S3 Access Logs setting it up

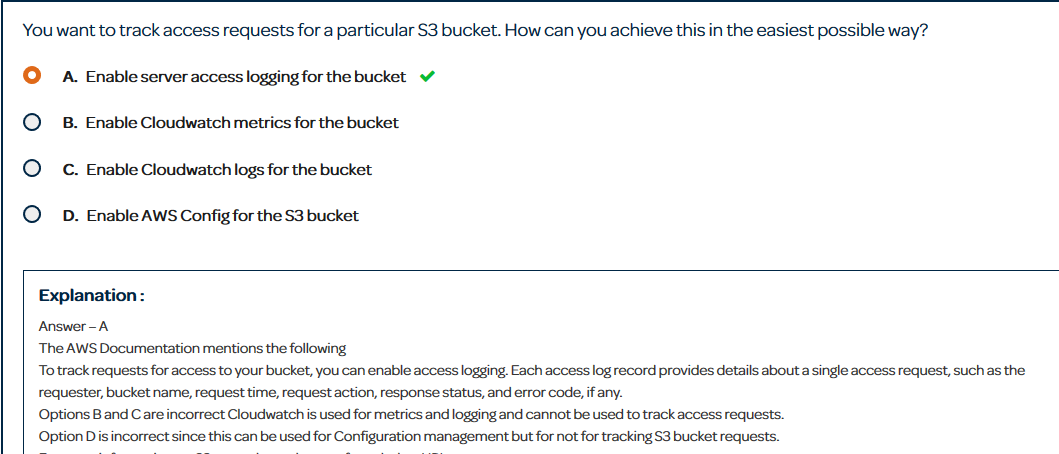
**S3 Access Logs-** Are an access mechanism in S3 that tracks access requests to buckets.

**Important features:**

* S3 access logs is configured for one bucket to log to another bucket in the same region
* The log delivery group must be granted write permissions on the target bucket
* This is not near real time
* Logs are delivered on a “Best Effort” basis and newly enabled access logs might not be displayed in the target bucket for up to an hour
* Changes to the target bucket might take up to an hour to propagate

How this might come up on your test.

The first time I took the test I did not get a question that I can remember about access logs.



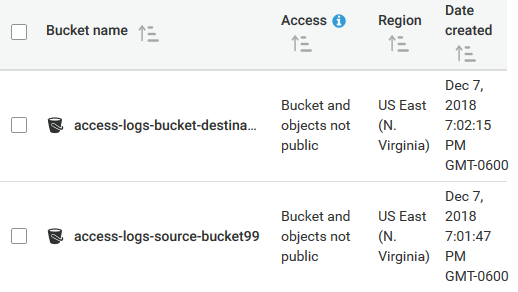
This is the kind of question where access logs would be the right answer. Keep in mind out of all the logging tools in AWS access logs are the slowest with logs coming in as slow as an hour. If you need any near-real-time logging think of a different answer.

One thing talked about in the video was, apache logs or access logs. In this scenario assume that the apache logs would be on an instance and the access logs would be enabled to log a static website. The answer would be the apache logs. The apache logs would be coming in via a CloudWatch agent which is much faster than access logs.

**Setting it up:**

**Step 1:**

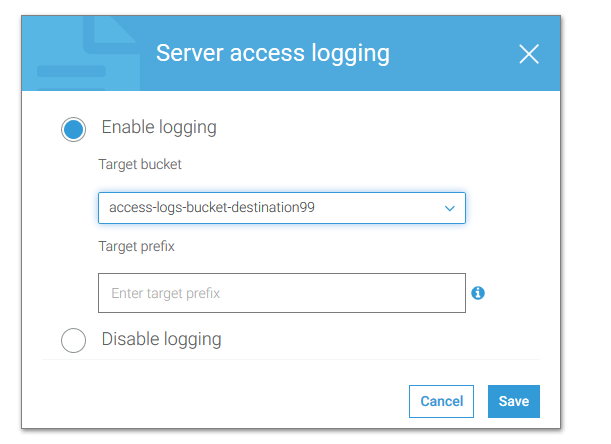
You first need two buckets. One in which you want to log and one in which you want to send the logs too. These buckets must be in the same region.



Considering that the best use case for using this is for when you are using S3 as a static website I will configure my source bucket to be a website.

**Step 2:**

Now we need to enable server access logging located in the properties tab. Click enable logging and choose the bucket you want to send too as shown below.

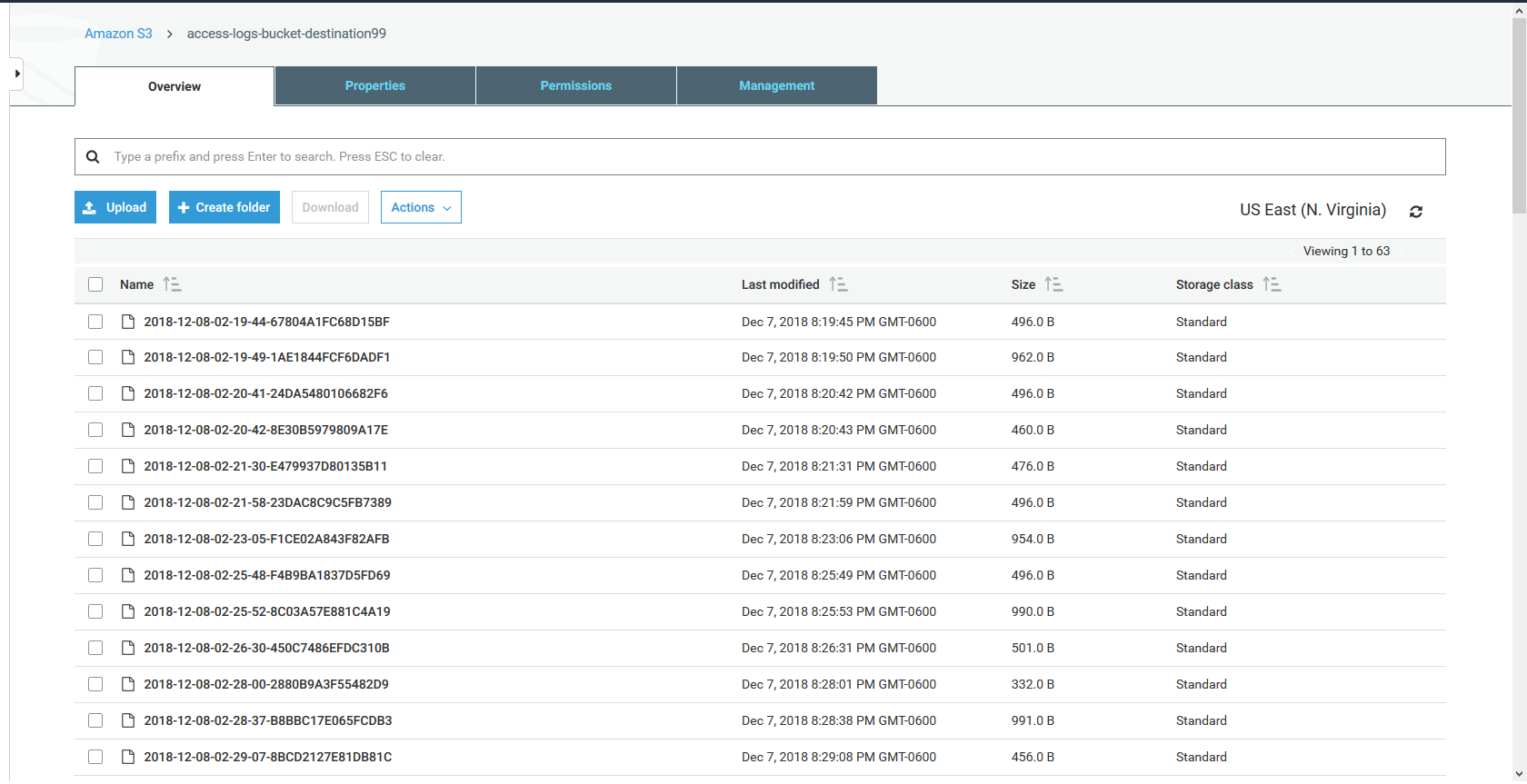


One thing I want to mention that may be important to the test. When I first enabled it last night I waited so long for it to start logging. In fact I went to sleep before it had begun to log. Remember that these logs are not real time and take some time to propagate.

Here is an example of what a log may look like.

4c98ce3929b43ce3636737d2e8284c20d894c8c4e017130c814f717118910a34 access-logs-source-bucket99 [08/Dec/2018:01:21:57 +0000] 104.54.200.158 arn:aws:iam::203647244731:user/Administrator 1FF1C5CAE90516C4 REST.GET.ENCRYPTION - "GET /access-logs-source-bucket99?encryption= HTTP/1.1" 404 ServerSideEncryptionConfigurationNotFoundError 370 - 16 - "-" "S3Console/0.4, aws-internal/3 aws-sdk-java/1.11.451 Linux/4.9.124-0.1.ac.198.71.329.metal1.x86\_64 OpenJDK\_64-Bit\_Server\_VM/25.192-b12 java/1.8.0\_192" –

Honestly these logs are super weird and hard to read in my opinion. They all just get thrown into a bucket with less than specific names so going through these logs one by one is not even a viable option. Check picture below for example.



One thing that came up a lot on my first test in terms of query CloudTrail logs was using Amazon Athena. Amazon Athena runs SQL queries on S3 buckets and I feel like that would be the only effective way to search through these logs.