**Hybrid Storage and Data Migration with AWS Storage Gateway File Gateway**

Here, you will attach a Network File System (NFS) mount to an on-premises data storage using the AWS Storage Gateway File Gateway service. After that, you'll copy the data to an Amazon S3 bucket. Next, you will set up more advanced Amazon S3 features like cross-region replication and Amazon S3 lifecycle policies.

At the end, you should be able to:

* Configure a File Gateway with an NFS file share and attach it to a Linux instance
* Migrate a set of data from the Linux instance to an S3 bucket
* Create and configure a primary S3 bucket to migrate on-premises server data to AWS
* Create and configure a secondary S3 bucket to use for cross-region replication
* Create an S3 lifecycle policy to manage data in a bucket automatically

**Pre-requisites for this lab**

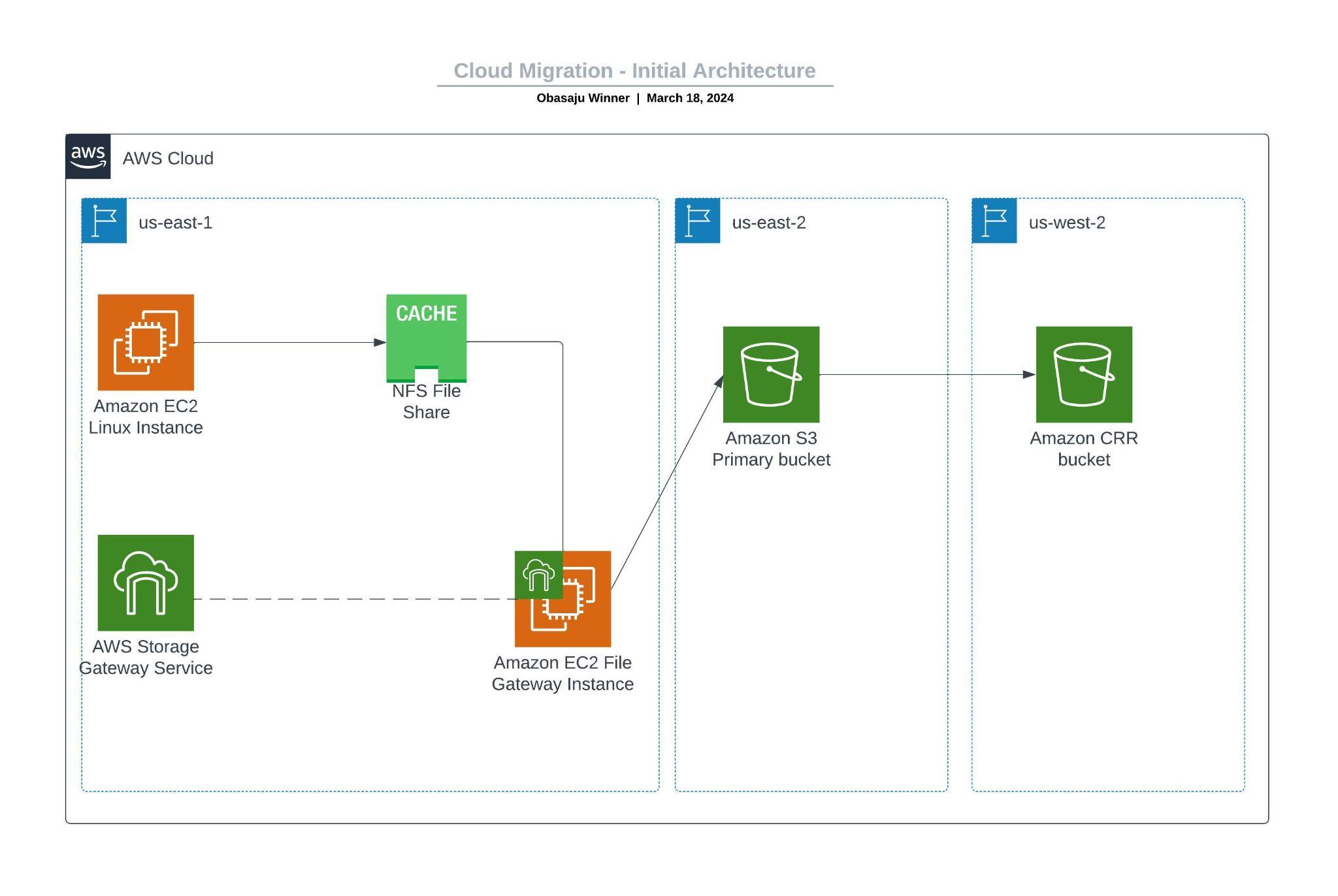
Three AWS Regions are used in this lab setting. To simulate an on-premises server, a Linux EC2 instance is deployed to the us-east-1 (N.Virginia) Region. The Linux server and Storage Gateway virtual appliance are both deployed to the same region. The appliance would be installed as a physical Storage Gateway appliance or in an environment using Microsoft Hyper-V or VMware vSphere.

The primary S3 bucket is created in the US-east-2 (Ohio) Region. Data from the Linux host is copied to the primary S3 bucket. This bucket can also be called the *source*.

The secondary S3 bucket is created in the US-west-2 region (Oregon). This secondary bucket is the target for the cross-region replication policy. It can also be called the *destination.*

Here’s the initial architecture

**Creating the primary and secondary S3 buckets**



Before you configure the File Gateway, you must create the primary S3 bucket (or the source) where you will replicate the data. You will also create the secondary bucket (or the destination) that will be used for cross-Region replication.

* In the search box to the right of **Services**, search for and choose **S3** to open the S3 console.
* Choose ***Create bucket*** then configure these settings:
  + **Bucket name:** Create a name that you can remember easily. It must be globally unique. e.g (my-source)
  + **Region:** US East (Ohio) *us-east-2*
  + **Bucket Versioning:** *Enable*

You must enable versioning for both the source and destination buckets for cross-Region replication.

* Choose ***Create bucket***

Repeat the previous steps in this task to create a second bucket with the following configuration:

* **Bucket name:** Create a name you can easily remember. It must be globally unique. e.g (my-destination)
* **Region** US West (Oregon) us-west-2
* **Versioning:** Enable

**Enabling cross-Region replication**

Now that you have created your two S3 buckets and enabled versioning on them, you can create a replication policy.

* Select the name of the *source* bucket (*my-source*) that you created in the US East (Ohio) Region.
* Select the **Management** tab and under *Replication* rules select ***Create replication rule***
* Configure the Replication rule:
  + **Replication rule name:** crr-full-bucket
  + **Status:** *Enabled*
  + **Source bucket:**
    - For **Choose a rule scope**, select *Apply to all objects in the bucket*
  + **Destination:**
    - *Choose a bucket in this account*
    - Choose **Browse S3** and select the bucket you created in the US West (Oregon) Region (*my-destination*).
    - Select **Choose path**
    - **IAM role:** S3-CRR-Role
      * **Note:** To find the AWS IAM role, in the search box, enter: S3-CRR (This role was pre-created with the required permissions for this lab)
* Choose **Save**. When prompted, if you want to replicate existing objects, choose **No**, and then choose **Submit**
* Return to and select the link to the bucket you created in the US East (Ohio) Region (my-source).
* Choose **Upload** to upload a file from your local computer to the bucket.

For this lab, use a small file that does not contain sensitive information, such as a blank text file.

* Choose **Add files**, locate and open the file, then choose **Upload**
* Wait for the file to upload, then choose **Close**. Return to the bucket you created in the US West (Oregon) Region (my-destination).

The file that you uploaded should also now have been copied to this bucket.

**Note:** You may need to refresh the console for the object to appear.

**Configuring the File Gateway and creating an NFS file share**

In this task, you will set up the File Gateway appliance as an Amazon EC2 instance. You will next set up a cache disk, choose an S3 bucket to synchronize your on-premises files with, and select an IAM policy to utilize. Finally, you'll set up an NFS file sharing on the File Gateway.

* In the search box to the right of **Services,** search for and choose **Storage Gateway** to open the Storage Gateway console.
* At the top-right of the console, verify that the current Region is *N. Virginia*.
* Choose **Create Gateway** then begin configuring the **Step 1: Set up gateway** settings:
  + Gateway name: *File Gateway*
  + Gateway time zone: Choose **GMT -5:00 Eastern Time (US & Canada), Bogota, Lima**
  + Gateway type: **Amazon S3 File Gateway**
  + Host platform: Choose **Amazon EC2**, then choose the **Launch instance** button.

A new tab opens to the EC2 instance launch wizard. This link automatically selects the correct Amazon Machine Image (AMI) that must be used for the File Gateway appliance.

* In the Launch an Instance screen, begin configuring the gateway as described:
  + Name: *File Gateway Appliance*
  + **AMI from catalog**: Accept the default aws-storage-gateway AMI
  + **Instance type:** Select the **t2.xlarge** instance type.
  + **Key pair name - *required*:** Choose or create a key pair (**vockey**).
* Configure the network and security group settings for the gateway.
  + Next to *Network settings*, choose **Edit**, then configure:
    - **VPC**: On-Prem-VPC
    - **Subnet**: On-Prem-Subnet
    - **Auto-assign public IP**: *Enable*
    - Under Firewall (security groups), choose **select an existing security group**
  + For **Common security groups:**
    - Select the security group with **FileGatewayAccess** in the name

**Note**: This security group is configured to allow traffic through ports 80 (HTTP), 443 (HTTPS), 53 (DNS), 123 (NTP), and 2049 (NFS). These ports enable the activation of the File Gateway appliance. They also enable connectivity from the Linux server to the NFS share that you will create on the File Gateway.

* + - Also, select the security group with **OnPremSshAccess** in the name.

**Note**: This security group is configured to allow Secure Shell (SSH) connections on port 22.

* + - Verify that both security group now appear as selected (details on each will appear in boxes in the console).

**Tip**: You may need to choose **Show all selected to** see them both.

* Configure the storage settings for the gateway
  + In the Configure storage panel, notice there is already an entry to create one *80GiB* root volume
  + Choose **Add new volume**
  + Set the size of the EBS volume to *150GiB*
* Finish creating the gateway
  + In the *Summary* panel on the right, keep the number of instances set to 1, and choose **Launch instance**

A Success message displays

* + Choose **View all instances**

Your *File Gateway Appliance* instance will take a few minutes to initialize.

* Monitor the status of the deployment and wait for **Status Checks** to complete.

**Tip:** Choose the refresh button to more quickly learn the status of the instance.

* Select your File Gateway instance, then in the *Details* tab below, locate the **Public IPV4 address** and copy it.

You will use this IP address when you complete the File Gateway deployment.

* Return to the **AWS Storage Gateway** tab in your browser. It should still be at the **Set up gateway on Amazon EC2** screen.
* Check the box next to *I completed all the steps above and launched the EC2 instance*, then choose **Next**
* Configure the **Step 2: Connect to AWS** settings:
  + For the **Service endpoint,** select **Publicly accessible**, and then choose **Next**
  + In the Gateway connection options:
  + For **IP address,** paste in the **IPV4 Public IP** address you copied from your File Gateway Appliance instance.
  + Choose **Next**
* In the **Step 3: Review and activate** settings screen choose **Next**
* Configure the **Step 4: Configure Gateway** settings:
  + CloudWatch log group: **Deactivate logging**
  + CloudWatch alarms: **No Alarm**
  + Choose **Configure**

A *Successfully activated gateway File Gateway Appliance* message displays. In the Configure cache storage panel, you will see a message showing the local disks loading.

* + Wait for the local disk's status to show that it finished processing (approximately 1 minute).
  + After the processing is complete, go to **Allocated to** and select **Cache.**
  + Choose **Save changes**
* Start creating a file share.
  + Wait for the File Gateway status to change to *Running (approximately* 1-2 minutes*)*
  + From the left side panel, choose **File Shares.**
  + Choose **Create file share**
* On the **File share settings configuration** screen, configure these settings:
  + **Gateway:** Select the name of the File Gateway that you created (which should be *File Gateway Appliance*)
  + **Amazon S3 bucket name:** Enter the name of the source bucket that you created in the US East (Ohio) us-east-2 Region in Task 1 (my-source).
  + **AWS region:** *US East (Ohio) us-east-2*
  + **Access objects using:** *Network File System (NFS)*
  + Choose **Next**
* On the **Amazon S3 storage settings** screen, configure these settings:
  + **Storage class for new objects:***S3 Standard*
  + **Object metadata:**
    - Check box ***Guess MIME type***
    - Check box ***Give bucket owner full control***
    - Uncheck *Enable Requester pays*
  + **Access your S3 bucket:** *Use an existing IAM role*
  + **IAM role:** Paste the *FgwIamPolicyARN,* which you can retrieve by following these instructions -
    - Choose the **Details** dropdown menu above these instructions
    - Select **Show**
    - Copy the *FgwIamPolicyARN* value
  + Choose **Next**
* In the File access settings screen, accept the default settings.

**Note:** You might get a warning message that the file share is accessible from anywhere. For this lab, you can safely disregard this warning. In a production environment, you should always create policies that are as restrictive as possible to prevent unwanted or malicious connections to your instances.

* + Choose **Next**
* Scroll to the bottom of the *Review and Create* screen, then select **Create**

Monitor the status of the deployment and wait for **Status** to change to *Available*, which takes less than a minute.

**Note:** You can choose the refresh button occasionally to notice more quickly when the status has changed.

This completes your **Storage gateway** creation.

* Select the file share that you just created by choosing the link.
* At the bottom of the screen, note the command to mount the file share on Linux. You will need it for the next task.

**Mounting the file share to the Linux instance and migrating the data**

Before you can migrate data to the NFS share that you created, you must first mount the share. In this task, you will mount the NFS share on a Linux server, and then copy data to the share.

* Connect to the **On-Prem Linux Server** instance.

For Windows users, choose the **Download PPK** button and save the **labsuser.ppk** file. Note the **OnPremLinuxInstance** address, if it is displayed.

For Linux and MacOS users, choose the **Download PEM** button and save the **labsuser.pem** file. Note the **OnPremLinuxInstance** address, if it is displayed.

* Open a terminal window, and change the directory to the directory where the *labsuser.pem* file was downloaded by using the cd command.

For example, if the labsuser.pem file was saved to your **Downloads** directory, run this command:

cd ~/Downloads

* Change the permissions on the key to be read-only, by running this command:

chmod 400 labsuser.pem

* Run the following command (replace **<public-ip>** with the **OnPremLinuxInstance** address that you copied earlier).
  + Alternatively, to find the IP address of the on-premises instance, return to the Amazon EC2 console and select **Instances**
  + Select the **On-Prem Linux Server** instance that you want to connect to
  + In the **Details** tab, copy the **Public IPV4 address** value

ssh -i labsuser.pem ec2-user@<public-ip>

* When you are prompted to allow the first connection to this remote SSH server, enter *yes*.

Because you are using a key pair for authentication, you are not prompted for a password.

**You should now be connected to the instance.**

* On the Linux instance, to view the data that exists on this server, enter the following command:

ls /media/data

You should see 20 image files in the .png format.

* Create the directory that will be used to synchronize data with your S3 bucket by using the following command:

sudo mkdir -p /mnt/nfs/s3

* Mount the file share on the Linux instance by using the command that you located in the Storage Gateway file shares details screen at the end of the last task.

sudo mount -t nfs -o nolock,hard <File-Gateway-appliance-private-IP-address>:/<S3-bucket-name> /mnt/nfs/s3

Notice that the command starts with sudo and ends with /mnt/nfs/s3

For example:

sudo mount -t nfs -o nolock,hard 10.10.1.33:/my-source /mnt/nfs/s3

* Verify that the share was mounted correctly by entering the following command:

df -h

The output of the command should similar to the following example:

[ec2-user@ip-10-10-1-210 ~]$ df -h

Filesystem Size Used Avail Use% Mounted on

devtmpfs 483M 64K 483M 1% /dev

tmpfs 493M 0 493M 0% /dev/shm

/dev/xvda1 7.8G 1.1G 6.6G 14% /

10.10.1.33:/my-source 8.80E 0 8.0E 0% /mnt/nfs/s3

* Now that you created the mount point, you can copy the data that you want to migrate to Amazon S3 into the share by using this command:

sudo cp -v /media/data/\*.png /mnt/nfs/s3

**Verifying that the data is migrated**

You have finished configuring the gateway and copying data into the NFS share. Now, you will verify that the configuration works as intended.

* In the **Services** search box, search for and choose **S3** to open the S3 console.
* Select the bucket that you created in the *US East (Ohio)* Region.
  + Verify that the 20 image files are listed.

**Note:** You might need to choose the refresh icon in the S3 console.

* Return to the **Buckets** page and select the bucket that you created in the *US West (Oregon)* Region.
  + Verify that the image files were replicated to this bucket, based on the policy that you created earlier.

**Note:** S3 Object replication can take up to 15 minutes to complete. Keep refreshing until you see the replicated objects.

Congratulations, you successfully migrated data to Amazon S3 by using AWS Storage Gateway in the File Gateway mode. After your data is stored in Amazon S3, you can act on it like native Amazon S3 data. In this lab, you created a replication policy to copy the data to a secondary Region. You could also perform other operations, such as configuring a lifecycle policy. For example, you could migrate infrequently used data automatically from S3 Standard to Amazon S3 Glacier for long-term storage, which can reduce costs.