MSBA Storage Lab

Team 1 (Andrea Ayala, Patty Kingery, Bryce Williams, Miguel Garcia, Patrick Grodach)

6/8/2024

Methodology

To start this lab we created an S3 bucket through AWS to store 10 jpeg images. Upon first upload we tried to access the photo through the URL. Through this process we learned that by default, the file was not shared for the general public to access. The reason for this is a safety measure to ensure files are not accidentally uploaded that should be private. In order to solve this, we had to edit the permission requirements so that anyone with the URL could access the image. For example, below is the link to a photo uploaded to the S3 bucket. (My dog banjo).

[Public URL for the original version of the image](https://mendozacloud-bwilli37.s3.amazonaws.com/Banjo+on+Doritos.jpg?versionId=N614d7dRyJqYfoOh7lf7tNFA9kRSZ_MQ)

The next part of this lab is changing an object in the S3 bucket with a new version. To do this, I took the image from the URL above and edited the image. I saved this to my desktop under the same file name before. If you don’t use the same file name it will create a new object in this bucket. Once it has been uploaded the image will change. Below is my example of the edited photo.

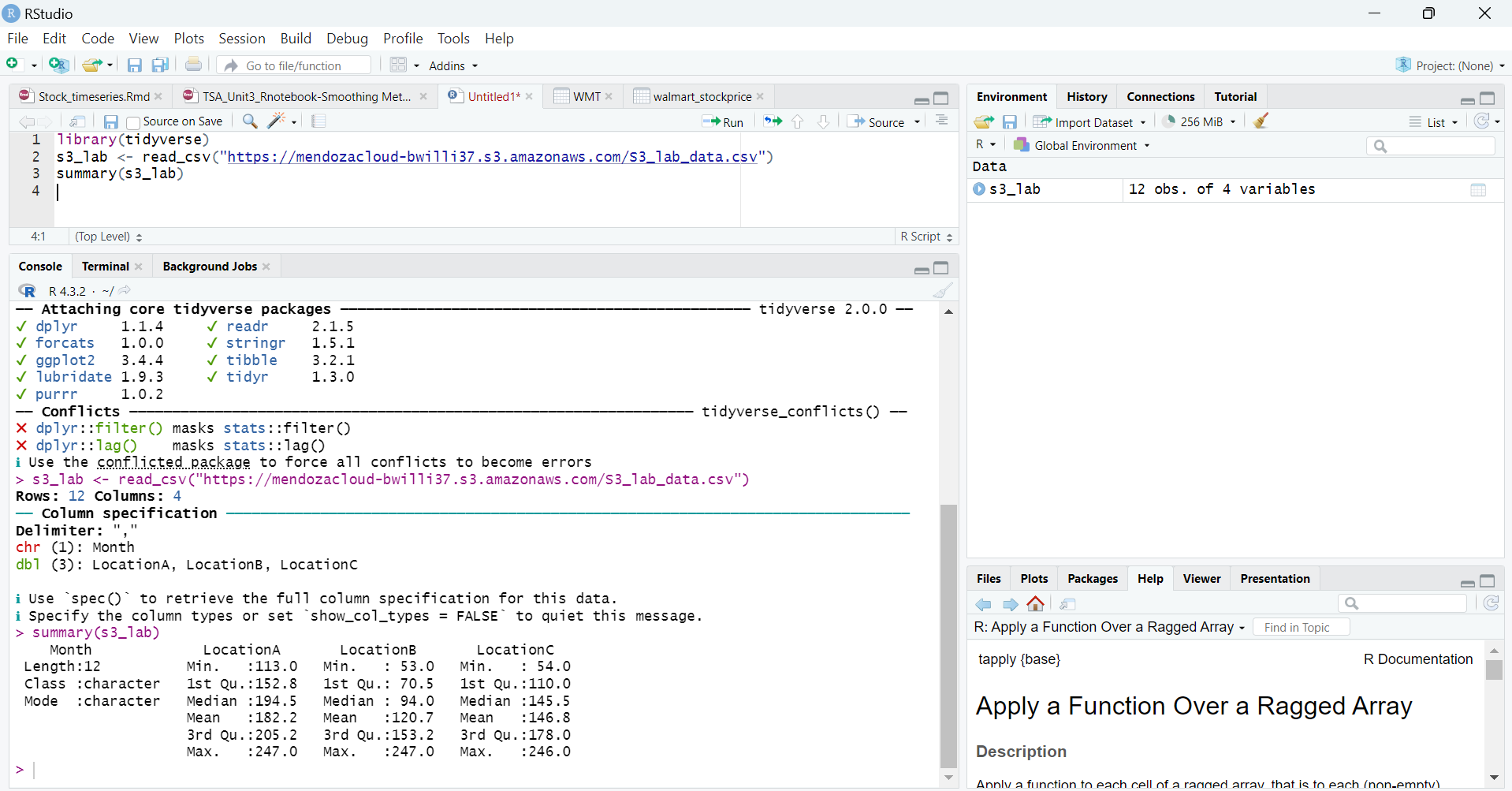
[Public URL for the modified image that you created](https://mendozacloud-bwilli37.s3.amazonaws.com/Banjo+on+Doritos.jpg)

Another feature we used in this lab was utilizing AWS to store data sets. For this lab we created a set of random data and created a new object in the S3 bucket as a csv file.

[Public URL for the dataset you created in Part VI](https://mendozacloud-bwilli37.s3.amazonaws.com/S3_lab_data.csv)

To practice access this data we created a line of code in R to pull this data set. Once it was imported into R, I ran the summary function on the data set. This ensured that the data could be accessed by the public. It also would make it very easy to store a variety of CSV files for later access.

*Screenshot of RStudio that you created in part VI*



Another useful feature used in this lab was creating an access log. This allows us to identify how the public is accessing these objects, what time of day they click the link, and what the IP address is for their device. To do this, we had to create a new bucket in S3. It had similar settings to the other bucket we created, however we did need to edit the ACL access to allow it to read and write access to our original bucket. By doing this, it started to create a log of when the public URL is accessed. This could be very useful if a user wants to know how often their URL is accessed.To generate this log we opened the link of our classmates' public URL.

Review Log Entries

Log #1

fae1fd3b4b92e6cf64aead62787b0e3bae0ea0accd932cefd674afeb43ff78c5 mendozacloud-bwilli37 [14/Jun/2024:02:57:28 +0000] 98.226.44.53 - 4557N11GTHS2WBD1 REST.GET.OBJECT Demo%2BDining%2BRoom.jpg "GET /Demo+Dining+Room.jpg HTTP/1.1" 403 AccessDenied 243 - 7 - "https://us-east-1.console.aws.amazon.com/s3/object/mendozacloud-bwilli37?region=us-east-1&bucketType=general&prefix=Demo+Dining+Room.jpg" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/125.0.0.0 Safari/537.36" - 6kjZkNhaO7FhcjXMmiiPjAnuYZXA0YE7qd7nmLnNwerJ2fNmxJqRdSr+VQrdT1SO1wcoIxuI0/s= - TLS\_CHACHA20\_POLY1305\_SHA256 - mendozacloud-bwilli37.s3.amazonaws.com TLSv1.3 - -

1. What file was accessed?
   1. File: Demo Dining Room.jpg
2. What is the IP address of the device that accessed the file?
   1. IP: 98.226.44.53
3. What time (in ET) did the access occur?
   1. Time: 10:57pm ET
4. What web browser was used to access the file?
   1. Web Browser: Google Chrome

Log #2

fae1fd3b4b92e6cf64aead62787b0e3bae0ea0accd932cefd674afeb43ff78c5 mendozacloud-bwilli37 [14/Jun/2024:02:57:39 +0000] 98.226.44.53 - 2GQBWV383B3984XQ REST.GET.OBJECT favicon.ico "GET /favicon.ico HTTP/1.1" 403 AccessDenied 243 - 8 - "https://mendozacloud-bwilli37.s3.amazonaws.com/Dining+Room.jpg" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/125.0.0.0 Safari/537.36" - kgJwXvgl9hfJhLL24s8V3xSV4eNmw3O9TR9PIuz/pwq80wiZFKqSvCIff4jOxF12LhKFwtxW5No= - TLS\_AES\_128\_GCM\_SHA256 - mendozacloud-bwilli37.s3.amazonaws.com TLSv1.3 - -

1. What file was accessed?
   1. File: Dining room.jpg
2. What is the IP address of the device that accessed the file?
   1. IP: 98.226.44.53
3. What time (in ET) did the access occur?
   1. Time: 10:57 pm ET
4. What web browser was used to access the file?
   1. Web Browser: Google Chrome

Analysis

1. *Question - 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?*

Answer- Per AWS’s official documentation, server access logging “provides detailed records for the requests that are made to a bucket”. The primary reason that Amazon cites for having server access logs is for security and audit purposes. The information is also helpful to perform analytics on to learn more about an organization’s customer base.

The downsides that we discussed related to server access logging is the performance reduction caused by the additional computing and storage required to create and save the access logs, as well as the cost of storing them. The data should only be housed as long as it is adding value or reducing risk – there are circumstances where the cost and effort to maintain the access logs will exceed the potential benefits. Additionally, there could be privacy concerns in certain industries related to logging access log data that need to be considered.

1. *Question - 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?*

Answer- Versioning was one of the most impressive parts of the Storage Lab for me (Patrick). Working at an organization where we handle large amounts of complex data *without* any versioning capabilities highlights the value in this technology – accidental deletions and overwrites can wreak havoc for large engagements. Being able to see changes throughout the lifecycle of a document is also a very helpful for efficiently following updates.

The primary downsides to versioning relate to cost and complexity. Difficult decisions need to be made in terms of how much version data should be retained and which documents should be prioritized in order to control costs. Additionally, version control can be difficult – its important for an organization to understand how it will handle version control issues prior to making the determination to turn on versioning.

1. *Question - 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?*

Answer- S3 is best for simple static websites. Per Amazon’s web hosting homepage, S3 is best for “websites that do not contain server-side scripting, like PHP or ASP.NET and customers that do not want to manage infrastructure”. S3 is ideal for websites that require scalability, but have few authors and only occasional intervals of high traffic. S3 does provide a reliable and available storage capability, so it is great for blogs and portfolio websites with consistent data. S3 is also cost-effective, and would therefore be an appropriate hosting environment for smaller businesses on tight budgets.

Websites that are not suitable for S3 include websites that need to dynamically scale resources or have other dynamic elements. High traffic websites and those with advances features like database integration should not be hosted using S3. In order to obtain these capabilities via AWS, the user would need to add other AWS services and features to its platform. There are other AWS resources available (EC2, for instance) that would better fit a user’s advanced needs rather than adding on to S3. Additionally, the lack of control and flexibility with S3 as compared to EC2 provides further reason that S3 would not be the right choice for websites requiring specific custom configurations.

1. *Question - 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:*
   1. *S3 Standard*
   2. *S3 Standard-IA*
   3. *S3 Intelligent-Tiering (assume that you have not enabled any of the optional tiers of service)*
   4. *S3 Glacier Flexible Retrieval*

Answer -

1. S3 Standard: This storage is great for files that will be accessed frequently and if the users need lower latency. An example of using this type of storage would be hosting a website.
2. S3 Standard-IA: This type of storage would be great for cheaper long term storage. For example, maybe a business has files that they don’t plan to access on a frequent basis, but still need to have a cheaper way to store this data. Standard-IA isn’t the fastest storage system, but it will allow users to store files for lower rates compared to S3 Standard.
3. S3 Intelligent-Tiering: This type of storage is best for a company that has a large amount of files that are accessed frequently and infrequently. The largest benefit of this file storage format is it will automatically store these files in long-term storage or short-term storage based on how often the users use the files. This would be beneficial for a company that doesn’t have someone who is managing the storage of a company’s files. Also, there is no fee for retrieving these files.
4. S3 Glacier Flexible Retrieval: This storage format is designed for longer term storage. Files are not instantly accessible, but the benefit of this format compared to Glacier deep archive is you can access these files 1 to 2 times a year for no fees. If a company had an annual report that they needed access to this would be a great use.
5. *Question - 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:* 
   * 1. *S3 Standard*
     2. *S3 Standard-IA*
     3. *S3 Intelligent-Tiering (assume that you have not enabled any of the optional tiers of service)*

*You will need to review the* [*S3 pricing*](https://aws.amazon.com/s3/pricing/?nc=sn&loc=4) *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*

Answer - Using the AWS pricing calculator, the price of hosting the 10GB file in S3 using the US East Region (N. Virginia) is as follows:

S3 Standard:

##### **Pricing calculations**

10 GB per month / 10 GB average item size = 1.00 unrounded number of objects

Round up by 1 (1.0000) = 1 number of objects

Tiered price for: 10 GB

10 GB x 0.023 USD = 0.23 USD

Total tier cost = 0.23 USD (S3 Standard storage cost)

30,000 GET requests in a month x 0.0000004 USD per request = 0.012 USD (S3 Standard GET requests cost)

0.23 USD + 0.012 USD = 0.24 USD (Total S3 Standard Storage, data requests, S3 select cost)

S3 Standard cost (monthly): 0.24 USD

1 number of objects x 0.000005 USD = 0.000005 USD (Cost for PUT, COPY, POST requests for initial data)

S3 Standard cost (upfront): 0.00 USD

S3 Standard-IA:

##### **Pricing calculations**

10 GB per month / 10 GB average item size = 1.00 unrounded number of objects

Round up by 1 (1.0000) = 1 number of objects

10 S3 Standard-IA Storage x 0.0125 USD = 0.125 USD (S3 Standard-IA storage cost)

30,000 GET requests for S3 Standard-IA Storage x 0.000001 USD per request = 0.03 USD (S3 Standard-IA GET requests cost)

0.125 USD + 0.03 USD = 0.15 USD (Total S3 Standard-IA Storage and other costs)

S3 Standard - Infrequent Access (S3 Standard-IA) cost (monthly): 0.15 USD

1 number of objects x 0.00001 USD = 0.00001 USD (Cost for PUT, COPY, POST requests for initial data)

S3 Standard - Infrequent Access (S3 Standard-IA) cost (upfront): 0.00 USD

S3 Intelligent-Tiering

##### **Pricing calculations**

10 GB per month / 10 GB average item size = 1.00 unrounded number of objects

Round up by 1 (1.0000) = 1 number of objects

1 frequent access multiplier x 10 GB = 10.00 GB (total frequent access storage)

Tiered price for: 10.00 GB

10 GB x 0.023 USD = 0.23 USD

Total tier cost = 0.23 USD (S3 INT Storage, Frequent Access Tier cost)

**Frequent Access Tier cost: 0.23 USD**

**Infrequent-Access Tier cost: 0 USD**

**Archive Instant Access Tier cost: 0 USD**

Round up by 1 (1.00) = 1 number of objects (total number of objects archive access storage)

1 number of objects (total number of objects archive access storage) x 8 KB = 8.00 KB overhead

8.00 KB overhead / 1048576 KB in a GB = 0.000008 GB overhead

Tiered price for: 0.000008 GB

0.000008 GB x 0.023 USD = 0.00 USD

Total tier cost = 0.0000002 USD (S3 Standard storage overhead cost for metadata)

1 number of objects (total number of objects archive access storage) x 32 KB = 32.00 KB overhead

32.00 KB overhead / 1048576 KB in a GB = 0.000031 GB overhead

0.000031 GB overhead x 0.0036 USD = 0.0000001 USD (S3-INT Archive Access storage overhead cost for metadata)

**Archive Access Tier cost: 0 USD**

Round up by 1 (1.00) = 1 number of objects (total number of objects deep archive access storage)

1 number of objects (total number of objects deep archive access storage) x 8 KB = 8.00 KB overhead

8.00 KB overhead / 1048576 KB in a GB = 0.000008 GB overhead

Tiered price for: 0.000008 GB

0.000008 GB x 0.023 USD = 0.00 USD

Total tier cost = 0.0000002 USD (S3 Standard storage overhead cost for metadata)

1 number of objects (total number of objects deep archive access storage) x 32 KB = 32.00 KB overhead

32.00 KB overhead / 1048576 KB in a GB = 0.000031 GB overhead

0.000031 GB overhead x 0.00099 USD = 0.00 USD (S3-INT Deep Archive storage overhead cost for metadata)

**Deep Archive Access Tier cost: 0 USD**

1 number of objects x 0.0000025 USD per object = 0.00 USD (Monitoring and automation objects cost)

30,000 GET requests in a month x 0.0000004 USD per request = 0.012 USD (S3 INT GET requests cost)

0.23 USD + 0.012 USD = 0.24 USD (Total S3 INT Storage, requests, select, scanned and retrieval cost)

S3 INT cost (monthly): 0.24 USD

1 number of objects x 0.000005 USD = 0.000005 USD (Cost for PUT, COPY, POST requests for initial data)

S3 INT cost (upfront): 0.00 USD

1. *Question - 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.*  
   1. *S3 Standard*
   2. *S3 Standard-IA*
   3. *S3 Intelligent-Tiering (assume that you have not enabled any of the optional tiers of service)*
   4. *S3 Glacier Flexible Retrieval*
   5. *S3 Glacier Deep Archive*

Answer - Using the AWS pricing calculator, the price of hosting the 75 TB of data using the US East (N. Virginia) region is as follows:

S3 Standard:

##### **Pricing calculations**

Tiered price for: 76,800 GB

51,200 GB x 0.023 USD = 1,177.60 USD

25,600 GB x 0.022 USD = 563.20 USD

Total tier cost: 1,177.60 USD + 563.20 USD = 1,740.80 USD (S3 Standard storage cost)

S3 Standard cost (monthly): 1,740.80 USD

S3 Standard Cost (annual): 20,890 USD

S3 Standard-IA:

##### **Pricing calculations**

76,800 S3 Standard-IA Storage x 0.0125 USD = 960.00 USD (S3 Standard-IA storage cost)

S3 Standard - Infrequent Access (S3 Standard-IA) cost (monthly): 960.00 USD

S3 Standard-IA Cost (annual): 11,520 USD

S3 Intelligent-Tiering

##### **Pricing calculations**

76,800 GB per month / 0.768 GB average item size = 100,000.00 unrounded number of objects

Round up by 1 (100000.0000) = 100000 number of objects

0 frequent access multiplier x 76,800 GB = 0.00 GB (total frequent access storage)

Tiered price for: 0.00 GB

Total tier cost = 0.00 USD (S3 INT Storage, Frequent Access Tier cost)

**Deep Archive Access Tier cost: 76.0526 USD**

100,000 number of objects x 0.0000025 USD per object = 0.25 USD (Monitoring and automation objects cost)

76.0526 USD + 0.25 USD = 76.30 USD (Total S3 INT Storage, requests, select, scanned and retrieval cost)

S3 INT cost (monthly): 76.30 USD

S3 INT Cost (annual): 916 USD

S3 Glacier Flexible Retrieval

##### **Pricing calculations**

76,800 GB per month / 0.768 GB average item size = 100,000.00 unrounded number of objects

Round up by 1 (100000.0000) = 100000 number of objects

100,000 number of objects x 32 KB = 3,200,000.00 KB overhead

3,200,000.00 KB overhead / 1048576 KB in a GB = 3.051758 GB overhead

3.051758 GB overhead x 0.0036 USD = 0.0109863 USD (Glacier Flexible Retrieval storage overhead cost for metadata)

**Glacier Flexible Retrieval storage overhead cost for metadata: 0.0109863 USD**

100,000 number of objects x 8 KB = 800,000.00 KB overhead

800,000.00 KB overhead / 1048576 KB in a GB = 0.762939 GB overhead

Tiered price for: 0.762939 GB

0.762939 GB x 0.023 USD = 0.02 USD

Total tier cost = 0.0175476 USD (S3 Standard storage overhead cost for metadata)

**S3 Standard storage overhead cost for metadata: 0.0175476 USD**

76,800 GB per month x 0.0036 USD = 276.48 USD (Glacier Flexible Retrieval storage cost)

**Glacier Flexible Retrieval storage cost: 276.48 USD**

0.0109863 USD + 0.0175476 USD + 276.48 USD = 276.508534 USD (Total Glacier Flexible Retrieval storage cost)

276.508534 USD + 0 USD + 0 USD + 0 USD + 0 USD + 0 USD + 0 USD + 0 USD + 0 USD + 0 USD = 276.51 USD (Total S3 Glacier Flexible Retrieval cost )

S3 Glacier Flexible Retrieval cost (monthly): 276.51 USD

S3 Glacier Flexible Retrieval cost (annual): 3,318 USD

S3 Glacier Deep Archive

##### **Pricing calculations**

76,800 GB per month / 0.768 GB average item size = 100,000.00 unrounded number of objects

Round up by 1 (100000.0000) = 100000 number of objects

100,000 number of objects x 32 KB = 3,200,000.00 KB overhead

3,200,000.00 KB overhead / 1048576 KB in a GB = 3.051758 GB overhead

3.051758 GB overhead x 0.00099 USD = 0.0030212 USD (Glacier Deep Archive storage overhead cost for metadata)

**Glacier Deep Archive storage overhead cost: 0.0030212 USD**

100,000 number of objects x 8 KB = 800,000.00 KB overhead

800,000.00 KB overhead / 1048576 KB in a GB = 0.762939 GB overhead

Tiered price for: 0.762939 GB

0.762939 GB x 0.023 USD = 0.02 USD

Total tier cost = 0.0175476 USD (S3 Standard storage overhead cost)

**S3 Standard storage overhead cost: 0.0175476 USD**

76,800 GB per month x 0.00099 USD = 76.032 USD (Glacier Deep Archive storage cost)

**Glacier Deep Archive storage cost: 76.032 USD**

0.0030212 USD + 0.0175476 USD + 76.032 USD = 76.052569 USD (Total Glacier Deep Archive storage cost)

S3 Glacier Deep Archive cost (monthly): 76.05 USD

S3 Glacier Deep Archive cost (annual): 913 USD

Screenshot from Storage Lab Section VI - screenshot of the showing the R Script and successful execution:

See the script on the left and the DB added to the Global Environment on the right.

