

MSBA Cloud Computing Storage Lab

In this lab, you will work interactively with several AWS storage services. We will be using the Simple Storage Service (S3), an object-based file store to store the photos you collected and host them on the web. You will also explore the logging and versioning capabilities of S3. Finally, you will participate in class exercises to load data onto an AWS Snowball device for migration to the cloud.

I. Creating an S3 Bucket

S3 stores objects in buckets. You can think of these as quite similar to a file folder. Each bucket must have a name that is globally unique, meaning that it must not be used by any other AWS user, anywhere in the world.

1. From the AWS Console, navigate to the S3 section and create an S3 bucket.
 - Use the following naming convention: mendozacloud-*netid* (e.g. mendozacloud-mchapple)
 - Create your bucket in the US East (N. Virginia) region. (Your starter account is limited to creating resources in this region).
 - Configure your bucket to enable ACLs with the “Bucket owner preferred” setting, as shown below:

Object Ownership [Info](#)

Control ownership of objects written to this bucket from other AWS accounts and granted using access control lists (ACLs). Object ownership determines who can specify access to objects.

☐ **ACLs disabled (recommended)**
All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.

☒ **ACLs enabled**
Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.

Object Ownership

☒ **Bucket owner preferred**
If new objects written to this bucket specify the bucket-owner-full-control canned ACL, they are owned by the bucket owner. Otherwise, they are owned by the object writer.

☐ **Object writer**
The object writer remains the object owner.

- Enable versioning for your bucket. Otherwise, accept the default options and create the bucket.
2. Upload your ten photos to the bucket that you just created. Use the Standard storage class and accept all other default options.
 3. Each object that you store in an S3 bucket is assigned a URL. Find the URL for one of the photos that you uploaded and attempt to access it in a web browser. What

happened?

4. By way of comparison, here is a photo that I shared in an S3 bucket. Try to access it:
 - <https://mendozcloud.s3.amazonaws.com/mcobship.jpeg>

II. Making an S3 Object Public

The reason that you were not able to access your photo in step 3 above while you were able to access my photo in step 4 is that S3 objects are private by default. To access an object on the web, you must deliberately make it publicly accessible. Select one of your photos that you don't mind making publicly accessible and do so.

Before you make a photo public, you will need to edit the public access settings for the bucket itself. AWS wants to be sure that you really intend to publish materials on the web, so they require that you set the public access settings for the bucket to allow objects to be publicly published. You will find these settings on the permissions tab for your bucket. Change your bucket settings so that they appear as below:

Block public access (bucket settings)

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to all your S3 buckets and objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

☐ **Block all public access**
 Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

☐ **Block public access to buckets and objects granted through new access control lists (ACLs)**
 S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.

☐ **Block public access to buckets and objects granted through any access control lists (ACLs)**
 S3 will ignore all ACLs that grant public access to buckets and objects.

☐ **Block public access to buckets and objects granted through new public bucket or access point policies**
 S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.

☐ **Block public and cross-account access to buckets and objects through any public bucket or access point policies**
 S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

Cancel Save changes

Next, find the correct place to adjust the settings for the specific photo of your choice and do so. (Note that you will want to adjust the settings for Objects in the bucket. You will also see the option to adjust settings for Object ACLs -- that's not what you're looking for.)

After you have done this, update the [Class Photo List](#) with a link to your photo.

III. Bucket Logging

With publicly-accessible images, you may want to keep track of who is accessing them. This can be done by enabling server access logging, which stores details about requests. In order to set up access logging, you first need to create a bucket to contain the logs.

1. Before beginning, read this [overview of S3 bucket access logs](#).
2. Create a bucket for logging, using the following naming convention: mendozacloud-*netid*-accesslogs (e.g. mendozacloud-mchapple-accesslogs). While you may use all of the same settings as the bucket you previously created, there is no need to enable bucket versioning.
3. On the Permissions tab, edit the access control list (ACL) for the accesslogs bucket to grant Amazon S3 Log Delivery group read and write access to this bucket and Objects list and write access to this bucket, as shown below:

Edit access control list (ACL) [info](#)

Access control list (ACL)
Grant basic read/write permissions to other AWS accounts. [Learn more](#)

Grantee	Objects	Bucket ACL
Bucket owner (your AWS account) Canonical ID: a35a8ac67285089159919f1964720f879f544c97a5b0893372adb53dc0db1c7b	<input checked="" type="checkbox"/> List <input checked="" type="checkbox"/> Write	<input checked="" type="checkbox"/> Read <input checked="" type="checkbox"/> Write
Everyone (public access) Group: http://acs.amazonaws.com/groups/global/AllUsers	<input type="checkbox"/> List <input type="checkbox"/> Write	<input type="checkbox"/> Read <input type="checkbox"/> Write
Authenticated users group (anyone with an AWS account) Group: http://acs.amazonaws.com/groups/global/AuthenticatedUsers	<input type="checkbox"/> List <input type="checkbox"/> Write	<input type="checkbox"/> Read <input type="checkbox"/> Write
S3 log delivery group Group: http://acs.amazonaws.com/groups/s3/LogDelivery	<input checked="" type="checkbox"/> List <input checked="" type="checkbox"/> Write	<input checked="" type="checkbox"/> Read <input checked="" type="checkbox"/> Write

Access for other AWS accounts
No other AWS accounts associated with the resource.

[Add grantee](#)

4. Change the properties of your original S3 bucket to enable logging to the bucket you just created.

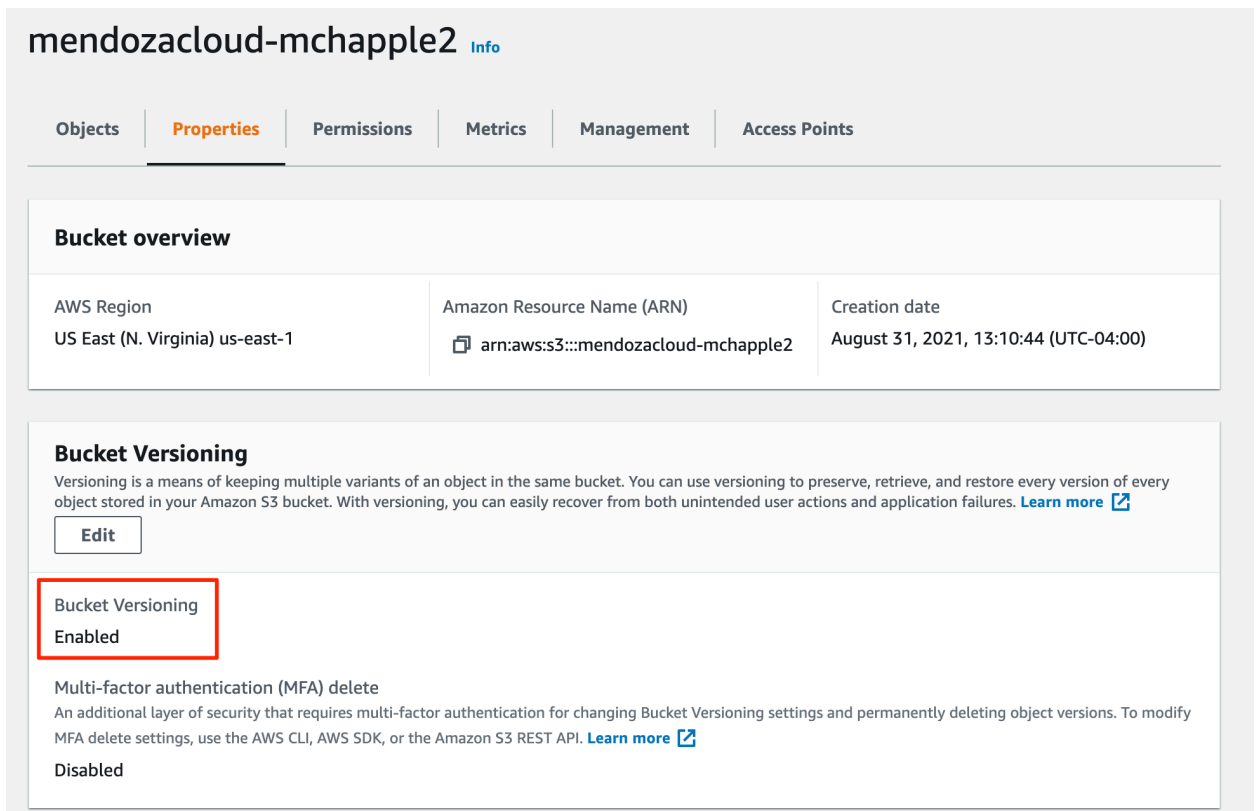
5. Use the public URL for the image you created earlier to access the image in a web browser. This will create at least one log entry for you to examine later.

Don't worry if you don't see log files delivered immediately to your accesslogs bucket. S3 begins generating log entries as soon as you enable logging, but there may be a delay of up to several hours before they are delivered to your bucket.

IV. Bucket Versioning

There are times where it makes sense to use the versioning feature of S3 to store changes to objects over time.


1. Before beginning, read this [overview of S3 versioning](#).
2. In a new browser tab, open the photo you made public up above.
3. Look at the properties on your mendozacloud-netid bucket to verify that versioning is enabled:




mendozacloud-mchapple2 [Info](#)

Objects | **Properties** | Permissions | Metrics | Management | Access Points

Bucket overview

AWS Region US East (N. Virginia) us-east-1	Amazon Resource Name (ARN)  arn:aws:s3::mendozacloud-mchapple2	Creation date August 31, 2021, 13:10:44 (UTC-04:00)
---	--	--


Bucket Versioning

Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures. [Learn more](#) 

[Edit](#)

Bucket Versioning
Enabled

Multi-factor authentication (MFA) delete

An additional layer of security that requires multi-factor authentication for changing Bucket Versioning settings and permanently deleting object versions. To modify MFA delete settings, use the AWS CLI, AWS SDK, or the Amazon S3 REST API. [Learn more](#) 

Disabled

4. On your local computer, edit one of the photos you uploaded to S3 so that it is obviously different from the original. This does not need to be wonderfully artistic, but you are

welcome to go crazy here if you'd like.

5. Upload the locally-modified photo to your mendozacloud-*netid* bucket. Make sure that you use the *exact* same filename.
6. Refresh the tab with the picture to ensure your local edits are visible. If you receive an error message, troubleshoot it and correct the problem.
7. Make another local edit to the image and upload it again (once again using the same filename). Refresh the browser tab once again - it should show the third version of the photograph.
8. Using S3 versioning, make the second version of your photograph the current version of the S3 object. If you're having trouble figuring out how to do this, [read working with objects in a versioning-enabled bucket](#). Since we are using the web console, you will need to download the version of the object that you want to display to your computer and then reupload it with the same filename.
9. Refresh your browser tab and verify that you now see the second version of your photo.

Note: Your browser may initially display a cached version of the image. If this happens, try opening the URL in a private mode/incognito tab.

V. Viewing Other Photos

Visit the [Class Photo List](#) again and take a few minutes to click on each photo on the list. This will generate the access log entries required for everyone to complete their lab report.

VI. Storing a Dataset in S3

In the previous parts of this lab, you used S3 to store photos that you will incorporate into web content for future labs. S3 may also be used to store datasets. In fact, the datasets that you used in the Data Management course all used CSV files that were stored in a public S3 bucket.

Now that you are familiar with the use of S3, use it to store a dataset of your choice. You may use any dataset of your choosing, or simply create a CSV file containing fictitious data. Be sure that whatever dataset you choose is one that you have permission to share publicly (i.e. don't use a confidential dataset from your job!)

1. Put the dataset in your S3 bucket and set it to allow public access.
2. Write an R script that reads the dataset directly from S3 and performs a summary() command on it. Take a screenshot showing the script and its successful execution. You will need this screenshot for your lab report.

You are now finished with the in-class portion of the lab.

VII. Reviewing Log Entries

Wait at least four hours to allow the log files from class to be delivered to the accesslogs bucket that you created.

Review the files that were delivered to the bucket. You may find it helpful to review this [description of the S3 log format](#). Locate two log entries that correspond to someone accessing one of your photo files. You will see the keyword GET in the log entry as well as the name of your file.

A word of caution: S3 logs are quite noisy. You will find many of the files contain only a single entry that does not correspond to an actual file access.

When you find those entries, write a short description of each that answers the following questions:

1. What file was accessed?
2. What is the IP address of the device that accessed the file?
3. What time (in ET) did the access occur?
4. What web browser was used to access the file?

VIII. Deliverable

Your deliverable from this in-class lab is a brief lab report describing your results, submitted to Canvas as a PDF document. Your report should contain two sections:

Methodology

Provide a concise narrative explaining what you did in the lab. You should include the following elements from your work:

1. Your name
2. Public URL for the modified image that you created.
3. Public URL for the original version of the image.
4. Public URL for the dataset you created in part VI.
5. Screenshot of RStudio that you created in part VI.
6. Cut-and-paste the two log entries that you selected in part VII.
7. Your interpretation of the two log entries from part VII.

Please note that this section should not simply be the items above pasted into a report. You should provide a concise narrative explaining what you did and demonstrating that you understood the steps you were following, rather than simply clicking through the lab report.

Analysis

Answer each of the following questions. This section of the lab report should be divided into sections with a separate section answering each question.

- a. In this lab, we turned on server access logging. Why do you think organizations would want to turn this feature on? Can you think of scenarios where it might not be a good idea to turn logging on? Why?
- b. In this lab, we turned on versioning. Why do you think organizations would want to turn this feature on? Can you think of scenarios where it might not be a good idea to turn versioning on? Why?
- c. People often use S3 to host websites. In what scenarios do you think S3 would be an appropriate hosting environment? Why? In what scenarios do you think S3 would not meet the needs of an organization? Why?
- d. Research the different S3 storage classes. Describe one scenario where it would be appropriate to use each of the following storage classes and explain why it is the best choice for that scenario:
 - i. S3 Standard
 - ii. S3 Standard-IA
 - iii. S3 Intelligent-Tiering (assume that you have not enabled any of the optional tiers of service)
 - iv. S3 Glacier Flexible Retrieval
- e. Imagine that you are hosting a single 10GB file in S3 using the US East (N. Virginia) region. This file is accessed 1,000 times per day by Internet users using GET requests. Determine the cost of hosting this file in each of the following services during the month of June, assuming that the account you are using is not eligible for the AWS Free Tier:
 - i. S3 Standard
 - ii. S3 Standard-IA
 - iii. S3 Intelligent-Tiering (assume that you have not enabled any of the optional tiers of service)

You will need to review the [S3 pricing](#) page to answer this question. For the purposes of simplifying the calculation, assume that 1 TB = 1,000 GB. Include the following costs in your analysis:

Include the following costs in your analysis (where applicable):

- Storage costs
 - GET request costs
 - Data transfer costs
 - Monitoring and automation costs
- f. Imagine that you are storing 75 TB of data in S3 using the US East (N. Virginia) region, across 100,000 objects. This data is archival data that will not be accessed regularly, if at all. Determine the annual cost of hosting this data in each of the following services, assuming that the data is not accessed that year. Assume that the data has already been stored in the S3 bucket during the prior year and was not accessed during that time period either. Also assume that the account is not eligible for the AWS Free Tier.
- i. S3 Standard
 - ii. S3 Standard-IA
 - iii. S3 Intelligent-Tiering (assume that you have not enabled any of the optional tiers of service)
 - iv. S3 Glacier Flexible Retrieval
 - v. S3 Glacier Deep Archive