DEPLOYMENT SCRIPT EXPLANATION

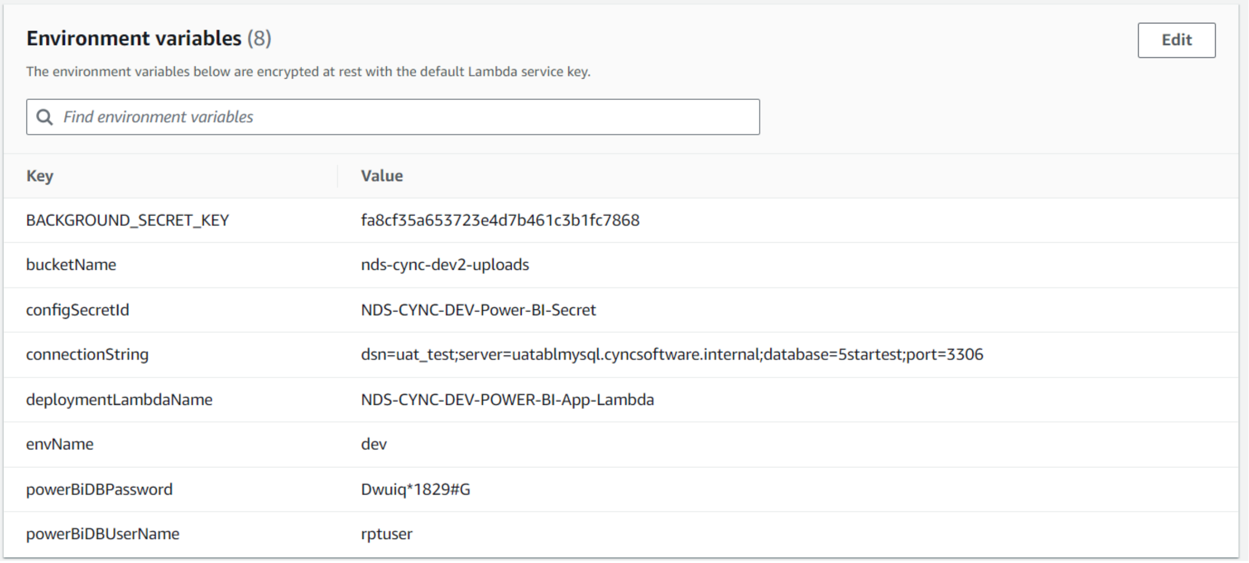
This document explains everything about this deployment script, how you can deploy and check if the report is successfully deployed.

What the Script Does

This script simplifies the process of moving reports and related components from one environment to another within a software system, for example, from "dev2" to "staging". It's not just about moving reports; it also handles the deployment of associated menus. In other words, you can think of this script as a tool for deploying reports based on specific menus.

prerequisite

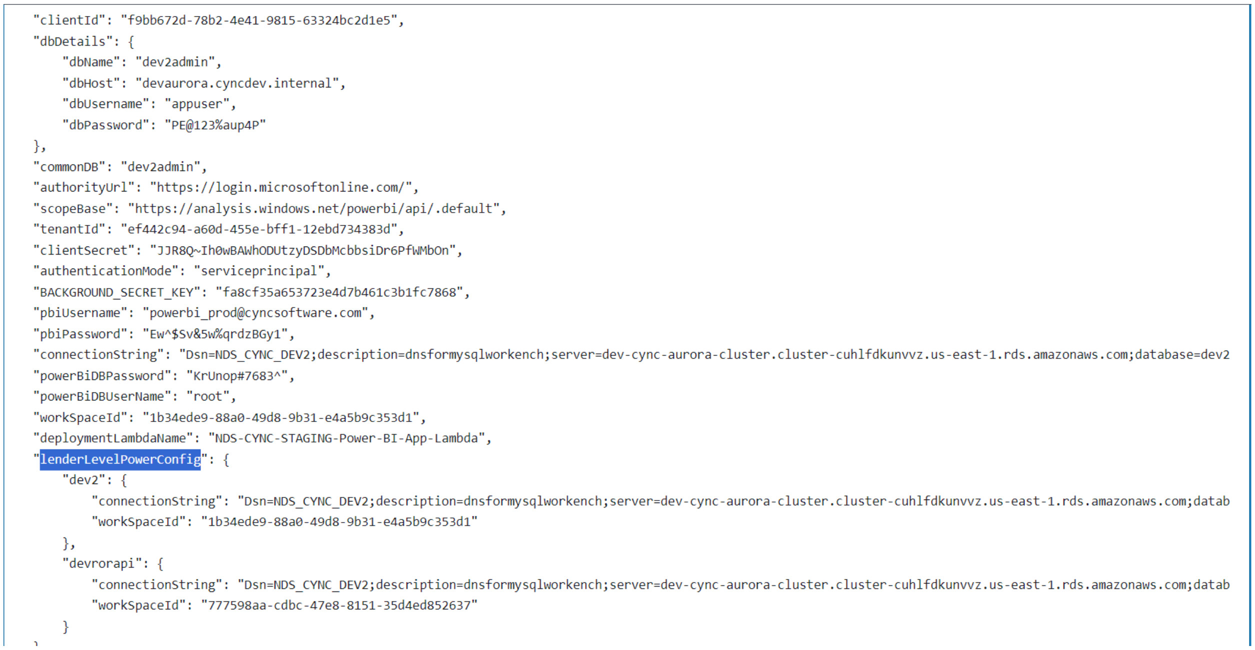
- Add key deploymentLambdaName in Environment variable .
- Add powerBiDBUserName in Environment Variable .
- Add powerBiDBPassword in Environment Variable .
- Give access lambda access for self invocation .
- Add `workSpaceId in secretManager .`
- Add `connectionString key in secretManager .`
- Add pbiusername `key in secretManager .`
- Add pbiPassword `key in secretManager .`
- `lenderLevelPowerConfig in secretManager .`



Environment variables (8)

The environment variables below are encrypted at rest with the default Lambda service key.

Key	Value
BACKGROUND_SECRET_KEY	fa8cf35a653723e4d7b461c3b1fc7868
bucketName	nds-cync-dev2-uploads
configSecretId	NDS-CYNC-DEV-Power-BI-Secret
connectionString	dsn=uat_test;server=uatablmysql.cyncsoftware.internal;database=5startest;port=3306
deploymentLambdaName	NDS-CYNC-DEV-POWER-BI-App-Lambda
envName	dev
powerBiDBPassword	Dwuiq*1829#G
powerBiDBUserName	rptuser



```
    "clientId": "f9bb672d-78b2-4e41-9815-63324bc2d1e5",
    "dbDetails": {
        "dbName": "dev2admin",
        "dbHost": "devaurora.cyncdev.internal",
        "dbUsername": "appuser",
        "dbPassword": "PE@123%aup4P"
    },
    "commonDB": "dev2admin",
    "authorityUrl": "https://login.microsoftonline.com/",
    "scopeBase": "https://analysis.windows.net/powerbi/api/.default",
    "tenantId": "ef442c94-a60d-455e-bff1-12ebd734383d",
    "clientSecret": "JJR8Q~Ih0wBAWhODUtzyDSDbMcbbsiDr6PfWMbOn",
    "authenticationMode": "serviceprincipal",
    "BACKGROUND_SECRET_KEY": "fa8cf35a653723e4d7b461c3b1fc7868",
    "pbiUsername": "powerbi_prod@cyncsoftware.com",
    "pbiPassword": "Ew^$Sv&5w%qrdzBGy1",
    "connectionString": "Dsn=NDS_CYNC_DEV2;description=dnsformysqlworkench;server=dev-cync-aurora-cluster.cluster-cuhlfdkunvvz.us-east-1.rds.amazonaws.com;database=dev2
    "powerBiDBPassword": "KrUnop#7683^",
    "powerBiDBUserName": "root",
    "workSpaceId": "1b34ede9-88a0-49d8-9b31-e4a5b9c353d1",
    "deploymentLambdaName": "NDS-CYNC-STAGING-Power-BI-App-Lambda",
    "lenderLevelPowerConfig": {
        "dev2": {
            "connectionString": "Dsn=NDS_CYNC_DEV2;description=dnsformysqlworkench;server=dev-cync-aurora-cluster.cluster-cuhlfdkunvvz.us-east-1.rds.amazonaws.com;datab
            "workSpaceId": "1b34ede9-88a0-49d8-9b31-e4a5b9c353d1"
        },
        "devrorapi": {
            "connectionString": "Dsn=NDS_CYNC_DEV2;description=dnsformysqlworkench;server=dev-cync-aurora-cluster.cluster-cuhlfdkunvvz.us-east-1.rds.amazonaws.com;datab
            "workSpaceId": "777598aa-cdbc-47e8-8151-35d4ed852637"
        }
```

Before you start:

1. Make sure you have DeploymentScript and package.json .

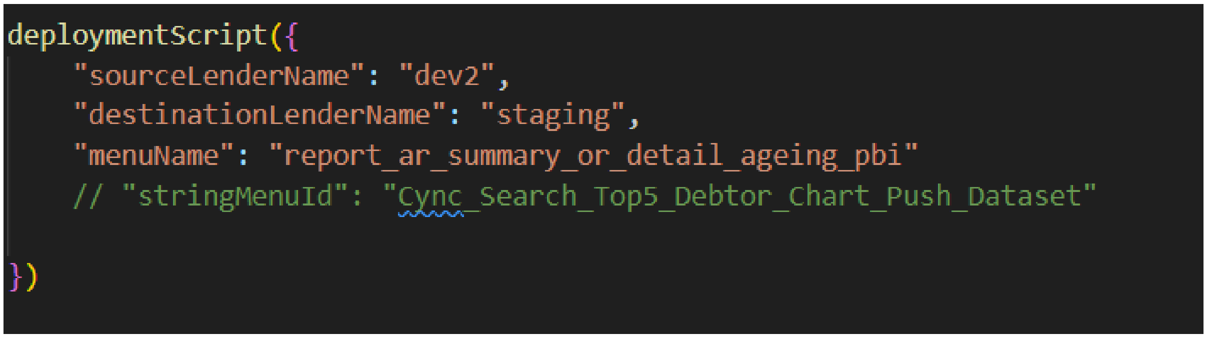
2. You have to install Node js You can download it from here https://nodejs.org/dist/v20.12.2/node-v20.12.2-x64.msi .

1. Additionally, the script relies on certain packages to work correctly. These packages are installed using Node.js's package manager, run this command npm install axios .
2. You'll need to use a terminal, like the one in VS Code, to run the script.

How to Use It

Utilizing the script is straightforward:

1. Specify which reports require movement and identify their current location (e.g., "dev2") and intended destination (e.g., "staging").
2. Specify which reports you want to deploy from one environment to another environment like dev2 to staging .

3. 

```
deploymentScript({
    "sourceLenderName": "dev2",
    "destinationLenderName": "staging",
    "menuName": "report_ar_summary_or_detail_ageing_pbi"
    // "stringMenuId": "Cync_Search_Top5_Debtor_Chart_Push_Dataset"

})
```

   In the above Image as you can see we are passing sourceLenderName , destinationLenderName and either menuName or else stringMenuId if it is pushDataset .

4. In SourceLenderName you should pass Source Lender from where you are taking report.
5. In destinationLenderName you have to pass lender where you want to deploy the report.
6. In menuName /stringMenuId either you have to menu name if it is normal paginated report and stringMenuId if it is push dataset report .

After completion of script execution

Requirements for Operation

- For optimal functionality, the script relies on specialized files containing pertinent information regarding report locations and destination specifications. These files serve as crucial references for the script's operation.

Illustrative Example

Consider a scenario where new reports are created within the development area and subsequently require testing in the staging environment before dissemination to stakeholders. By indicating the reports' current location (development) and the desired destination (staging), initiating the script initiates the seamless transfer of reports between designated areas.

Conclusion

In summary, the deployment script significantly simplifies the process of managing and transferring reports within the CyncSoftware ecosystem. By automating intricate backend processes, the script empowers users to focus on their core responsibilities without the burden of manual file management tasks