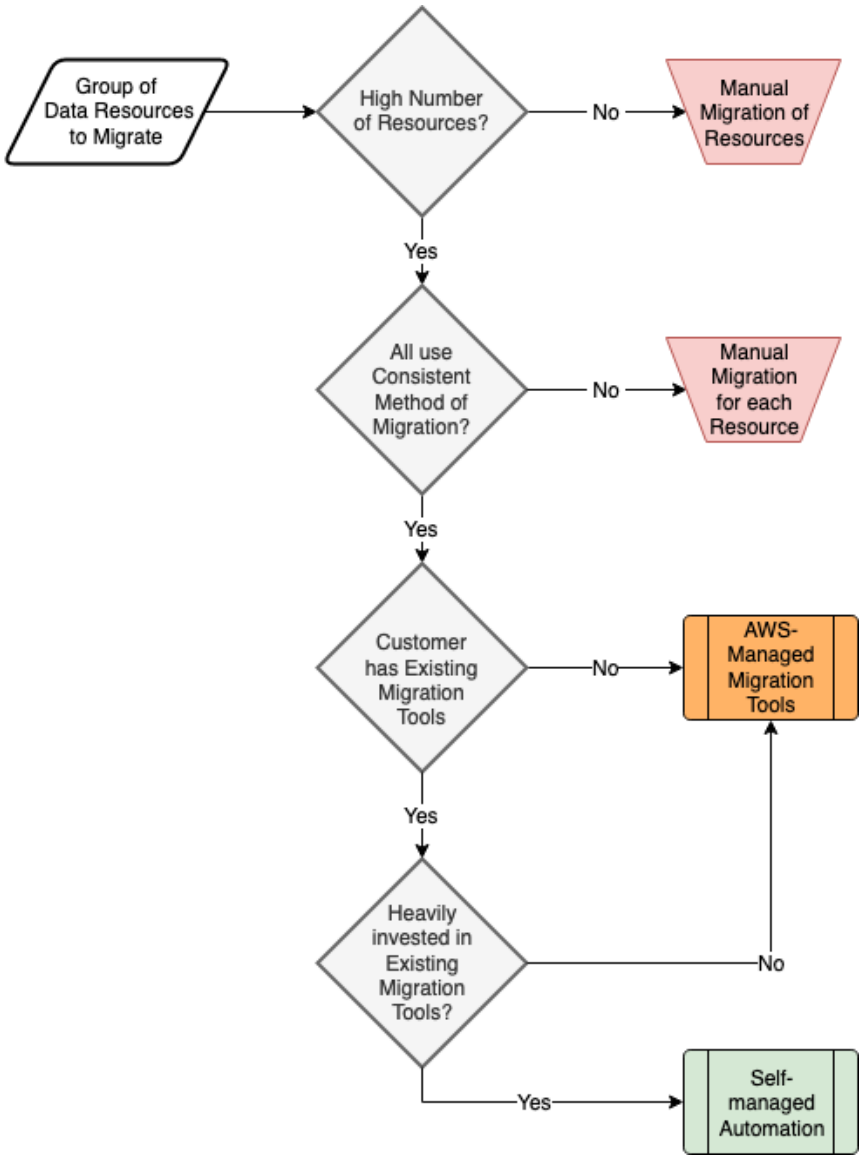

Figure 1 - Migration Method Decision Tree

Self-managed Migration Methods

Small, one-time data transfers on limited bandwidth connections may be accomplished using these very simple tools.

Amazon S3 AWS Command Line Interface

For migrating small amounts of data, you can use the Amazon S3 AWS Command Line Interface to write commands that move data into an Amazon S3 bucket. You can upload objects up to 5 GB in size in a single operation. If your object is greater than 5 GB, you can use multipart upload. Multipart uploading is a three-step process: You initiate the upload, you upload the object parts, and after you have uploaded all the parts, you complete the multipart upload. Upon receiving the complete multipart upload request, Amazon S3 constructs the object from the uploaded parts. Once complete, you can access the object just as you would any other object in your bucket.

Amazon Glacier AWS Command Line Interface

For migrating small amounts of data, you can write commands using the Amazon Glacier AWS Command Line Interface to move data into Amazon Glacier. In a single operation, you can upload archives from 1 byte to up to 4 GB in size. However, for archives greater than 100 MB in size, we recommend using multipart upload. Using the multipart upload API, you can upload large archives, up to about 40,000 GB (10,000 * 4 GB).

Storage Partner Solutions

Multiple Storage Partner solutions work seamlessly to access storage across on-premises and AWS Cloud environments. Partner hardware and software solutions can help customers do tasks such as backup, create primary file storage/cloud NAS, archive, perform disaster recovery, and transfer files.

AWS-Managed Migration Tools

AWS has designed several sophisticated services to help with cloud data migration.

AWS Direct Connect

AWS Direct Connect lets you establish a dedicated network connection between your corporate network and one AWS Direct Connect location. Using this connection, you can create virtual interfaces directly to AWS services. This bypasses Internet service providers (ISPs) in your network path to your target AWS region. By setting up private connectivity over AWS Direct Connect, you could reduce network costs, increase bandwidth throughput, and provide a more consistent network experience than with Internet-based connections.

Using AWS Direct Connect, you can easily establish a dedicated network connection from your premises to AWS at speeds starting at 50 Mbps and up to 100 Gbps. You can use the connection to access Amazon Virtual Private Cloud (Amazon VPC) as well as AWS public services, such as Amazon S3.

AWS Direct Connect in itself is not a data transfer service. Rather, AWS Direct Connect provides a high bandwidth connection that can be used to transfer data between your corporate network and AWS with more consistent performance and without ever having the data routed over the Internet. Encryption methods may be applied to secure the data transfers over the AWS Direct Connect such as AWS Site-to-Site VPN

AWS APN Partners can help you set up a new connection between an AWS Direct Connect location and your corporate data center, office, or colocation facility. Additionally, many of our partners offer AWS Direct Connect Bundles that provide a set of advanced hybrid architectures that can reduce complexity and provide peak performance. You can extend your on-premises networking, security, storage, and compute technologies to the AWS Cloud using managed hybrid architecture, compliance infrastructure, managed security, and converged infrastructure.

With 108 Direct Connect locations worldwide and more than 50 Direct Connect delivery partners, you can establish links between your on-premises network and AWS Direct Connect locations.

With AWS Direct Connect, you only pay for what you use, and there is no minimum fee associated with using the service. AWS Direct Connect has two pricing components: port-hour rate (based on port speed), and data transfer out (per GB per month). Additionally, if you are using an APN partner to facilitate an AWS Direct Connect connection, contact the partner to discuss any fees they may charge. For information about pricing, see Amazon Direct Connect Pricing.

AWS Snow Family

The AWS Snow Family accelerates moving large amounts of data into and out of AWS using AWS managed hardware and software. The Snow Family, comprised of AWS Snowcone, AWS Snowball, and AWS Snowmobile, are various physical devices each with different form factors and capacities. They are purpose-built for efficient data storage and transfer and have built-in compute capabilities. The AWS Snowcone device is a lightweight, handheld storage device that accommodates field environments where access to power may be limited and WiFi is necessary to make the connection. An AWS Snowball Edge device is rugged enough to withstand a 70 G shock and at 49.7 pounds (22.54 kg), it is light enough for one person to carry. It is entirely self-contained, with

110-240 VAC power, ships with country-specific power cables, as well as an E Ink display and control panel on the front. Each AWS Snowball Edge appliance is weather-resistant and serves as its own shipping container.

With AWS Snowball, you have the choice of two devices as of the date of this writing, Snowball Edge Compute Optimized with more computing capabilities, suited for higher performance workloads, or Snowball Edge Storage Optimized with more storage, which is suited for large-scale data migrations and capacity-oriented workloads.

Snowball Edge Compute Optimized provides powerful computing resources for use cases such as machine learning, full motion video analysis, analytics, and local computing stacks. These capabilities include 52 vCPUs, 208 GiB of memory, and an optional NVIDIA Tesla V100 GPU. For storage, the device provides 42 TB usable HDD capacity for S3 compatible object storage or EBS-compatible block volumes, as well as 7.68 TB of usable NVMe SSD capacity for EBS-compatible block volumes. Snowball Edge Compute Optimized devices run Amazon EC2 sbe-c and sbe-g instances, which are equivalent to C5, M5a, G3, and P3 instances.

Snowball Edge Storage Optimized devices are well suited for large-scale data migrations and recurring transfer workflows, as well as local computing with higher capacity needs. Snowball Edge Storage Optimized provides 80 TB of HDD capacity for block volumes and Amazon S3-compatible object storage, and 1 TB of SATA SSD for block volumes. For computing resources, the device provides 40 vCPUs, and 80 GiB of memory to support Amazon EC2 sbe1 instances (equivalent to C5).

AWS transfers your data directly onto Snowball Edge device using on-premises high-speed connections, ships the device to AWS facilities, and transfers data off of AWS Snowball Edge devices using Amazon's high-speed internal network. The data transfer process bypasses the corporate Internet connection and mitigates the requirement for an AWS Direct Connect services. For datasets of significant size, AWS Snowball is often faster than transferring data via the Internet and more cost-effective than upgrading your data center's Internet connection. AWS Snowball supports importing data into and exporting data from Amazon S3 buckets. From there, the data can be copied or moved to other AWS services such as Amazon Elastic Block Store (Amazon EBS), Amazon Elastic File System (Amazon EFS), Amazon FSx File Gateway, and Amazon Glacier.

AWS Snowball is ideal for transferring large amounts of data, up to many petabytes, in and out of the AWS cloud securely. This approach is effective, especially in cases where you don't want to make expensive upgrades to your network infrastructure; if you frequently experience large backlogs of data; if you are in a physically isolated

environment; or if you are in an area where high-speed Internet connections are not available or cost-prohibitive. In general, if loading your data over the Internet would take a week or more, you should consider using AWS Snow Family.

Common use cases include cloud migration, disaster recovery, data center decommission, and content distribution. When you decommission a data center, many steps are involved to make sure valuable data is not lost, and the AWS Snow Family can help ensure data is securely and cost-effectively transferred to AWS. In a content distribution scenario, you might use Snowball Edge devices if you regularly receive or need to share large amounts of data with clients, customers, or business partners. Snowball appliances can be sent directly from AWS to client or customer locations.

If you need to move massive amounts of data, AWS Snowmobile is an Exabyte-scale data transfer service. Each Snowmobile is a 45-foot long ruggedized shipping container hauled by a trailer truck with up to 100 PB data storage capacity. Snowmobile also handles all of the logistics. AWS personnel transport and configure the Snowmobile. They will also work with your team to connect a temporary high-speed network switch to your local network. The local high-speed network facilitates rapid transfer of data from within your datacenter to the Snowmobile. Once you've loaded all your data, the Snowmobile drives back to AWS where the data is imported into Amazon S3.

Moving data at this massive scale requires additional preparation, precautions, and security. Snowmobile uses GPS tracking, round the clock video surveillance, and dedicated security personnel. Snowmobile offers an optional security escort vehicle while your data is in transit to AWS. Management of and access to the shipping container and data stored within is limited to AWS personnel using hardware secure access control methods.

AWS Snow Family might not be the ideal solution if your data can be transferred over the Internet in less than one week, or if your applications cannot tolerate the offline transfer time.

With AWS Snow Family, as with most other AWS services, you pay only for what you use. Snowball has three pricing components: a service fee (per job), extra day charges as required, and data transfer out. The first 5 days of Snowcone usage and the first 10 days of onsite Snowball includes 10 days of device use. For the destination storage, the standard Amazon S3 storage pricing applies. For pricing information, see AWS Snowball pricing. Snowmobile pricing is based on the amount of data stored on the truck per month. For more information about AWS Regions and availability, see AWS Regional Services.

AWS Storage Gateway

AWS Storage Gateway makes backing up to the cloud extremely simple. It connects an on-premises software appliance with cloud-based storage to provide seamless and secure integration between an organization's on-premises IT environment and the AWS storage infrastructure. The service enables you to securely store data in the AWS Cloud for scalable and cost-effective storage. AWS Storage Gateway supports three types of storage interfaces used in on-premises environment including file, volume, and tape. It uses industry-standard network storage protocols such as Network File System (NFS) and Server Message Block (SMB) that work with your existing applications enabling data storage using S3 File Gateway function to store data in Amazon S3. It provides low-latency performance by maintaining an on-premises cache of frequently accessed data while securely storing all of your data encrypted in Amazon S3. Once data is stored in Amazon S3, it can be archived in Amazon S3 Glacier. For disaster recovery scenarios, AWS Storage Gateway, together with Amazon Elastic Compute Cloud (Amazon EC2), can serve as a cloud-hosted solution that mirrors your entire production environment.

You can download the AWS Storage Gateway software appliance as a virtual machine (VM) image that you install on a host in your data center or as an EC2 instance. After you've installed your gateway and associated it with your AWS account through the AWS activation process, you can use the AWS Management Console to create gateway-cached volumes, gateway-stored volumes, or a gateway–virtual tape library (VTL), each of which can be mounted as an iSCSI device by your on-premises applications.

Volume Gateway supports iSCSI connections that enable storing of volume data in S3. With caching enabled, you can use Amazon S3 to hold your complete set of data, while caching some portion of it locally for on-premises frequently accessed data. Gateway-cached volumes minimize the need to scale your on-premises storage infrastructure, while still providing your applications with low-latency access to frequently accessed data. You can create storage volumes up to 32 TiB in size and mount them as iSCSI devices from your on-premises application servers. Each gateway configured for gateway-cached volumes can support up to 32 volumes and total volume storage per gateway of 1,024 TiB. Data written to these volumes is stored in Amazon S3, with only a cache of recently written and recently read data stored locally on your on-premises storage hardware.

Gateway-stored volumes store your locally sourced data in cache, while asynchronously backing up data to AWS. These volumes provide your on-premises applications with low-latency access to their entire datasets, while providing durable, off-site backups.

You can create storage volumes up to 16 TiB in size and mount them as iSCSI devices from your on-premises application servers. Each gateway configured for gateway-stored volumes can support up to 32 volumes, with a total volume storage of 512 TiB. Data written to your gateway-stored volumes is stored on your on-premises storage hardware, and asynchronously backed up to Amazon S3 in the form of Amazon EBS snapshots.

A gateway-VTL allows you to perform offline data archiving by presenting your existing backup application with an iSCSI-based VTL consisting of a virtual media changer and virtual tape drives. You can create virtual tapes in your VTL by using the AWS Management Console, and you can size each virtual tape from 100 GiB to 5 TiB. A VTL can hold up to 1,500 virtual tapes, with a maximum aggregate capacity of 1 PiB. After the virtual tapes are created, your backup application can discover them using its standard media inventory procedure. Once created, tapes are available for immediate access and are stored in Amazon S3.

Virtual tapes you need to access frequently should be stored in a VTL. Data that you don't need to retrieve frequently can be archived to your virtual tape shelf (VTS), which is stored in Amazon Glacier, further reducing your storage costs.

Organizations are using AWS Storage Gateway to support a number of use cases. These use cases include corporate file sharing, enabling existing on-premises backup applications to store primary backups on Amazon S3, disaster recovery, and mirroring data to cloud-based compute resources and then later archiving the data to Amazon Glacier.

With AWS Storage Gateway, you pay only for what you use. AWS Storage Gateway has the following pricing components: gateway usage (per gateway appliance per month), and data transfer out (per GB per month). Based on type of gateway appliance you use there are snapshot storage usage (per GB per month), and volume storage usage (per GB per month) for gateway-cached volumes/gateway-stored volumes, and virtual tape shelf storage (per GB per month), virtual tape library storage (per GB per month), and retrieval from virtual tape shelf (per GB) for Gateway-Virtual Tape Library. For information about pricing, see AWS Storage Gateway pricing.

Amazon S3 Transfer Acceleration (S3TA)

Amazon S3 Transfer Acceleration (S3TA) enables fast, easy, and secure transfers of files over long distances between your client and your Amazon S3 bucket. Transfer Acceleration leverages Amazon CloudFront globally distributed AWS edge locations. As

data arrives at an AWS edge location, data is routed to your Amazon S3 bucket over an optimized network path.

Transfer Acceleration helps you fully utilize your bandwidth, minimize the effect of distance on throughput, and ensure consistently fast data transfer to Amazon S3 regardless of your client's location. Acceleration primarily depends on your available bandwidth, the distance between the source and destination, and packet loss rates on the network path. Generally, you will see more acceleration when the source is farther from the destination, when there is more available bandwidth, and/or when the object size is bigger. You can use the online speed comparison tool to get the preview of the performance benefit from uploading data from your location to Amazon S3 buckets in different AWS Regions using Transfer Acceleration.

Organizations are using Transfer Acceleration on a bucket for a variety of reasons. For example, they have customers that upload to a centralized bucket from all over the world, transferring gigabytes to terabytes of data on a regular basis across continents, or having underutilized the available bandwidth over the Internet when uploading to Amazon S3. The best part about using Transfer Acceleration on a bucket is that the feature can be enabled by a single click of a button in the Amazon S3 console; this makes the accelerate endpoint available to use in place of the regular Amazon S3 endpoint.

With Transfer Acceleration, you pay only for what you use and for transferring data over the accelerated endpoint. Transfer Acceleration has the following pricing components: data transfer in (per GB), data transfer out (per GB), and data transfer between Amazon S3 and another AWS Region (per GB). Transfer acceleration pricing is in addition to data transfer (per GB per month) pricing for Amazon S3. For information about pricing, see Amazon S3 pricing.

AWS Kinesis Data Firehose

Amazon Kinesis Data Firehose is the easiest way to load streaming data into AWS. The service can capture and automatically load streaming data into Amazon S3, Amazon Redshift, Amazon Elasticsearch Service, or Splunk. Amazon Kinesis Data Firehose is a fully managed service, making it easier to capture and load massive volumes of streaming data from hundreds of thousands of sources. The service can automatically scale to match the throughput of your data and requires no ongoing administration. Additionally, Amazon Kinesis Data Firehose can also batch, compress, transform, and encrypt data before loading it. This process minimizes the amount of storage used at the destination and increases security.

You can use Data Firehose by creating a delivery stream and sending the data to it. The streaming data originators are called data producers. A producer can be as simple as a PutRecord() or PutRecordBatch() API call, or you can build your producers using Kinesis Agent. You can send a record (before base64-encoding) as large as 1000 KiB. Additionally, Firehose buffers incoming streaming data to a certain size called a *Buffer Size* (1 MiB to 128 MiB) or for a certain period of time called a *Buffer Interval* (60 to 900 seconds) before delivering to destinations.

With Amazon Kinesis Data Firehose, you pay only for the volume of data you transmit through the service. Amazon Kinesis Data Firehose has a single pricing component: data ingested (per GiB), which is calculated as the number of data records you send to the service, times the size of each record rounded up to the nearest 5 KiB. There may be charges associated with PUT requests and storage on Amazon S3 and Amazon Redshift, and Amazon Elasticsearch instance hours based on the destination you select for loading data. For information about pricing see, Amazon Kinesis Data Firehose pricing.

AWS Transfer Family

If you are looking to modernize your file transfer workflows for business processes that are heavily dependent on FTP, SFTP, and FTPS; the AWS Transfer Family service provides fully managed file transfers in and out of Amazon S3 buckets and Amazon EFS shares. The AWS Transfer Family uses a highly available multi-AZ architecture that automatically scales to add capacity based on your file transfer demand. This means no more FTP, SFTP, and FTPS servers to manage. The AWS Transfer Family allows the authentication of users through multiple methods including self-managed, AWS Directory Service, on-premises Active Directory systems through AWS Managed Microsoft AD connectors, or custom identity providers. Custom identity providers may be configured through the Amazon API Gateway enabling custom configurations. DNS entries used by existing users, partners, and applications are maintained using Route 53 for minimal disruption and seamless migration. With your data residing in Amazon S3 or Amazon EFS, you can use other AWS services for analytics and data processing workflows.

There are many use cases that require a standards-based file transfer protocol like FTP, SFTP, or FTPS. AWS Transfer Family is a good fit for secure file sharing between an organization and third parties. Examples of data that are shared between organizations are large files such as audio/video media files, technical documents, research data, and EDI data such as purchase orders and invoices. Another use case is providing a central location where users can download and globally access your data

securely. A third use case is to facilitate data ingestion for a data lake. Organizations and third parties can FTP, SFTP, or FTPS research, analytics, or business data into an Amazon S3 bucket, which can then be further processed and analyzed.

With the AWS Transfer Family, you only pay for the protocols you have enabled for access to your endpoint, and the amount of data transferred over each of the protocols. There are no upfront costs and no resources to manage yourself. You select the protocols, identity provider, and endpoint configuration to enable transfers over the chosen protocols. You are billed on an hourly basis for each of the protocols enabled to access your endpoint, until the time you delete it. You are also billed based on the amount of data (Gigabytes) uploaded and downloaded over each of the protocols. For more details on pricing per region, see AWS Transfer Family pricing.

Third-Party Connectors

Many of the most popular third-party backup software packages, such as CommVault Simpana and Veritas NetBackup, include Amazon S3 connectors. This allows the backup software to point directly to the cloud as a target while still keeping the backup job catalog complete. Existing backup jobs can simply be rerouted to an Amazon S3 target bucket, and the incremental daily changes are passed over the Internet. Lifecycle management policies can move data from Amazon S3 into lower-cost storage tiers for archival status or deletion. Eventually, and invisibly, local tape and disk copies can be aged out of circulation and tape and tape automation costs can be entirely removed.

These connectors can be used alone, or they can be used with a gateway provided by AWS Storage Gateway to back up to the cloud without affecting or re-architecting existing on-premises processes. Backup administrators will appreciate the integration into their daily console activities, and cloud architects will appreciate the behind-the-scenes job migration into Amazon S3.

Cloud Data Migration Use Cases

Use Case 1: One-Time Massive Data Migration

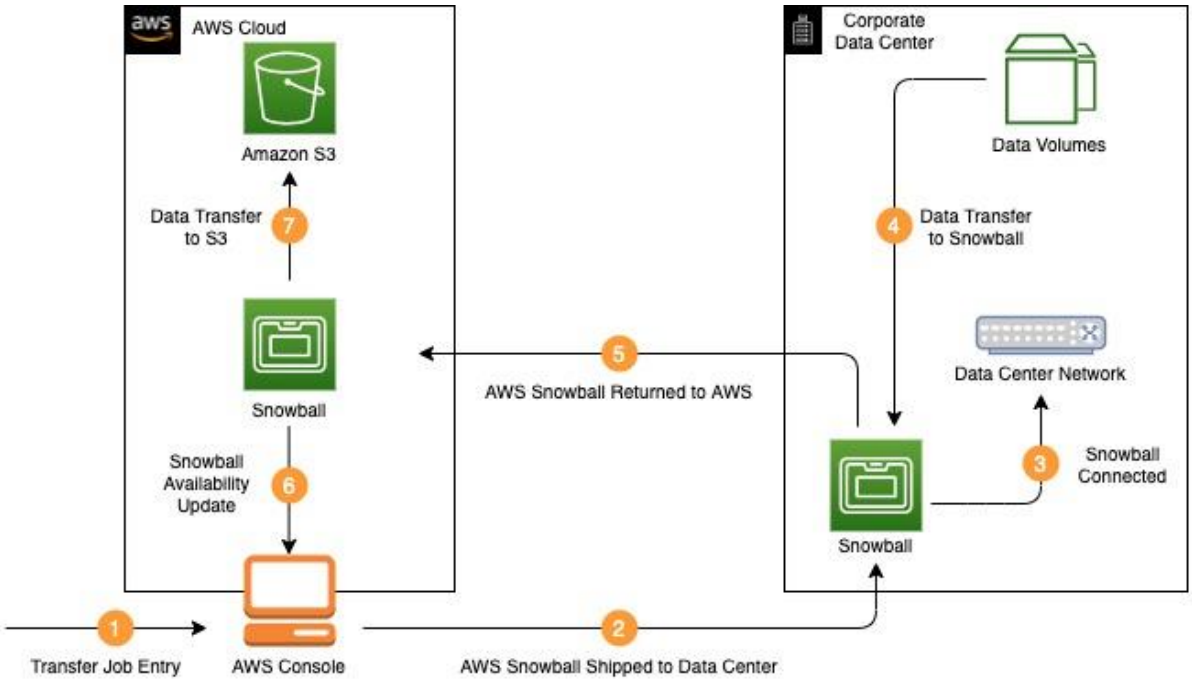

Figure 2 - One-time massive data migration

In use case 1, a customer goes through the process of decommissioning a data center and moving the entire workload to the cloud. First, all the current corporate data needs to be migrated. To complete this migration AWS Snowball appliances are used to move the data from the customer's existing data center to an Amazon S3 bucket in the AWS Cloud.

1. Customer creates a new data transfer job in the AWS Snowball Management Console by providing the following information.
    a. Choose Import into Amazon S3 to start creating the import job.
    b. Enter the shipping address of the corporate data center, and shipping speed (one or two day).
    c. Enter job details, such as name of the job, destination AWS Region, destination Amazon S3 bucket to receive the imported data, and Snowball Edge device type.

d. Enter security settings indicating the IAM role Snowball assumes to import the data and AWS KMS master key used to encrypt the data within Snowball.

e. Set Amazon Simple Notification Service (SNS) notification options and provide a list of comma-separated email addresses to receive email notifications for this job. Choose which job status values trigger notifications.

f. Download AWS OpsHub for Snow family to manage your devices and their local AWS services. With AWS OpsHub you can unlock and configure single or clustered devices, transfer files, and launch/manage instances running on Snow Family devices.

2. After the job is created, AWS ships the Snowball Appliances to the customer data center by AWS. In this example, the customer is importing 200 TB of data into Amazon S3, they will need to create three Import jobs of 80 TB Snowball Edge Storage Optimized capacity.

3. After receiving the Snowball appliance, the customer performs the following tasks.

   a. Customer connects the powered-off appliance to their internal network, and uses the supplied power cables to connect to a power outlet.

   b. After the Snowball is ready, the customer uses the E-Ink display to choose the network settings and assign an IP address to the appliance.

4. The customer transfers the data to the Snowball appliance using the following steps.

   a. Download the credentials consisting of a manifest file and an unlock code for a specific Snowball job from AWS Snow Family Management Console.

   b. Install the Snowball Client on an on-premises machine to manage the flow of data from the on-premises data source to the Snowball.

   c. Access the Snowball client using the terminal or command prompt on the workstation and typing the following command:

```
snowballEdge unlock-device --endpoint [https://Snowball IP Address]
--manifest [Path/to/manifest/file] –unlock-code [29 character
unlock code]
```

   d. Begin transferring data onto the Snowball using the following tools:

i. Version 1.16.14 or earlier of the AWS CLI `s3 cp` or `s3 sync` commands. Detailed installation and command syntax are found here

ii. AWS OpsHub, which was installed in step 1f. Detailed commands and instructions on managing S3 Storage can be found here.

5. After the data transfer is complete, disconnect the Snowball from your network and seal the Snowball. After being properly sealed, the return shipping label appears on the E-Ink display. Arrange UPS pickup of the appliance for shipment back to AWS.
6. UPS automatically reports back a tracking number for the job to the AWS Snowball Management Console. The customer can access that tracking number, and a link to the UPS tracking website by viewing the job's status details in the console.
7. After the appliance is received at the AWS Region, the job status changes from ***In transit to AWS*** to ***At AWS***. On average, it takes a day for data import into Amazon S3 to begin. When the import starts, the status of the job changes to **Importing**. From this point on, it takes an average of two business days for your import to reach **Completed** status. You can track status changes through the AWS Snowball Management Console or by Amazon SNS notifications.

Use Case 2: Continuous On-premises Data Migration

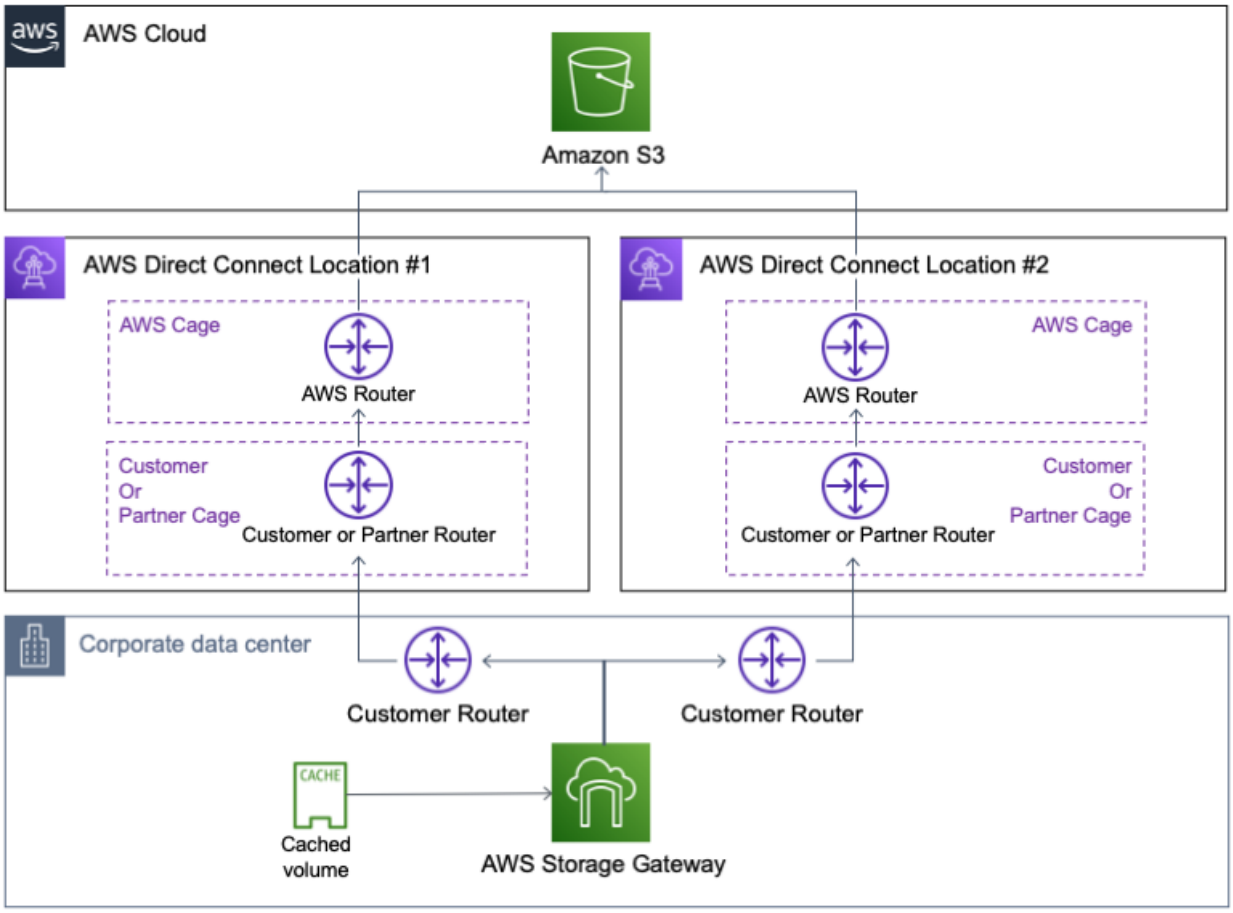

Figure 3 - Ongoing data migration from on-premises storage solution

In use case 2, a customer has a hybrid cloud deployment with data being used by both an on-premises environment and systems deployed in AWS. Additionally, the customer wants a dedicated connection to AWS that provides consistent network performance. As part of the on-going data migration, AWS Direct Connect acts as the backbone, providing a dedicated connection that bypasses the Internet to connect to AWS cloud. Additionally, the customer deploys AWS Storage Gateway with Gateway-Cached Volumes in the data center, which sends data to an Amazon S3 bucket in their target AWS region. The following steps describe the required steps to build this solution:

e. The customer creates an AWS Direct Connect connection between their corporate data center and the AWS Cloud.

a. To set up the connection using the Connection Wizard ordering type, the customer provides the following information using the AWS Direct Connect Console:

    i. Choose a resiliency level.

        1. Maximum Resiliency (for critical workloads): You can achieve maximum resiliency for critical workloads by using separate connections that terminate on separate devices in more than one location. This topology provides resiliency against device, connectivity, and complete location failures.

        2. High Resiliency (for critical workloads): You can achieve high resiliency for critical workloads by using two independent connections to multiple locations. This topology provides resiliency against connectivity failures caused by a fiber cut or a device failure. It also helps prevent a complete location failure.

        3. Development and Test (non-critical or test/dev workloads): You can achieve development and test resiliency for non-critical workloads by using separate connections that terminate on separate devices in one location. This topology provides resiliency against device failure, but does not provide resiliency against location failure.

    ii. Enter connection settings:

        1. Bandwidth – choose from 1Gbps to 100Gbps
        2. First location – the first physical location for your first Direct Connect connection
        3. First location service provider
        4. Second location – the second physical location for your second Direct Connect connection
        5. Second location service provider

    iii. Review and create menu: confirm your selections and click create.

b. After the customer creates a connection using the AWS Direct Connect console, AWS will send an email within 72 hours. The email will include a

Letter of Authorization and Connecting Facility Assignment (LOA-CFA). After receiving the LOA-CFA, the customer will forward it to their network provider so they can order a cross connect for the customer. The customer is not able to order a cross connect for themselves in the AWS Direct Connect location if the customer does not already have equipment there. The network provider will have to do this for the customer.

   c. After the physical connection is set up, the customer creates the virtual interfaces within AWS Direct Connect to connect to AWS public services, such Amazon S3.

   d. After creating virtual interfaces, the customer runs the AWS Direct Connect failover test to make sure that traffic routes to alternate online virtual interfaces.

2. After the AWS Direct Connect connection is setup, the customer creates an Amazon S3 bucket into which the on-premises data can be backed up.

3. The customer deploys the AWS Storage Gateway in their existing data center using following steps:

   a. Deploy a new gateway using AWS Storage Gateway console.

   b. Select Volume Gateway-Cached volumes for the type of gateway.

   c. Download the gateway virtual machine (VM) image and deploy on the on-premises virtualization environment.

   d. Provision two local disks to be attached to the VM.

   e. After the gateway VM is powered on, record the IP address of the machine, and then enter the IP address in the AWS Storage Gateway console to activate the gateway.

4. After the gateway is activated, the customer can configure the volume gateway in the AWS Storage Gateway console:

   a. Configure the local storage by selecting one of the two local disks attached to the storage gateway VM to be used as the upload buffer and cache storage.

   b. Create volumes on the Amazon S3 bucket.

5. The customer connects the Amazon S3 gateway volume as an iSCSI connection through the storage gateway IP address on a client machine.

6. After setup is completed and the customer applications write data to the storage volumes in AWS, the gateway at first stores the data on the on-premises disks (referred to as *cache storage*) before uploading the data to Amazon S3. The cache storage acts as the on-premises durable store for data that is waiting to upload to Amazon S3 from the upload buffer. The cache storage also lets the gateway store the customer application's recently accessed data on-premises for low-latency access. If an application requests data, the gateway first checks the cache storage for the data before checking Amazon S3. To prepare for upload to Amazon S3, the gateway also stores incoming data in a staging area, referred to as an *upload buffer.* Storage Gateway uploads this buffer data over an encrypted Secure Sockets Layer (SSL) connection to AWS, where it is stored encrypted in Amazon S3.

Use Case 3: Continuous Streaming Data Ingestion

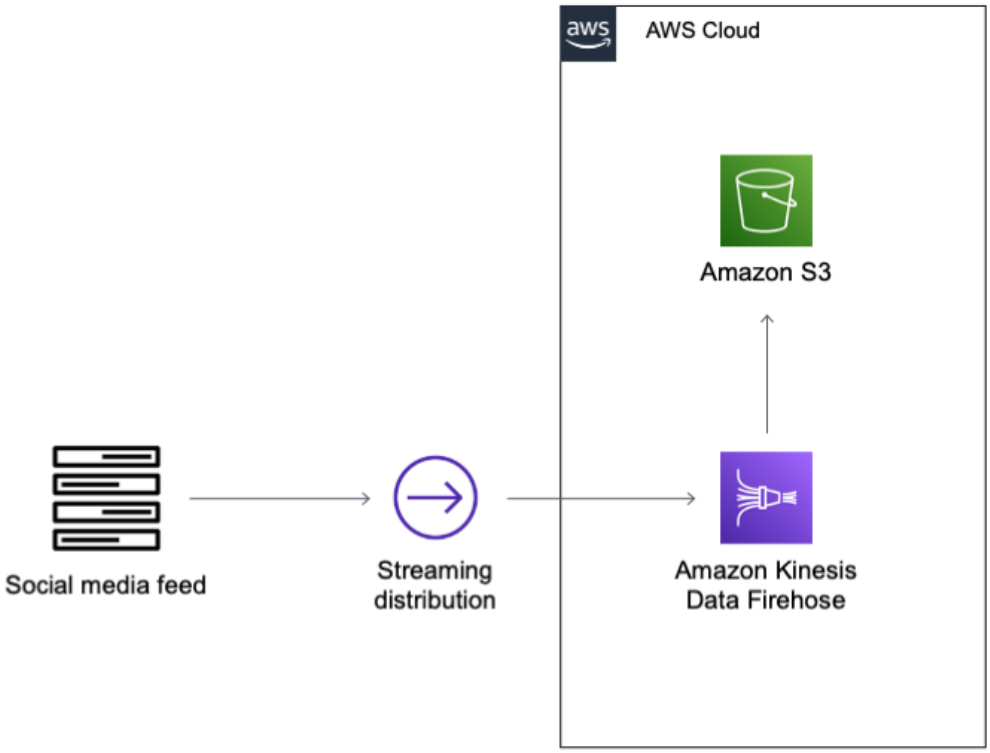

Figure 4 - Continuous streaming data ingestion

In use case 3, the customer wants to ingest a social media feed continuously in Amazon S3. As part of the continuous data migration, the customer uses Amazon Kinesis Data Firehose to ingest data without having to provision a dedicated set of servers.

1. The customer creates an Amazon Kinesis Data Firehose Delivery Stream, using the following steps in the Amazon Kinesis Data Firehose console:
    a. Choose the Delivery Stream name.
    b. Choose the Amazon S3 bucket; choose the IAM role that grants Firehose access to Amazon S3 bucket.
    c. Firehose buffers incoming records before delivering the data to Amazon S3. The customer chooses Buffer Size (1-128 MBs) or Buffer Interval (60-900 seconds). Whichever condition is satisfied first triggers the data delivery to Amazon S3.

   d. The customer chooses from three compression formats (GZIP, ZIP, or SNAPPY), or no data compression.

   e. The customer chooses whether to encrypt the data or not, with a key from the list of AWS Key Management Service (AWS KMS) keys that they own.

2. The customer sends the streaming data to an Amazon Kinesis Firehose delivery stream by writing appropriate code using AWS SDK.

Conclusion

This whitepaper walked you through different AWS managed and self-managed storage migration options. Additionally, the paper covered different use cases showing how multiple storage services can be used together to solve different migration needs.

Contributors

Contributors to this document include:

- Shruti Worlikar, Solutions Architect, Amazon Web Services
- Kevin Fernandez, Sr. Solutions Architect, Amazon Web Services
- Scott Wainner, Sr. Solutions Architect, Amazon Web Services

Further Reading

For additional information, see:

- AWS Direct Connect
- AWS Snow Family
- AWS Storage Gateway
- AWS Kinesis Data Firehose
- Storage Partner Solutions

Document revisions

Date	Description
July 13, 2021	Repaired broken links. Updated Time/Performance characteristics. Added decision tree. Added AWS Transfer Family. Updated with new AWS Snow Family services. Updated procedures in use cases.
May 2016	First publication