

Denodo Data Virtualization Security Architecture & Protocols



Denodo Unified Security

Data virtualization offers a single logical point of access, avoiding point-to-point connections from consuming applications to the information sources. As a single point of access for applications, it is the ideal place to enforce access security restrictions that can be defined in terms of the canonical model with a very fine granularity (e.g., access to “Bill,” “Order,” and so on).

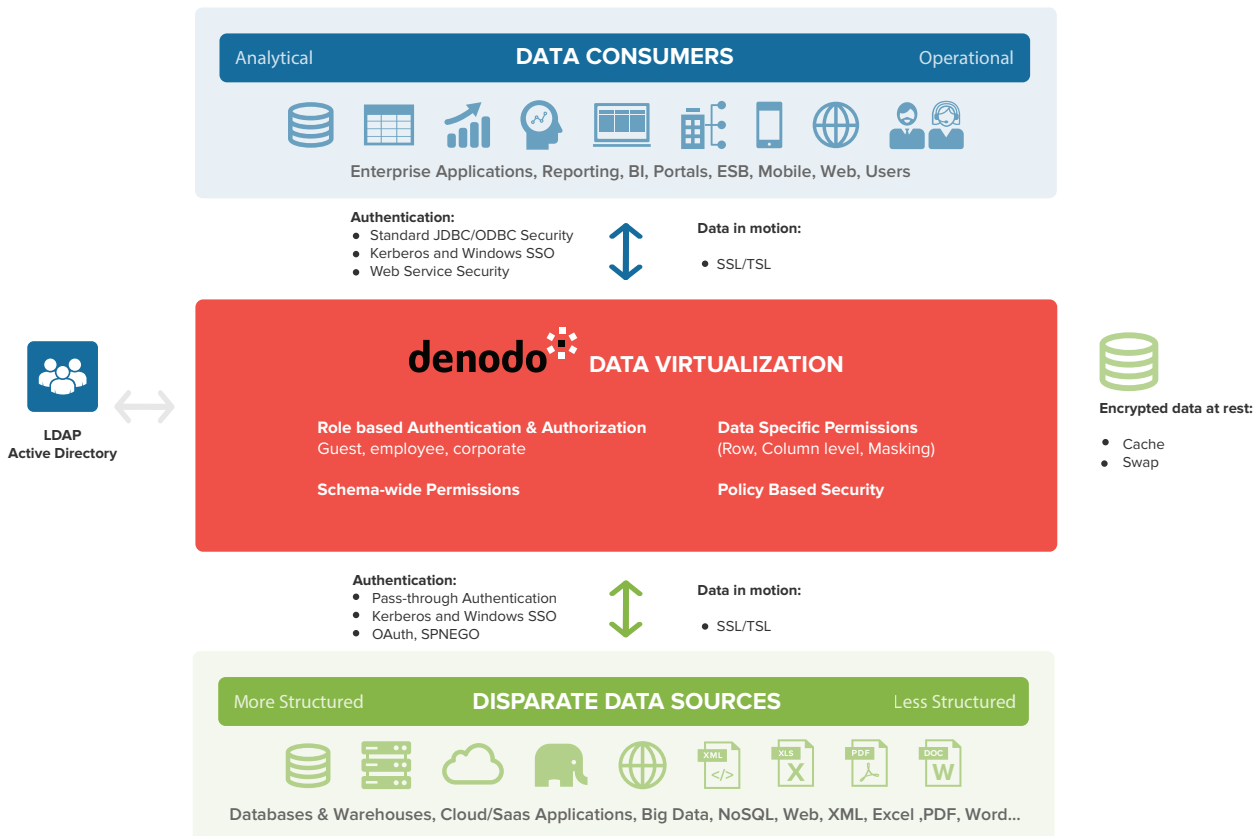


Fig. 1 - Denodo Platform Security

Denodo secures access from consumer applications to final data sources end-to-end. Typically this is established via SSL connections between the consumer and the Denodo Platform and by the specific data source security protocol between the Denodo Platform and the data sources (e.g., SSL, HTTPs, or sFTP).

The Denodo Platform includes its own user and role-based authentication and authorization mechanism with both schema-wide permissions (e.g., to access Denodo databases and views) and data-specific permissions (e.g., to access the specific rows or columns in a virtual view). Denodo offers very fine-grained access up to the cell level (applying both row-based and column-based security) including the possibility of masking specific cells (e.g., managers are not allowed to view the “salary” column of higher-level management, so those cells would appear masked in the results). Denodo row-based security does not require any coding, and it can be defined graphically with the Denodo Administration tool.

When there is an authentication mechanism in place, Denodo can delegate authentication to an external LDAP or Active Directory server, so there is no need to define users in the built-in user management system, and the LDAP/AD system would provide the role for the user, which would be used to constrain the user’s access to any database or view within the data virtualization server.

As a third alternative, it is also possible to connect to external custom entitlement services (through Custom Policies).

To grant access to a given data source through the data virtualization layer, the user can choose whether to make use of a service account for the source, in which case the Denodo Platform would always use the service account credentials to access the source, or to apply pass-through authentication and authorization in such a way that the security guardrails in the Denodo Platform are bypassed, and user credentials are directly used in the data source.

Security Architecture & Protocols

We include hereinafter a description of the security support in the Denodo Platform. The following diagram highlights the Denodo Platform Security Architecture.

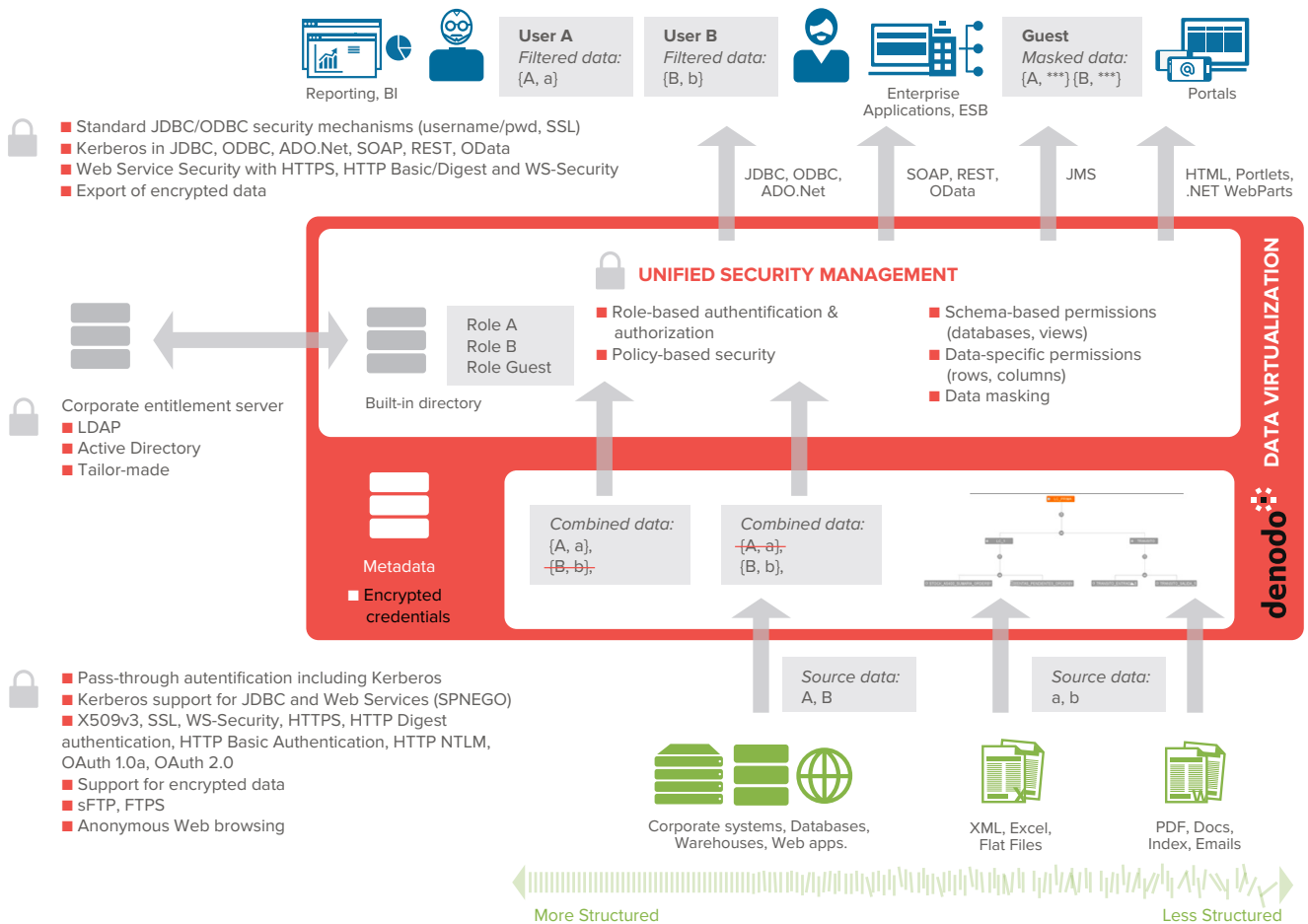


Fig. 2 - Denodo Platform Security Architecture and Protocols

The Denodo Platform Security Architecture has the following features:

- Unified Security Management offering a single point to control the access to any piece of information.
- Role based security access: the grey boxes depict a scenario where different users (User A, User B, and Guest) are only allowed to access either 'filtered' or 'masked' data by using the Denodo role-based security model. Denodo offers a low granularity level with regard to security that can be established in the following way:
 - Schema-wide permissions: each user/role can be assigned permissions (e.g. connect, read, write and write) to specific schemas (Denodo databases and views) so that different user profiles obtain different levels of control on the virtual schemas and the data they access.
 - Data-specific permissions: Denodo supports access permissions to specific rows (row-based security) or columns (column-based security) of virtual schema views, so that users/roles can be restricted from accessing specific pieces of data in the system. For example, non-managers may not be allowed to view the "salary" column on a virtual "Employee" view.

- Authentication credentials can be stored either internally in the Denodo Platform in a built-in repository or externally in a corporate entitlements server such as an LDAP / AD repository. Also it is possible to use 'custom policies' to connect to any other corporate security system (see "tailor-made" capability in the picture).
- Security protocols used to connect to data sources (southbound) or to publish data to consuming applications (northbound) are showed in the figure. In particular Denodo supports the following standards:
 - Northbound / Publish Interface:
 - » Standard JDBC/ODBC security mechanisms (username/password, SSL).
 - » Kerberos support for the JDBC, ODBC, ADO.Net, SOAP (SPNEGO), REST (SPNEGO), OData (SPNEGO) interfaces and for the VDP administration tool.
 - » Web Service security with HTTPS, HTTP Basic/Digest and WS-Security protocols.
 - » Export of encrypted data: Denodo Task Scheduler can be used to export an encrypted CSV or SQL file using 'Password-Based.Encryption with MD5 and DES' (PBE with MD5 and DES).
 - Southbound / Data Source Connection:
 - » Pass-through authentication including Kerberos.
 - » Kerberos support for JDBC, Web Services (SPNEGO).
 - » X509v3, SSL, WS-Security, HTTPS, HTTP Digest authentication, HTTP Basic Authentication, HTTP NTLM, OAuth 1.0a and 2.0.
 - » Support for encrypted data: Password-Based Encryption with MD5 and DES (PBE with MD5 and DES). In addition to the PBE with MD5 and DES algorithms Denodo provides support for custom decryption algorithms.
 - » sFTP, FTPS.
 - » Anonymous web browsing: through the use of an anonymizer server (i.e. anonymous proxy).
- Denodo offers three different alternatives to integrate with identity, authentication and authorization services:
 - Denodo built-in security.
 - Integration with external entitlement service (LDAP/AD).
 - Integration with external custom entitlement service with specific security policies (Custom Policies).
- The Denodo Platform provides an audit trail of all the information about the queries and other actions executed on the system. Denodo will generate an event for each executed sentence that causes any change in the Denodo Catalog (store of all the server metadata). With this information it is possible to check who has accessed specific resources, what changes have been made or what queries have been executed. All Denodo components include configurable multi-level logs based on the Log4J standard, and they can be configured to log specific information that can be later used for compliance and auditing purposes.

User & Role Management

Denodo's fine-grained, role-based access control includes row- and column-level permissions, full integration with LDAP/Active Directory for sourcing user identities and group-based authorizations to virtual views. Denodo Roles, defined in a Denodo Virtual Database, aggregate permissions on individual users (defined externally in LDAP/AD or natively as Denodo virtual database) for accessing virtual database schemas (data sources, views, web services, stored procedures), etc.

Denodo Virtual DataPort distinguishes two types of users:

- 'Administrators' can create, modify and delete databases without any limitation. Likewise, they can also create, modify and delete users. When the server is installed, a default administrator user is created with user name admin and password admin. There must always be at least one administrator user.
- 'Normal users' cannot create, modify or delete users or databases. Administrators can grant them connection, read, create or write privileges to one or several databases or to specific views contained in them. An administrator can promote a normal user to 'local administrator' or of one or more databases, which means that this user will be able to perform administration tasks over these databases.

A 'Normal user' can have 'Roles', sets of access rights over virtual databases and their virtual views and stored procedures. Roles allow administrators to manage user privileges easily because by changing the privileges assigned to a role, they change the privileges of all the users assigned that role.

Virtual DataPort access rights are applied to a specific user or a role, to delimit the tasks they can perform over databases, views and stored procedures. Access rights can be applied globally to a database or specifically to a view/stored procedure in a specific database. Virtual DataPort supports the following types of global database access rights: Read, Create, Write, and Connection Access. Virtual DataPort also supports individual privileges to specific views and stored procedures. The privileges that can be applied to a specific view and/or stored procedure of a database are: Read, Write, Insert, Update, and Delete. Within "Read privileges" on a view, a user (or a role granting same), can query this view to obtain all its data. However, if certain users or roles should have access to only some of the columns of a specific view, you can use 'Column privileges' and 'Row privileges' to further constrain "Read" access.

Hierarchical Roles

Roles can be "hierarchical". Once a baseline role is established, another role can be created which "inherits" and refines it. These "role hierarchies" can be built to any depth within Denodo.

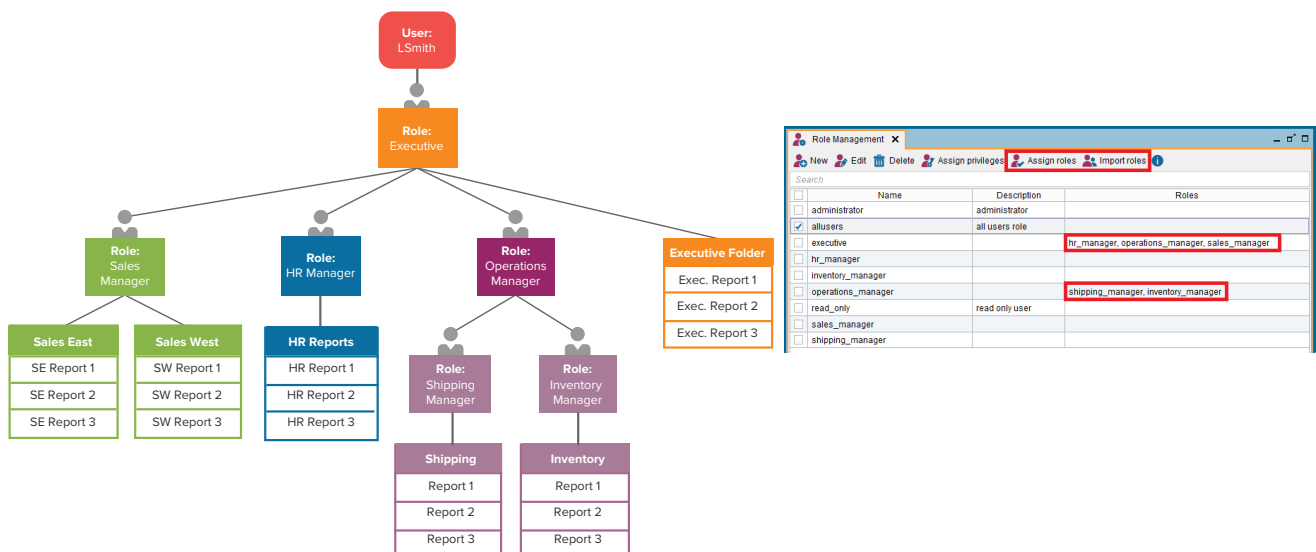
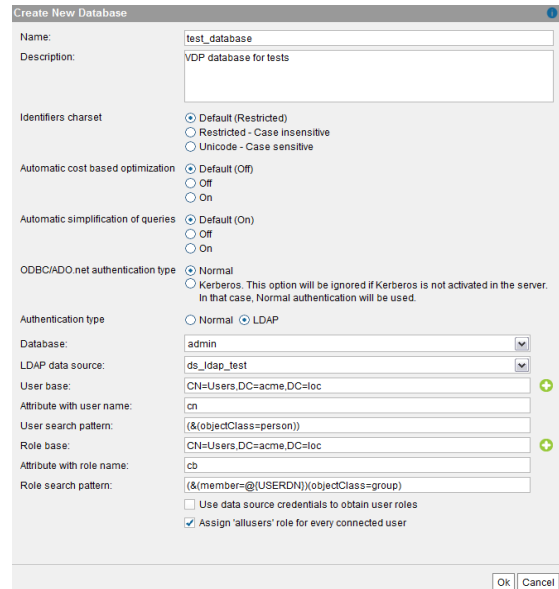


Fig. 3 - Creating role hierarchies

Authentication Through Ldap/Ad

Denodo can delegate the authentication to the designated LDAP/Active Directory system. Roles can be imported from either third-party source at any time, and then configured to constrain access to Denodo's virtual databases and views. A database with LDAP authentication delegates the authentication of users to an LDAP server. The benefit over the 'Normal' authentication is that you can rely on an LDAP server such as the Microsoft Windows Active Directory, to authenticate users without having to create them in Virtual DataPort. The Server also obtains the names of the roles that the users belong to, from the LDAP server and uses them to check which actions the users can do. When a user tries to connect to a LDAP database, the Server checks first if the user is a Virtual DataPort "administrator". If not, it connects to an LDAP server to check the credentials and obtain the roles of the user.



The screenshot shows the 'Create New Database' dialog box with the following configuration:

- Name: test_database
- Description: VDP database for tests
- Identifiers charset: Default (Restricted), Restricted - Case insensitive, Unicode - Case sensitive
- Automatic cost based optimization: Default (Off), Off, On
- Automatic simplification of queries: Default (On), Off, On
- ODBC/ADO.net authentication type: Normal, Kerberos. This option will be ignored if Kerberos is not activated in the server. In that case, Normal authentication will be used.
- Authentication type: Normal, LDAP
- Database: admin
- LDAP data source: ds_ldap_test
- User base: CN=Users,DC=acme,DC=loc
- Attribute with user name: cn
- User search pattern: (&(objectClass=person))
- Role base: CN=Users,DC=acme,DC=loc
- Attribute with role name: cb
- Role search pattern: (&(member=@(USERDN))(objectClass=group))
- Use data source credentials to obtain user roles
- Assign 'allusers' role for every connected user

Fig. 4 - Configuring a virtual database with LDAP authentication

Row And Column Level Security And Data Masking

Denodo can enforce strict, fine-grained user and role-based permissions for each and every element defined within it. Denodo's authorization policies can implement row and column security. These policies are specified by view, by row (specified with a selection condition), by column or by row-column combination (i.e. rows restricted but only for restricted columns), and are evaluated prior to each query execution to determine if the customer is allowed to see particular results or not. For example, Denodo's row-level security would allow hiding the salary of people with position='manager' when querying a 'salary' view from users with an 'employee' role (thus not entitled to access salary information).

Single Sign-On

Denodo supports delegating authentication / authorization to LDAP, Active Directory. It also supports pass-through of user credentials to data sources for principal propagation, therefore allowing to leverage existing authentication infrastructures. Also it supports sources with OAuth authorization. Denodo also include support for SSO of client applications using Kerberos.

Cache

When accessing cached data, the same security restrictions of the user/role on a given database, view, columns and/or rows are taken into account.

Policy Based Security

Custom policies allow developers to provide their own access control policies. Similar concepts already exist in some systems such as Oracle (Virtual Private Database Policies). Developers can code their own custom access control policies and the administrator can assign them to one (or several) users/roles in a view in Denodo. When a given user/role executes a query on that view, Denodo invokes the code of the policy. The policy may return:

- An indication that the query is not allowed.
- Nothing, indicating that the query is allowed as is.

- A filtering condition (indicating that the query is allowed but results must be filtered according to the returned condition).

The implementation of the policy has available the context of the query (user, projected fields, VQL statement, ...). Custom policies can be used to integrate an external Policy Server (e.g. Axiomatics) to provide dynamic authorization based on policies defined in such server.

Data Consuming Users, Apps

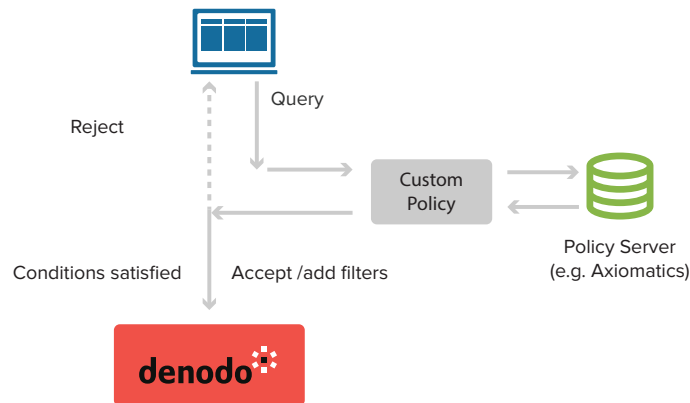


Fig. 5 - Policy-based security

Examples of possible uses: set limits to the number of queries executed by a certain user/role; determine if a query can be executed depending on the time of the day or leveraging the access policies in an external policy server.

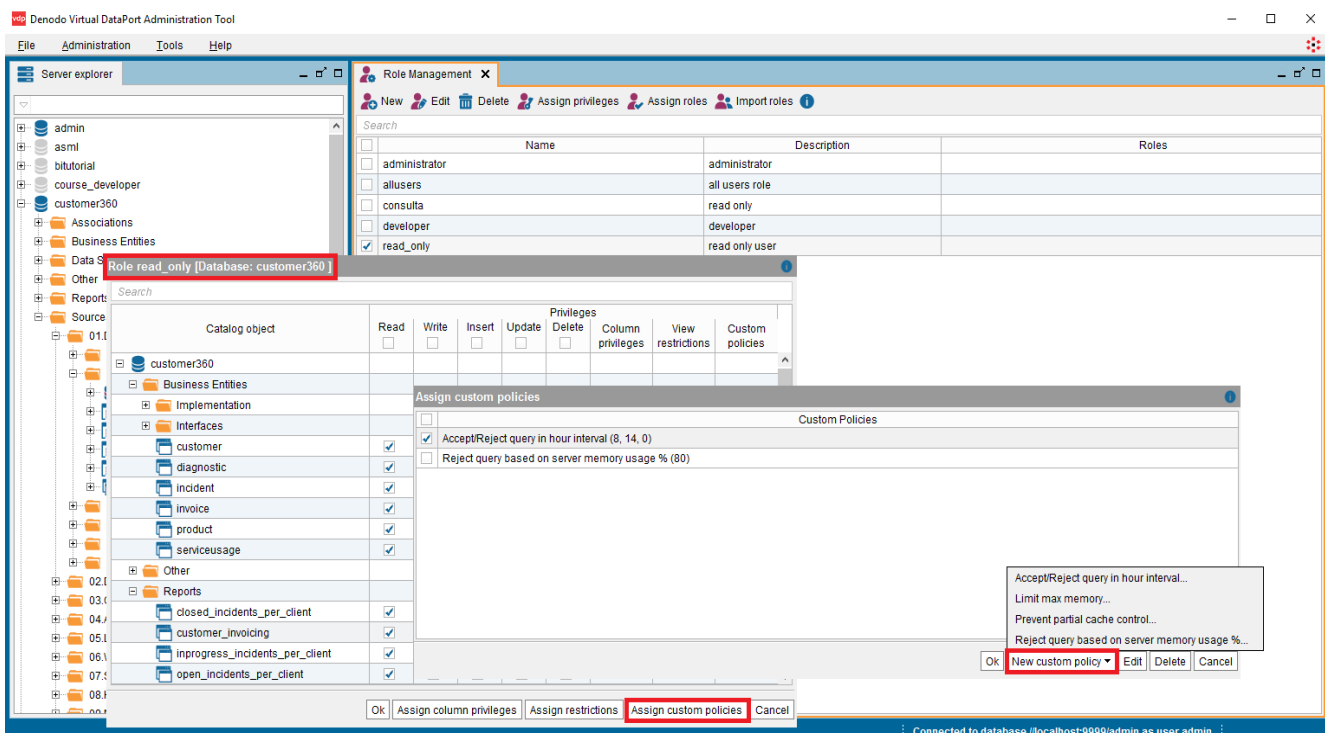


Fig. 6 - Dynamic Authorization based on policies

Encryption

Denodo's hybrid approach to data integration, allows different data access & delivery modes, all of which may involve securely accessing sensitive data: real-time from the data sources; from the Denodo cache; or from a staging area (i.e. ETL-like process where data is moved from its original data source to an external repository).

In order to cover all possible scenarios, Denodo supports the application of strategies on a per view basis to guarantee secured access to sensitive data through encryption/decryption at different levels.

Data at rest (secured caching of sensitive data or storage in staging area)

- When working in cached mode, Denodo will transparently leverage any encryption mechanism available in the selected Cache System. For example, Oracle Transparent Data Encryption (TDE) allows sensitive data to be encrypted within the data files to prevent access to it from the operating system.
- Denodo can access encrypted data files with the algorithm 'Password-Based Encryption with MD5 and DES' (PBE with MD5 and DES). This encryption method is described in the Java Cryptography Architecture Reference Guide (<http://download.oracle.com/javase/6/docs/technotes/guides/security/crypto/CryptoSpec.html>). In addition, the Denodo Platform can be extended with others standard/customer tailored encryption algorithms.
- The Denodo Scheduler can also export encrypted CVS or SQL files using the same algorithm.
- When security at the dataset level is not required, it's possible to selectively apply encryption/decryption only to sensitive fields using Denodo's built-in functions. These functions support any encryption algorithm supported by the default JCE of the Denodo Platform JRE, or by any additional provider registered as part of the Denodo Platform JRE.

Data in motion (securely accessing and delivering data)

- All communication between the Denodo Platform and the Data Consumers/Data Sources, as well as between the different modules within the Denodo Platform, can be secured through SSL at the connection level.
- If security at the connection level is not required, Denodo's built-in functions for encryption/decryption can be selectively applied to sensitive fields prevent unauthorized accesses.

About Denodo

Denodo is the leader in data virtualization providing unmatched performance, unified data access and agile data services at lower cost than traditional data integration. Denodo's customers gain business agility through a unified virtual data layer for enterprise-wide agile BI, big data analytics, cloud integration, single-view applications, and SOA data services.

Visit www.denodo.com Email info@denodo.com twitter.com/denodo

NA & APAC (+1) 877 556 2531 | EMEA (+44) (0) 20 7869 8053 | DACH (+49) (0) 89 203 006 441 | Iberia & Latin America (+34) 912 77 58 55