

Getting Started

Workshop Details

In this workshop, you will add encryption and decryption features to the Busy Engineer's Document Bucket to learn about some real world AWS patterns for integrating client-side encryption using AWS Key Management Service (AWS KMS) and the AWS Encryption software development kit (ESDK) in application code. You will learn how to leverage features like multiple AWS KMS Customer Master Key (CMK) support and Encryption Context to secure your application.

To begin, the Document Bucket supports storing objects (documents or files or other blobs of data) in a private Amazon S3 bucket, and indexing them in Amazon DynamoDB. This allows Document Bucket users to share objects with other users, and store them for retrieval later. The DynamoDB entries provide metadata context for each Document Bucket object, as well as fast lookups to the provided context.

This context allows users to store additional information about the S3 object like the origin user, the destination fleet, the project, or any other tag that would be useful to know without having to download and examine the object.

DynamoDB is configured to allow you to index on the keys associated with the S3 objects. So, for example, it's a quick query to find out which objects have been tagged with metadata context that includes "configuration".

The Document Bucket supports the following APIs:

- `list`: This operation queries DynamoDB for all entries for all objects in the Document Bucket, and their metadata. It returns the `set` of items that have been stored.

- `store` : This operation accepts a blob of bytes and a `map` of metadata context. It generates a unique identifier (UID) for the object. The UID and associated metadata are written to DynamoDB. The bytes of the object are written to S3 under a key of that UID. Any context metadata keys in DynamoDB are updated to include that new UID.
- `retrieve` : This operation accepts a unique identifier (UID) as an argument. It looks up the UID in DynamoDB to identify the associated context and then retrieves the referenced object from S3. It returns the Document Bundle (UID, context, and referenced object) to the caller.
- `search` : This operation accepts a metadata key as search input. It queries DynamoDB for the `set` of objects in the Document Bucket with context matching that key. It returns the `set` of UIDs and their associated metadata.
 - Once you have identified the desired object(s), you can pass the UIDs to `retrieve` to fetch the objects.

This is a start for sharing, storing, and searching various objects.

But what about sensitive documents? Or protecting, say, important configuration files from accidental corruption during storing or retrieving?

The workshop exercises will help you address some of these challenges.

Background

In this section, we will give you step-by-step instructions to prepare your AWS Environment to work with the Busy Engineer's Document Bucket.

To set up your environment, you will:

- Deploy a CloudFormation stack to launch an AWS Cloud9 integrated development environment (IDE)
- AWS Cloud9 lets you write, run, and debug your code with just a browser
- Launch the Cloud9 IDE that you will use for the rest of the workshop

- Launch resource stacks using the AWS Cloud Development Kit (CDK), including:
 - The `BusyEngineersDocumentBucket` stack, with your DynamoDB table and S3 bucket
 - One CMK in one region, called Faythe, that you will use in encryption and decryption operations
 - One CMK in another region, called Walter, that you will use in encryption and decryption operations
- Bootstrap the development environment in Cloud9

Let's Go!

1. Sign in to your AWS Account for the workshop
 - If you are using your own account, make sure this is **not** a production account! This is a workshop for learning and experimentation. Don't put production at risk!
2. Click this link to [load the CloudFormation template for your Cloud9 IDE](https://us-east-2.console.aws.amazon.com/cloudformation/home?region=us-east-2#/stacks/quickcreate?templateUrl=https%3A%2F%2Fbusy-engineers-cfn.s3.us-east-2.amazonaws.com%2Fdocument-bucket-cloud9-bootstrap.yaml&stackName=BusyEngineersDocumentBucketEnvironment)
[https://us-east-2.console.aws.amazon.com/cloudformation/home?region=us-east-2#/stacks/quickcreate?templateUrl=https%3A%2F%2Fbusy-engineers-cfn.s3.us-east-2.amazonaws.com%2Fdocument-bucket-cloud9-bootstrap.yaml&stackName=BusyEngineersDocumentBucketEnvironment]
3. Click **Create Stack** to launch the stack
4. Do not change your region settings from the default and do not switch regions in the console
5. It will take about three minutes to launch your Cloud9 IDE
6. Click this link to [Open the Cloud9 Console and find your Cloud9 IDE](https://us-east-2.console.aws.amazon.com/cloud9/home?region=us-east-2#) [https://us-east-2.console.aws.amazon.com/cloud9/home?region=us-east-2#]
7. You may need to wait a minute and refresh while CloudFormation spins up the resources
8. There will be a blue tile with your workshop IDE when it's ready

9. At the bottom of the tile, click **Open IDE** to launch Cloud9
10. Type `cd ~/environment/workshop` and hit `Enter`
11. Type `make bootstrap` and hit `Enter` to set up your workshop environment
12. Wait until you see `*** BOOTSTRAP COMPLETE ***`
13. `make bootstrap` will take approximately 5 minutes. If you are in a live workshop, you can run this step during the presentation. If you are working on your own, grab a cup of your favorite beverage while you wait.
14. At this point you have deployed your workshop stacks using CDK and set up language environments for all of the workshop languages
15. In Cloud9, close your Terminal window and open a new one (`Window -> New Terminal`) to pick up the changes `make bootstrap` installed
16. Choose your workshop language, and `cd` to its folder under `exercises`

Java	Typescript Node.JS	JavaScript Node.JS	Python
<pre>1 cd ~/environment/workshop/exercises/java</pre>			

Your environment is ready!

Start the workshop with [Exercise 1: Add the Encryption SDK](#) [`./add-the-encryption-sdk/`].

Explore Further

- Check out the `~/environment/workshop/cdk` directory to see how the workshop resources are described using CDK
- [Who are Faythe and Walter?](#)
[https://en.wikipedia.org/wiki/Alice_and_Bob#Cast_of_characters]

Next exercise

Now that you have your environment and language selected, you can [Add the Encryption SDK](#) [../add-the-encryption-sdk/].