An Overview of AWS Cloud Data Migration Services

May 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

Cloud is a new normal in today's IT industry. One of the most challenging steps required to deploy an application infrastructure in the cloud involves the physics of moving data into and out of the cloud. Amazon Web Services (AWS) provides a number of services for moving data, and each solution offers various levels of speed, security, cost, and performance. This paper outlines the different AWS services that can help seamlessly transfer on-premises data to and from the AWS Cloud.

Introduction

As you plan your data migration strategy you'll 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 like one-time large batches, constant device streams, intermittent updates, or even hybrid data storage combining the AWS Cloud and on-premises data stores. These ways can be used individually or together to help streamline the realities of cloud data migration projects.

Cloud Data Migration Challenges

When you plan for a data migration, you need to determine how much data is being moved and how long the transfer will take over your existing Internet connection. The bandwidth that is used for data migration will not be available for your organization's typical Internet traffic. In addition, your organization might be concerned with moving sensitive business information from your internal network to a secure AWS environment. By determining the security level for your organization, you will be able to determine the AWS services you will need to use for your data migration.

Security/Data Sensitivity

When customers migrate data off-premises, they are concerned about the security of data while in transit. At AWS, we take security very seriously and build

security features into all of our data migration services. The following table lists these features.

AWS Services	Security Feature
AWS Direct Connect	• Provides a dedicated physical connection with no data transfer over the Internet. • Uses AWS Identity and Access Management (IAM) so you can control access to the AWS Direct Connect Management Console. • Integrates with AWS CloudTrail to capture API calls made by or on behalf of a customer account.
AWS Import/Export Snowball	• Uses IAM so you can control access to the AWS Import/Export Snowball console. • Integrates with the AWS Key Management Service (KMS) to encrypt data-at-rest that is stored on AWS Snowball. • 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 Snowball device.
AWS Storage Gateway	• IAM helps you provide security in controlling access to 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. • Authentication between your gateway and iSCSI initiators can be secured by using Challenge-Handshake Authentication Protocol (CHAP).
Amazon S3 Transfer Acceleration	• IAM helps you provide security in controlling access to Amazon Simple Storage Service (S3). • 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.

AWS Services	Security Feature
	• 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 Firehose	• IAM helps you provide security in controlling access to AWS Kinesis Firehose. • Data in transit can be secured by using SSL/TLS.

Cloud Data Migration Tools

As mentioned previously, you can select managed or unmanaged migration tools. You will make your choice based on your specific use case. This section discusses managed and unmanaged migration tools, with a brief description of how each solution works.

Time/Performance

When you migrate data from your on-premises storage to AWS storage services you want to take the least possible 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 = (Total Bytes)/(Megabits per second * 125 *
1000 * Network Utilization * 60 seconds * 60 minutes * 24
hours)
```

For example, if you have a T1 connection (1.544 Mbps) and 1 TB of data to move to AWS, theoretically, the minimum time it would take over the network connection at 80 percent utilization is 82 days.

```
(1024 * 1024 * 1024 * 1024)/(1.544 * 125 * 1000 * 0.80 * 60
* 60 * 24) = 82.42 days
```

If this amount of time is not practical for you, there are many ways you can migrate large amounts of data to AWS without taking days to do it. You can use AWS managed migration tools, which are suites of services that can help you optimize or replace your Internet connection to the AWS Cloud, or are able to provide friendly interfaces to Amazon Simple Storage Service (Amazon S3). For moving small amounts of data from your on-site location to the AWS Cloud, you can use easy, one-and-done methods discussed in the unmanaged AWS migration tools section.

For the best results we suggest the following:

Connection	Data Scale	Method
Less than 10Mbps	Less than 500GB	Unmanaged
More than 10Mbps	More than 500GB	Managed

Unmanaged Migration Tools

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 low 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 low amounts of data you can write commands using the Amazon Glacier AWS Command Line Interface to move data into Amazon Glacier. For archives greater than 100 MB in size, we recommend using multipart upload.

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 more sophisticated services specifically 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 will allow you to bypass Internet service providers (ISPs) in your network path. 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 1 GB or 10 GB network connection from your premises to AWS. You can use the connection to access Amazon Virtual Private Cloud (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 backbone that can be used to transfer data between your corporate network and AWS securely without ever having the data routed over the Internet.

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, which 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.

You can provision a single connection to any AWS Direct Connect location in the United States (US) and use it to access any of the AWS Regions in the US. The Regions are US East (Northern Virginia), US West (Northern California), US West (Oregon), and AWS GovCloud (US). Data transferred between AWS Regions flows over a network infrastructure maintained by AWS and does not flow across the public Internet.

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. You can find pricing information at the Amazon Direct Connect Pricing page.

AWS Import/Export Snowball

AWS Import/Export Snowball (AWS Snowball) accelerates moving large amounts of data into and out of AWS using secure Snowball appliances. The AWS Snowball appliance is purpose-built for efficient data storage and transfer. It is rugged enough to withstand a 6 G shock, and, at 50 pounds, it is light enough for one person to carry. It is entirely self-contained, with 110-volt power and a 10 GB network connection on the back, as well as an E Ink display and control panel on the front. Each AWS Snowball appliance is weather-resistant and serves as its own shipping container.

The AWS Snowball appliance is available in two capacity models (as of this writing) which can store either 50 or 80 terabytes of data. If you want to transfer

more data than that, you can use multiple appliances. For Amazon S3, individual files are loaded as objects and can range up to 5 TB in size. There is no limit to the number of objects you can place in Amazon S3, and the aggregate total amount of data that can be imported is virtually unlimited.

AWS transfers your data directly onto and off of AWS Snowball storage devices using Amazon's high-speed internal network, bypassing the Internet. For datasets of significant size, AWS Snowball is often faster than Internet transfer is and more cost-effective than upgrading your connectivity. 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 (EBS) 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 Import/Export Snowball.

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 Snowball can help ensure data is securely and cost-effectively transferred to AWS. In a content distribution scenario, you might use Snowball appliances if you regularly receive or need to share large amounts of data with clients, customers, or business associates. Snowball appliances can be sent directly from AWS to client or customer locations.

AWS Import/Export Snowball 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 Import/Export Snowball, 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 (the first 10 days of onsite usage are free), and

data transfer out. For the destination storage, the standard Amazon S3 storage pricing applies. You can find pricing information at the AWS Import/Export pricing page.

AWS Storage Gateway

The AWS Storage Gateway service 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 industry-standard storage protocols that work with your existing applications. It provides low-latency performance by maintaining frequently accessed data on-premises while securely storing all of your data encrypted in Amazon S3 or Amazon Glacier. For disaster recovery scenarios, AWS Storage Gateway, together with Amazon Elastic Compute Cloud (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.

With gateway-cached volumes, you can use Amazon S3 to hold your primary data, while retaining some portion of it locally in a cache for 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 TB 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 20 volumes and total volume storage of 150 TB. 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 primary data locally, 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 1 TB 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 12 volumes, with a total volume storage of 12 TB. 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 GB to 2.5 TB. A VTL can hold up to 1,500 virtual tapes, with a maximum aggregate capacity of 150 TB. 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. You can find pricing information at the AWS Storage Gateway pricing page.

Amazon S3 Transfer Acceleration (S3-XA)

Amazon S3 Transfer Acceleration (S3-XA) enables fast, easy, and secure transfers of files over long distances between your client and your Amazon S3 bucket. Transfer Acceleration leverages Amazon CloudFront's 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. You can find pricing information at the Amazon S3 pricing page.

AWS Kinesis Firehose

Amazon Kinesis 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, and Amazon Elasticsearch Service. Amazon Kinesis 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 Firehose can also batch, compress, and encrypt data before loading it. This process minimizes the amount of storage used at the destination and increases security.

You can use 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 as large as 1000 KB. Additionally, Firehose buffers incoming streaming data to a certain size called a *Buffer Size* (1 MB to 128 MB) or for a certain period of time called a *Buffer Interval* (60 to 900 seconds) before delivering to destinations.

With Amazon Kinesis Firehose, you pay only for the volume of data you transmit through the service. Amazon Kinesis Firehose has a single pricing component: data ingested (per GB), 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 KB. 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. You can find detailed pricing information at the Amazon Kinesis Firehose pricing page.

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

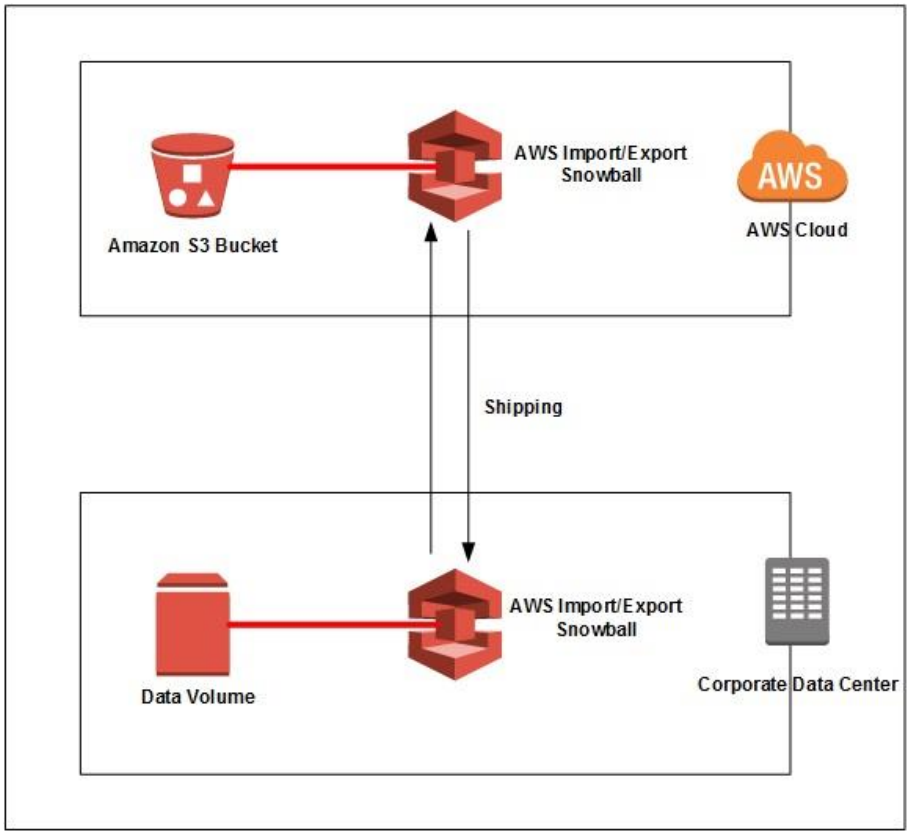

Figure 1: One-time massive data migration

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 capacity of 50 TB or 80 TB.
    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.
2. After the job is created, AWS ships the Snowball Appliances to the customer data center by AWS. Since the customer is importing 200 TB of data in Amazon S3, they will need to create three Import jobs of 80 TB Snowball capacity.
3. After receiving the Snowball appliance, the customer does the following tasks.
    a. Customer connects the powered-off appliance to their internal network, and attaches the power on the appliance.

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 Import/Export Snowball 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:

```
snowball start -i [Snowball IP Address] -m
[Path/to/manifest/file] -u [29 character unlock code]
```

   d. Begin transferring data onto the Snowball using the following copy (`cp`) command:

```
snowball cp [options] [path/to/data/source]
s3:///[path/to/data/destination]
```

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, and arrange UPS pickup for 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 also 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: Ongoing Data Migration from On-Premises Storage Solution

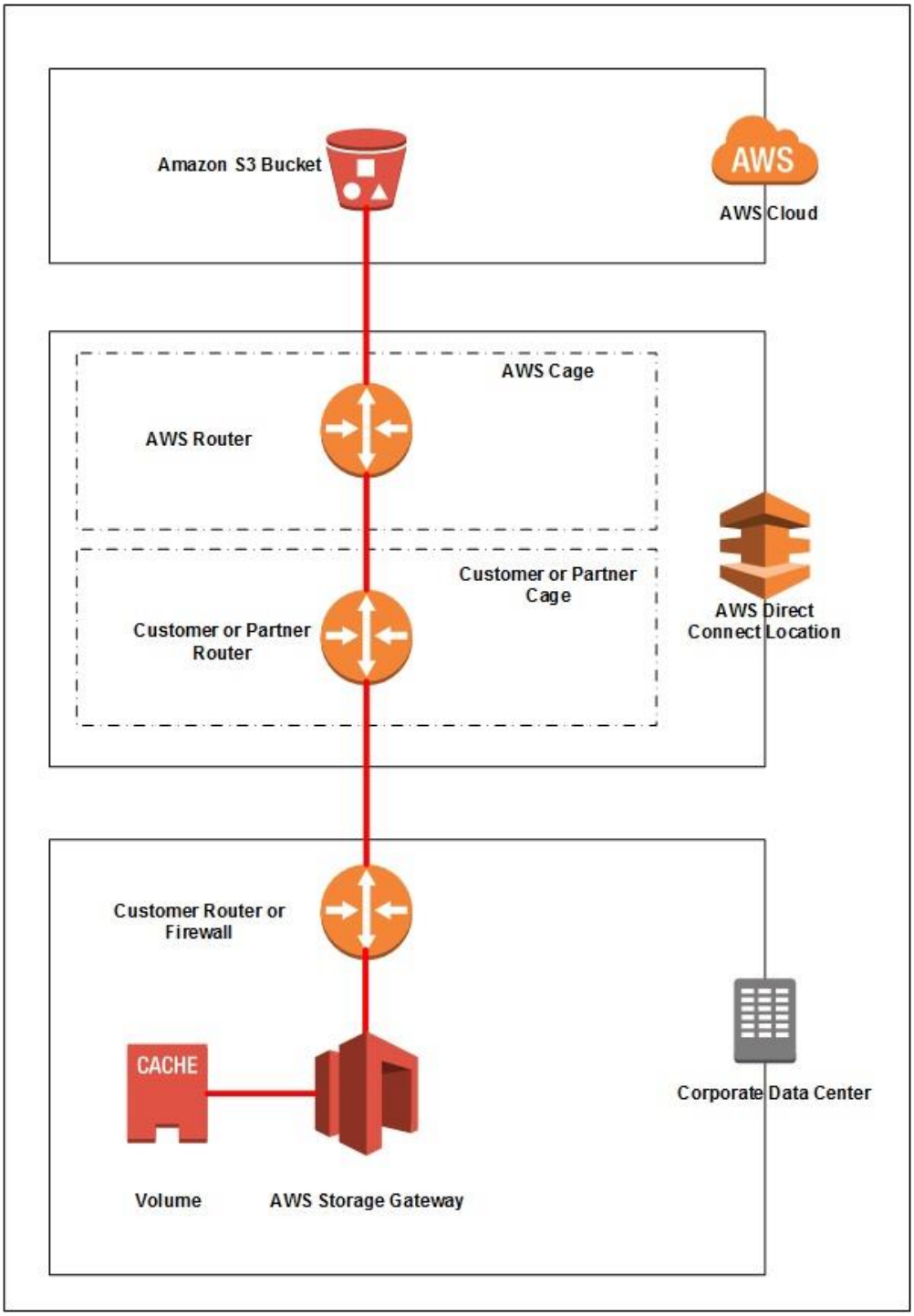

Figure 2: Ongoing data migration from on-premises storage solution

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 ISP to connect to AWS cloud. Additionally, AWS Storage Gateway is used to have a Gateway-Cached Volume deployed in the customer's data center, while backing up to an Amazon S3 bucket.

1. The customer creates an AWS Direct Connect connection between its own corporate data center and the AWS cloud.

    a. To set up the connection, the customer provides the following information using the AWS Direct Connect console:

        i. Customer contact information.

        ii. AWS Direct Connect location to connect to. If a customer doesn't have a presence in one of the AWS Direct Connect locations, they will need to work with partner in the AWS Partner Network (APN) to help establish network circuits between an AWS Direct Connect location and their data center, office, or colocation environment. Or, provide a colocation space within the same facility as the AWS Direct Connect location.

        iii. If the services of an AWS Direct Connect partner who is a member of the AWS Partner Network (APN) are needed.

        iv. The port speed that is required, either 1 Gbps or 10 Gbps. For port speeds less than 1 G, contact an APN partner who supports AWS Direct Connect.

    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 they do not 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 interface within AWS Direct Connect to connect to AWS public services, such Amazon S3.

d. After creating the virtual interface, the customer downloads the router configuration file based on the vendor, platform, and Software of the router.

e. The customer uses the appropriate router configuration based on the file downloaded to ensure connectivity to AWS Direct Connect.

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 Gateway-Cached volumes for the type of gateway.

   c. Download the gateway virtual machine (VM) 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

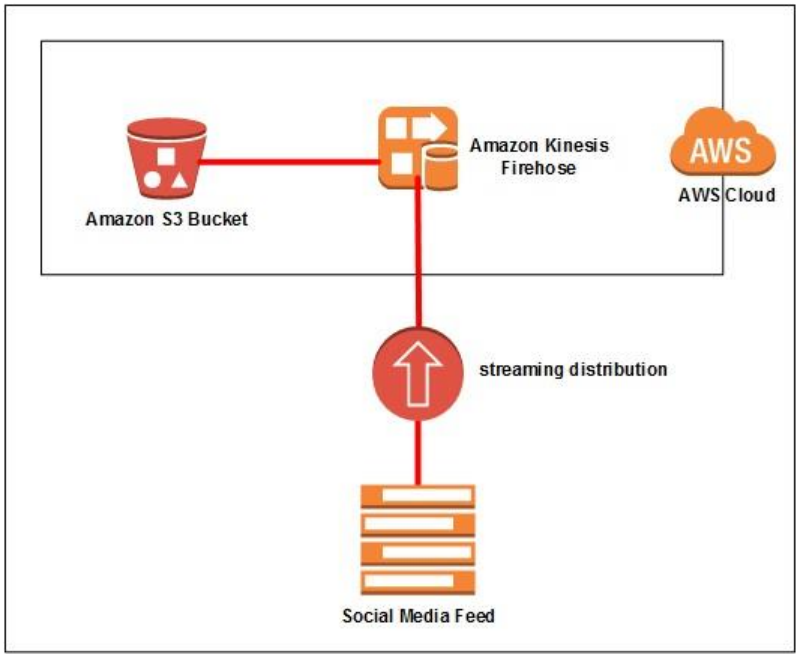

Figure 3: Continuous streaming data ingestion

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 Firehose to ingest data without having to provision a dedicated set of servers.

1. The customer creates an Amazon Kinesis Firehose Delivery Stream, using the following steps in the Amazon Kinesis 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 customers 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

In this whitepaper we walked through different AWS managed and unmanaged storage migration options. Additionally we covered different use cases showing how multiple storage services can be used together to solve different migration needs.

Contributors

The following individuals and organizations contributed to this document:

- Shruti Worlikar, solutions architect, Amazon Web Services

Further Reading

For additional help, please consult the following sources:

- AWS Direct Connect
- AWS Import/Export Snowball
- AWS Storage Gateway
- AWS Kinesis Firehose
- Storage Partner Solutions