

#FABCONSQLCON2026

FABCON
Microsoft Fabric
COMMUNITY CONFERENCE

SQLCON
Microsoft SQL
COMMUNITY CONFERENCE

ATLANTA MARCH 16 - 20, 2026



Guardians of the lake: securing your data with OneLake

Cole Haddock
Solutions Architect, Vanguard, USA

Meet the speaker



Cole Haddock, Solution Architect, Vanguard

#Traveler, #DynamicsNerd, #ProblemSolver #DiveMaster,
#BloggerInTrainingWheels #FabricNerd



Show of hands

- Who is currently using Dataverse or Dynamics 365 and is familiar with the security model?
- Who is using Microsoft Fabric today?
- Who is using Fabric Link with Dataverse or Dynamics 365?
- Who is using OneLake security?

Reflection of the past

- Business unit
- Enhanced business units
- Security Roles
 - Org
 - Parent Child BU
 - BU
 - User
- Field Security Profiles

Table	Create	Read	Write	Delete	Append	Append To	Assign	Share
Account	🟡	🟡	🟡	🟡	🟡	🟡	🟡	🟡
ACIViewMapper	🟢	🟢	🟢	🟢				
Action Card	🟡	🟡	🟡	🔴	🟡	🟢	🔴	
Action Card User Settings	🟡	🟡	🟡	🟡				🟡
Activity	🟡	🟡	🟡	🟡	🟡	🟡	🟡	🟡
Advanced Similarity Rule	🔴	🔴	🔴	🔴	🔴	🔴	🔴	
Announcement	🔴	🔴	🔴	🔴		🔴		
Application File	🟢	🟢	🟢	🟢				
Azure Service Connection	🟢	🟢	🟢	🔴	🟢	🟢		

← Back Copy security role Rename security role Save Configure column view

ⓘ The Basic User role privileges cannot be adjusted due to its non-customizable nature. [Learn More](#)

Security Role: Basic User

Search by table name or table privilege

Details

Tables Miscellaneous privileges Privacy-related privileges

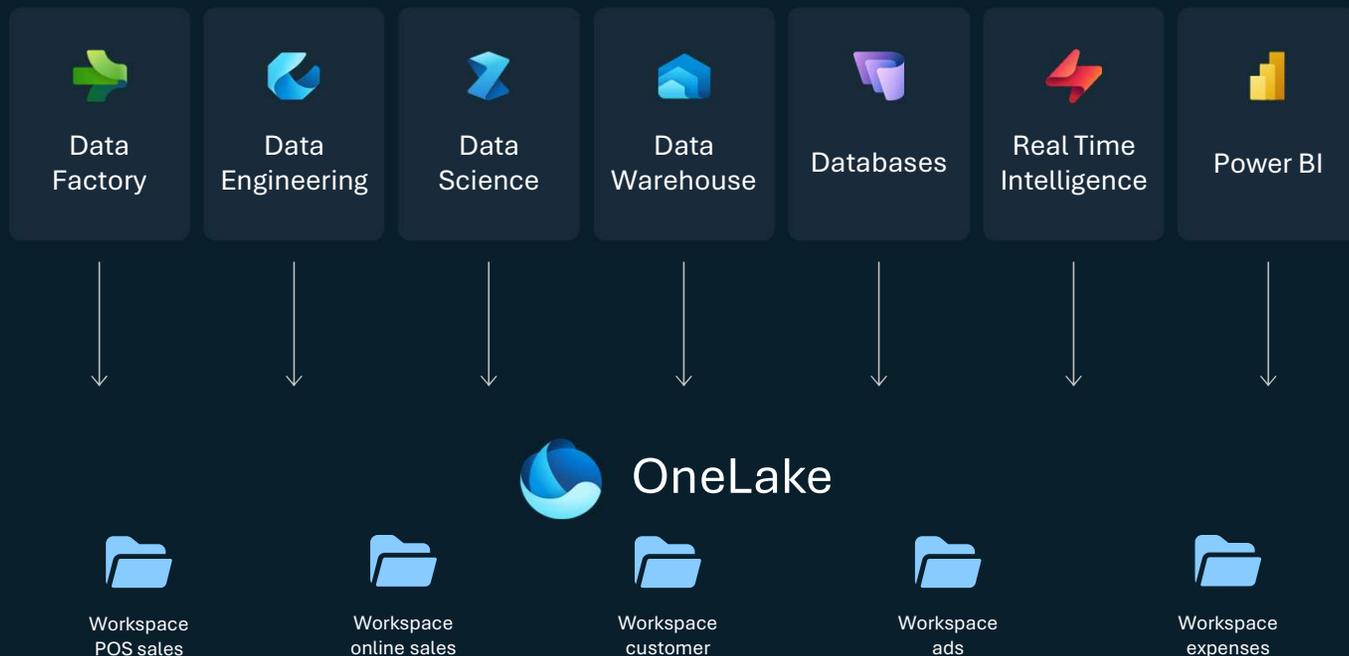
Show only assigned tables

Compact Grid View On

Table	Name	Record owner...	Permission Set...	Create	Read	Write	Delete	Append	Append to	Assign
Business Unit	businessunit	Business Unit	Reference	None	Organization	None	None	None	None	None
Channel Property Group	channelpropertygroup	Organization	Custom	None	Organization	None	None	Organization	Organization	
Currency	transactioncurrency	Organization	Custom	None	Organization	None	None	Organization	Organization	
Document Template	documenttemplate	Organization	Reference	None	Organization	None	None	None	None	None
Email Server Profile	emailserverprofile	User or Team	Custom	None	Organization	None	None	None	Organization	None
Goal	goal	User or Team	Custom	None	User	None	None	None	None	None

A single unified SaaS data lake

“No Silos”



Unified Security and Governance

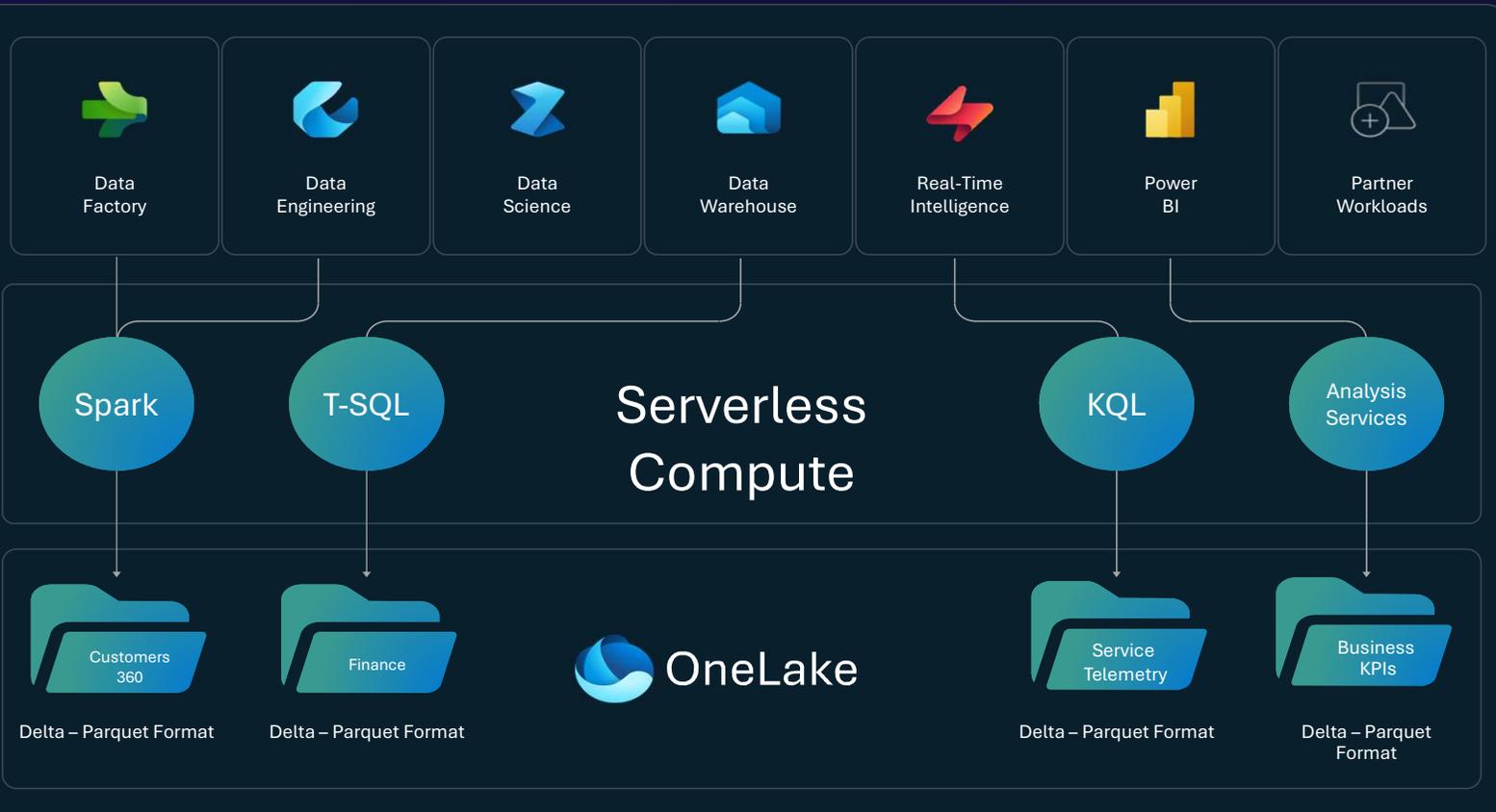
Provisioned automatically with the tenant.

Any data in OneLake works with out-of-the-box governance such as data lineage, data protection, certification, catalog integration, etc. All data is ultimately under the control of a tenant admin.

OneLake enables distributed ownership. Different workspaces allow different parts of the organization to work independently while still contributing to the same data lake. Each workspace can have its own administrator, access control, region and capacity for billing.

One copy for all computes

Real separation of compute and storage



All the compute engines store their data automatically in OneLake as data items.

The data is stored in a single common format.

Delta, an open standards format, and it is the storage format for all tabular data in Fabric.

All the compute engines have been fully optimized to work with Delta Parquet as their native format.

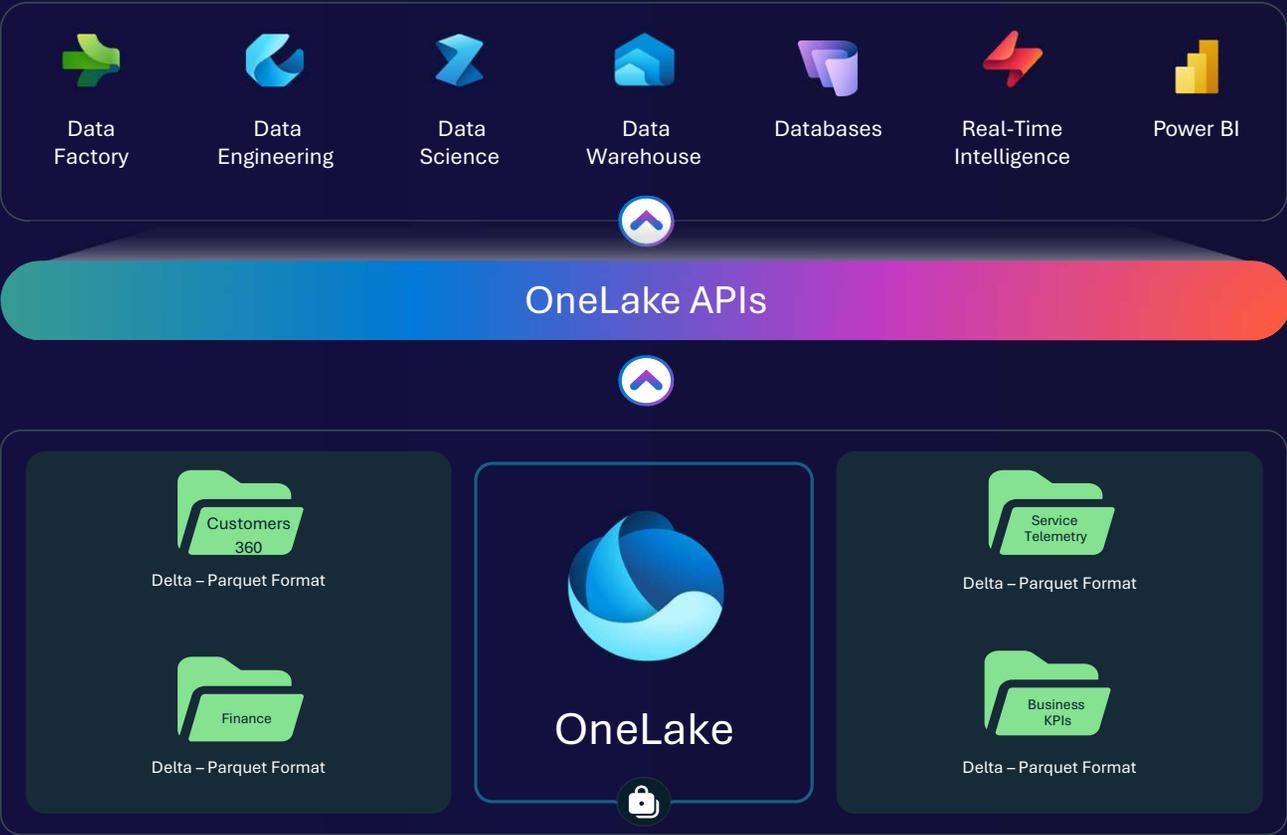
Current Workspace Security

- Admin – Full control, including adding/removing members and deleting items
- Member – Can reshare items and add Contributors or Viewers
- Contributor – Builds reports and dashboards
- Viewer – Read-only access; can run T-SQL queries

How do we provide better security to Lakehouse

OneLake Security

Defined once, enforced everywhere



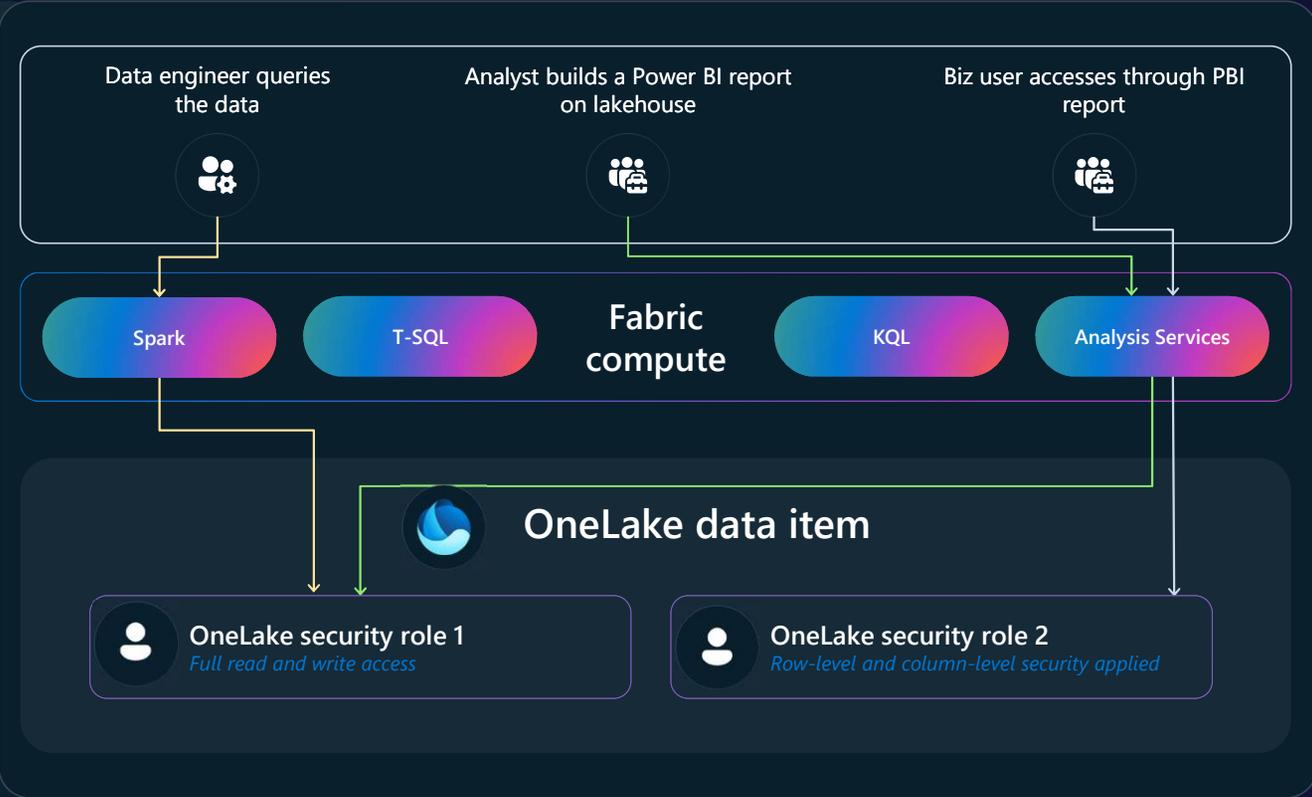
-  Define security roles in OneLake using powerful features like table, row, or column level security.
-  Data is secured consistently across experiences inside and outside of Fabric.
-  OneLake security roles can be managed in a single place, providing end to end coverage for data access for your entire data estate.

OneLake Security

03 Users access data through different Fabric engines

02 The permissions are checked when specific users are accessing data from the data item

01 Data owner sets up roles with defined permissions and assigns user groups to each role



OneLake Security Principles

Defined once, enforced everywhere

Security lives with the data

Enforce permissions safely

OneLake Security Details

FABCON | SQLCON

ATLANTA26

JOIN THE
CONVERSATION

#FABCONSQLCON26

Concept and overview

- Roles
 - The core concept of permissions management in OneLake. Roles represent a set of permissions, applied to a scope, for a given set of members.
- Permissions
 - A set of actions that users are allowed to taken on a specified scope. Permissions are things like Read, ReadWrite, etc.
- Scope
 - A scope is the set of items you are granting permissions to. It can be a table, folders, or an artifact. It can also have constraints that restrict access to specific rows or columns.
- Members
 - Members are the users or identities that are accessing data. Members are given the permissions of a role by being assigned to that role. Members can be groups or non-user identities such as service principals.

OneLake Security types

OneLake security supports three primary types of security.

- **Folder**
 - The base security type in OneLake, define permissions for a Folder in OneLake. Permissions inherit to subfolders for easy management.
 - Folders in the Tables path of a Fabric artifact contain tabular data and can be assigned additional permissions.
- **Row**
 - Limit access to specific rows of data in a table. Users specify SQL predicates that are used to restrict access.
- **Column**
 - Restrict access to data in certain columns using column level security. Define specific permissions on individual columns to hide sensitive data. Data masking is also an option as part of column level security.

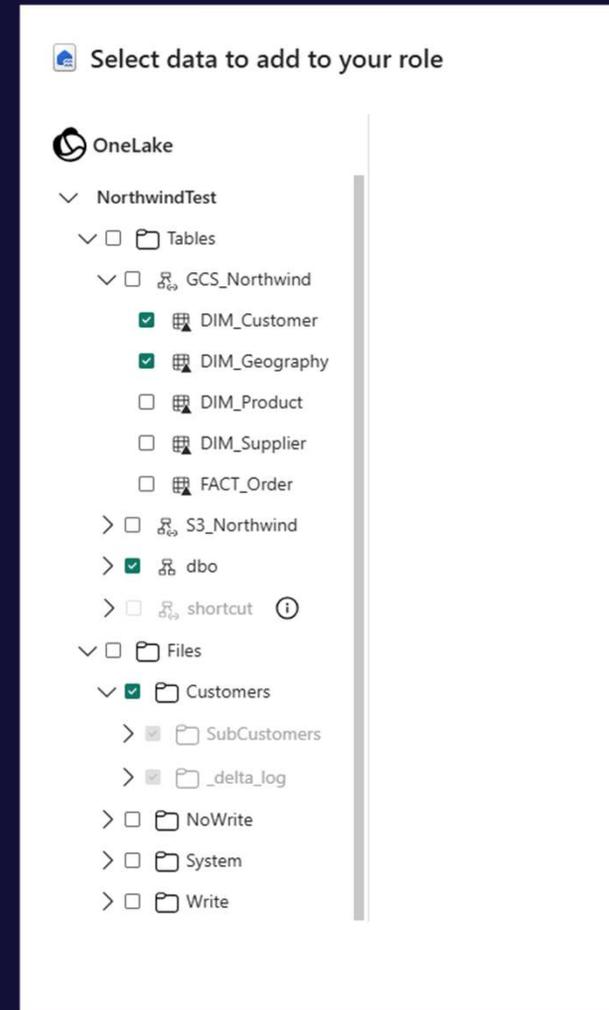
OneLake – object level security

Folder level security allows you to set permissions on a folder or folders in OneLake.

Security is based on Windows and SharePoint access models, as opposed to POSIX style ACLs.

Permissions inherit automatically to any sub folders.

Tables are a subset of folder security that can also have more granular permission types applied to them.



OneLake – column level security

Column level security allows for limiting access to specific columns in a table.

Access to columns can be left to default or columns can be removed, preventing users from seeing the column.

Hidden columns will be removed automatically when querying in Spark and Power BI. SQL requires excluding the columns from a query.

Data masking is not available yet.

All Lakehouse roles > Survived > Row and column security (Preview)

Manage table level access for

Manage access to your data by adding row and column level security to a table in a role. Members of this role will only be able to see specific data based on the rules you set below. [Learn more](#)

Row security (Preview) Column security (Preview)

Column level security Enabled

Column	Visibility
<input type="text" value="Id"/>	<input type="text" value="Read"/>
<input type="text" value="SinkCreatedOn"/>	<input type="text" value="Read"/>
<input type="text" value="SinkModifiedOn"/>	<input type="text" value="Read"/>
<input type="text" value="statecode"/>	<input type="text" value="Read"/>
<input type="text" value="statuscode"/>	<input type="text" value="Read"/>
<input type="text" value="createdby"/>	<input type="text" value="Read"/>
<input type="text" value="createdby_entitytype"/>	<input type="text" value="Read"/>
<input type="text" value="createdonbehalfby"/>	<input type="text" value="Read"/>
<input type="text" value="createdonbehalfby_entitytype"/>	<input type="text" value="Read"/>
<input type="text" value="modifiedby"/>	<input type="text" value="Read"/>

OneLake – row level security

Row level security allows for defining security predicates to govern access to select rows within a table.

Predicates use T-SQL syntax for definition and allow for multiple conditions to be set.

RLS can be dynamic per user or lookup data across multiple tables. (coming soon)

The screenshot shows the 'Row and column security (Preview)' configuration page for the 'Perished' role. The breadcrumb path is 'All Lakehouse roles > Perished > Row and column security (Preview)'. The main heading is 'Manage table level access for' followed by a dropdown menu showing 'contact'. Below this, there is explanatory text: 'Manage access to your data by adding row and column level security to a table in a role. Members of this role will only be able to see specific data based on the rules you set below. [Learn more](#)'. There are two tabs: 'Row security (Preview)' (which is selected and underlined) and 'Column security (Preview)'. A 'Save' button is located below the tabs. At the bottom, there is a text area with the instruction 'Show data to members of your role if the following rules apply:' and a single SQL rule: '1 SELECT * FROM contact WHERE ws1_outcome = '0''.

All Lakehouse roles > Perished > Row and column security (Preview) ✕

Manage table level access for

Manage access to your data by adding row and column level security to a table in a role. Members of this role will only be able to see specific data based on the rules you set below. [Learn more](#)

Row security (Preview) Column security (Preview)

Show data to members of your role if the following rules apply:

```
1 SELECT * FROM contact WHERE ws1_outcome = '0'
```

Shortcut User vs Delegation

OneLake security – User Identity (Passthrough)

Shortcuts are either passthrough or delegated

User Identity (Passthrough)



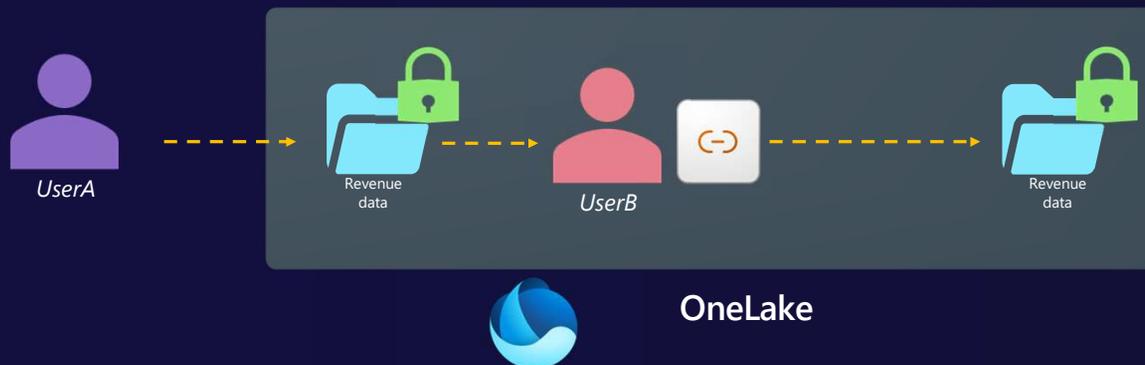
Shortcuts in OneLake that are passthrough:

- OneLake shortcuts

OneLake security - Delegated

Shortcuts are either passthrough or delegated

Delegated



Shortcuts in OneLake that are delegated:

- AWS S3
- GCS
- ADLS Gen2
- S3 Compatible

Demo

FABCON | SQLCON

ATLANTA26

JOIN THE
CONVERSATION

#FABCONSQLCON26

OneLake security – API

Interacting with OneLake security programmatically

- **GET/PUT dataAccessRoles**
- GET/PUT
<https://api.fabric.microsoft.com/v1/workspaces/{workspaceId}/items/{itemId}/dataAccessRoles>
- This is a bulk API. The existing role payload in the item gets replaced by the new one being sent.

Interacting with OneLake security programmatically

Lesson learned

- Adopt CI/CD practices early in the implementation
- Treat OneLake security as a foundational design decision
- Use user-level pass-through wherever possible
- Align semantic models with Direct Lake mode
- Groups / Users used in OneLake should be added to the Lake house level.
- [OneLake Security for SQL analytics endpoints \(Preview\) - Microsoft Fabric | Microsoft](#)

Sound off.
The mic is all yours.
Influence the product roadmap.

Join the Fabric User Panel



Share your feedback directly with our Fabric product group and researchers.

<https://aka.ms/JoinFabricUserPanel>

Join the SQL User Panel



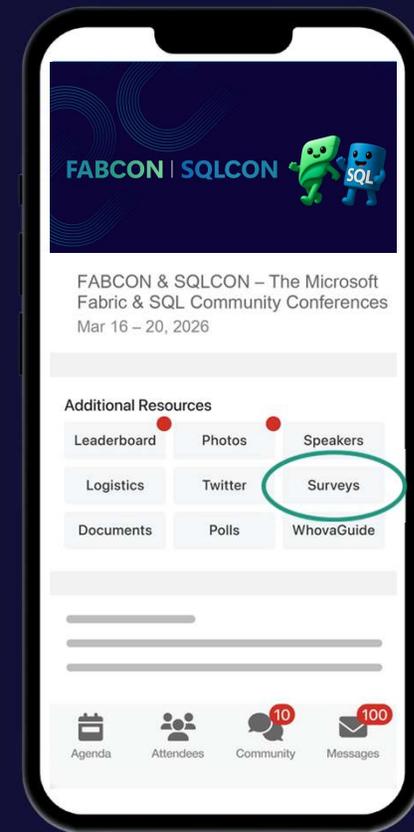
Influence our SQL roadmap and ensure it meets your real-life needs

<https://aka.ms/JoinSQLUserPanel>

How was the session?



Complete Session Surveys in *Whova* for your chance to WIN PRIZES!



Get Two Fabric Certifications for FREE

Attendees of FABCON can take the Fabric Analytics Engineer or Fabric Data Engineer exam for free. Be part of the 2 fastest growing role-based certifications in Microsoft history.

Request your voucher by March 23, 2026.

<https://aka.ms/fabcon/cert100>

