Dynamic Authorization Service™ for SQL Databases

Authorize digital interactions, at scale, extend identity security enterprise-wide

Identity-Aware Security for SQL Databases

SQL databases are used widely by enterprises to store everything from analytical to operational data. With the rapidly growing volume of data, there is increasing pressure on enterprises to ensure data is not exposed to unauthorized users.

Authorization has traditionally been built into individual, homegrown applications. However, this method is not scalable nor secure. Controls have become too separated and too siloed to ensure proper authorization and compliance. Rules that were built for original databases do not apply to end users of applications and services.

PlainID eliminates these challenges by externalizing authorization and centralizing its management via **The PlainID Dynamic Authorization Service™.** Enforcement is then distributed to the **PlainID Authorizer™ for SQL Databases.** The Authorizer sits at the data layer, inside the applications, or internal to your microservices layer and addresses data security at scale, and dynamically, for all users who access SQL databases.

When a user attempts to access data, the Authorizer intercepts data requests made to SQL databases and checks whether they meet the prescribed conditions (e.g. location, risk score, etc.). Stricter conditions can be set for sensitive data down to the row, column, and even cell-level.

Ultimately, PlainID strengthens data security for SQL databases by ensuring users are only able to see certain data based on who they are, and where they're accessing the data from.

Improve Data Security & Privacy in Real-time

Data is often fetched using a service account, with too many permissions. Applications will typically retrieve all data and mask information at the middleware or service mesh layer, which creates vulnerabilities for sensitive data.

The **PlainID Authorizer for SQL Databases** dynamically modifies the query at runtime to return only data the user is authorized to see – enforcing continuous, and contextual access to digital assets within databases such as PostgreSQL, Microsoft SQL Server, Amazon Redshift and more. This empowers enterprises with:

- Standardization of access controls to protect their most sensitive assets for both analytical and operational data
- Low-code/No-code development through the use of Spring Boot and .NET libraries provided by PlainID
- Enterprise-wide data access control in any layer of the technology stack through the use of REST APIs

Business Impact

Central Policy Management
Manage access to SQL database data objects with a business-driven UI to provide consistent access decisions across enterprise apps.

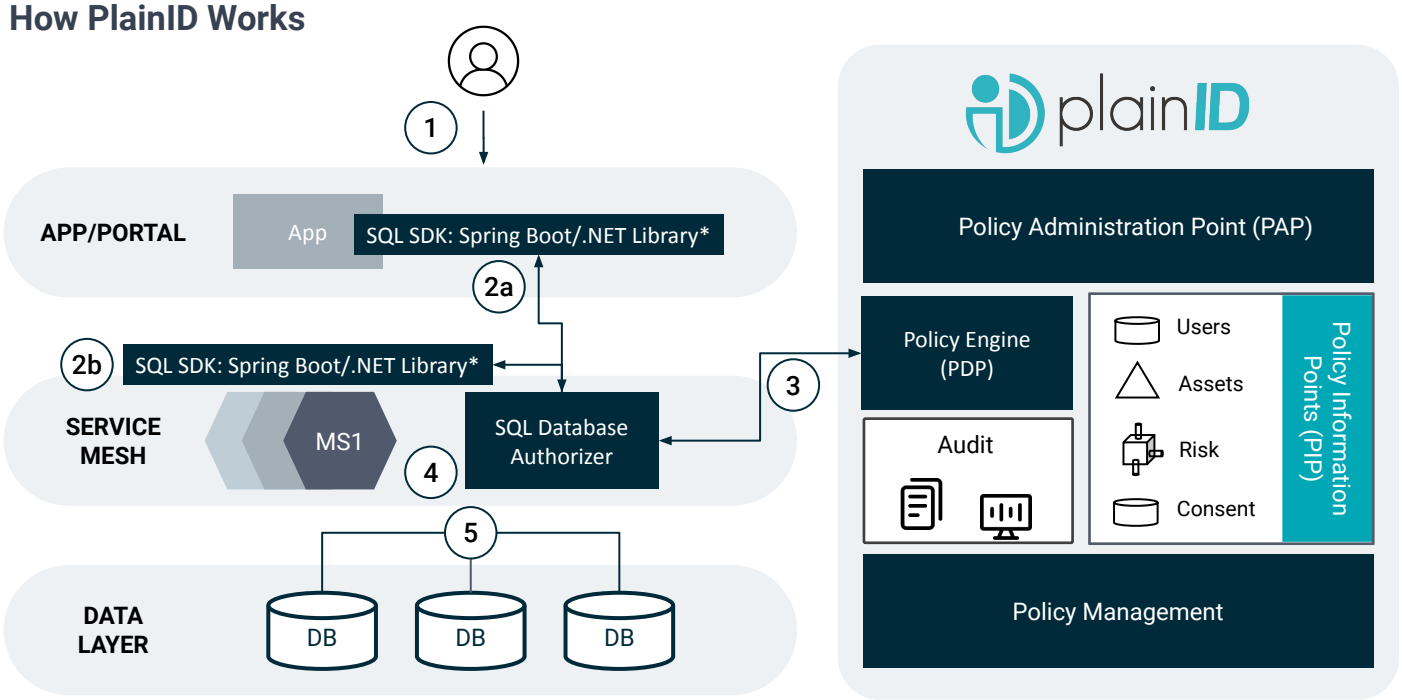
Coarse & Fine-grained Authorization
Enforce fine-grained controls at a granular level by determining what rows/columns/cell data are exposed to the authorized user.

Low to no-code Integration
Accelerate application development with PlainID's SQL SDK while ensuring robust access controls.

Consistently Control Data
Standardize data access controls across enterprise layers, from applications, to APIs and microservices, down to the data tier

Google BigQuery
Microsoft SQL Server
amazon REDSHIFT
PostgreSQL
MariaDB
MySQL
trino
Starburst

How PlainID Works



1. A User goes to the app that will allow the user to interact with digital assets.
2. At either the app-level, or in the microservice layer that fetches data from the database, the PlainID libraries initialize and call the PlainID SQL Database Authorizer's REST API. **Neither Spring Boot nor .NET libraries are required to use this service. These libraries help developers apply Data Access Control with minimal to no code.*
3. The SQL Database Authorizer looks at the original statement the microservice or app wanted to use (such as SELECT * FROM CLIENTS_TABLE) with the user context and reaches out to PlainID PDP REST API for an authorization decision.
4. The SQL Database Authorizer changes the original SQL statement, utilizing the identity-aware decision from the PDP, and returns a new SQL statement to the microservice or app layer such as SELECT FNAME, LNAME, ADDRESS from CLIENTS_TABLE where CLIENTS TABLE.REGION='US'
5. The SQL statement from the SQL Database Authorizer is executed and the appropriate data is fetched from the database.

Future-proof Your Enterprise

The **PlainID Integration Hub** is designed to address the complex challenges of enterprise access control. By offering out-of-the-box Authorizers™ and Integrations, it allows for a standardized approach across varied and distributed infrastructures – unifying disparate access controls under one platform.

PostgreSQL · amazon REDSHIFT · Google BigQuery · denodo · apigee · envoy · MariaDB · MySQL · Microsoft SQL Server · trino · Starburst · Kong · HashiCorp Consul · snowflake · Power BI · Kuma · Istio

Visit PlainID.com/integration-hub for more information

ABOUT PLAINID

PlainID is The Identity Security Company™. We help identity-centric enterprises defend themselves from adversaries who use identity-based attacks. Our Identity Security Posture Management Platform provides Identity Insights, SaaS Authorization Management, and Dynamic Authorization Services to create identity-centric security across SaaS, APIs, microservices, apps, and data powered by policy-based access control. Visit PlainID.com for more information.

plainID