**Stripe Reader**

To add new device to stripe dashboard navigate:

Payments => select Readers tab => select existing Location or add New => register New reader.

For WisePOS-E reader swipe right from left edge to access settings. Default PIN: 07139

Generate new pairing key and enter it into Stripe dashboard to add new reader device.

For testing, record secret stripe test key into SSM parameter store. Key starts with sk\_test\_,,,

**SSM Parameter Store**

Parameters and variables are stored in SSM parameter store in a tree format, in order to retrieve multiple variables by path.

|  |  |
| --- | --- |
| Variable | Description |
| /taptodonate/benevity/client\_id | Benevity client id |
| /taptodonate/benevity/client\_secret | Benevity client secret |
| /taptodonate/benevity/o-auth-url | Benevity oAuth server url |
| /taptodonate/rds/db-name | RDS Database name |
| /taptodonate/rds/endpoint-url | RDS endpoint url |
| /taptodonate/rds/user | RDS user name = admin |
| /taptodonate/rds/password | Rds admin password |
| /taptodonate/rds/port | Rds port |
| /taptodonate/sqs/endpoint-url | Queue endpoint url |
| /taptodonate/stripe/test-key | Stripe test key. sk\_test… |

**Lambdas**

Lambdas to communicate with RDS database:

|  |  |  |
| --- | --- | --- |
| Lambda | Role | Layer |
| assignConfigToReader | ssm-read-and-db-vpc-access | mysql-node-module |
| getReadersTable | ssm-read-and-db-vpc-access | mysql-node-module |
| getDonationRecords | ssm-read-and-db-vpc-access | mysql-node-module |
| getConfigsTable | ssm-read-and-db-vpc-access | mysql-node-module |
| getActiveConfigsTable | ssm-read-and-db-vpc-access | mysql-node-module |
| getConfig | ssm-read-and-db-vpc-access | mysql-node-module |
| removeConfig | ssm-read-and-db-vpc-access | mysql-node-module |
| postConfig | ssm-read-and-db-vpc-access | mysql-node-module |

Lambdas to call Benevity API’s

|  |  |  |
| --- | --- | --- |
| Lambda | Role | Layers |
| capturePaymentIntent | Capture-payment-intent-role | Aws-sdk-node-module  Axios-node-module  Mysql-node-module  Stripe-node-module |
| searchForCause | ssm-read-access | axios-node-module |
| getCauseDetails | ssm-read-access | axios-node-module |
| sendReceipt | Send-receipt-role | Axios-node-module  Aws-sdk-node-module |

Lambdas to communicate with Stripe servers:

|  |  |  |
| --- | --- | --- |
| Lambda | Role | Layer |
| retrieveConnectionToken | Secret-acces-role | stripe-node-module  aws-sdk-node-module |
| createPaymentIntent | Secret-acces-role | Stripe-node-module |

**Roles**

Create custom roles with following permissions.

|  |  |
| --- | --- |
| Role Name | Permissions Policies |
| ssm-read-access | AmazonSSMReadOnlyAccess |
| capture-payment-intent-role | AmazonSSMReadOnlyAccess AmazonSQSFullAccess  AWSLambdaSQSQueueExecutionRole  CloudWatchFullAccess |
| send-receipt-role | AmazonSSMReadOnlyAccess AmazonSQSFullAccess  AWSLambdaSQSQueueExecutionRole |
| ssm-read-and-db-vpc-access | AWSLambdaVPCAccessExecutionRole  AmazonSSMReadOnlyAccess |

**Layers**

Layers are used to upload node-modules.

To create a layer zip up nodejs folder and upload to AWS.

|  |  |
| --- | --- |
| Layer Name | Dependencies |
| axios-node-module | Aixos: 0.27.2 |
| mysql-node-module | Mysql: 2.18.1 |
| stripe-node-moduel | Stripe: 9.13.0 |

**API Gateway**

GoodnessKiosk-API

|  |  |  |
| --- | --- | --- |
| Routes | Method | Lambda Integrations |
| /assignConfigToReader | POST | assignConfigToReader |
| /capturePaymentIntent | POST | capturePaymentIntent |
| /createPaymentIntent | POST | createPaymentIntent |
| /getActiveConfigsTable | POST | getActiveConfigsTable |
| /getCauseInfo | POST | getCauseDetails |
| /getConfig | POST | getConfig |
| /getConfigsTable | POST | getConfigsTable |
| /getDonationRecords | POST | getDonationRecords |
| /getReadersTable | POST | getReadersTable |
| /postNewConfig | POST | postConfig |
| /removeConfig | POST | removeConfig |
| /retrieveConnectionToken | GET | retrieveConnectionToken |
| /searchForCauses | POST | searchForCause |

**RDS Database**

goodness-kiosk-rds.cpe0spwckwwh.us-east-1.rds.amazonaws.com

admin

Pa$$w0rd123

Security group

Add inbound rule to security group: 0.0.0.0/0

Database is created with public access, in order to connect via MySQL workbench.

To create and seed the database, run .sql script which will create donation\_kiosk\_rds and populate with initial data.

**SQS Standard Queue**

Standard queue is used to delay sending out email receipts.

It takes up to 30 seconds for donation to be updated with receipt\_id, so we need to delay calling get donations/{id} API. After a minute, message becomes visible and lambda sendReceipt is triggered.

donation-ids-standard-queue

<https://sqs.us-east-1.amazonaws.com/635071431249/donation-ids-standard-queue>

|  |  |
| --- | --- |
| Configuraiton | Value |
| Type | Standard |
| Visibility Timeout | 1 min |
| Delivery Delay | 1 min |
| Receive Messege Wait Time | 20 sec |
| Message Retention Period | 4 days |
| Maxium Message Size | 256 kb |