Setting Up IAM Policies for CDK Execution Role

To enable AWS CDK to deploy resources, you must create and attach specific IAM policies to the **CDK execution role**.

1. Create IAM Policies

The following IAM policies must be created:

- neuraflash-desktop-cdk-appsync
- neuraflash-desktop-cdk-common
- neuraflash-desktop-cdk-connect
- neuraflash-desktop-cdk-database
- neuraflash-desktop-cdk-iam
- neuraflash-desktop-cdk-lambda
- neuraflash-desktop-cdk-s3-cloudfront

2. Attach Policies to the CDK Execution Role

Once created, these policies must be attached to the **CDK execution role**, which is automatically created when AWS CDK is bootstrapped. The role follows this naming pattern:

```
JavaScript
cdk-hnb659fds-cfn-exec-role-<account_id>-<region>
```

Example: cdk-hnb659fds-cfn-exec-role-123456789012-us-east-1

The exact role name depends on the AWS **CDK bootstrap version** and **region**.

3. Prerequisite: Bootstrap AWS CDK

Before using the CDK execution role, you must **bootstrap the AWS region** for CDK.
Bootstrapping sets up necessary resources, including the execution role. Run the following command:

```
JavaScript
cdk bootstrap aws://<account_id>/<region>
```

Example: cdk bootstrap aws://123456789012/us-east-1

Once bootstrapped, attach the required policies to the execution role to grant CDK the necessary permissions for deployment.

Create a New IAM Role for Additional Permissions

After configuring the CDK execution role, create a separate **IAM role** that assumes a **restricted CDK role** but has additional permissions to load the initial application configuration.

1. **Create a new IAM role**
2. **Attach the following policies to this role:**
    - neuraflash-desktop-cdk-sts
    - neuraflash-desktop-initial-load

This role ensures that the application can assume restricted permissions while still being able to initialize necessary configurations.

AWS IAM Policy Management

This IAM policy grants the necessary permissions to manage AWS AppSync APIs within your AWS account. It allows you to securely create, modify, and remove AppSync resources while adhering to security and compliance standards.

- **Copy the policy JSON**
- **Replace** YOUR_ACCOUNT_ID with your actual AWS account ID
- **Create the policy in AWS IAM : neuraflash-desktop-cdk-appsync**

AppSync Policy Breakdown

Base Permissions for AppSync API Management

- Grants permissions to create, delete, and retrieve AppSync GraphQL APIs.
- Allows tagging and untagging of resources for better resource organization.
- The scope is restricted to APIs within your AWS account.

Additional Permissions for Data Sources, Resolvers, and API Keys

- Provides permissions to manage data sources, resolvers, and API keys associated with AppSync.
- Includes actions for retrieving schema creation statuses and updating API keys.
- Covers WebACL settings for security and compliance.
- Resources are set to "*" to accommodate various API components dynamically.

JavaScript

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "appsyncBase",
            "Effect": "Allow",
            "Action": [
                "appsync:DeleteGraphqlApi",
                "appsync:StartSchemaCreation",
                "appsync:CreateGraphqlApi",
                "appsync:GetGraphqlApi",
                "appsync:UntagResource",
```

```
            "appsync:TagResource",
            "appsync:ListTagsForResource"
        ],
        "Resource": [
            "arn:aws:appsync:*:YOUR_       :apis/*",
            "arn:aws:appsync:*:YOUR_ACCOUNT_ID:/v1/apis/*"
        ]
    },
    {
        "Sid": "appsyncStar",
        "Effect": "Allow",
        "Action": [
            "appsync:CreateDataSource",
            "appsync:CreateResolver",
            "appsync:CreateApiKey",
            "appsync:GetDataSource",
            "appsync:GetResolver",
            "appsync:GetSchemaCreationStatus",
            "appsync:DeleteApiKey",
            "appsync:DeleteDataSource",
            "appsync:DeleteResolver",
            "appsync:UpdateApiKey",
            "appsync:SetWebACL"
```

```
            ],
            "Resource": "*"
        }
    ]
}
```

AWS IAM Policy for EventBridge, Kinesis, SSM, KMS, and WAFv2 Management

This IAM policy defines the necessary permissions for managing AWS resources related to EventBridge, Kinesis, AWS Systems Manager (SSM), AWS Key Management Service (KMS), and AWS WAFv2 within the specified AWS account. The policy is structured to ensure secure and controlled access to critical resources while allowing flexibility for operations.

- **Copy the policy JSON**
- **Replace** YOUR_ACCOUNT_ID with your actual AWS account ID
- **Create the policy in AWS IAM : neuraflash-desktop-cdk-common**

Policy Breakdown for EventBridge Rule Management

- Grants permissions to create, modify, enable, and delete EventBridge rules.
- Allows tagging and untagging of EventBridge rules for better organization.
- Scope is limited to rules matching the prefix nf-desktop*.

Policy Breakdown for Kinesis Stream Access

- Provides read access to Kinesis streams, including listing shards and retrieving records.
- ListStreams permission is applied globally to allow querying available streams.

Policy Breakdown for AWS Systems Manager (SSM) Parameter Store

- Grants permissions to create, update, and delete parameters related to nf-desktop*.

- Allows read access to SSM parameters under cdk-bootstrap/* for initialization.

Policy Breakdown for AWS Key Management Service (KMS)

- Grants full permissions for key management operations, including creating, tagging, and describing keys.
- Allows encryption and decryption of data using keys specific to nf-desktop*.
- Broader permissions (kmsStar) enable management of aliases and key policies.

Policy Breakdown for AWS WAFv2 WebACL Management

- Allows creation and association of WebACLs for resources under nf-desktop*.
- Provides read access to WebACLs across the account.

JavaScript

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "eventBridgeRuleBase",
            "Effect": "Allow",
            "Action": [
                "events:PutTargets",
                "events:DescribeRule",
                "events:EnableRule",
                "events:PutRule",
                "events:ListRules",
                "events:DeleteRule",
                "events:RemoveTargets",
                "events:TagResource",
                "events:ListTagsForResource",
                "events:UntagResource"
            ],
            "Resource": "arn:aws:events:*:YOUR_ACCOUNT_ID:rule/nf-desktop*"
        },
        {
            "Sid": "kinesisBase",
            "Effect": "Allow",
            "Action": [
                "kinesis:DescribeStreamSummary",
                "kinesis:ListShards",
```

```
                "kinesis:GetShardIterator",
                "kinesis:GetRecords",
                "kinesis:DescribeStream"
            ],
            "Resource": "arn:aws:kinesis:*:YOUR_ACCOUNT_ID:stream/*"
        },
        {
            "Sid": "kinesisStar",
            "Effect": "Allow",
            "Action": ["kinesis:ListStreams"],
            "Resource": "*"
        },
        {
            "Sid": "ssmBase",
            "Effect": "Allow",
            "Action": [
                "ssm:AddTagsToResource",
                "ssm:ListTagsForResource",
                "ssm:RemoveTagsFromResource",
                "ssm:PutParameter",
                "ssm:LabelParameterVersion",
                "ssm:GetParametersByPath",
                "ssm:GetParameters",
                "ssm:GetParameter",
                "ssm:DeleteParameter",
                "ssm:DeleteParameters"
            ],
            "Resource": ["arn:aws:ssm:*:YOUR_ACCOUNT_ID:parameter/nf-desktop*"]
        },
        {
            "Sid": "ssmBootstrap",
            "Effect": "Allow",
            "Action": ["ssm:GetParameters", "ssm:GetParameter"],
            "Resource":
["arn:aws:ssm:*:YOUR_ACCOUNT_ID:parameter/cdk-bootstrap/*"]
        },
        {
            "Sid": "kmsStar",
            "Effect": "Allow",
            "Action": [
                "kms:GetKeyRotationStatus",
                "kms:ListAliases",
                "kms:CreateKey",
                "kms:CreateAlias",
```

```
            "kms:CreateGrant",
            "kms:PutKeyPolicy",
            "kms:TagResource",
            "kms:DescribeKey",
            "kms:GenerateDataKeyWithoutPlaintext",
            "kms:ListResourceTags",
            "kms:GetKeyPolicy"
        ],
        "Resource": ["*"]
    },
    {
        "Sid": "kmsBase",
        "Effect": "Allow",
        "Action": [
            "kms:Decrypt",
            "kms:Encrypt",
            "kms:DeleteAlias",
            "kms:ScheduleKeyDeletion",
            "kms:EnableKeyRotation"
        ],
        "Resource": [
            "arn:aws:kms:*:YOUR_ACCOUNT_ID:key/*",
            "arn:aws:kms:*:YOUR_ACCOUNT_ID:alias/nf-desktop*"
        ]
    },
    {
        "Sid": "wafv2Base",
        "Effect": "Allow",
        "Action": [
            "wafv2:CreateWebACL",
            "wafv2:GetWebACL",
            "wafv2:ListTagsForResource",
            "wafv2:GetWebACLForResource",
            "wafv2:AssociateWebACL"
        ],
        "Resource": [
            "arn:aws:wafv2:*:YOUR_ACCOUNT_ID:*/managedruleset/*/*",
            "arn:aws:wafv2:*:YOUR_ACCOUNT_ID:*/webacl/nf-desktop*"
        ]
    },
    {
        "Sid": "wafv2Star",
        "Effect": "Allow",
        "Action": ["wafv2:GetWebACLForResource"],
```

```
            "Resource": ["arn:aws:wafv2:*:YOUR_ACCOUNT_ID:*/webacl/*/*"]
        }
    ]
}
```

IAM Policy for Amazon Connect Integration with Lambda

- **Copy the policy JSON**
- **Replace** YOUR_ACCOUNT_ID with your actual AWS account ID
- **Replace** YOUR_INSTANCE_ID with your actual Amazon Connect Instance Id
- **Create the policy in AWS IAM : neuraflash-desktop-cdk-connect**

Policy Breakdown for Amazon Connect

- Associate and disassociate Lambda functions with an Amazon Connect instance.
- List existing Lambda function integrations.
- Create and delete integration associations.

Scope of Permissions for Amazon Connect

- **Amazon Connect Instance**: Your Amazon Connect instance.
- **Integration Associations**: Your Amazon Connect instance.

This ensures the policy is restricted to only the required Connect instance and its integrations.

JavaScript

```
{
   "Version": "2012-10-17",
   "Statement": [
       {
           "Sid": "connectBase",
           "Effect": "Allow",
           "Action": [
```

```
                "connect:AssociateLambdaFunction",
                "connect:ListLambdaFunctions",
                "connect:DisassociateLambdaFunction",
                "connect:CreateIntegrationAssociation",
                "connect:DeleteIntegrationAssociation"
            ],
            "Resource": [
                "arn:aws:connect:*:YOUR_ACCOUNT_ID:instance/YOUR_INSTANCE_ID",
"arn:aws:connect:*:YOUR_ACCOUNT_ID:instance/YOUR_INSTANCE_ID/integration-association/*"
            ]
        }
    ]
}
```

IAM Policy for Amazon DynamoDB Table Management

This policy grants permissions to manage Amazon DynamoDB tables within the specified AWS account. It allows the creation, updating, deletion, and configuration of tables, including tagging and backup settings. The scope is restricted to tables with the prefix nf-desktop*.

- **Copy the policy JSON**
- **Replace** YOUR_ACCOUNT_ID with your actual AWS account ID
- **Create the policy in AWS IAM: neuraflash-desktop-cdk-database**

Policy Breakdown for AWS Dynamo DB

- **Table Management**: Create, update, describe, and delete DynamoDB tables.
- **Tagging**: Add, list, and remove resource tags.
- **Time to Live (TTL) Settings**: Update and describe TTL configurations.
- **Continuous Backups**: Enable and describe backup settings.

Scope of Permissions:

Restricted to **DynamoDB tables** prefixed with nf-desktop* within the AWS **account**.

This policy ensures controlled access to DynamoDB table management while maintaining security and compliance.

JavaScript

```
{
   "Version": "2012-10-17",
   "Statement": [
       {
           "Sid": "dynamoBase",
           "Effect": "Allow",
           "Action": [
               "dynamodb:CreateTable",
               "dynamodb:DescribeTable",
               "dynamodb:DeleteTable",
               "dynamodb:TagResource",
               "dynamodb:ListTagsOfResource",
               "dynamodb:UntagResource",
               "dynamodb:UpdateTable",
               "dynamodb:UpdateTimeToLive",
               "dynamodb:DescribeTimeToLive",
               "dynamodb:UpdateContinuousBackups",
               "dynamodb:DescribeContinuousBackups"
           ],
           "Resource": "arn:aws:dynamodb:*:YOUR_ACCOUNT_ID:table/nf-desktop*"
       }
   ]
}
```

IAM Policy for Managing IAM Roles and Policies

This policy grants permissions to manage IAM roles and policies within the specified AWS account, allowing their creation, modification, deletion, and tagging. It also enables permission management and the creation of service-linked roles for specific AWS services.

- **Copy the policy JSON**
- **Replace** YOUR_ACCOUNT_ID with your actual AWS account ID
- **Create the policy in AWS IAM: neuraflash-desktop-cdk-iam**

Policy Breakdown for IAM Roles & Policies

- **IAM Role Management**: Create, update, delete, and manage IAM roles and their permissions.
- **IAM Policy Management**: Create, update, delete, and manage IAM policies.
- **Tagging**: Add, list, and remove tags for IAM users, roles, and policies.
- **Role Permissions**: Attach, detach, and manage role policies, including inline and managed policies.
- **Service-Linked Roles**: Allows creation of service-linked roles for **Amazon Timestream** and **AWS Lambda**.

Scope of Permissions:

- Restricted to **IAM roles and policies** within AWS account YOUR_ACCOUNT_ID, specifically those prefixed with nf-desktop* and LogRetention*.
- Allows service-linked role creation for AWS-managed services.

This policy ensures controlled access to the IAM role and policy management while maintaining security and compliance.

JavaScript

```
{
   "Version": "2012-10-17",
   "Statement": [
       {
           "Sid": "iamBase",
```

```
            "Effect": "Allow",
            "Action": [
                "iam:UntagUser",
                "iam:UntagRole",
                "iam:ListRoleTags",
                "iam:TagRole",
                "iam:UntagPolicy",
                "iam:ListPolicyTags",
                "iam:TagPolicy",
                "iam:TagUser",
                "iam:ListUserTags",
                "iam:DetachRolePolicy",
                "iam:DeleteRole",
                "iam:DeleteRolePolicy",
                "iam:DeletePolicy",
                "iam:CreateRole",
                "iam:GetRole",
                "iam:AttachRolePolicy",
                "iam:PutRolePolicy",
                "iam:CreatePolicy",
                "iam:GetPolicy",
                "iam:ListPolicyVersions",
                "iam:ListRolePolicies",
                "iam:ListAttachedRolePolicies",
                "iam:ListEntitiesForPolicy",
                "iam:GetPolicyVersion",
                "iam:GetRolePolicy",
                "iam:PassRole"
            ],
            "Resource": [
                "arn:aws:iam::YOUR_ACCOUNT_ID:role/nf-desktop*",
                "arn:aws:iam::YOUR_ACCOUNT_ID:role/LogRetention*",
                "arn:aws:iam::YOUR_ACCOUNT_ID:role/service-role/nf-desktop*",
                "arn:aws:iam::YOUR_ACCOUNT_ID:policy/nf-desktop*",
```

```
                "arn:aws:iam::YOUR_ACCOUNT_ID:policy/service-role/nf-desktop*"
            ]
        },
        {
            "Sid": "iamLinkedRole",
            "Effect": "Allow",
            "Action": "iam:CreateServiceLinkedRole",
            "Resource": [

"arn:aws:iam::*:role/aws-service-role/timestream.amazonaws.com/*",
                "arn:aws:iam::*:role/aws-service-role/lambda.amazonaws.com/*"
            ]
        }
    ]
}
```

IAM Policy for AWS Lambda Management

This policy grants permissions to manage AWS Lambda functions, layers, and event source mappings within the specified AWS account. It allows function creation, configuration management, invocation, versioning, aliasing, tagging, and deletion. Additionally, it provides permissions for handling event source mappings globally.

- **Copy the policy JSON**
- **Replace** YOUR_ACCOUNT_ID with your actual AWS account ID
- **Create the policy in AWS IAM: neuraflash-desktop-cdk-lambda**

Policy Breakdown for Lambda Management

- **Lambda Function Management**: Create, delete, update, and configure Lambda functions.
- **Layer Management**: Manage Lambda layers, including publishing new versions and retrieving existing ones.

- **Alias and Versioning**: Create, publish, list, and manage aliases and versions of Lambda functions.
- **Invocation & Permissions**: Invoke functions and manage permissions for function access.
- **Tagging**: Add, remove, and list tags for Lambda resources.
- **Event Source Mappings**:
    - **Scoped Permissions**: Create, delete, and manage event source mappings for specific resources.
    - **Global Permissions**: Allows managing event source mappings for all Lambda functions.

Scope of Permissions:

- Restricted to **Lambda functions and layers** within AWS account YOUR_ACCOUNT_ID, specifically those prefixed with nf-desktop* and nf-desktop*.
- Global access ("*" resource) for managing event source mappings.

This policy ensures full control over AWS Lambda functions and layers while maintaining flexibility for event-driven architectures.

JavaScript

```
{
   "Version": "2012-10-17",
   "Statement": [
       {
           "Sid": "lambdaBase",
           "Effect": "Allow",
           "Action": [
               "lambda:CreateFunction",
               "lambda:GetLayerVersion",
               "lambda:GetFunctionConcurrency",
               "lambda:GetLayerVersionPolicy",
               "lambda:CreateAlias",
               "lambda:PublishVersion",
               "lambda:PublishLayerVersion",
```

```
            "lambda:ListLayerVersions",
            "lambda:ListLayers",
            "lambda:ListVersionsByFunction",
            "lambda:ListAliases",
            "lambda:GetFunctionConfiguration",
            "lambda:GetAlias",
            "lambda:ListFunctions",
            "lambda:GetFunctionCodeSigningConfig",
            "lambda:GetRuntimeManagementConfig",
            "lambda:ListProvisionedConcurrencyConfigs",
            "lambda:TagResource",
            "lambda:ListTags",
            "lambda:UntagResource",
            "lambda:GetFunction",
            "lambda:AddPermission",
            "lambda:RemovePermission",
            "lambda:InvokeFunction",
            "lambda:DeleteFunctionConcurrency",
            "lambda:DeleteFunction",
            "lambda:DeleteAlias",
            "lambda:DeleteLayerVersion",
            "lambda:DeleteProvisionedConcurrencyConfig",
            "lambda:DeleteFunctionEventInvokeConfig"
        ],
        "Resource": [
            "arn:aws:lambda:*:YOUR_ACCOUNT_ID:layer:nf-desktop*",
            "arn:aws:lambda:*:YOUR_ACCOUNT_ID:function:nf-desktop*",
            "arn:aws:lambda:*:YOUR_ACCOUNT_ID:layer:nfdesktop*",
            "arn:aws:lambda:*:YOUR_ACCOUNT_ID:event-source-mapping:*"
        ]
    },
    {
        "Sid": "lambdaStar",
        "Effect": "Allow",
```

```
            "Action": [
                "lambda:CreateEventSourceMapping",
                "lambda:GetEventSourceMapping",
                "lambda:DeleteEventSourceMapping"
            ],
            "Resource": "*"
        }
    ]
}
```

IAM Policy for S3 and CloudFront Management

This policy grants permissions to manage Amazon S3 buckets and CloudFront distributions within the specified AWS account. It includes bucket creation, policy management, object retrieval, notification configurations, and CloudFront operations such as listing, updating, and deleting distributions.

- **Copy the policy JSON**
- **Replace** YOUR_ACCOUNT_ID with your actual AWS account ID
- **Create the policy in AWS IAM: neuraflash-desktop-cdk-s3-cloudfront**

Policy Breakdown for S3 and CloudFront Management

Amazon S3 Permissions:

- **Bucket Management**: Create, delete, update, and configure S3 buckets (nf-desktop*).
- **Object Retrieval**: Access objects in the cdk bucket.
- **Contact Lens Data Access**: Manage notifications and list the amazon-connect-5d076362867f bucket.

Amazon CloudFront Permissions:

- **Read-Only Access**: Allows listing and retrieving CloudFront configurations.

- **Management Permissions**: Create, update, tag, and delete distributions and origin access controls.
- **Conditional Deletion**: CloudFront distributions can only be deleted if tagged with "neuraflash-desktop": "nf-desktop*"

This policy ensures control over S3 and CloudFront while maintaining security through resource scoping and tagging conditions.

JavaScript

```
{
   "Version": "2012-10-17",
   "Statement": [
       {
           "Sid": "s3Base",
           "Effect": "Allow",
           "Action": [
               "s3:CreateBucket",
               "s3:PutBucketPolicy",
               "s3:PutBucketTagging",
               "s3:PutBucketWebsite",
               "s3:DeleteBucketPolicy",
               "s3:DeleteBucketWebsite",
               "s3:DeleteBucket",
               "s3:PutEncryptionConfiguration",
               "s3:GetLifecycleConfiguration",
               "s3:PutLifecycleConfiguration",
               "s3:PutBucketOwnershipControls",
               "s3:PutBucketAcl",
               "s3:PutBucketVersioning",
               "s3:PutBucketPublicAccessBlock",
               "s3:PutBucketLogging",
               "s3:GetBucketAcl",
               "s3:PutBucketAcl"
           ],
```

```
            "Resource": ["arn:aws:s3:::nf-desktop*"]
        },
        {
            "Sid": "s3GetObject",
            "Effect": "Allow",
            "Action": "s3:GetObject",
            "Resource": ["arn:aws:s3:::cdk-bucket*"]
        },
        {
            "Sid": "s3ContactLens",
            "Effect": "Allow",
            "Action": ["s3:PutBucketNotification", "s3:ListBucket",
"s3:GetBucketNotification"],
            "Resource": "arn:aws:s3:::s3-bucket"
        },
        {
            "Sid": "cloudfrontReadOnly",
            "Effect": "Allow",
            "Action": ["cloudfront:List*", "cloudfront:Get*"],
            "Resource": [
                "arn:aws:cloudfront::YOUR_ACCOUNT_ID:distribution/*",
                "arn:aws:cloudfront::YOUR_ACCOUNT_ID:origin-access-identity/*"
            ]
        },
        {
            "Sid": "cloudfrontBase",
            "Effect": "Allow",
            "Action": [
                "cloudfront:CreateDistribution",
                "cloudfront:CreateCloudFrontOriginAccessIdentity",
                "cloudfront:TagResource",
                "cloudfront:CreateStreamingDistributionWithTags",
                "cloudfront:UpdateDistribution",
                "cloudfront:DeleteCloudFrontOriginAccessIdentity",
```

```
                "cloudfront:CreateOriginAccessControl"
            ],
            "Resource": [
                "arn:aws:cloudfront::YOUR_ACCOUNT_ID:distribution/*",
                "arn:aws:cloudfront::YOUR_ACCOUNT_ID:origin-access-identity/*",
                "arn:aws:cloudfront::YOUR_ACCOUNT_ID:origin-access-control/*"
            ]
        },
        {
            "Sid": "cloudfrontDelete",
            "Effect": "Allow",
            "Action": "cloudfront:DeleteDistribution",
            "Resource": "arn:aws:cloudfront::YOUR_ACCOUNT_ID:distribution/*",
            "Condition": {
                "StringLike": {
                    "aws:ResourceTag/neuraflash-desktop": "nf-desktop*"
                }
            }
        }
    ]
}
```

IAM Policy for Initial Data Load

This policy grants permissions to interact with Amazon Connect, AWS Directory Service, and Amazon DynamoDB within the specified AWS account. It enables controlled access to describe instances, retrieve directory details, and perform read/write operations on DynamoDB tables.

- **Copy the policy JSON**
- **Replace** YOUR_ACCOUNT_ID with your actual AWS account ID
- **Replace** YOUR_INSTANCE_ID with your actual Amazon Connect Instance ID

- **Create the policy in AWS IAM: neuraflash-desktop-initial-load**

Policy Breakdown for Initial Data Load

Amazon Connect Permissions:

- Allows describing a specific Amazon Connect instance.

AWS Directory Service Permissions:

- Enables describing directories across the account.

Amazon DynamoDB Permissions:

- Allows inserting, batch writing, querying, and updating tables (nf-desktop*).
- Grants permission to describe table structures.

This policy ensures minimal necessary access for Amazon Connect, directory information retrieval, and DynamoDB table operations while maintaining security and operational efficiency.

JavaScript

```
{
   "Version": "2012-10-17",
   "Statement": [
       {
           "Sid": "connectBase",
           "Effect": "Allow",
           "Action": "connect:DescribeInstance",
           "Resource":
"arn:aws:connect:*:YOUR_ACCOUNT_ID:instance/YOUR_INSTANCE_ID"
       },
       {
           "Sid": "directoriesBase",
           "Effect": "Allow",
           "Action": "ds:DescribeDirectories",
           "Resource": "*"
       },
```

```
        {
            "Sid": "dynamoBase",
            "Effect": "Allow",
            "Action": [
                "dynamodb:PutItem",
                "dynamodb:BatchWriteItem",
                "dynamodb:Query",
                "dynamodb:DescribeTable",
                "dynamodb:UpdateTable"
            ],
            "Resource": "arn:aws:dynamodb:*:YOUR_ACCOUNT_ID:table/nf-desktop*"
        }
    ]
}
```

IAM Policy for AWS STS (Security Token Service) Role Assumption

This policy grants permissions to assume specific IAM roles using AWS STS (Security Token Service), enabling secure and temporary access to AWS resources.

- **Copy the policy JSON**
- **Replace** YOUR_ACCOUNT_ID with your actual AWS account ID
- Replace AWS_REGION with your desired AWS region
- **Create the policy in AWS IAM: neuraflash-desktop-cdk-sts**

Policy Breakdown AWS STS

- Allows assuming roles using sts:AssumeRole and sts:AssumeRoleWithSAML.
- Enables access to specific IAM roles for deployment, CloudFormation execution, and file publishing.

This policy ensures controlled access to deployment-related IAM roles, supporting AWS CDK and CloudFormation workflows while maintaining security best practices.

JavaScript

```
{
   "Version": "2012-10-17",
   "Statement": [
       {
           "Sid": "stsBase",
           "Effect": "Allow",
           "Action": ["sts:AssumeRole", "sts:AssumeRoleWithSAML"],
           "Resource": [

"arn:aws:iam::YOUR_ACCOUNT_ID:role/cdk-hnb659fds-deploy-role-YOUR_ACCOUNT_ID-AWS_REGION",

"arn:aws:iam::YOUR_ACCOUNT_ID:role/cdk-hnb659fds-cfn-exec-role-YOUR_ACCOUNT_ID-AWS_REGION",

"arn:aws:iam::YOUR_ACCOUNT_ID:role/cdk-hnb659fds-file-publishing-role-YOUR_ACCOUNT_ID-AWS_REGION"
           ]
       }
   ]
}
```