



Encrypt Data with AWS KMS



Nchindo Boris

The screenshot shows the AWS DynamoDB 'Explore items' interface. On the left, the navigation menu includes 'Dashboard', 'Tables', 'Explore items' (which is selected), 'PartiQL editor', 'Backups', 'Exports to S3', 'Imports from S3', 'Integrations', 'Reserved capacity', and 'Settings'. Below that is the 'DAX' section with 'Clusters', 'Subnet groups', 'Parameter groups', and 'Events'. The main panel has a search bar at the top with 'Find tables' and a dropdown for 'Any tag value'. It shows a table named 'nextwork-kms-table'. Under the table, there are two tabs: 'Scan' (selected) and 'Query'. A dropdown for 'Select a table or index' also points to 'nextwork-kms-table'. To the right of the table name is a dropdown for 'Select attribute projection' set to 'All attributes'. Below these are 'Filters - optional' fields and 'Run' and 'Reset' buttons. A red box highlights an error message: 'Access denied to kmsDecrypt. You don't have permission to kmsDecrypt. To request access, copy the following text and send it to your AWS administrator. Learn more about troubleshooting access denied errors.' The message details the user ('arn:aws:iam::550744777562:user/nextrwork-kms-user'), action ('kmsDecrypt'), resource ('arn:aws:kms:eu-north-1:550744777562:key/63535e62-22dd-49ec-b053-238482b8e48'), and context ('Context: no identity-based policy allows the action'). At the bottom of the main panel, it says 'Table: nextwork-kms-table - Items returned (0)' and has 'Actions' and 'Create item' buttons. The status bar at the bottom shows 'CloudShell Feedback' and various system icons.

Introducing Today's Project!

In this project, I will; - Create encryption keys with AWS KMS (Key Management Service). - Encrypt a DynamoDB database with a KMS key. - Add and retrieve data from your database to test your encryption. - Observe how AWS stops unauthorized access to your data. - Give a user encryption access. The goal is to understand how to safeguard cloud environments and set up secure access to resources.

Tools and concepts

Services I used include; - AWS Key Management Service (KMS) - Amazon DynamoDB
- AWS IAM Concepts I learnt include encryption, .

Project reflection

This project took me approximately 1 hour. It was most rewarding to actually see a security practice work well and it being understood by me

I did this project as part of my journey to understanding cloud computing and it definitely met one of my goals



Encryption and KMS

Encryption is the process that uses algorithms to convert data into a secure format called ciphertext. Companies and developers do this to secure user data, transactions, files and more. Encryption keys tell the algorithm exactly how it would transform plain text into the jumbled up format called cipher text.

AWS KMS is a secure vault for your encryption keys. You use KMS to create, manage, and use encryption keys that protect the data in your AWS resources. Key management systems are important because they help to manage all your encryption keys, like what it encrypts or who has access, in one place.

Encryption keys are broadly categorized as symmetric and asymmetric encryption keys. I set up a symmetric key because they are generally faster and more efficient for encrypting large amounts of data, which is why I am using one for my DynamoDB table.



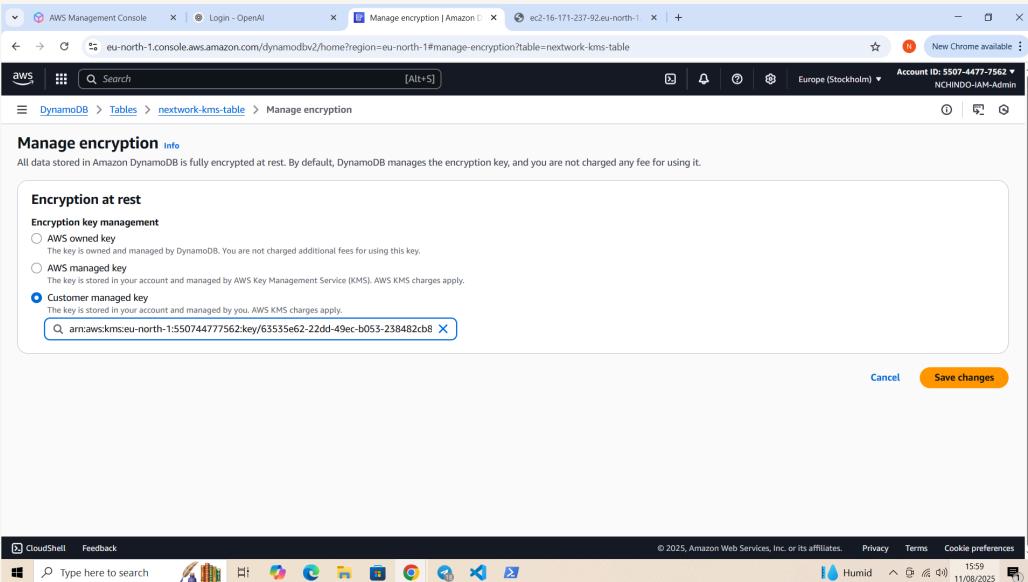
The screenshot shows the AWS Management Console interface for the Key Management Service (KMS). The user is in the 'Customer-managed keys' section of the 'Custom key store'. A success message at the top indicates that a new AWS KMS key was created with alias 'nextwork-kms-key' and key ID '63535e62-22dd-49ec-b053-238482cb0e48'. The table below lists the key details:

Aliases	Key ID	Status	Key type	Key spec	Key usage
nextwork-kms-key	63535e62-22dd-49ec-b053-238482cb0e48	Enabled	Symmetric	SYMMETRIC	Encrypt and decrypt

Encrypting Data

My encryption key will safeguard data in DynamoDB, which is one of AWS's database services.

The different encryption options in DynamoDB include; - Owned by Amazon
DynamoDB - AWS managed key - Stored in your account, and owned and managed by you Their differences are based on who has control and management. I selected Customer managed key to have full control





Data Visibility

Rather than controlling who has access to the key, KMS manages user permissions by giving the test user the permission to see that a KMS key is available in my AWS environment, but the test user still wouldn't have the permissions to decrypt the data it protects.

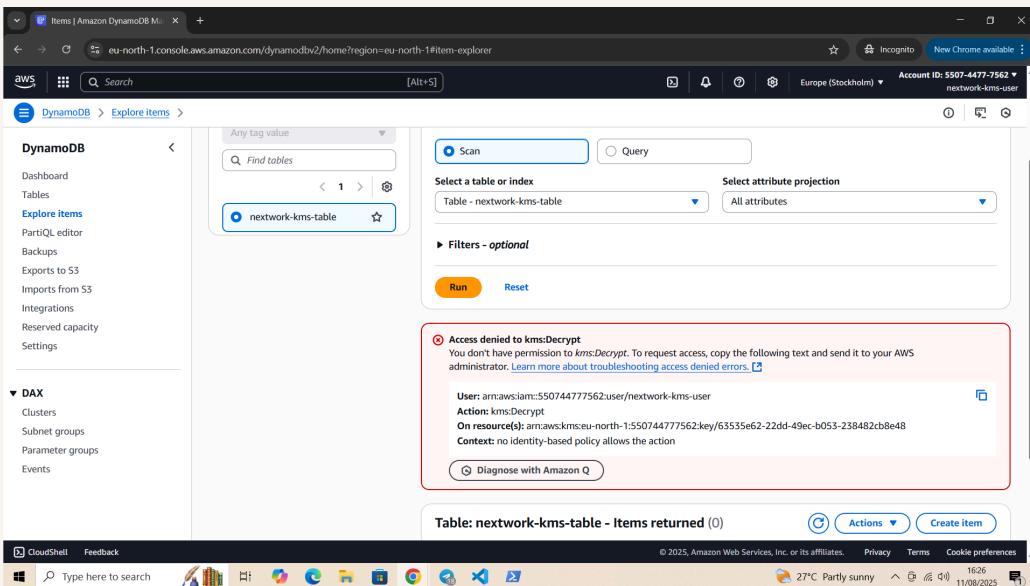
Despite encrypting my DynamoDB table, I could still see the table's items because as an authorised user I have permissions to use the encryption key in KMS. DynamoDB uses transparent data encryption, which makes sure that my data is secure at rest, yet still accessible to authorized users that have the right permissions.

The screenshot shows the AWS Management Console interface for the DynamoDB service. The left sidebar navigation includes 'Dashboard', 'Tables', 'Explore items' (which is selected), 'PartiQL editor', 'Backups', 'Exports to S3', 'Imports from S3', 'Integrations', 'Reserved capacity', and 'Settings'. Below this is a 'DAX' section with 'Clusters', 'Subnet groups', 'Parameter groups', and 'Events'. The main content area is titled 'Items | Amazon DynamoDB' and shows a table named 'nextwork-kms-table'. The table has one item with the primary key 'id' having the value '1'. At the top of the table view, there are buttons for 'Scan' (which is highlighted) and 'Query', and dropdown menus for 'Select a table or index' (set to 'Table - nextwork-kms-table') and 'Select attribute projection' (set to 'All attributes'). Below these are 'Filters - optional' fields and 'Run' and 'Reset' buttons. A green status bar at the bottom of this panel indicates a completed scan: 'Completed - Items returned: 1 - Items scanned: 1 - Efficiency: 100% - RCU consumed: 2'. Below the table, a message states 'Table: nextwork-kms-table - Items returned (1)' and 'Scan started on August 11, 2025, 16:08:03'. There are 'Actions' and 'Create item' buttons. The bottom of the screen shows the Windows taskbar with various pinned icons and the system tray indicating the date and time as 11/08/2025.

Denying Access

I configured a new IAM user to grant the user full access to DynamoDB but not to my KMS key. The permission policies I granted this user are AmazonDynamoDBFullAccess but not KMS key access

After accessing the DynamoDB table as the test user, I was denied access because the new IAM user I am logged in as (nextwork-kms-user) does not have the permission to decrypt the data. This confirmed that a KMS key can be accessible to many users, but only those with the right permissions can use it to do specific actions like encryption or decryption.





EXTRA: Granting Access

To let my test user use the encryption key, I added the test user to be one of the key users. My key's policy was updated to Allow use of key for test user also

Using the test user, I retried to access the data again. I observed could now see the data which confirmed that I now have access to the data with permissions to decrypt and encrypt

Encryption secures data instead of security groups or permission policies. I could combine encryption with security groups or permission policies to better secure data.

The screenshot shows the AWS Management Console interface for the KMS service. A specific key named 'arn:aws:kms:eu-north-1:550744777562:key/63535e62-22dd-49ec-b053-238482cb8e48' is selected. The 'Key policy' tab is active, displaying the JSON policy:

```
39  {
40    "Sid": "Allow use of the key",
41    "Effect": "Allow",
42    "Principal": {
43      "AWS": [
44        "arn:aws:iam::550744777562:user/NCHINDO-IAM-Admin",
45        "arn:aws:iam::550744777562:user/nextwork-kms-user"
46      ]
47    },
48    "Action": [
49      "kms:Encrypt",
50      "kms:Decrypt",
51      "kms:ReEncrypt*",
52      "kms:GenerateDataKey*",
53      "kms:DescribeKey"
54    ]
55  }
```

The left sidebar shows navigation options like 'Customer-managed keys' and 'Custom key stores'. The top right corner displays account information: Account ID: 5507-4477-7562 and IAM Admin.



nextwork.org

The place to learn & showcase your skills

Check out nextwork.org for more projects

