# Getting Started with Baffle Data Protection Services (AWS)

Release 1.4.0.34

Document version 2.2

# Overview

This guide provides a walkthrough for getting started with Baffle Data Protection Services in AWS. It describes Baffle Manager and Baffle Shield system requirements and architecture, followed by configuration steps to set up Baffle's column level encryption. The configuration steps are divided into five main sections:

1. Configure Baffle Manager — the administrative console *(page 6)*

2. Connect to your Keystore — the source for encryption keys *(page 11)*

3. Connect to your data store *(page 13)*

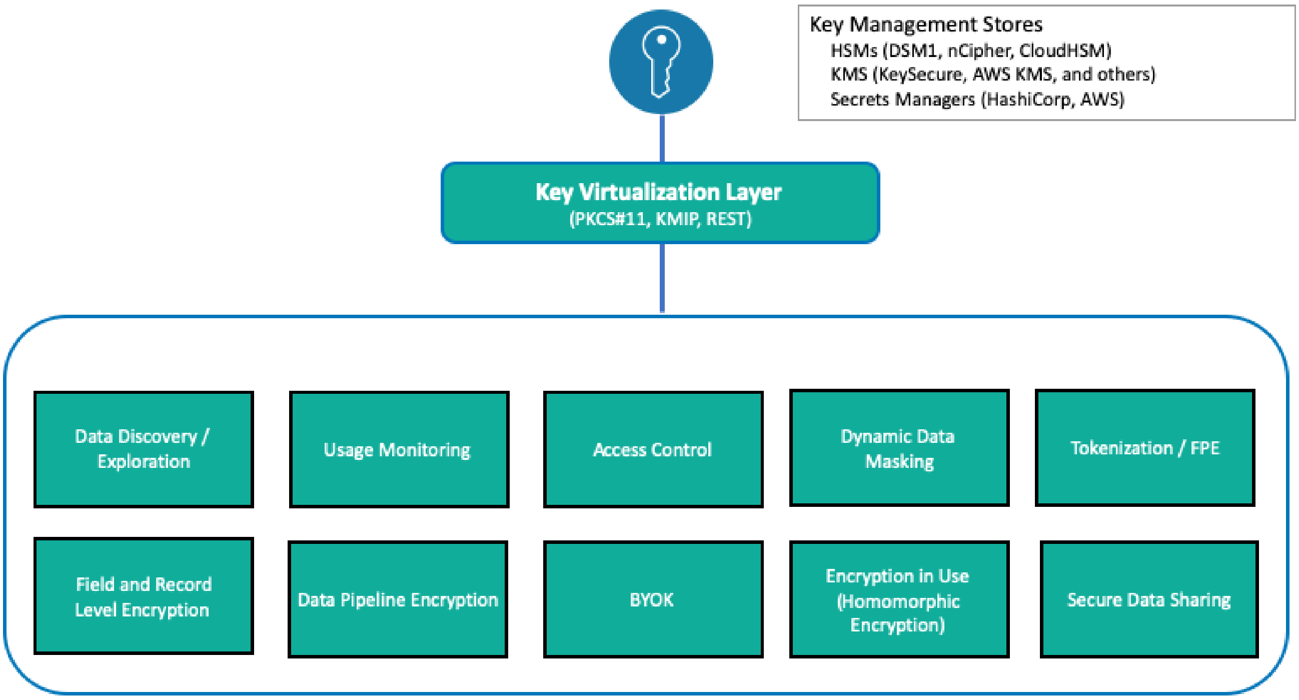
4. Configure a Baffle Shield — the encryption machine *(page 15)*

5. Define a data protection policy to encrypt your chosen fields *(page 18)*

## Background Information

Baffle Data Protection Services provide a range of data encryption, tokenization and de-identification methods to protect data in data stores and cloud storage environments. Common methods that Baffle employs include column or field level encryption, tokenization, format preserving encryption (FPE), dynamic data masking, and record level encryption.

Baffle integrates with key management stores via a key virtualization layer. It can also provide its own local key store, for customers to use their own keys to apply data protection in the cloud.


Key Management Stores
HSMs (DSM1, nCipher, CloudHSM)
KMS (KeySecure, AWS KMS, and others)
Secrets Managers (HashiCorp, AWS)
Key Virtualization Layer
(PKCS#11, KMIP, REST)
Data Discovery / Exploration
Usage Monitoring
Access Control
Dynamic Data Masking
Tokenization / FPE
Field and Record Level Encryption
Data Pipeline Encryption
BYOK
Encryption in Use (Homomorphic Encryption)
Secure Data Sharing

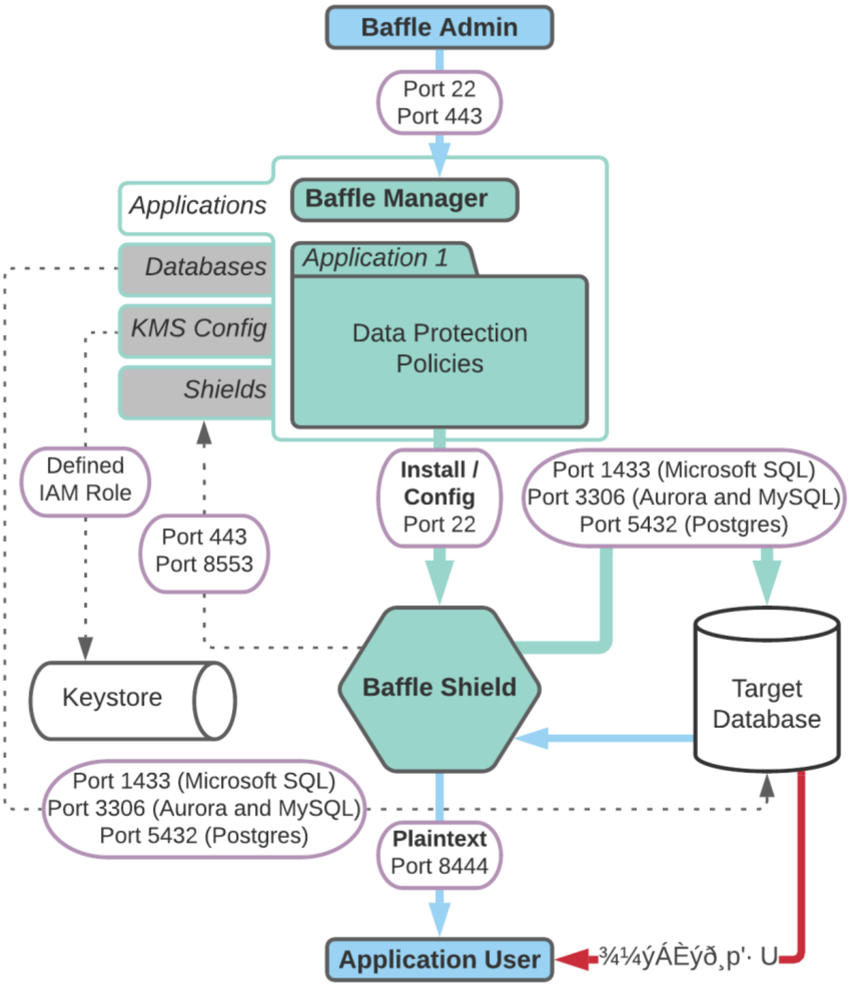
## Pre-Requisites and Minimum System Requirements

Whether you use Baffle Professional Services to perform your deployment testing, or your organization does so independently as part of planning, ensure that your test environment meets the following minimum system requirements.

<table>
<tr><th>Baffle Component</th><th>Operating System</th><th>vCPU</th><th>Memory</th><th>Initial Space</th><th>Java</th></tr>
<tr><td>Baffle Manager</td><td>CentOS 7</td><td>2</td><td>8 GB</td><td>64 GB</td><td>OpenJDK Java 1.8</td></tr>
<tr><td>Baffle Shield</td><td>RHEL 7 or CentOS 7 equivalent</td><td>4</td><td>8 GB</td><td>64 GB[1]</td><td>OpenJDK Java 1.8</td></tr>
<tr><td>Database Platform</td><td>AWS RDS, Azure SQL and other supported database platforms[1]</td><td>16</td><td>256 GB</td><td>512 GB</td><td>OpenJDK Java 1.8</td></tr>
<tr><th colspan="6">Prerequisite Information for Data Encryption</th></tr>
<tr><td>Data Schema</td><td colspan="5">• Number of columns to be encrypted<br>• Data types and column field names<br>• Number of rows in table(s)<br>• Database size; Indexing, if any</td></tr>
<tr><td>Application</td><td colspan="5">• Identify the application and associated data for testing (for example, Microsoft SQL Server 2014 or later)<br>• Set aside a copy of the application and data to expedite troubleshooting and diagnostics.<br>• Provide test data that is encoded using UTF-8 character set.</td></tr>
<tr><td>Key Storage</td><td colspan="5">• Provide a supported key storage solution (see Key Management Support in the Baffle support center)<br>• Provide associated encryption keys<br>• Host in AWS and make available to Baffle infrastructure</td></tr>
</table>

[1] Additional supported database platforms are listed in the Baffle support center

## Baffle Architecture and Communication



## Port Requirements

Baffle Manager enables encryption policies and configurations by communicating with the Baffle Shield and your databases. Baffle Manager constructs a privacy schema that maps key IDs to data columns, thus enabling encryption in a simplified manner.

The following table lists the ports that must allow connections in order for Baffle Manager to communicate.

| Host | Port Required | Direction | Purpose |
|---|---|---|---|
| Baffle Manager | 22 | Inbound | Console access for admin |
| Baffle Manager | 443 | Inbound | Web interface access for admin |
| Baffle Manager | 8553 | Inbound | Baffle Shield client access |
| Baffle Manager | 22 | Outbound | Baffle Shield configuration |
| Baffle Manager | 1433 | Outbound | Database schema mapping |
| Baffle Manager | 5696 | Outbound | (Optional) KeySecure access |
| Baffle Shield | 22 | Inbound | Console and Baffle Manager access |
| Baffle Shield | 8444 | Inbound | Application communication |
| Baffle Shield | 1433 | Outbound | Database access[1] |
| Baffle Shield | 3306 | Outbound | Database access[2] |
| Baffle Shield | 5432 | Outbound | Database access[3] |
| Baffle Shield | 5696 | Outbound | KeySecure access |
| Baffle Shield | 8553 | Outbound | Baffle Manager communications |
| Database Server[1] | 1433 | Inbound | Baffle Manager and Baffle Shield access |
| Database Server[2] | 3306 | Inbound | Baffle Manager and Baffle Shield access |
| Database Server[3] | 5432 | Inbound | Baffle Manager and Baffle Shield access |
| KeySecure | 5696 | Inbound | (Optional) Baffle Manager and Baffle Shield key config and retrieval |

[1] For Microsoft SQL Server default port communications

[2] For MySQL, MariaDB or Aurora server default port communications

[3] For PostgreSQL server default port communications

# Configuration Walkthrough (AWS)

## Section 1. Launch and configure the Baffle Manager AMI from AWS Marketplace

1. Search for Baffle in the AWS Marketplace, or click the following link to begin setup – Baffle Data Protection Services.

2. Launch an EC2 instance for Baffle Manager with the following settings.

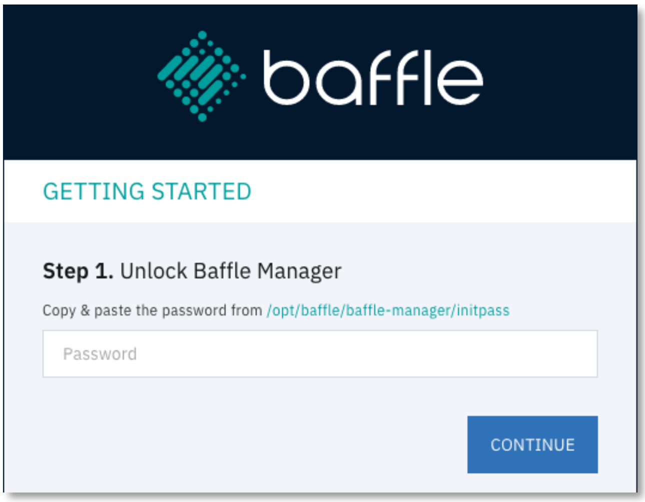
    a. Create a new security group on the VPC based on ‘seller settings’. This configuration opens the necessary ports for Baffle Manager. Set the range of IP addresses that will be permitted access.

    b. Ensure you have saved the selected key pair to access the Baffle Manager.

3. Once the instance is running, connect to it with a web browser via HTTPS. Use the public IP address of the instance. For example, https://192.168.1.1 as an address.

    If you are unable to connect to the instance via HTTPS, check your security group inbound rules. Also ensure that your instance has finished initializing.

    Because the instance is bootstrapped with a self-signed certificate, you will receive an invalid CA warning. Select the browser option to “proceed”. (You will have the opportunity to upload and use your organization’s certificate later in this section.) The following window should appear:



GETTING STARTED

**Step 1.** Unlock Baffle Manager

Copy & paste the password from /opt/baffle/baffle-manager/initpass

    This indicates that the Baffle Manager is in a locked state.

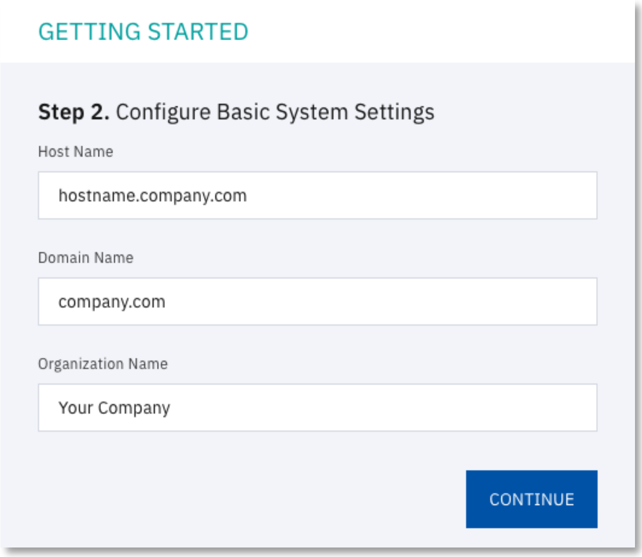
4. To unlock the Baffle Manager, access the system via SSH. Use “baffle” as the username for the SSH connection, followed by the public IP address (for example, baffle@192.168.1.1). You will also need the key pair file that you selected when you launched the instance.

5. Once you have connected to the instance via SSH, issue the following command to retrieve the unlock code.

```
sudo more /opt/baffle/baffle-manager/initpass
```

6. In your browser, paste the unlock code into the password field and click CONTINUE.

7. **Configure System Settings.** You will be prompted for hostname and domain settings. All system users must have this domain name as part of this email going forward.



GETTING STARTED

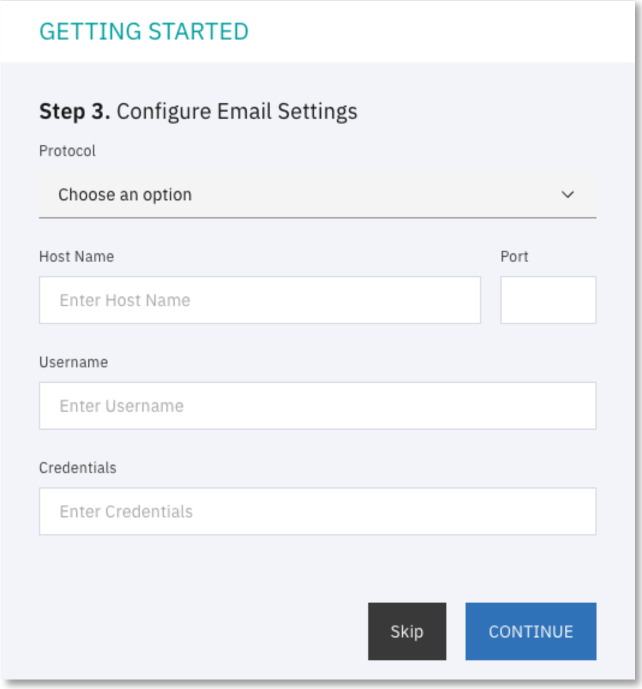
**Step 2.** Configure Basic System Settings

Host Name: hostname.company.com

Domain Name: company.com

Organization Name: Your Company

8. **Configure Email Settings.** This allows Baffle Manager to send emails to provide notifications and for password resets. Enter the SMTP server to use as well as the credential to use to authentication to the SMTP server.



GETTING STARTED

**Step 3.** Configure Email Settings

Protocol: Choose an option

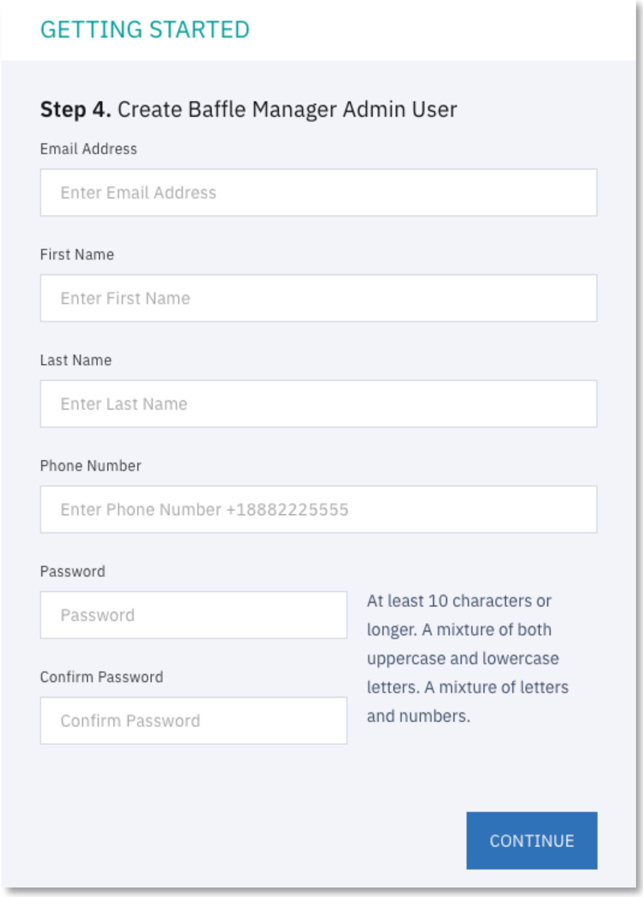
Host Name: Enter Host Name

Port:

Username: Enter Username

Credentials: Enter Credentials

9. **Create Admin Account.** The screen below prompts you to create the initial Baffle Manager administrator account. This account is used to configure the subsequent components such as the key management store, data store connections, and Baffle Shields.



GETTING STARTED

**Step 4.** Create Baffle Manager Admin User

Email Address

Enter Email Address

First Name

Enter First Name

Last Name

Enter Last Name

Phone Number

Enter Phone Number +18882225555

Password

Password

Confirm Password

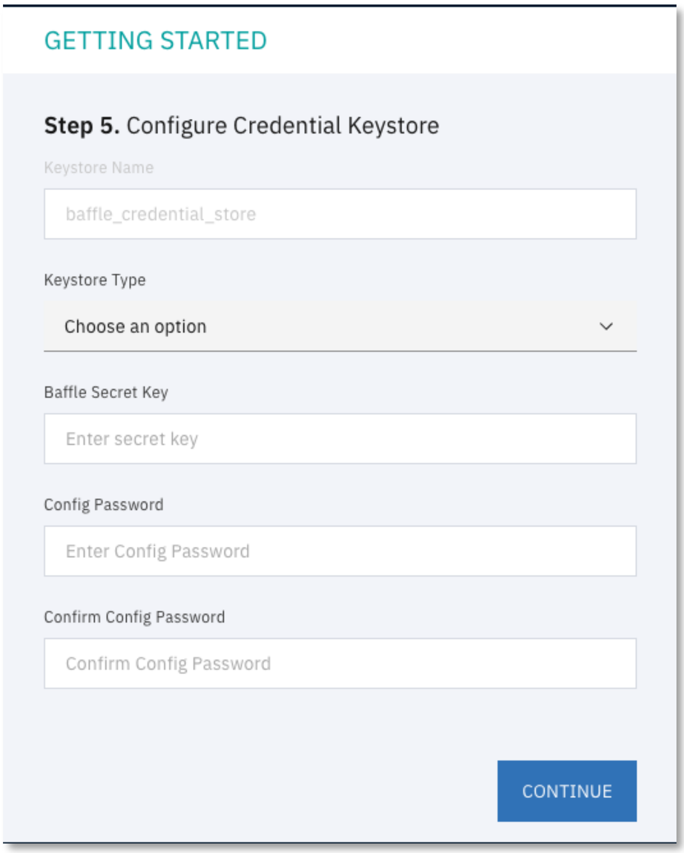
Confirm Password

At least 10 characters or longer. A mixture of both uppercase and lowercase letters. A mixture of letters and numbers.

CONTINUE

10. **Configure Credential Keystore.** This configuration screen establishes an encrypted credential store for any system access credential or access key that the Baffle Manager or Baffle Shield utilize. The default name is “baffle_credential_store” and cannot be changed.

    Select LOCAL for Keystore type.  For Secret Key, enter any random string which will be used to generate a random key to encrypt the Keystore Config Password.  For Config Password, enter a secure password or passphrase to secure the actual keystore.



GETTING STARTED

**Step 5.** Configure Credential Keystore

Keystore Name

baffle_credential_store

Keystore Type

Choose an option

Baffle Secret Key

Enter secret key

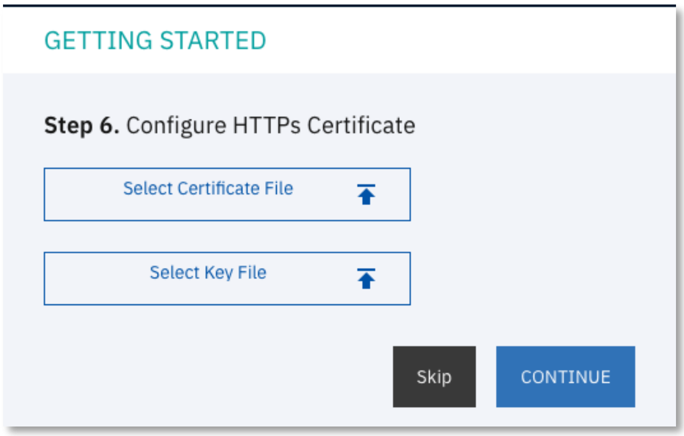
Config Password

Enter Config Password

Confirm Config Password

Confirm Config Password

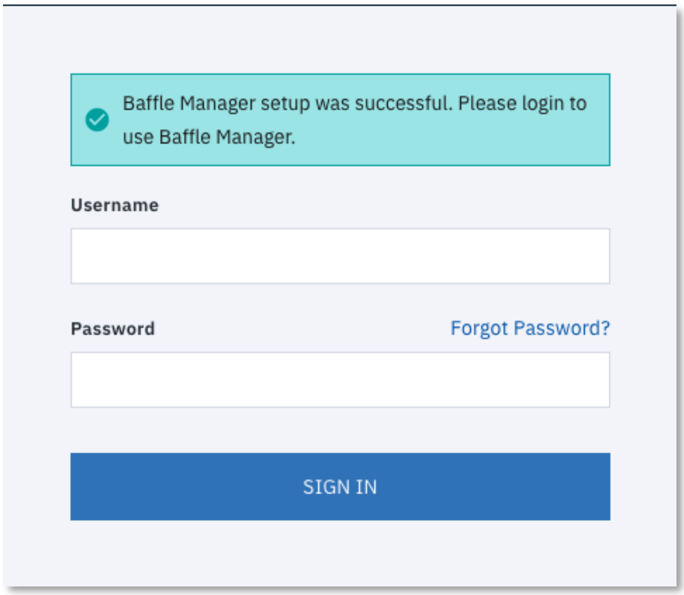
11. **Install SSL Certificate.** This configuration step allows you to install an SSL certificate to secure access to the Baffle Manager web interface.  Upload the certificate and key file for your organization or respective CA to enable SSL for the Baffle Manager console.



GETTING STARTED

**Step 6.** Configure HTTPs Certificate

12. This should complete the initial setup process and bring you to the login page.



✓ Baffle Manager setup was successful. Please login to use Baffle Manager.

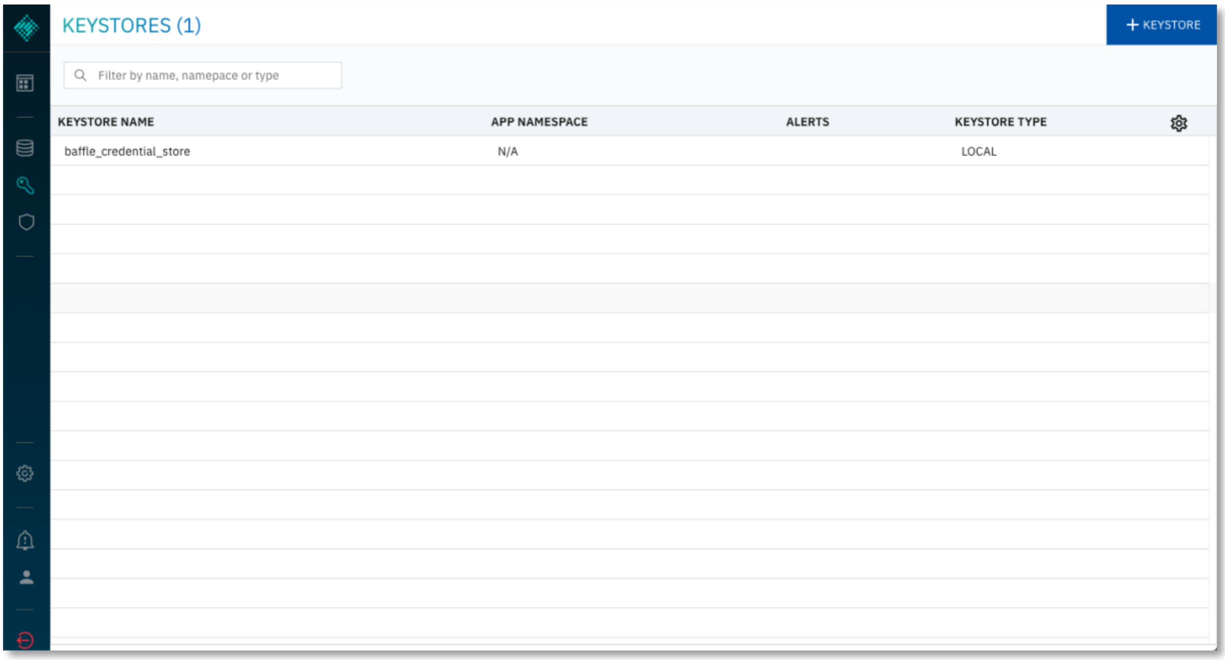
13. Enter the credentials for the administrator account you created in Step 9 to login and continue the configuration process.

## Section 2. Connect to a Keystore

Before you can enroll your applications, add databases and enable encryption, you must enroll your Keystore so that Baffle Manager can access and/or create data encryption keys (DEKs) that will be used to protect your data.

Baffle Data Protection Services supports various Keystore vendors using industry standard protocols such as KMIP, PKCS#11, and REST APIs. Follow the steps below to enroll a Keystore for use with Baffle Shields and databases.

1. **Display a list of configured keystores**. After logging into Baffle Manager, click the key icon on the left hand navigation panel. If this is the first time you are enrolling a Keystore, there will only exist the "baffle_credential_store" that was created in the previous section.



| KEYSTORE NAME | APP NAMESPACE | ALERTS | KEYSTORE TYPE |
|---|---|---|---|
| baffle_credential_store | N/A | | LOCAL |

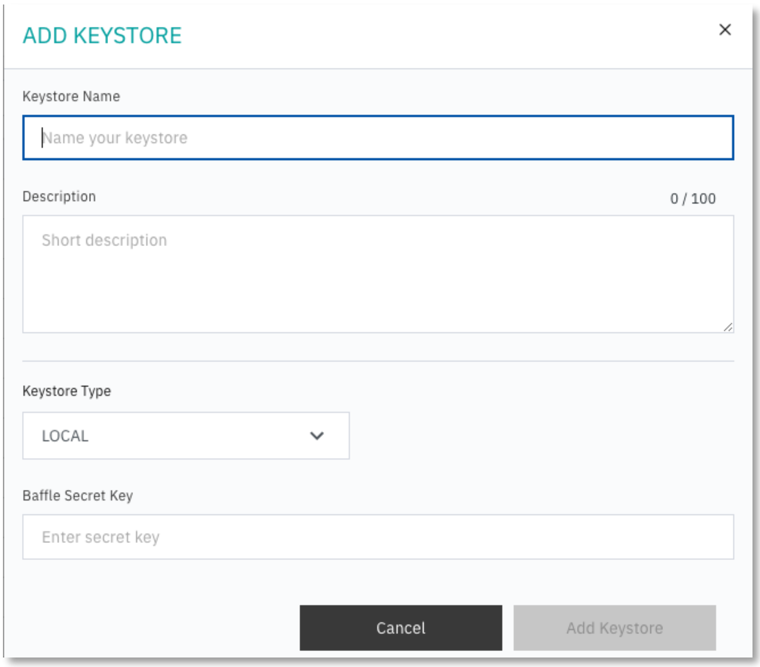
Click on the +KEYSTORE button in the top right corner to add a new Keystore.

2. **Enter a Keystore name** and description.

3. **Specify the Keystore Type** from the dropdown menu and enter respective parameters for the Keystore selected.
   Keystore parameters are specific to the Keystore type or vendor.

4. When completed, click on "**Add Keystore**".

Example of a Local Keystore configuration:



ADD KEYSTORE

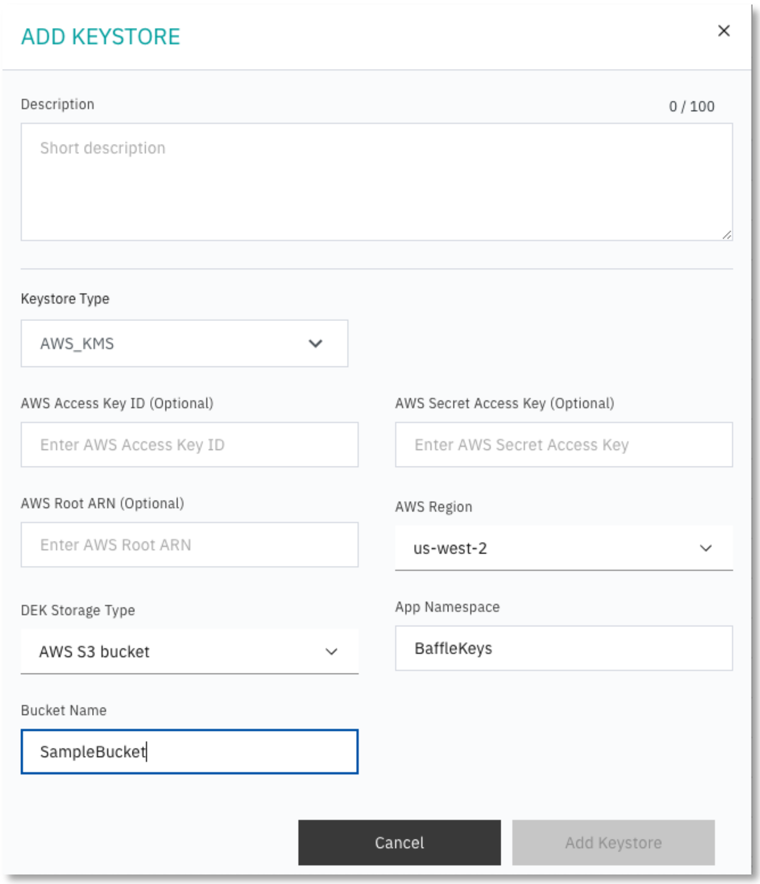
Keystore Name: Name your keystore

Description 0 / 100: Short description

Keystore Type: LOCAL

Baffle Secret Key: Enter secret key

Example of an AWS KMS configuration:



ADD KEYSTORE

Description 0 / 100: Short description

Keystore Type: AWS_KMS

AWS Access Key ID (Optional): Enter AWS Access Key ID

AWS Secret Access Key (Optional): Enter AWS Secret Access Key

AWS Root ARN (Optional): Enter AWS Root ARN

AWS Region: us-west-2

DEK Storage Type: AWS S3 bucket

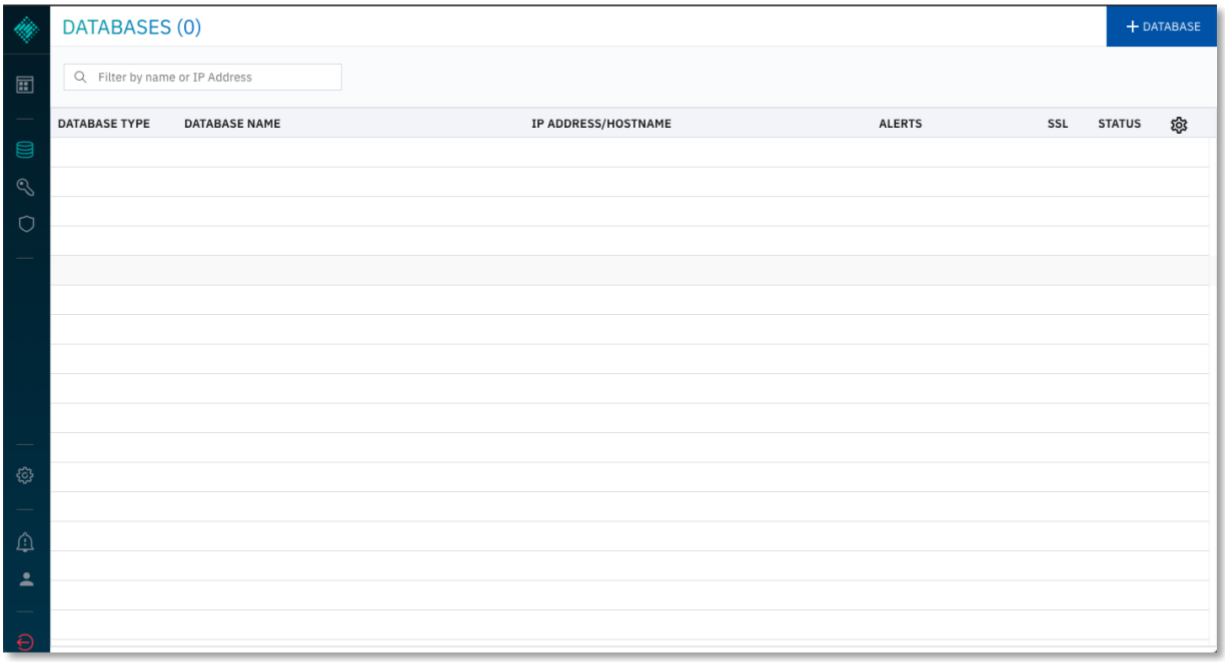
App Namespace: BaffleKeys

Bucket Name: SampleBucket

## Section 3. Connect to a Data Store

In this section, you will configure a connection to a database. This connection will allow Baffle Manager to enumerate fields or columns that can be selected as part of a data privacy policy, in order to enable column level encryption.

1. **Display the list of configured databases.** Click on the database icon on the left hand navigation panel to display a list of configured databases.



2. **Enroll a database**. Click on the +DATABASE button to add a Data Store. Enter a database name and description.

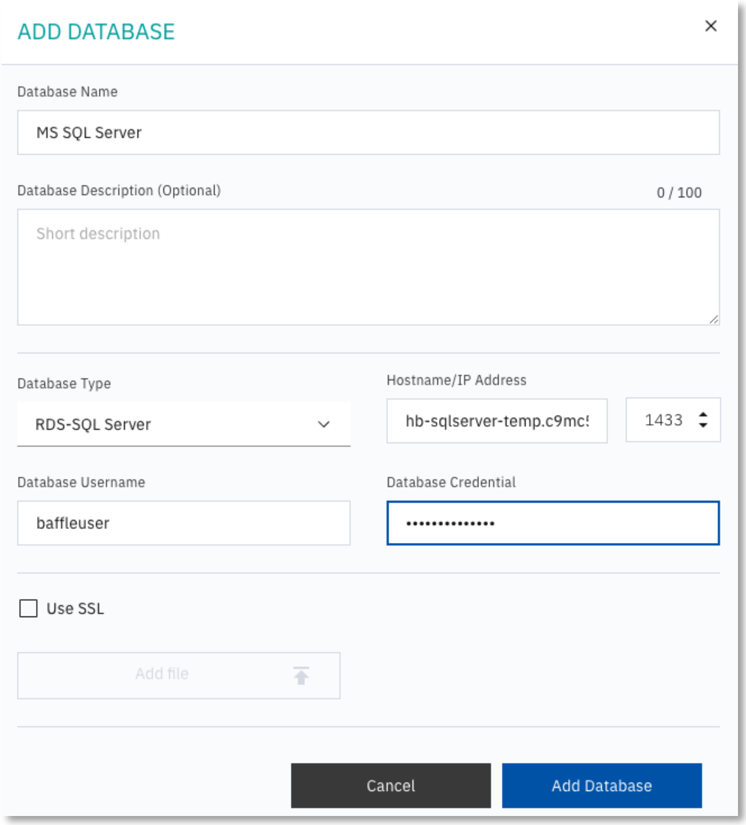
    a. Specify the database type. Then enter the hostname or IP and port of the database. Default database ports are found on page 5.

    b. Enter the database user credentials. **It is recommended that you create a new user on your database for use with Baffle.** See Appendix A (page 26) for details.

    To allow users on your database with less privileges to access the encrypted data, see Appendix B (page 27).

    c. Select Use SSL to enable an SSL/TLS connection to the database.

Below is an example of a Microsoft SQL Server configuration.

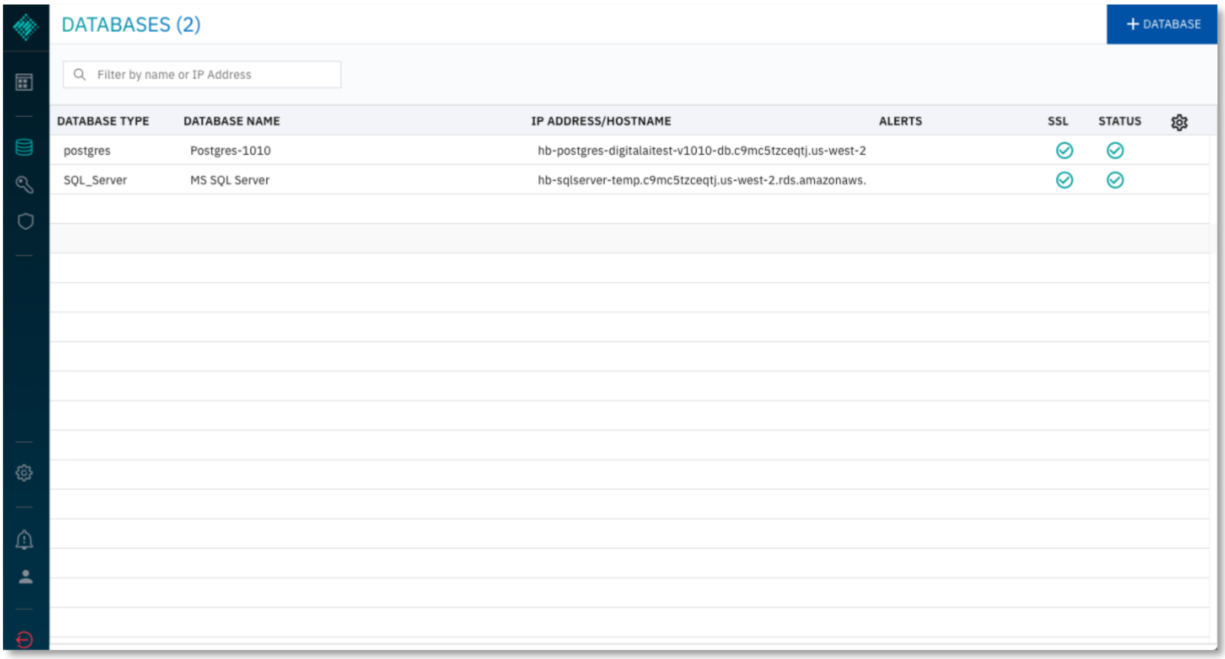


ADD DATABASE

| | |
|---|---|
| Database Name | MS SQL Server |
| Database Description (Optional) | Short description (0 / 100) |
| Database Type | RDS-SQL Server |
| Hostname/IP Address | hb-sqlserver-temp.c9mc! |
| Port | 1433 |
| Database Username | baffleuser |
| Database Credential | •••••••••••••• |

□ Use SSL

3. Click **Add Database** to complete enrollment. The new database should be listed along with the other configured databases as shown below.



DATABASES (2)

| DATABASE TYPE | DATABASE NAME | IP ADDRESS/HOSTNAME | ALERTS | SSL | STATUS |
|---|---|---|---|---|---|
| postgres | Postgres-1010 | hb-postgres-digitalaitest-v1010-db.c9mc5tzceqtj.us-west-2 | | ✓ | ✓ |
| SQL_Server | MS SQL Server | hb-sqlserver-temp.c9mc5tzceqtj.us-west-2.rds.amazonaws. | | ✓ | ✓ |

## Section 4. Launch and Configure a Baffle Shield AMI

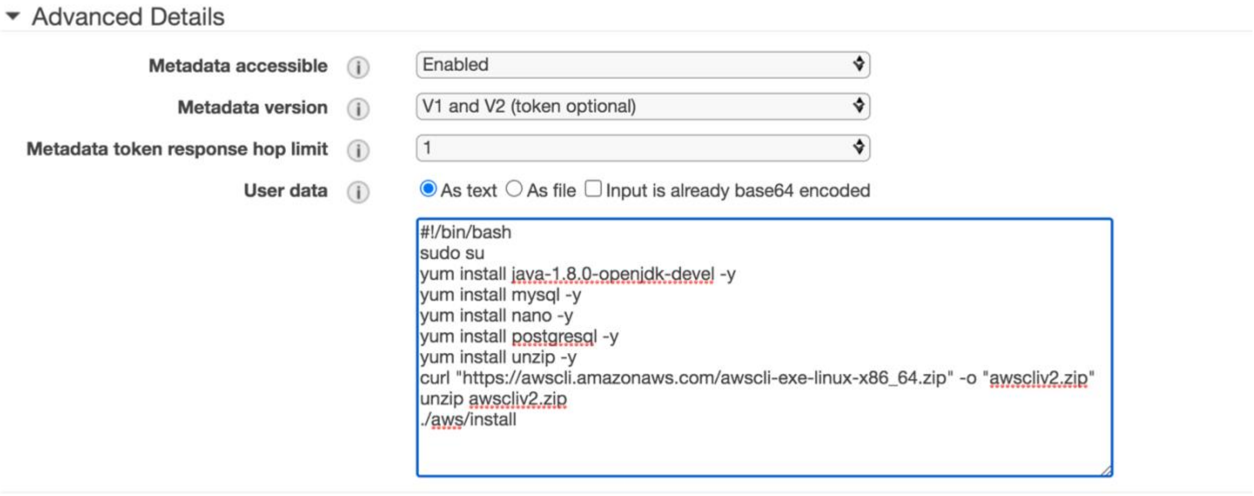
This section walks through the installation and configuration of a Baffle Shield. The Shield will be used to enforce a Data Protection Policy, encrypting the data in the databases that were configured in the previous section.

1. **Configure an AMI instance** to run the Baffle Shield.

   a. Launch a new AMI instance from EC2 that is appropriately sized for your environment. Run the AMI with a CentOS 7 operating system.

   b. Issue the following bootstrap commands in the Advanced Details section during the instance setup process.

```
#!/bin/bash
sudo su
yum install java-1.8.0-openjdk-devel -y
yum install mysql -y
yum install nano -y
yum install postgresql -y
yum install unzip -y
curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"
unzip awscliv2.zip
./aws/install
```



Advanced Details

Metadata accessible: Enabled

Metadata version: V1 and V2 (token optional)

Metadata token response hop limit: 1

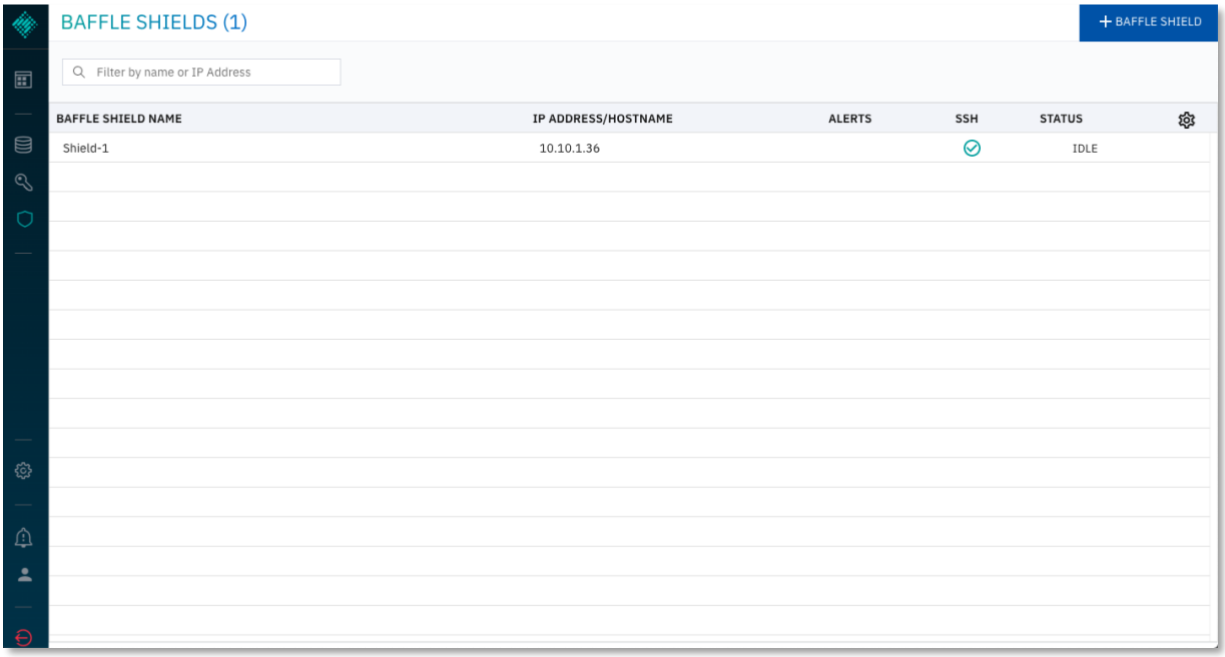
User data: ● As text ○ As file □ Input is already base64 encoded

```
#!/bin/bash
sudo su
yum install java-1.8.0-openjdk-devel -y
yum install mysql -y
yum install nano -y
yum install postgresql -y
yum install unzip -y
curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"
unzip awscliv2.zip
./aws/install
```

   c. Ensure the security group for your Baffle Shield allows inbound connections from Baffle Manager (on port 22) and from your own IP address (on port 8444 by default).

   d. Once you complete the setup process, allow the instance a few minutes to initialize.

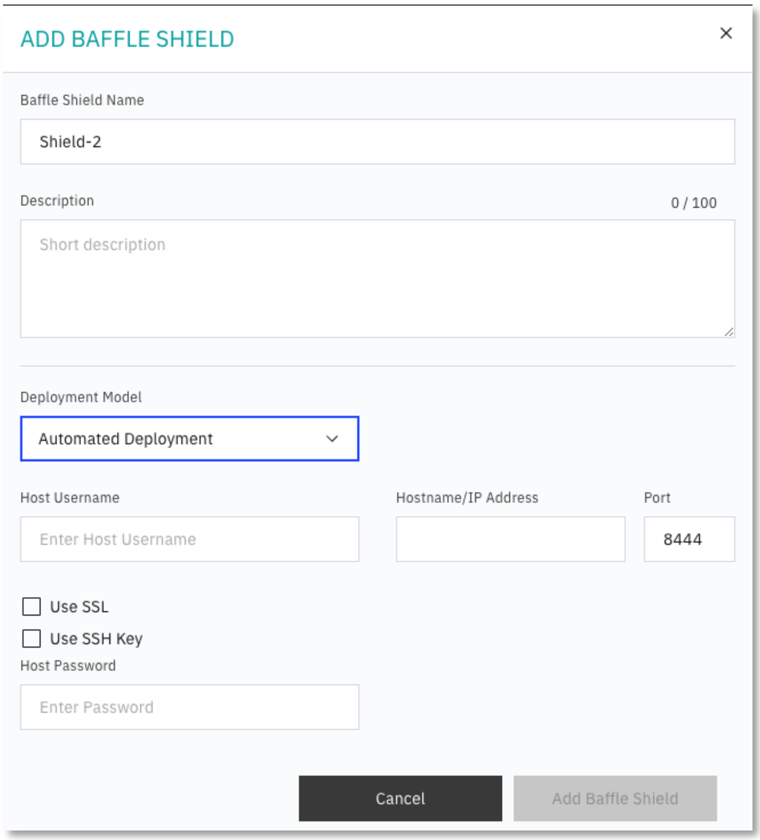
2. **Connect the Baffle Shield to Baffle Manager.** Once the instance is running, return to your Baffle Manager admin interface. Click on the shield icon on the left hand navigation panel. This will display a list of connected Baffle Shields. Click on the +BAFFLE SHIELD button in the upper right hand corner.



BAFFLE SHIELDS (1)

| BAFFLE SHIELD NAME | IP ADDRESS/HOSTNAME | ALERTS | SSH | STATUS |
|---|---|---|---|---|
| Shield-1 | 10.10.1.36 |  | ✓ | IDLE |

3. **Configure Baffle Shield.** Enter a name and description.



ADD BAFFLE SHIELD

Baffle Shield Name: Shield-2

Description 0 / 100

Deployment Model: Automated Deployment

Host Username

Hostname/IP Address

Port: 8444

□ Use SSL

□ Use SSH Key

Host Password

a. Select “Automated Deployment” for Deployment Model**.**

b. Enter the Host Username “centos” to access the Baffle Shield EC2 Instance.

c. Enter the IP Address of the Baffle Shield you have just launched. If your Shield runs in the same VPC as your Baffle Manager instance, it is recommended that you use the private IP address here.

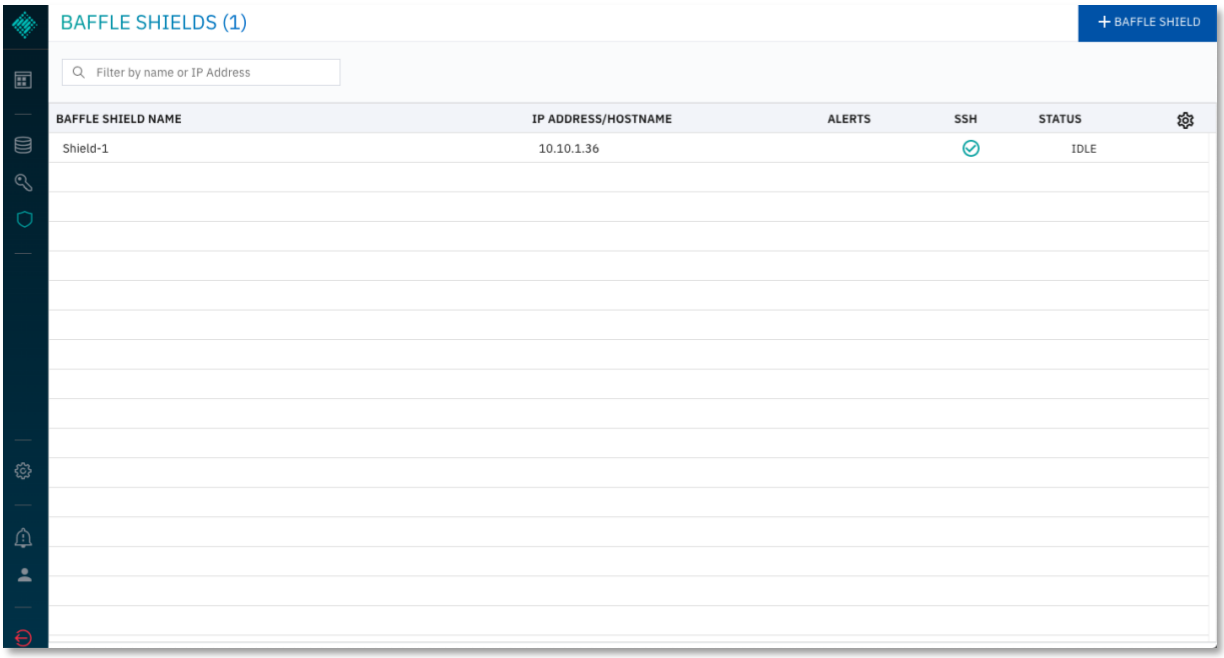
d. Enter a port number that the Baffle Shield will use to listen for application connections. The default port is 8444.

e. Select “Use SSL” if the data store connection uses SSL.

f. Select “Use SSH Key” and upload the key that you selected when you set up the Shield instance.

g. Optionally, a username and password can be used to access the Baffle Shield.

4. Click **Add Baffle Shield** to complete the process. The new Shield will be added to the list of configured Baffle Shields.



| BAFFLE SHIELD NAME | IP ADDRESS/HOSTNAME | ALERTS | SSH | STATUS |
|---|---|---|---|---|
| Shield-1 | 10.10.1.36 |  | ✓ | IDLE |

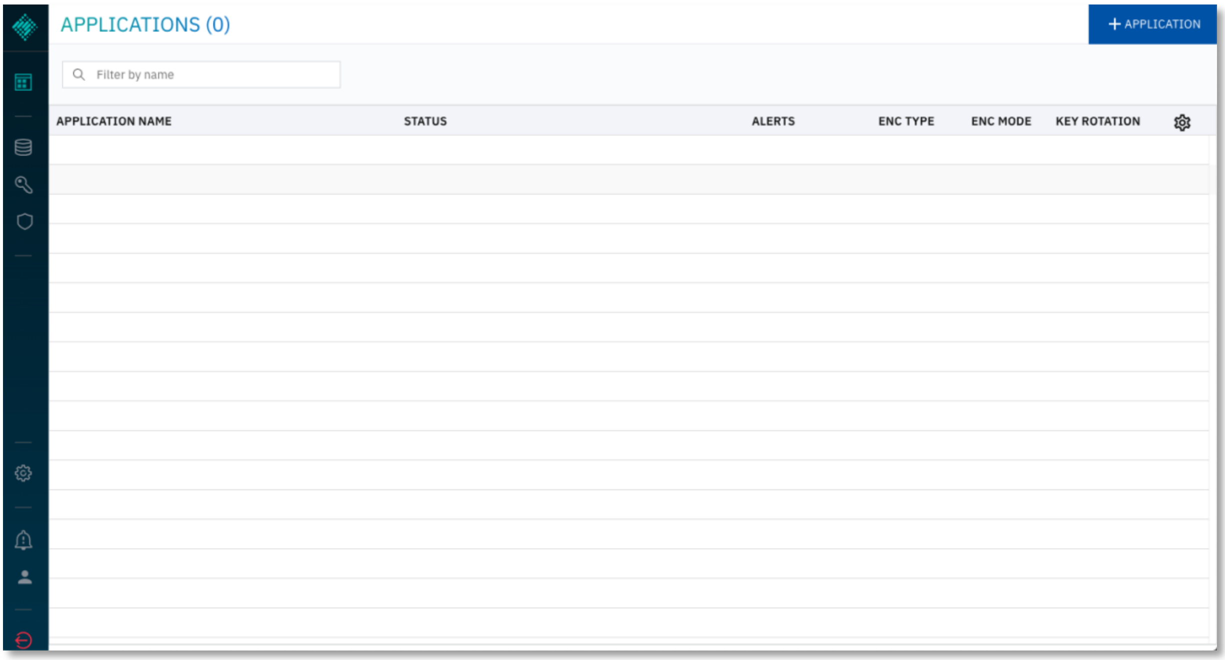
If the Baffle Manager is unable to connect to the shield, ensure that your Shield’s security group permits inbound access from Baffle Manager.

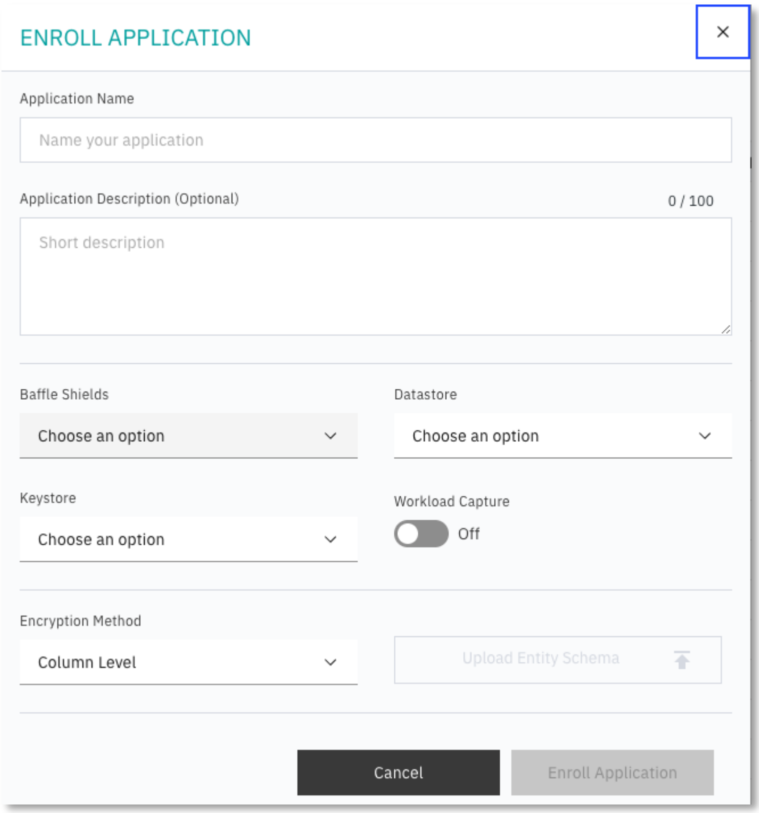
## Section 5. Define a Data Protection Policy and Encrypt your Data

Now, all the components of Baffle's Advanced Data Protection have been established. This section brings these components together, creating an application to execute a Data Protection Policy. The policy selects columns for encryption and keys that will be used for the encryption process. Upon completion of the Data Protection Policy, you can migrate data through a Baffle Shield and encrypt the existing data in your data store.

The creation of a Data Protection Policy establishes a Privacy Schema that Baffle Shields use to present the original data schema to a respective application while handling the encrypt and decrypt operations transparently for the configured fields.

1. **Add an Application to create a Data Protection Policy.** Click on the Applications Icon in the left hand navigation panel. The defined Data Protection Policies are displayed as Applications. Click on +APPLICATION.





ENROLL APPLICATION

Application Name

Name your application

Application Description (Optional) 0 / 100

Short description

Baffle Shields: Choose an option

Datastore: Choose an option

Keystore: Choose an option

Workload Capture: Off

Encryption Method: Column Level

2. **Enroll Application.** Enter a name and description.

   a. Choose the Baffle Shield from the drop down that was configured in the previous section.

   b. Select the Data Store which you will encrypt.

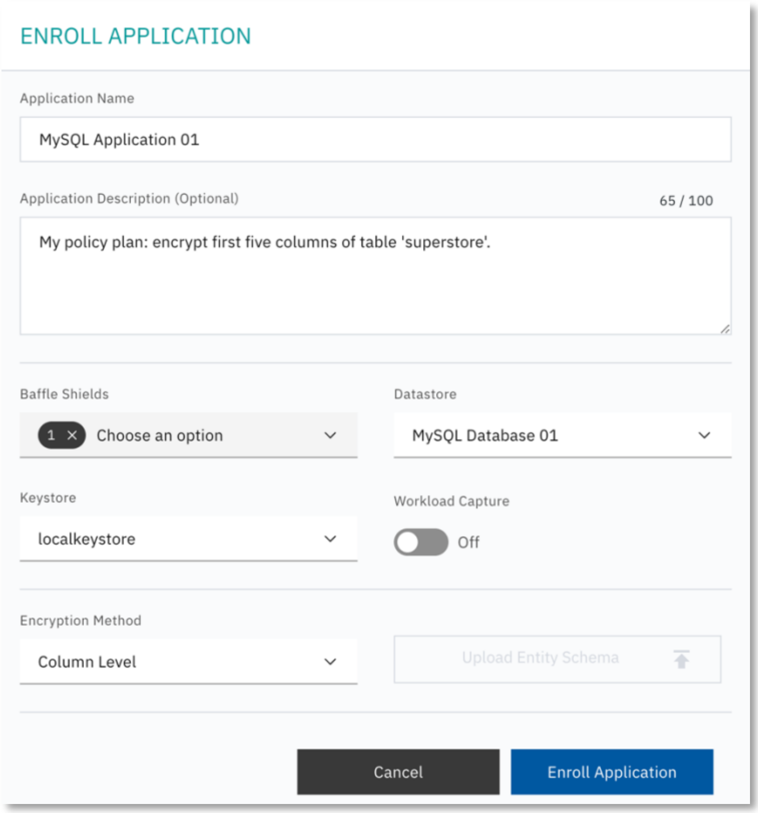
   c. Select the Keystore to be used as a source for data encryption keys.

   d. Specify the operational mode for the Baffle Shied. Leave Workload Capture Off, unless profiling an application.

   e. Specify Column Level for the Encryption Method.

   f. Click Enroll Application.

Below is an example of enrolling an application and deploying a Data Protection Policy for a MySQL database.



ENROLL APPLICATION

Application Name: MySQL Application 01

Application Description (Optional) 65 / 100

My policy plan: encrypt first five columns of table 'superstore'.

Baffle Shields: 1 × Choose an option

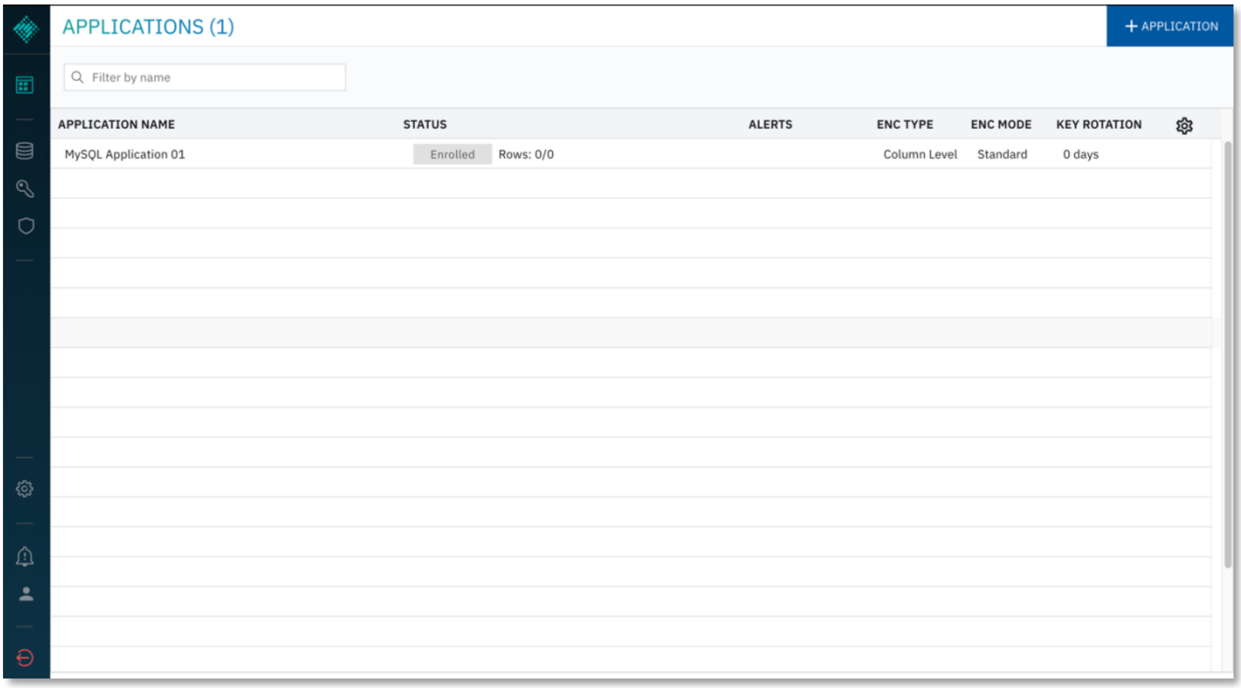
Datastore: MySQL Database 01

Keystore: localkeystore

Workload Capture: Off

Encryption Method: Column Level

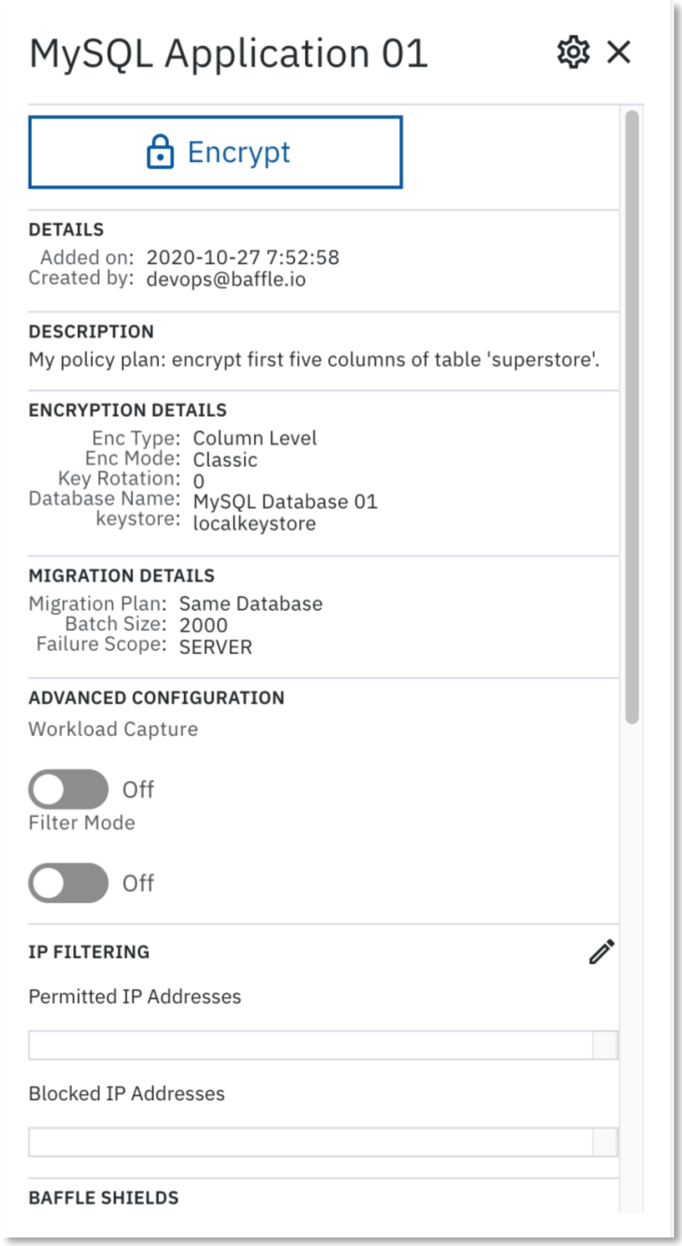
3. The Applications page now displays the new application.



APPLICATIONS (1)

| APPLICATION NAME | STATUS | ALERTS | ENC TYPE | ENC MODE | KEY ROTATION |
|---|---|---|---|---|---|
| MySQL Application 01 | Enrolled Rows: 0/0 | | Column Level | Standard | 0 days |

4. **Define the Data Protection Policy.** Click on the Application configured in the steps above. A side bar will display information about the application. Click on the ENCRYPT button to define the policy.



MySQL Application 01

DETAILS

Added on: 2020-10-27 7:52:58
Created by: devops@baffle.io

DESCRIPTION

My policy plan: encrypt first five columns of table 'superstore'.

ENCRYPTION DETAILS

Enc Type: Column Level
Enc Mode: Classic
Key Rotation: 0
Database Name: MySQL Database 01
keystore: localkeystore

MIGRATION DETAILS

Migration Plan: Same Database
Batch Size: 2000
Failure Scope: SERVER

ADVANCED CONFIGURATION

Workload Capture

Off

Filter Mode

Off

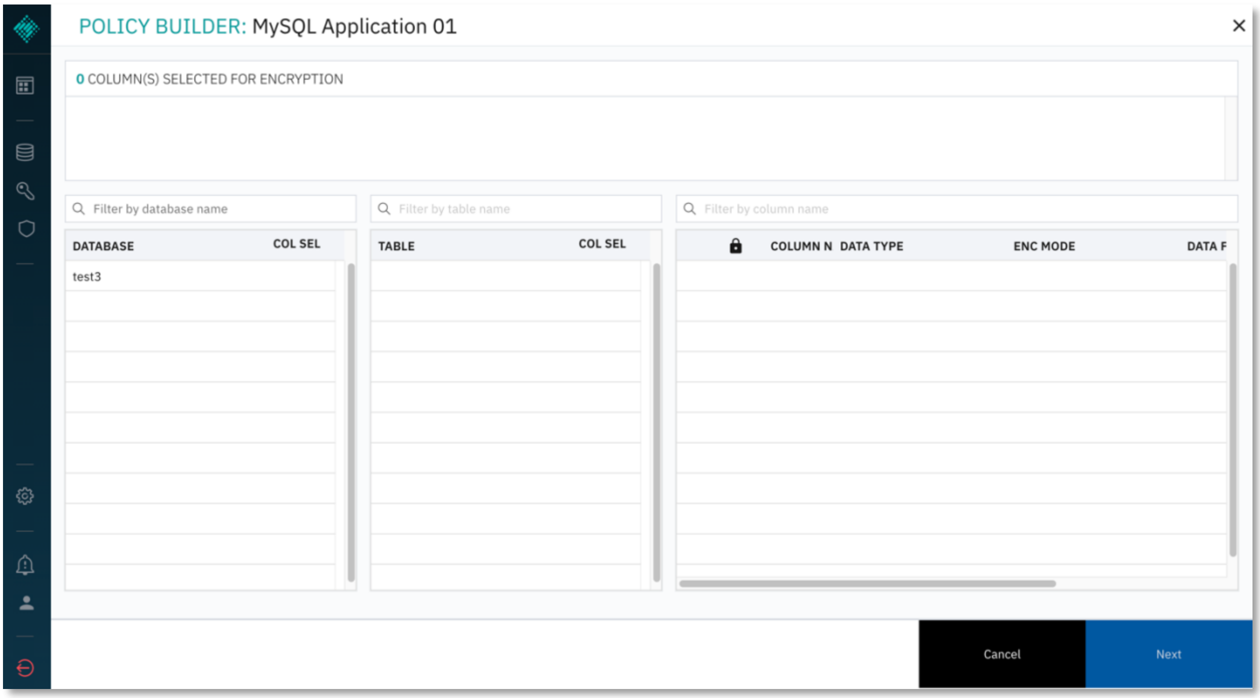
IP FILTERING

Permitted IP Addresses

Blocked IP Addresses

BAFFLE SHIELDS

5. **Select fields for encryption.** The Policy Builder will open for the configured data store.



POLICY BUILDER: MySQL Application 01

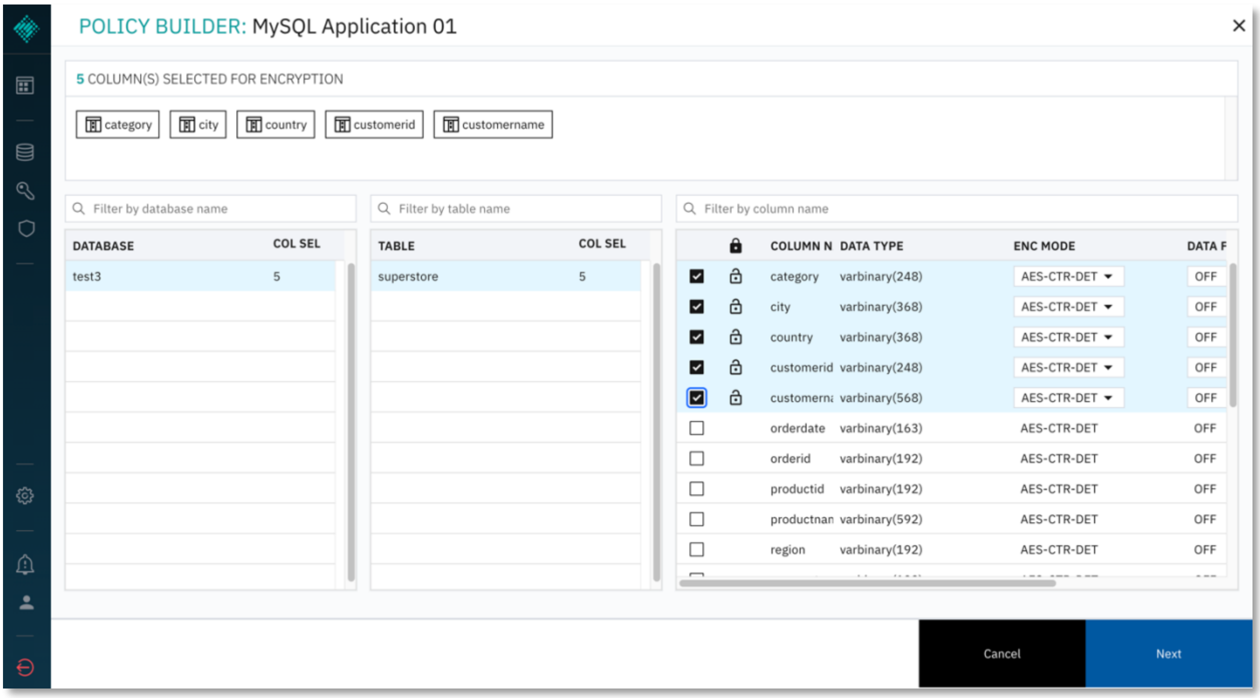
0 COLUMN(S) SELECTED FOR ENCRYPTION

| DATABASE | COL SEL |
|---|---|
| test3 | |

| TABLE | COL SEL |
|---|---|

| | 🔒 | COLUMN N | DATA TYPE | ENC MODE | DATA F |
|---|---|---|---|---|---|

6. **Select the database** and table you wish to encrypt..



POLICY BUILDER: MySQL Application 01

5 COLUMN(S) SELECTED FOR ENCRYPTION

category · city · country · customerid · customername

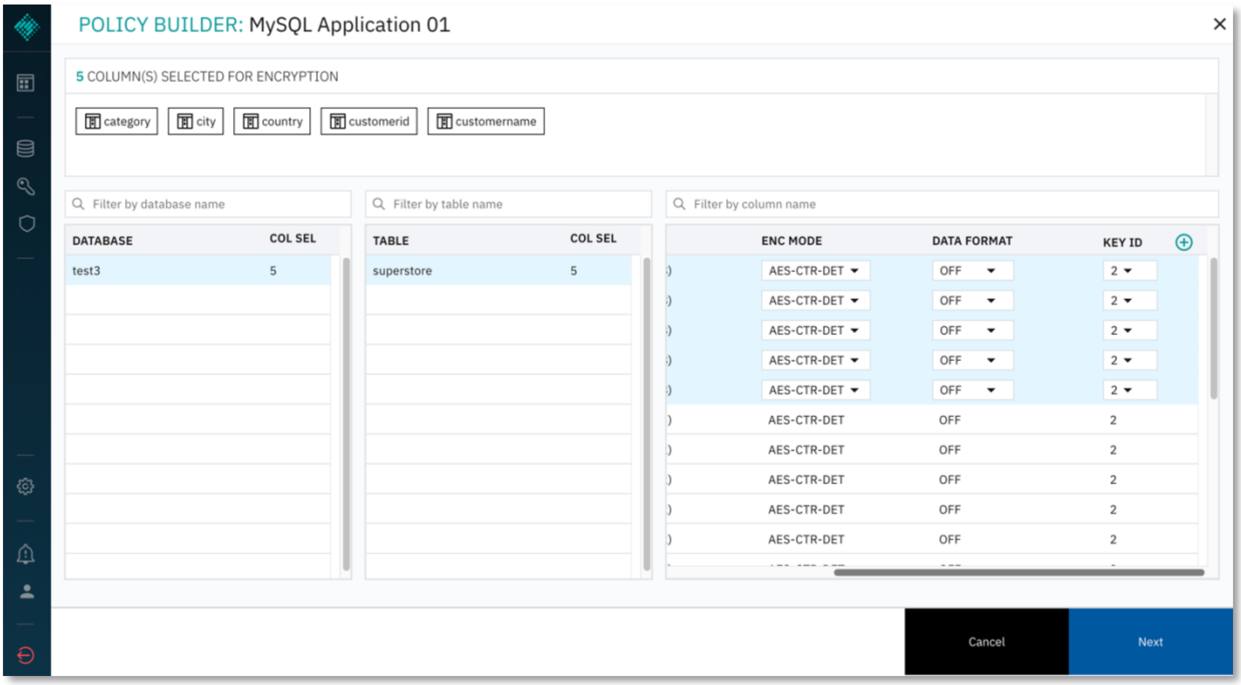
| DATABASE | COL SEL |
|---|---|
| test3 | 5 |

| TABLE | COL SEL |
|---|---|
| superstore | 5 |

| | 🔒 | COLUMN N | DATA TYPE | ENC MODE | DATA F |
|---|---|---|---|---|---|
| ☑ | 🔓 | category | varbinary(248) | AES-CTR-DET ▾ | OFF |
| ☑ | 🔓 | city | varbinary(368) | AES-CTR-DET ▾ | OFF |
| ☑ | 🔓 | country | varbinary(368) | AES-CTR-DET ▾ | OFF |
| ☑ | 🔓 | customerid | varbinary(248) | AES-CTR-DET ▾ | OFF |
| ☑ | 🔓 | customerna | varbinary(568) | AES-CTR-DET ▾ | OFF |
| □ | | orderdate | varbinary(163) | AES-CTR-DET | OFF |
| □ | | orderid | varbinary(192) | AES-CTR-DET | OFF |
| □ | | productid | varbinary(192) | AES-CTR-DET | OFF |
| □ | | productnan | varbinary(592) | AES-CTR-DET | OFF |
| □ | | region | varbinary(192) | AES-CTR-DET | OFF |

7. **Select columns** for encryption and the respective encryption mode.

8. Optional: Specify Key IDs for use to encrypt specific columns. Scroll to the right on the column selector and add more keys by clicking (+). The default value for Key ID is 2. Available Key IDs will be displayed in the Key ID dropdown menu for each column.



POLICY BUILDER: MySQL Application 01

5 COLUMN(S) SELECTED FOR ENCRYPTION

category | city | country | customerid | customername

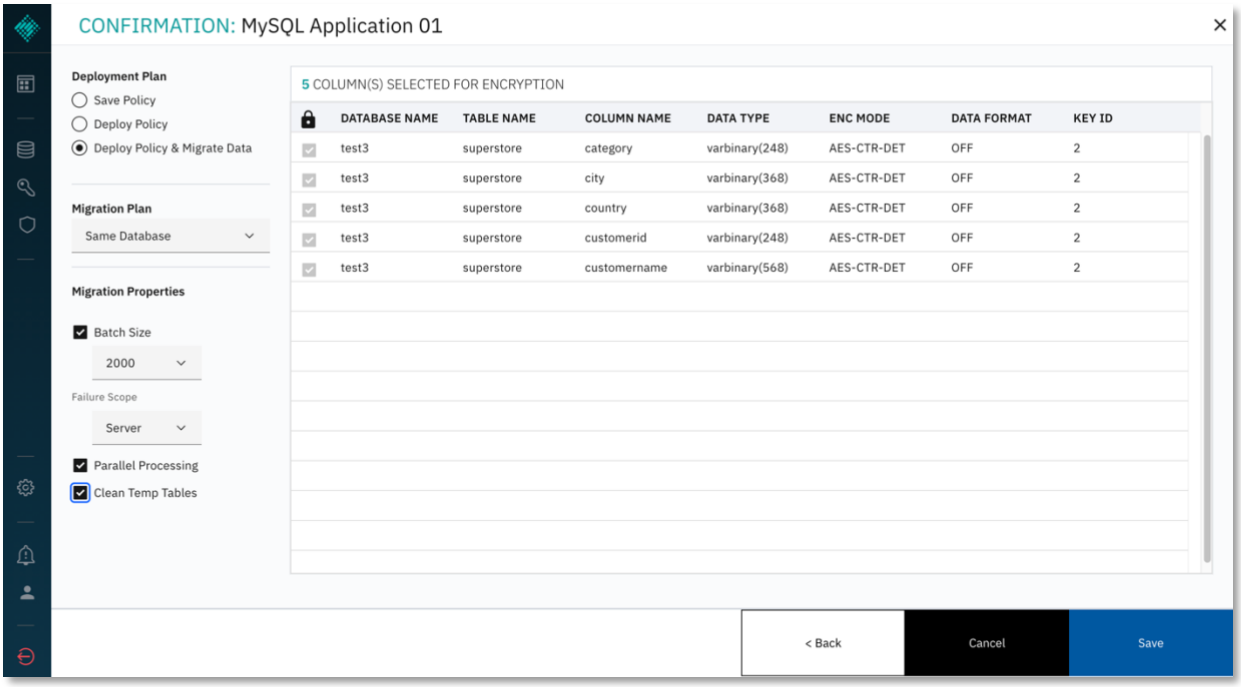
Filter by database name | Filter by table name | Filter by column name

| DATABASE | COL SEL |
|---|---|
| test3 | 5 |

| TABLE | COL SEL |
|---|---|
| superstore | 5 |

| | ENC MODE | DATA FORMAT | KEY ID |
|---|---|---|---|
| ) | AES-CTR-DET | OFF | 2 |
| ) | AES-CTR-DET | OFF | 2 |
| ) | AES-CTR-DET | OFF | 2 |
| ) | AES-CTR-DET | OFF | 2 |
| ) | AES-CTR-DET | OFF | 2 |
| ) | AES-CTR-DET | OFF | 2 |
| ) | AES-CTR-DET | OFF | 2 |
| ) | AES-CTR-DET | OFF | 2 |
| ) | AES-CTR-DET | OFF | 2 |
| ) | AES-CTR-DET | OFF | 2 |

9. Click NEXT to proceed. Under Deployment Plan, select **Deploy Policy & Migrate Data** to save and deploy the policy you have just configured, and to migrate the existing data in the columns you selected. Alternatively, you may simply save the policy to edit it later, or deploy the policy without migrating existing data.
    a. Select the option to **Clean Temp Tables** so the Baffle Shield deletes the temporary tables it will use to carry out encryption.
    b. Click SAVE to complete policy creation and execute the policy.



CONFIRMATION: MySQL Application 01

Deployment Plan

○ Save Policy

○ Deploy Policy

● Deploy Policy & Migrate Data

Migration Plan

Same Database

Migration Properties

☑ Batch Size

2000

Failure Scope

Server

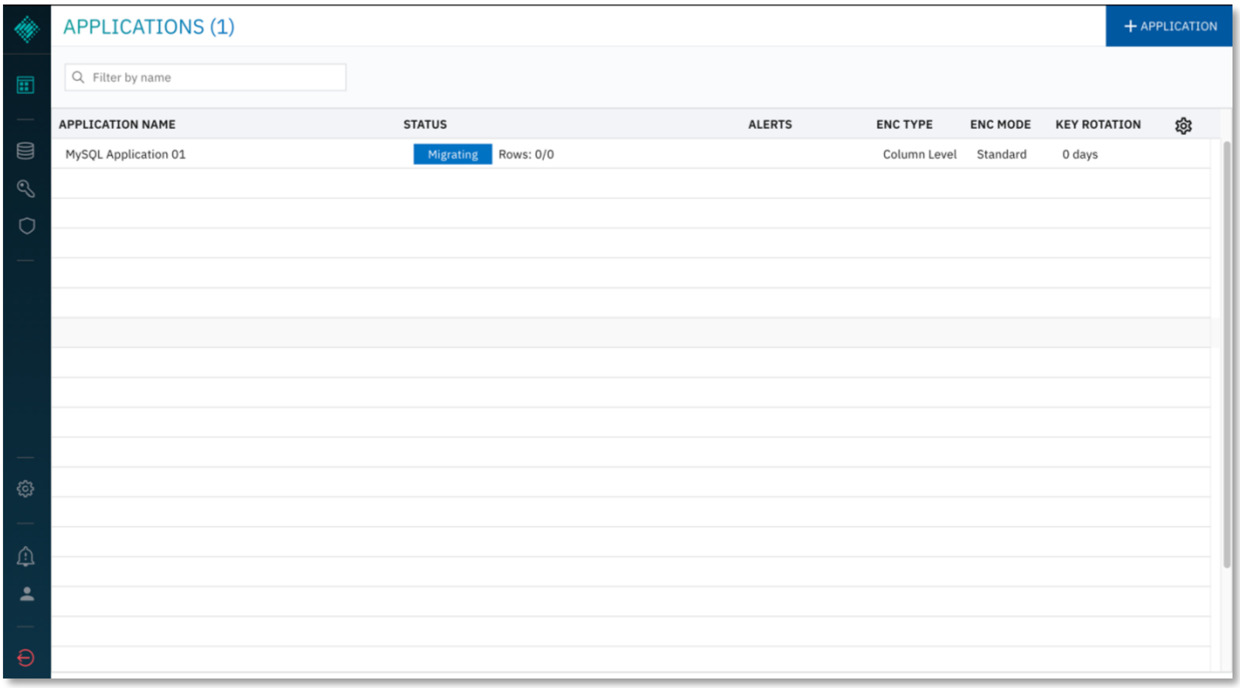
☑ Parallel Processing

☑ Clean Temp Tables

5 COLUMN(S) SELECTED FOR ENCRYPTION

| 🔒 | DATABASE NAME | TABLE NAME | COLUMN NAME | DATA TYPE | ENC MODE | DATA FORMAT | KEY ID |
|---|---|---|---|---|---|---|---|
| ☑ | test3 | superstore | category | varbinary(248) | AES-CTR-DET | OFF | 2 |
| ☑ | test3 | superstore | city | varbinary(368) | AES-CTR-DET | OFF | 2 |
| ☑ | test3 | superstore | country | varbinary(368) | AES-CTR-DET | OFF | 2 |
| ☑ | test3 | superstore | customerid | varbinary(248) | AES-CTR-DET | OFF | 2 |
| ☑ | test3 | superstore | customername | varbinary(568) | AES-CTR-DET | OFF | 2 |

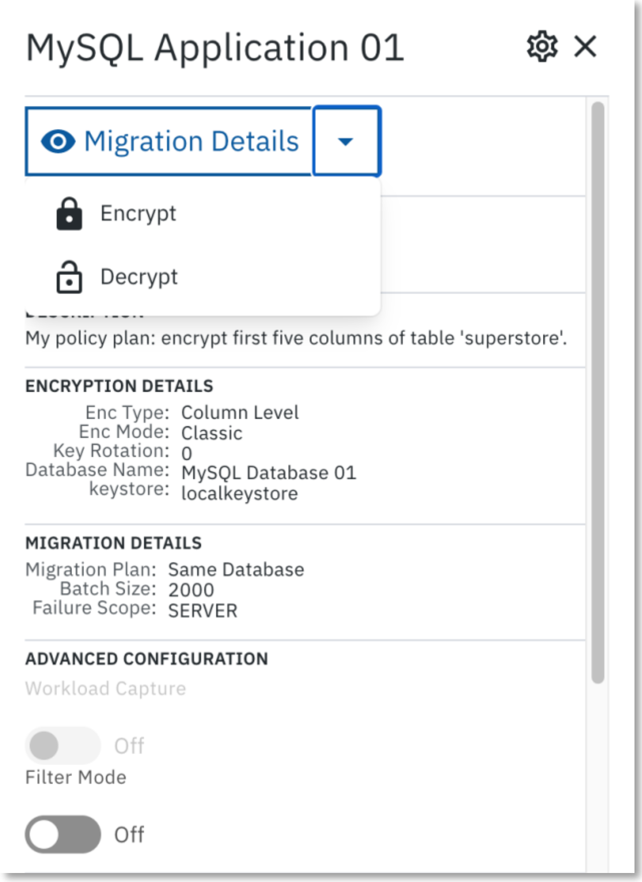
10. The Applications list now indicates the data migration is in progress. If migration does not initiate, you may have to configure your database user privileges. See **Appendix A** (page 26).



APPLICATIONS (1)

| APPLICATION NAME | STATUS | ALERTS | ENC TYPE | ENC MODE | KEY ROTATION |
|---|---|---|---|---|---|
| MySQL Application 01 | Migrating Rows: 0/0 | | Column Level | Standard | 0 days |

11. To Decrypt data, click on the application again, and select DECRYPT from the dropdown menu. The Policy Builder will re-open. Select the columns which you would like to decrypt and click NEXT to proceed. Only the columns that you have previously encrypted will be available to decrypt.



MySQL Application 01

My policy plan: encrypt first five columns of table 'superstore'.

ENCRYPTION DETAILS

Enc Type: Column Level
Enc Mode: Classic
Key Rotation: 0
Database Name: MySQL Database 01
keystore: localkeystore

MIGRATION DETAILS

Migration Plan: Same Database
Batch Size: 2000
Failure Scope: SERVER

ADVANCED CONFIGURATION

Workload Capture

Off

Filter Mode

Off

## Summary

You have now completed configuration of the Baffle Manager, Baffle Shield and created a Data Protection Policy to protect your data.

To confirm your data is encrypted, access the database normally with your SQL client. You should find the columns you selected are now encrypted.

To view the columns in the clear, use your SQL client to connect to the Baffle Shield. Connect using the public IP address of the Shield, port 8444, and the credentials for the database user you submitted in section 3, step 2b. Access the encrypted tables, and you should find the columns are visible.

# Appendix

## Appendix A: Database Privileges for encryption and migration

In order to carry out encryption and migration, Baffle Shield requires certain user permissions on the database. It is recommended that you create a new user on your database for Baffle Shield to use, rather than assign your database administrator.

Use your SQL client to issue the following grants. Enter the credentials of this new user in section 3, step 2b, so that Baffle Shield has full privileges to encrypt and decrypt the data you select.

1. To create a new user:

   a.
   ```
   create user '<baffle user>'@'%';
   ```

   b.
   ```
   set password for '<baffle user>' = password('<password>');
   ```

2. To grant the requisite permissions:

   a.
   ```
   GRANT USAGE ON *.* TO '<baffle user>'@'%';
   ```

   b.
   ```
   GRANT ALL PRIVILEGES ON shadow_information_schema.* TO '<baffle user>'@'%';
   ```

   c.
   ```
   GRANT ALL PRIVILEGES ON <target database>.* TO '<baffle user>'@'%' WITH GRANT OPTION;
   ```

   Repeat step c for each database you wish to encrypt. When completed, Baffle Shield has the necessary permissions in order to carry out encryption and migration. Use the credentials of the user specified here.

## Appendix B: Minimum Required Database Privileges

These are the minimum required grants for users on your database who need the least access privileges. Use your SQL client to issue the following commands with your admin user. These grants permit the restricted-access user to obtain only the data you specify.

For MySQL and Aurora databases:

1. Issue the following commands.

   a.
   ```
   GRANT USAGE ON *.* TO '<username>'@'%';
   ```

   b.
   ```
   GRANT ALL PRIVILEGES ON shadow_information_schema.* TO
   '<username>'@'%';
   ```

   c.
   ```
   GRANT SELECT ON <target database>.<target table> TO
   '<username>'@'%';
   ```

   Repeat step c for each table you wish to make accessible to the user. When completed, you may connect to the Baffle Shield proxy with this user.

   d. To confirm user privileges, use: `show grants;`

2. OPTIONAL: some databases may require additional information from the user. Take the hash of the user's password with the following:

   a.
   ```
   SELECT PASSWORD ('<user password>');
   ```

   Insert the hash back into the expressions:

   b.
   ```
   GRANT USAGE ON *.* TO '<username>'@'%' IDENTIFIED BY PASSWORD
   '<password hash>';
   ```

   c.
   ```
   GRANT ALL PRIVILEGES ON shadow_information_schema.* TO
   '<username>'@'%' IDENTIFIED BY PASSWORD '<password hash>';
   ```

   d.
   ```
   GRANT SELECT ON <target database>.<target table> TO
   '<username>'@'%' IDENTIFIED BY PASSWORD '<password hash>';
   ```

   As before, repeat step d for each table you have selected.