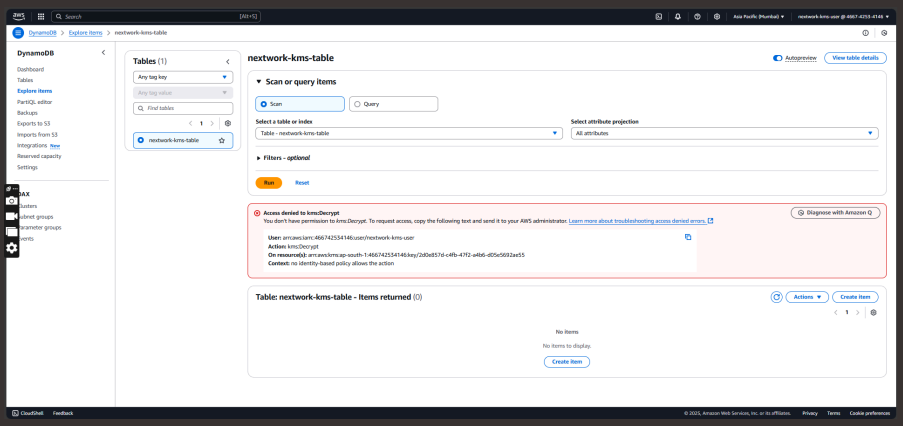
nextwork.org

Encrypt Data with AWS KMS

DI Dineshraj Dhanapathy



Introducing Today's Project!

In this project, I will demonstrate how to create encryption keys using AWS KMS (Key Management Service). I'll encrypt a DynamoDB database with a KMS key, add and retrieve data to test the encryption, and observe how AWS protects the data from unauthorized access. Finally, I'll give a user permission to use the encryption key. This hands-on project shows how AWS KMS strengthens security for databases and ensures only authorized users can access sensitive information.

Tools and concepts

Services I used include AWS Key Management Service (KMS), DynamoDB, and IAM (Identity and Access Management). Key concepts I learned include encryption, which protects sensitive data by transforming it into unreadable formats, and key management, which controls who can use and manage encryption keys. I also learned about user permissions, using KMS to grant and restrict access, and the importance of combining encryption with access control tools to secure data. These concepts ensure data protection and compliance in cloud environments.

Project reflection

This project took me approximately 1-2 hours to complete. The most challenging part was configuring the IAM policies and ensuring that the test user had the right permissions without compromising security. It was most rewarding to see how encryption in AWS KMS worked seamlessly with DynamoDB and how the permissions and key management protected sensitive data. The hands-on experience reinforced key cloud security concepts and gave me a deeper understanding of how AWS manages encryption and access control.

I chose to do this project today because I wanted to gain hands-on experience with AWS KMS, encryption, and access control. As part of my learning journey, this project aligned perfectly with my goal to understand how to secure data in AWS and manage permissions effectively. Yes, this project met my goals by allowing me to practice

Encryption and KMS

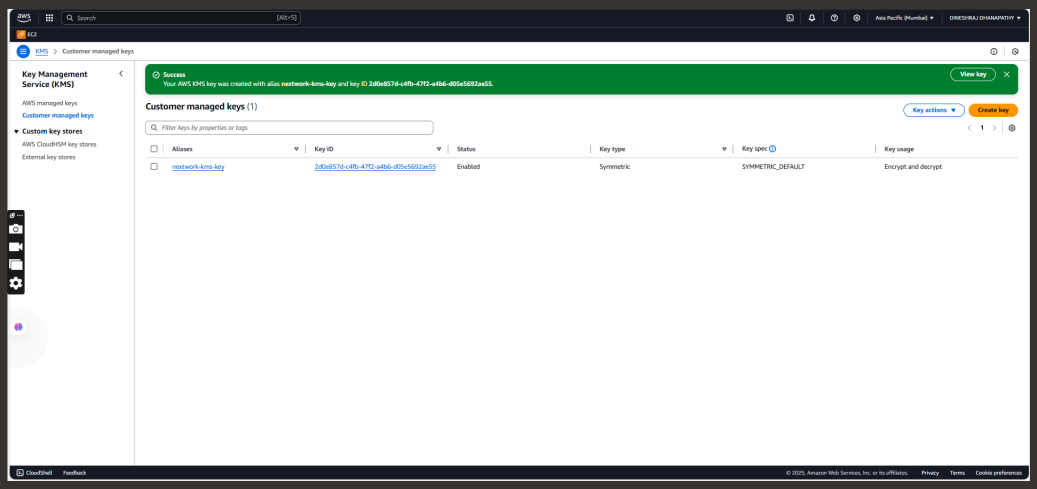
Encryption is the process of converting plain text into unreadable cipher text using an encryption algorithm and key. Companies and developers do this to protect sensitive data from unauthorized access and ensure privacy. Encryption keys are special codes that guide the algorithm on how to scramble and unscramble the data. Without the correct key, the data remains jumbled and unreadable. Encryption helps secure data at rest, in transit, and ensures compliance with security standards.

AWS KMS is a secure and managed service that helps you create, control, and use encryption keys to protect your data across AWS services. Key management systems are important because they centralize the management of encryption keys, ensure secure access control, and provide auditing capabilities. With AWS KMS, you can easily encrypt data, manage who can use or manage keys, and meet security and compliance requirements without building your own key storage solution.

Encryption keys are broadly categorized as symmetric and asymmetric keys. I set up a symmetric key because it uses the same key for both encryption and decryption, making it faster and more efficient for encrypting large amounts of data. Symmetric keys are ideal for services like DynamoDB where performance and speed are important. They are simpler to manage and integrate easily with AWS services through AWS KMS.

DI

Dineshraj Dhanapathy
NextWork Student



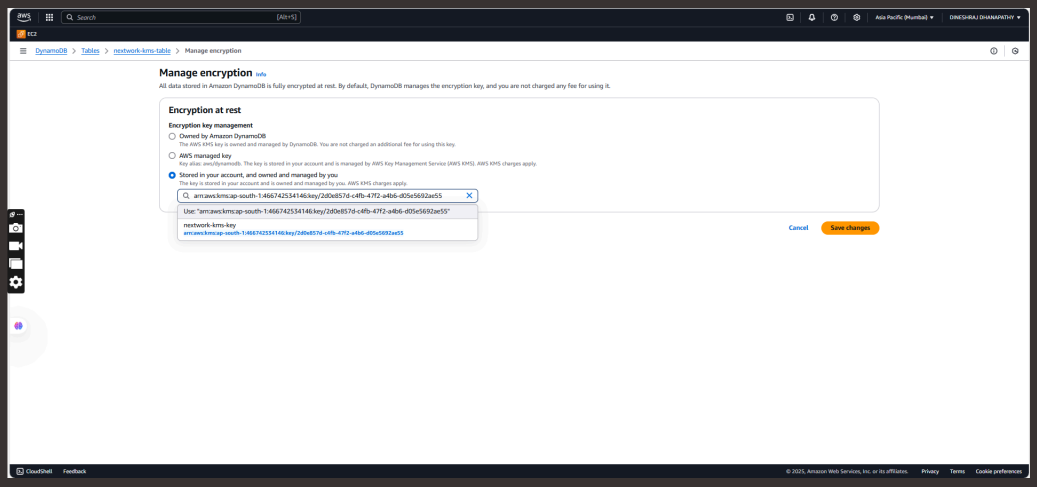
Encrypting Data

My encryption key will safeguard data in DynamoDB, which is a fully managed NoSQL database service offered by AWS. It provides fast and flexible performance for applications that need consistent, low-latency data access. DynamoDB automatically scales to handle large volumes of data and traffic. With built-in security, backup, and restore features, DynamoDB is ideal for modern applications that require high availability, scalability, and strong data protection.

The different encryption options in DynamoDB include keys owned by Amazon DynamoDB, AWS managed keys, and customer managed keys (CMKs). Their differences are based on the level of control and visibility you have. With Amazon-owned keys, you have no access to the keys. AWS managed keys offer some visibility but AWS handles the management. Customer managed keys, stored in your account, give you full control over key creation, permissions, and usage. I selected a customer managed key for maximum security and control.

DI

Dineshraj Dhanapathy
NextWork Student



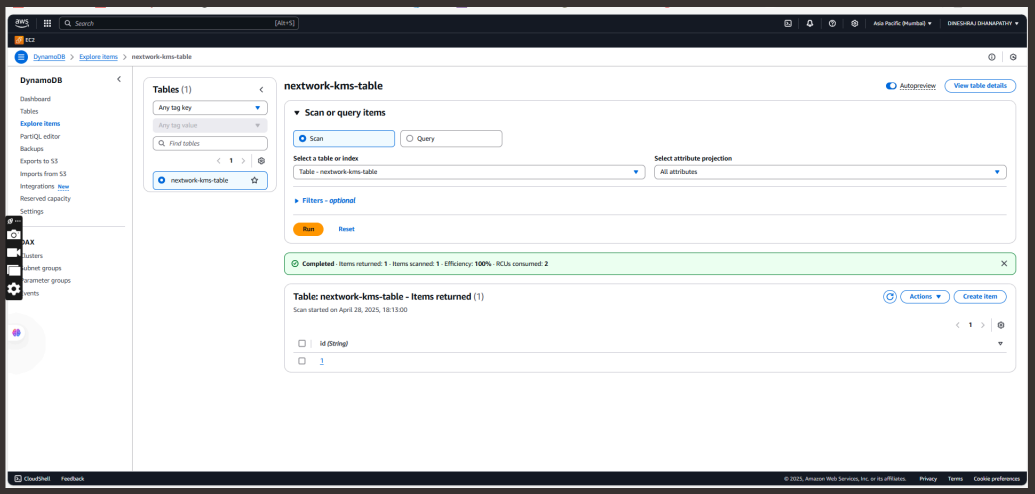
Data Visibility

Rather than controlling who has access to the key, KMS manages user permissions by defining policies and roles that specify who can use, manage, or administer the key. You can assign these permissions through AWS Identity and Access Management (IAM) policies, which control access to KMS operations like key creation, encryption, and decryption. This allows you to enforce the principle of least privilege, granting users only the permissions they need to perform specific actions on the key.

Despite encrypting my DynamoDB table, I could still see the table's items because DynamoDB uses transparent data encryption, which means the data is automatically encrypted at rest and decrypted during access for authorized users. This ensures that applications and users with proper permissions can work with the data normally, without needing to manually decrypt it. Encryption mainly protects data from unauthorized access and not from authorized users or applications.

DI

Dineshraj Dhanapathy
NextWork Student



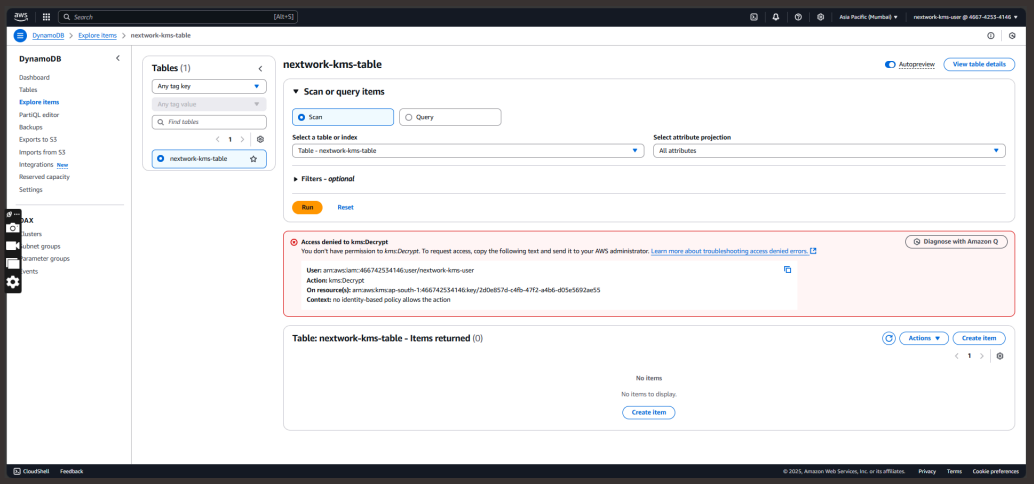
Denying Access

I configured a new IAM user to provide controlled access to my AWS resources. The permission policies I granted this user are full access to DynamoDB, allowing them to interact with the database, but not access to the KMS key. This ensures the user can perform necessary operations in DynamoDB while maintaining the security of the encryption key. I applied the principle of least privilege to restrict unnecessary access to sensitive resources.

After accessing the DynamoDB table as the test user, I encountered an "Access Denied" error for the access-analyzer:ListAnalyzers permission because the test user doesn't have the required permissions to list analyzers. This confirmed that while the user has full access to DynamoDB, their lack of access to the KMS key and other restricted permissions prevents them from performing unauthorized actions. The user can interact with the DynamoDB table but cannot access sensitive resources or perform restricted actions.

DI

Dineshraj Dhanapathy
NextWork Student



EXTRA: Granting Access

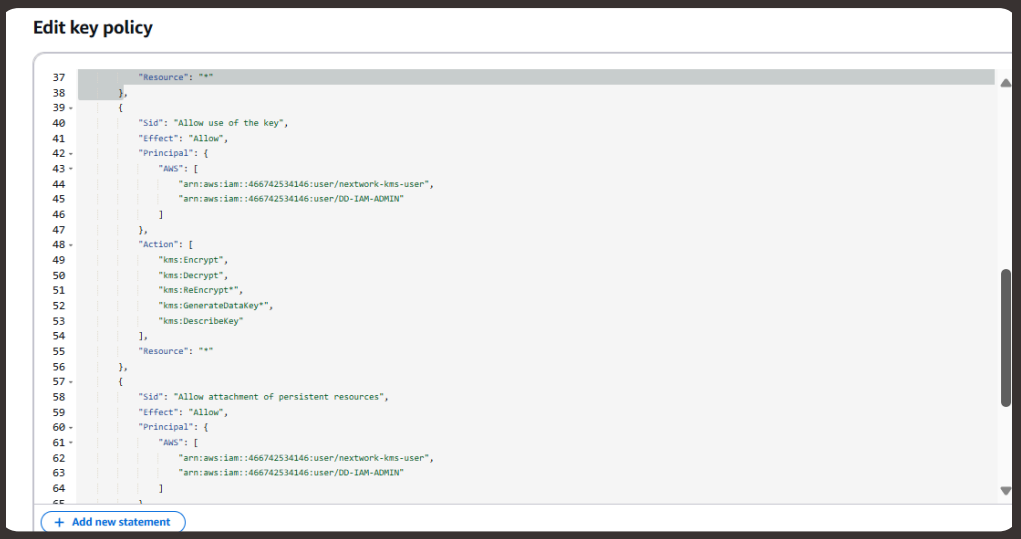
To let my test user use the encryption key, I added them as a key user in the KMS key policy. My key's policy was updated to grant the user specific permissions such as Encrypt, Decrypt, and DescribeKey, which allow them to encrypt and decrypt data with the key and view details about the key. Additionally, permissions like ReEncrypt and GenerateDataKey were included to enable more advanced operations, like transferring data between keys and creating temporary mini-keys for more efficient encryption.

Using the test user, I retried accessing the encrypted data in DynamoDB. I observed that the test user was able to interact with the database but could not decrypt or view the actual data. Instead, they received an "Access Denied" error when attempting to decrypt the items, which confirmed that the user had the appropriate access to DynamoDB but lacked the permissions to decrypt data without the necessary KMS key. This validated that the KMS encryption is working as expected.

Encryption secures data instead of controlling who can access a resource. While other access control tools, like security groups or permission policies, regulate who can interact with a service or resource, encryption protects the actual data within that resource. I could combine encryption with access control tools to ensure that only authorized users can access and read sensitive data. This way, even if someone bypasses access controls, the encrypted data remains unreadable without the proper decryption key.

DI

Dineshraj Dhanapathy
NextWork Student



nextwork.org

The place to learn & showcase your skills

Check out nextwork.org for more projects