

Domain 4: Administer and govern Microsoft Fabric

Administer a Microsoft Fabric environment

1. Introduction

<https://learn.microsoft.com/en-us/training/modules/administer-fabric/1-introduction>

Introduction

Completed

Administering a Microsoft Fabric environment involves tasks that are essential for ensuring the efficient and effective use of the Fabric platform within an organization.

As a Fabric administrator (admin), you need to know:

- Fabric architecture
- Security and governance features
- Analytics capabilities
- Deployment and licensing options

You also need to be familiar with the Fabric admin portal and other administrative tools, and be able to configure and manage the Fabric environment to meet the needs of your organization.

Fabric admins work with business users, data analysts, and IT professionals to deploy and use Fabric to meet business objectives and comply with organizational policies and standards.

By the end of this module, you'll have an understanding of the Fabric administrator role and the tasks and tools involved in administering Fabric.

2. Understand the Fabric Architecture

<https://learn.microsoft.com/en-us/training/modules/administer-fabric/2-fabric-architecture>

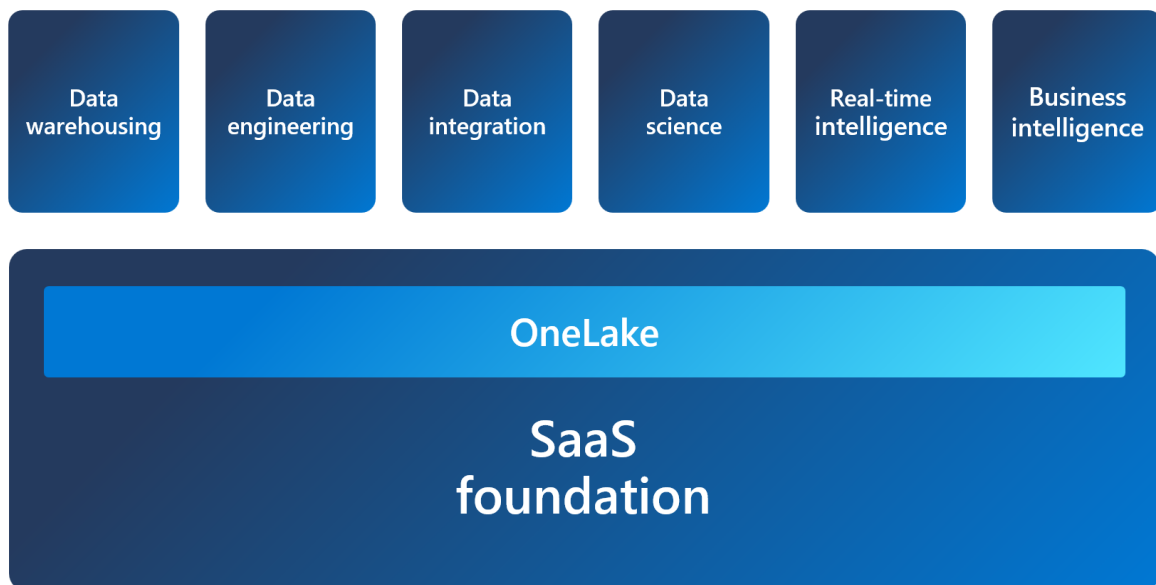
Understand the Fabric Architecture

Completed

Microsoft Fabric is a Software-as-a-Service platform, which provides a simple and integrated approach while reducing administrative overhead. Fabric provides an all-in-one analytics solution for enterprises that covers everything from data movement to data science, real-time analytics, and business intelligence. It offers a comprehensive suite of services, including:

- Data warehousing
- Data engineering
- Data integration
- Data science
- Real-time intelligence
- Business intelligence

All data in Fabric is stored in OneLake, which is built on Azure Data Lake Storage (ADLS) gen2 architecture. OneLake is hierarchical in nature to simplify management across your organization. There's only one OneLake per tenant and it provides a single-pane-of-glass file-system namespace that spans across users, regions, and even clouds.



Understand Fabric concepts

Tenant is a dedicated space for organizations to create, store, and manage Fabric items. There's often a single instance of Fabric for an organization, which is aligned with Microsoft Entra ID. The Fabric *tenant* maps to the root of OneLake and is at the top level of the hierarchy.

Capacity is a dedicated set of resources that is available at a given time to be used. A tenant can have one or more capacities associated with it. Capacity defines the ability of a resource to perform an activity or to produce output. Capacity needs vary by item and duration of use. Fabric offers capacity through the Fabric SKU and Trials.

Domain is a logical grouping of workspaces. Domains are used to organize items in a way that makes sense for your organization. You can group things together in a way that makes it easier for the groups of people to have access to workspaces. For example, you might have a domain for sales, another for marketing, and another for finance.

Workspace is a collection of items that brings together different functionality in a single tenant. It acts as a container that uses capacity for the work that is executed, and provides controls for who can access the items in it. For example, in a sales workspace, users associated with the sales organization can create a data warehouse, run notebooks, create datasets, create reports, and more.

Items are the building blocks of the Fabric platform. They're the objects that you create and manage in Fabric. There are different types of items, such as data warehouses, data pipelines, datasets, reports, and dashboards.

Understanding Fabric concepts is important for you as an admin, because it helps you understand how to manage the Fabric environment.

Note

For more information, see the [Start a Fabric trial](#) documentation.

3. Understand the Fabric administrator role

<https://learn.microsoft.com/en-us/training/modules/administer-fabric/3-admin-role-tools>

Understand the Fabric administrator role

Completed

Now that you understand the Fabric architecture and what you and your team might use Fabric for, let's look at the admin role and the tools used to manage the platform.

There are several roles that work together to administer Microsoft Fabric for your organization. If you're a Microsoft 365 admin, a Power Platform admin, or a Fabric capacity admin, you're involved in administering Fabric. The Fabric admin role was formerly known as Power BI admin.

As a Fabric admin, you work primarily in the Fabric admin portal. You might also need to familiarize yourself with the following tools:

- [Microsoft 365 admin center](#)
- [Microsoft 365 Security & Microsoft Purview compliance portal](#)
- [Microsoft Entra ID in the Azure portal](#)

- [PowerShell cmdlets](#)
- [Administrative APIs and SDK](#)

Note

For specific details on the different admin roles and responsibilities, see the [What is Microsoft Fabric administration?](#) documentation.

Describe admin tasks

As an admin, you might be responsible for a wide range of tasks to keep the Fabric platform running smoothly. These tasks include:

Security and access control: One of the most important aspects of Fabric administration is managing security and access control to ensure that only authorized users can access sensitive data. You can use role-based access control (RBAC) to:

- Define who can view and edit content.
- Set up data gateways to securely connect to on-premises data sources.
- Manage user access with Microsoft Entra ID.

Data governance: Effective Fabric administration requires a solid understanding of data governance principles. You should know how to secure inbound and outbound connectivity in your tenant and how to monitor usage and performance metrics. You should also know how to apply data governance policies to ensure data within your tenant is only accessible to authorized users.

Customization and configuration: Fabric administration also involves customizing and configuring the platform to meet the needs of your organization. You might configure private links to secure your tenant, define data classification policies, or adjust the look and feel of reports and dashboards.

Monitoring and optimization: As a Fabric admin, you need to know how to monitor the performance and usage of the platform, optimize resources, and troubleshoot issues. Examples include configuring monitoring and alerting settings, optimizing query performance, managing capacity and scaling, and troubleshooting data refresh and connectivity issues.

Specific tasks vary depending on the needs of your organization and the complexity of your Fabric implementation.

Describe admin tools

It's important to familiarize yourself with a few tools to effectively implement the tasks previously outlined. Fabric admins can perform most admin tasks using one or more of the following tools: the Fabric admin portal, PowerShell cmdlets, admin APIs and SDKs, and the admin monitoring workspace.

Fabric admin portal

Fabric's *admin portal* is a web-based portal where you can manage all aspects of the platform. You can centrally manage, review, and apply settings for the entire tenant or by capacity in the admin portal. You can also manage users, admins and groups, access audit logs, and monitor usage and performance.

The admin portal enables you to turn settings on and off. There are many settings located in the admin portal. One noteworthy setting is the Fabric on/off switch, located in tenant settings that lets organizations that use Power BI opt into Fabric. Here, you can enable Fabric for your tenant or allow capacity admins to enable Fabric.

Admin portal

The screenshot displays the Fabric Admin Portal interface. On the left is a sidebar with a list of settings categories: Tenant settings (marked 'New'), Usage metrics, Users, Premium Per User, Audit logs, Domains (marked 'New'), Workloads, Tags (marked 'New'), Capacity settings, Refresh summary, Embed Codes, Organizational visuals, Azure connections, Workspaces, Custom branding, Fabric identities, Featured content, and Help + support. The main content area shows a notification at the top: 'There are new or updated tenant settings. Expand to review the changes.' Below this, the 'Microsoft Fabric' section is active, displaying the setting 'Users can create Fabric items' which is 'Enabled for the entire organization'. A descriptive text explains that this allows production-ready features and that turning it off doesn't impact Power BI items. A toggle switch is currently in the 'Enabled' position. Under 'Apply to:', 'The entire organization' is selected with a radio button. There are also checkboxes for 'Specific security groups' and 'Except specific security groups'. A section for 'Delegate setting to other admins' includes a note about selecting admins and a checked checkbox for 'Capacity admins can enable/disable'. At the bottom are 'Apply' and 'Cancel' buttons.

PowerShell cmdlets

Fabric provides a set of *PowerShell cmdlets* that you can use to automate common administrative tasks. A PowerShell cmdlet is a simple command that can be executed in PowerShell.

For example, you can use cmdlets in Fabric to systematically create and manage groups, configure data sources and gateways, and monitor usage and performance. You can also use the cmdlets to manage the Fabric admin APIs and SDKs.

Note

See [Microsoft Power BI Cmdlets for Windows PowerShell and PowerShell Core](#) for more resources on PowerShell cmdlets that work with Fabric.

Admin APIs and SDKs

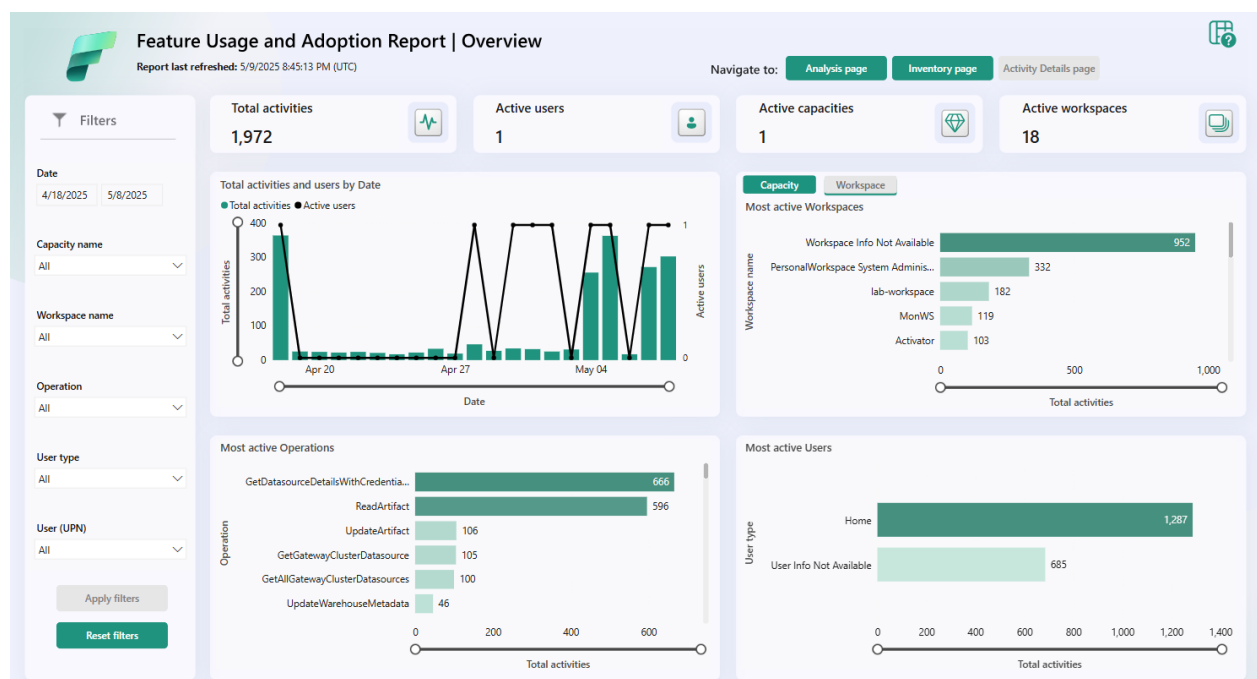
An admin *API and SDK* are tools that allow developers to interact with a software system programmatically. An API (Application Programming Interface) is a set of protocols and tools that enable communication between different software applications. An SDK (Software Development Kit) is a set of tools and libraries that helps developers create software applications that can interact with a specific system or platform. You can use APIs and SDKs to automate common administrative tasks and integrate Fabric with other systems.

For example, you can use APIs and SDKs to create and manage groups, configure data sources and gateways, and monitor usage and performance. You can also use the APIs and SDKs to manage the Fabric admin APIs and SDKs.

You can make these requests using any HTTP client library that supports OAuth 2.0 authentication, such as Postman, or you can use PowerShell scripts to automate the process.

Admin monitoring workspace

Fabric tenant admins have access to the *admin monitoring workspace*. You can choose to share access to the workspace or specific items within it with other users in your organization. The admin monitoring workspace includes the Feature Usage and Adoption dataset and report, which together provide insights on the usage and performance of your Fabric environment. You can use this information to identify trends and patterns, and troubleshoot issues.



Note

For more information about what is included in the Admin Monitoring Workspace, see [What is the Fabric admin monitoring workspace](#).

4. Manage Fabric security

<https://learn.microsoft.com/en-us/training/modules/administer-fabric/4-manage-security>

Manage Fabric security

Completed

As a Fabric admin, part of your role is to manage security for the Fabric environment, including managing users and groups, and how users share and distribute content in Fabric.

Manage users: assign and manage licenses

User licenses control the level of user access and functionality within the Fabric environment. Administrators ensure licensed users have the access they need to data and analytics to do their jobs effectively. They also limit access to sensitive data and ensure compliance with data protection laws and regulations.

Managing licenses allows administrators to monitor and control costs by ensuring that licenses are allocated efficiently and only to users who need them. This can help to prevent unnecessary expenses and ensure that the organization is utilizing its resources effectively.

Having the appropriate procedures in place to assign and manage licenses helps to control access to data and analytics, ensure compliance with regulations, and optimize costs.

License management for Fabric is handled in the Microsoft 365 admin center. For more information about managing licenses, see [Assign licenses to users](#).

Note

The *license type* in workspace settings is related to the user licenses listed here. Users can see reports depending on the user license and the workspace license. For detailed information, see the [Microsoft Fabric licenses](#) documentation.

Manage items and sharing

As an admin, you can manage how users share and distribute content. You can manage how users share content with others, and how they distribute content to others. You can also manage how users interact with items, such as data warehouses, data pipelines, datasets, reports, and dashboards.

Items in workspaces are best distributed through a workspace app or the workspace directly. Granting the least permissive rights is the first step in securing the data. Share the read only app for access to the reports or grant access to the workspaces for collaboration and development. Another aspect of managing and distributing items is enforcing these types of best practices.

You can manage sharing and distribution both internally and outside of your organization, in compliance with your organization's policies and procedures.

Note

For more information, see the [Security in Microsoft Fabric](#) documentation.

5. Govern data in Fabric

<https://learn.microsoft.com/en-us/training/modules/administer-fabric/5-govern-fabric>

Govern data in Fabric

Completed

Fabric includes built-in governance features to help you manage and control your data. *Endorsement* is a way for you as an admin to designate specific Fabric items as trusted and approved for use across the organization.

Admins can also make use of the *scanner API* to scan Fabric items for sensitive data, and the *data lineage* feature to track the flow of data through Fabric.

Endorse Fabric content

Endorsement is a key governance feature that builds trust in your data assets by marking Fabric items as reviewed and approved. Endorsed items display a badge, signaling to users that these assets are reliable. Endorsement helps users trust the data, and it also helps you as an admin manage the overall growth of items across your environment.

Promoted Fabric content appears with a Promoted badge in the Fabric portal. Workspace members with the contributor or admin role can promote content within a workspace. The Fabric admin can promote content across the organization.

Certified content requires a more formal process that involves a review of the content by a designated reviewer. Content appears with a Certified badge in the Fabric portal. Admins manage the certification process and can customize it to meet the needs of your organization.

If you aren't an admin, you need to request item certification from an admin. You can perform request certification by selecting the item in the Fabric portal, and then selecting **Request certification** from the **More** menu.

Note

For more detailed information on the content endorsement process, see [Promote or certify content](#).

Scan for sensitive data

Metadata scanning facilitates governance of data by enabling cataloging and reporting on all the metadata of your organization's Fabric items. The *scanner API* is a set of Admin REST APIs that allows you to scan Fabric items for sensitive data. Use the scanner API to scan data warehouses, data pipelines, datasets, reports, and dashboards for sensitive data. The scanner API can be used to scan both structured and unstructured data.

Important

Before metadata scanning can be run, it needs to be set up in your organization by an Admin. For more information, see the [Metadata scanning overview](#).

Track data lineage

Data lineage is the ability to track the flow of data through Fabric, also known as *impact analysis*. Data lineage allows you to see where data comes from, how it's transformed, and where it goes. The lineage view in workspaces helps you understand the data that is available in Fabric, and how it's being used.

Report on sensitive data

With the Microsoft Purview hub (preview) in Fabric, you can manage and govern your organization's Fabric data estate. It contains reports that provide insights about sensitive data, item endorsement, and domains, and also serves as a gateway to more advanced capabilities in the Microsoft Purview portal such as Data Catalog, Information Protection, Data Loss Prevention, and Audit.

6. Module assessment

<https://learn.microsoft.com/en-us/training/modules/administer-fabric/6-knowledge-check>

Module assessment

Completed

7. Summary

<https://learn.microsoft.com/en-us/training/modules/administer-fabric/7-summary>

Summary

Completed

In this module, you learned about the Fabric architecture and the role of an administrator for the Fabric platform. You also explored the different tools available for managing security and sharing, as well as the governance features that can be used to enforce standards and ensure compliance. Your understanding of how to manage a Fabric environment ensures that it's secure, compliant, and well-governed. With this knowledge, you're well-equipped to help your organization get the most out of Fabric and derive valuable insights from all your data.

For more information about data governance, complete the [Govern data in Microsoft Fabric with Purview](#) module.

Secure data access in Microsoft Fabric

1. Introduction

<https://learn.microsoft.com/en-us/training/modules/secure-data-access-in-fabric/1-introduction>

Introduction

Completed

Security in Microsoft Fabric is optimized for securing data for specific use cases. Different users need the ability to perform various actions in Fabric to fulfill their job responsibilities. Fabric facilitates this by allowing you to grant users access to specific data workloads through workspace and item permissions, compute permissions, and OneLake data access roles (preview).

Secure data by use case

A security use case in Fabric refers to a set of users needing data access and accessing data in a specific way. Once a use case is identified, Fabric permissions associated with that use case can be configured. Suppose you work at a healthcare company with multiple systems that store data, such as electronic health records (EHR), insurance claims data, clinical trial data, patient and disease registries, and administrative data. Different users within your company or partner organizations need to view, transform, analyze, aggregate, and use this data to derive business insights. Users need access to different Fabric compute engines, items, and workspaces to perform their jobs effectively:

- **Data engineers** need access to data in a lakehouse to develop downstream data products.
- **Business or data analysts** need to query data to answer business questions.
- **Data scientists** need to access data in a lakehouse and consume it through Apache Spark to create models and experiments.
- **Report creators** need to build reports to share with report consumers.
- **Report consumers** need to view data in Power BI reports to make decisions.

In this module, you'll learn how to use the Fabric access control features available to secure your data and provide your team with the necessary access within Fabric to perform their job duties. You'll explore Fabric's multi-layer security model and how to use it to manage data access.

By the end of this module, you'll know how to configure security for an entire workspace, for individual Fabric data items, and how to apply granular permissions within Fabric data items.

2. Understand the Fabric security model

<https://learn.microsoft.com/en-us/training/modules/secure-data-access-in-fabric/2-understand-fabric-security-model>

Understand the Fabric security model

Completed

Data access in organizations is often restricted by users' responsibilities, and roles and by an organization's Fabric deployment patterns, and data architecture. Fabric has a flexible, multi-layer security model that allows you to configure security to accommodate different data access requirements. Having the ability to control permissions at different layers means you can adhere to the principle of least privilege, restricting user permissions to only what's needed to perform job tasks.

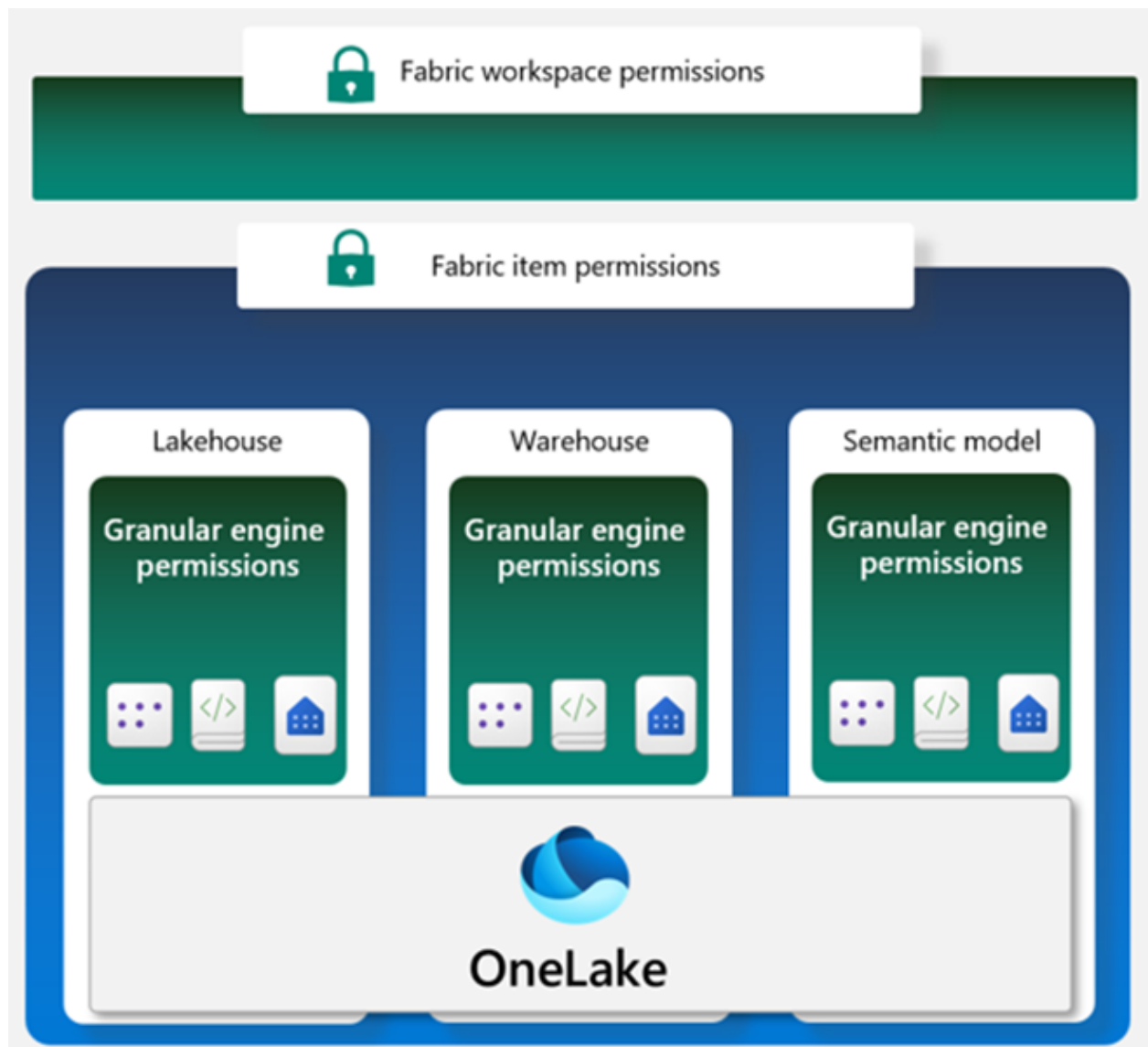
Fabric has three security levels and they're evaluated sequentially to determine whether a user has data access. The order of evaluation for access is:

1. Microsoft Entra ID authentication: checks if the user can authenticate to the Azure identity and access management service, Microsoft Entra ID.
2. Fabric access: checks if the user can access Fabric.
3. Data security: checks if the user can perform the action they've requested on a table or file.

The third level, data security, has several building blocks that can be configured individually or together to align with different access requirements. The primary access controls in Fabric are:

- Workspace roles
- Item permissions
- Compute or granular permissions
- OneLake data access controls (preview)

It's helpful to envision these building blocks in a hierarchy to understand how access controls can be applied individually or together.



A *workspace* in Fabric enables you to distribute ownership and access policies using *workspace roles*. Within a workspace, you can create Fabric data *items* like lakehouses, data warehouses, and semantic models. *Item permissions* can be inherited from a workspace role or set individually by sharing an item. When workspace roles provide too much access, items can be shared using item permissions to ensure proper security.

Within each data item, *granular engine permissions* such as Read, ReadData, or ReadAll can be applied.

Compute or granular permissions can be applied within a specific compute engine in Fabric, like the SQL Endpoint or semantic model.

Fabric data items store their data in OneLake. Access to data in the lakehouse can be restricted to specific files or folders using the role-based-access control (RBAC) feature called OneLake data access controls (preview).

3. Configure workspace and item permissions

<https://learn.microsoft.com/en-us/training/modules/secure-data-access-in-fabric/3-configure-workspace-and-item-permissions>

Configure workspace and item permissions

Completed

Workspaces are environments where users can collaborate to create groups of items. Items are the resources you can work with in Fabric such as lakehouses, warehouses, and reports. Workspace roles are preconfigured sets of permissions that let you manage what users can do and access in a Fabric workspace.

Item permissions control access to individual Fabric items within a workspace. Item permissions let you either adjust the permissions set by a workspace role or give a user access to one or more items within a workspace without adding the user to a workspace role.

Let's consider some scenarios where you would need to configure data access using workspace roles and item permissions.

Understand workspace roles

Suppose you work at a health care company as the Fabric security admin. You need to set up access for a new data engineer. The data engineer needs the ability to:

- Create Fabric items in an existing workspace
- Read all data in an existing lakehouse that's in the same workspace where they can create Fabric items

Workspace roles control what users can do and access within a Fabric workspace. There are four workspace roles and they apply to *all items* within a workspace. Workspace roles can be assigned to individuals, security groups, Microsoft 365 groups, and distribution lists. Users can be assigned to the following roles:

- **Admin** - Can view, modify, share, and manage all content and data in the workspace, and manage permissions.
- **Member** - Can view, modify, and share all content and data in the workspace.
- **Contributor** - Can view and modify all content and data in the workspace.
- **Viewer** - Can view all content and data in the workspace, but can't modify it.

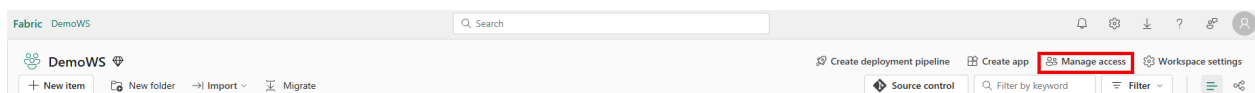
Tip

For a full list of the permissions associated with workspace roles, see: [Roles in workspaces](#)

To meet the access requirements for the new data engineer, you can assign them the workspace **Contributor** role. This gives them access to modify content in the workspace, including creating Fabric items like lakehouses. The contributor role would also allow them to read data in the existing lakehouse.

Assign workspace roles

Users can be added to workspace roles from the **Manage access** button from within a workspace. Add a user by entering the user's name and selecting the workspace role to assign them in the **Add people** dialogue.



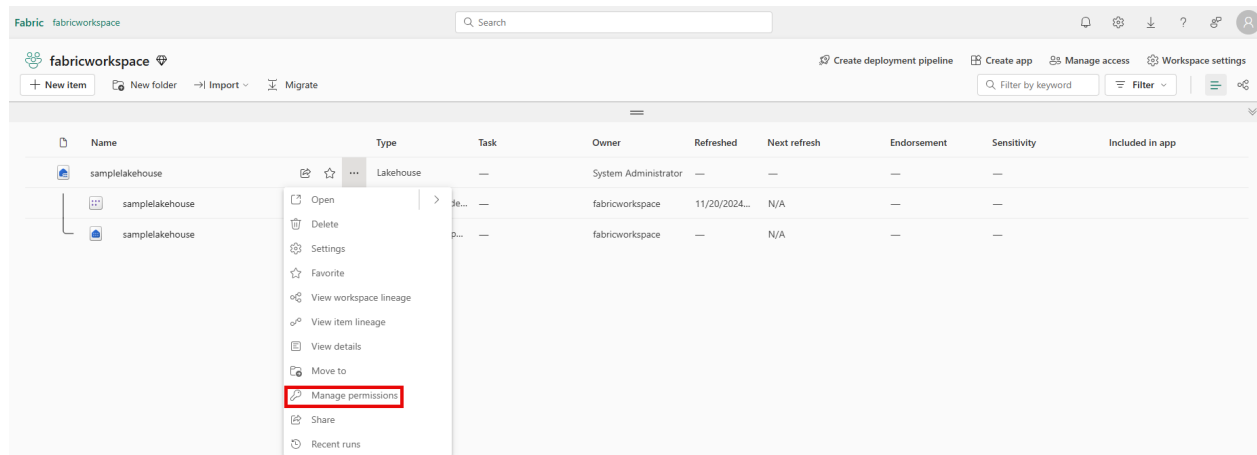
Configure item permissions

Item permissions control access to individual Fabric items within a workspace. Item permission can be used to give a user access to one or more items within a workspace without adding the user to a workspace role or can be used with workspace roles.

Suppose that after a few months of having **Contributor** access on a workspace, a data engineer no longer needs to create Fabric items and now only needs to view a single lakehouse and read data in it.

Since the engineer no longer needs to view all items in the workspace, the **Contributor** workspace role can be removed and item permissions on the lakehouse can be configured so the engineer will only be able to see the lakehouse metadata and data and nothing else in the workspace. This item access configuration helps you adhere to the principle of least privilege, where the engineer only has access to what's needed to perform their job duties.

An item can be shared and item permissions can be configured by selecting on the ellipsis (...) next to a Fabric item in a workspace and then selecting **Manage permissions**.



In the **Grant people access** window that appears after selecting **Manage permissions**, if you add the user and don't select any of the checkboxes under **Additional permissions**, the user will have read access to the lakehouse metadata. The user won't have access to the underlying data in the lakehouse. To grant the engineer the ability to read data and not just metadata, **Read all SQL endpoint data** or **Read all Apache Spark** can be selected.

Grant people access



samplelakehouse

People you share this Lakehouse with can open it and its SQL endpoint and read the default dataset. To allow them to read directly in the Lakehouse, grant additional permissions.

A

Alice

X

Additional permissions

- ☒ Read all SQL endpoint data ⓘ
- ☒ Read all Apache Spark ⓘ
- ☐ Build reports on the default semantic model

Notification Options

- ☒ Notify recipients by email

Add a message (optional)

i Depending on which additional permissions you select, recipients will have different access to the SQL endpoint, default dataset, and data in the lakehouse. For details, view lakehouse permissions documentation.

Grant

Back

Tip

Each Fabric data item has its own security model. To learn more about permissions that can be granted when a lakehouse or other Fabric data item is shared see:

- [Warehouse](#)
- [Lakehouse](#)
- [Semantic model](#)

4. Apply granular permissions

<https://learn.microsoft.com/en-us/training/modules/secure-data-access-in-fabric/4-apply-granular-permissions>

Apply granular permissions

Completed

When the permissions provided by workspace roles or item permissions are insufficient, granular permissions like table and row-level security and file and folder access can be set through the:

- SQL analytics endpoint
- OneLake data access roles (preview)
- Warehouse
- Semantic model

Configure data access through the SQL analytics endpoint in a lakehouse

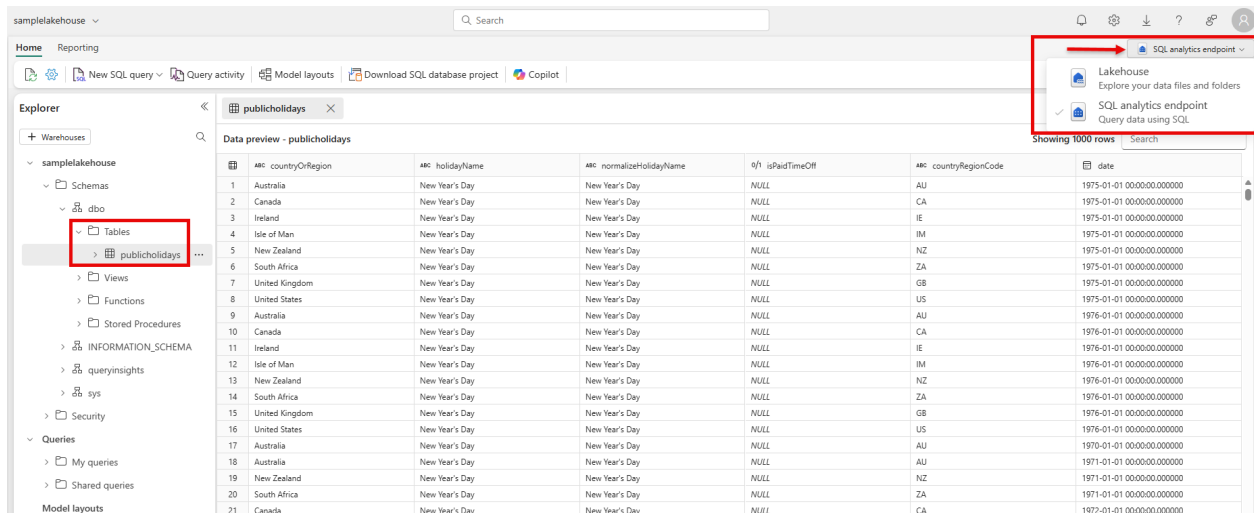
Data in a lakehouse can be read through the SQL analytics endpoint. Each Lakehouse has an autogenerated SQL analytics endpoint that can be used to transition between the *lake* view of the lakehouse and the *SQL* view of the lakehouse. The *lake* view supports data engineering and Apache Spark and the *SQL* view of the same lakehouse allows you to create views, functions, stored procedures and to apply SQL security and object level permissions.

Data in a Fabric lakehouse is stored with the following folder structure:

- /Files
- /Tables

View the SQL analytics endpoint view of the lakehouse

The SQL analytics endpoint is used to read data in the /Tables folder of the lakehouse using T-SQL.



Apply granular permissions to the lakehouse using T-SQL

Using the SQL analytics endpoint, granular T-SQL permissions can be applied to SQL objects using Data Control Language (DCL) commands such as:

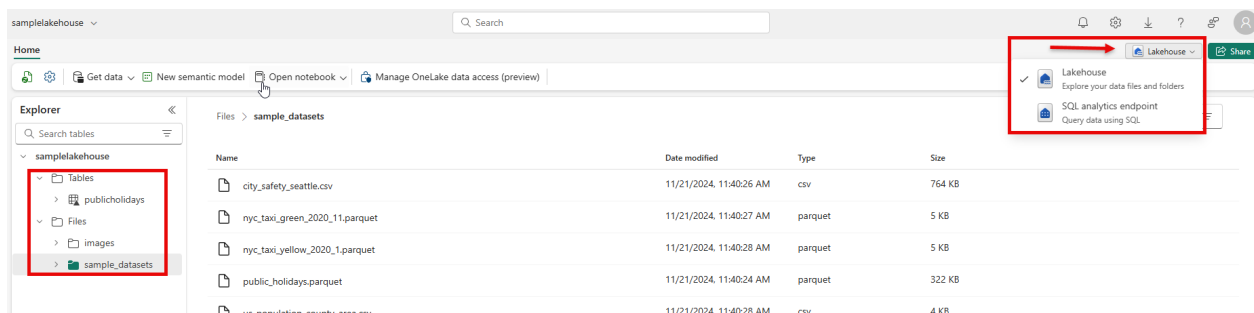
- [GRANT](#)
- [DENY](#)
- [REVOKE](#)

Row-level security, column-level security, and dynamic data masking can also be applied using the SQL analytics endpoint. See:

- [Row-level security](#)
- [Column-level security](#)
- [Dynamic data masking](#)

Configure data access through the lake view of the lakehouse

The lake view of the lakehouse is used to read data in the /Tables and /Files folder of the lakehouse.

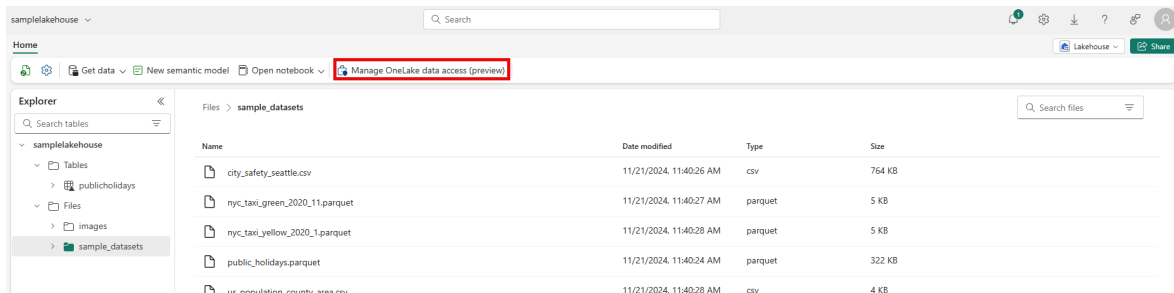


Use OneLake data access roles to secure data

Workspace and item permissions provide coarse access to data in a lakehouse. To further refine data access, folders in the lake view of the lakehouse can be secured using OneLake data access roles

(preview). You can create custom roles within a lakehouse and grant read permissions only to specific folders in OneLake. Folder security is inheritable to all subfolders. To create a custom OneLake data access role:

1. Select **Manage OneLake data access (preview)** from the menu in the lake view of the lakehouse.



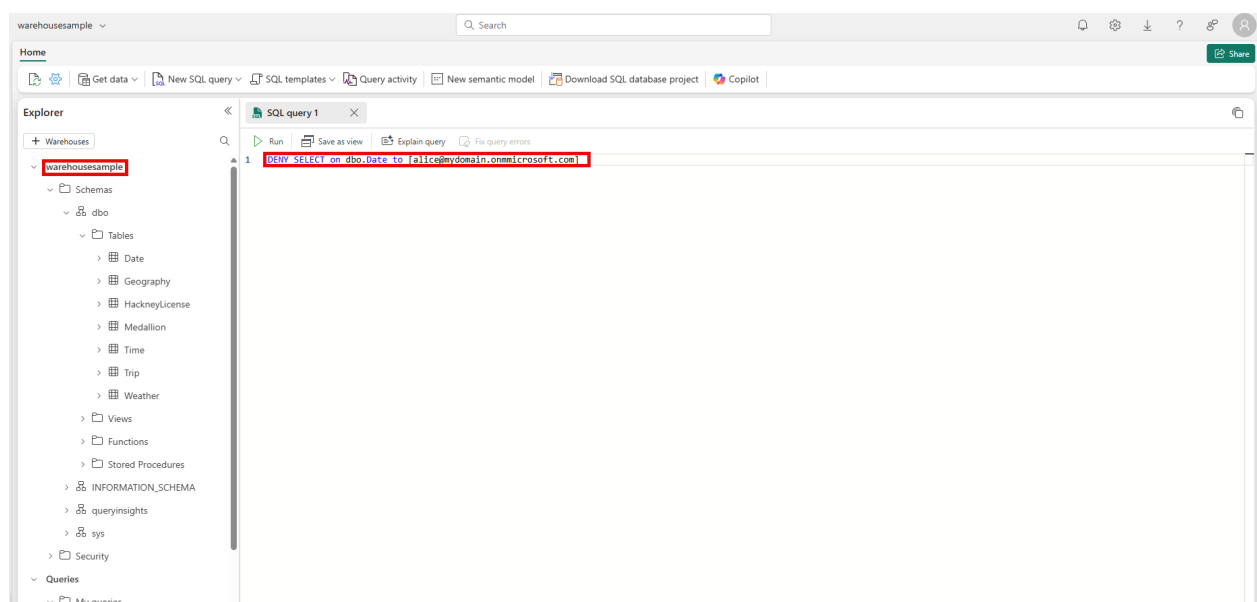
2. In the **New Role** window, create a new role name and select the folders to grant access to.
3. Once the role is created, assign a user or group to the role and select the permissions to assign.

Tip

For more information on how OneLake RBAC permissions are evaluated with workspace and item permissions, see: [How OneLake RBAC permissions are evaluated with Fabric permissions](#)

Configure granular warehouse permissions

Granular permissions can be applied to warehouses using the SQL analytics endpoint, similar to the way the endpoint is used for the lakehouse. The same permissions can be applied: GRANT, REVOKE, and DENY and row-level security, column-level security, and dynamic data masking.



Configure Semantic model permissions

A user's role in a workspace implicitly grants them permission on the semantic models in a workspace. Semantic models allow for security to be defined using DAX. More granular permission can be applied using row-level security (RLS). To learn more about the managing RLS or permissions on the semantic model see:

- [Semantic model permissions](#)
- [Row-level security \(RLS\) with Power BI](#)

5. Exercise: Secure data access in Microsoft Fabric

<https://learn.microsoft.com/en-us/training/modules/secure-data-access-in-fabric/5-exercise-secure-data-access>

Exercise: Secure data access in Microsoft Fabric

Completed

Now it's your chance to secure data access in Microsoft Fabric.

In this exercise, you learn how to secure data access in Fabric using the concepts explored in this module.

Note

- You need a Microsoft Fabric trial license with the Fabric preview enabled in your tenant. See [Getting started with Fabric](#) to enable your Fabric trial license.
- To complete the exercises in this lab, you'll need two users: one user should be assigned the Workspace Admin role, and the other will be assigned permissions throughout this lab. To assign roles to workspaces see [Give access to your workspace](#).

Launch the exercise and follow the instructions.

[Launch Exercise](#)

6. Module assessment

Module assessment

Completed

7. Summary

<https://learn.microsoft.com/en-us/training/modules/secure-data-access-in-fabric/7-summary>

Summary

Completed

In this module, you learned how to use the access control features available across various Fabric engines to secure your data and provide your team with the necessary access within Fabric to perform their job duties. You explored Fabric's multi-layer security model and how to use it to manage data access.

The main takeaways from this module include understanding how to configure security for an entire workspace, and for individual Fabric data items, and how to apply granular permissions.

For more reading, you can refer to the following URLs:

- [Microsoft Fabric security white paper](#)
- [Microsoft Fabric permission model](#)
- [Build common data architectures with OneLake in Microsoft Fabric](#)
- [How to secure data for common Fabric data architectures](#)
- [End-to-end security scenario](#)

Secure a Microsoft Fabric data warehouse

1. Introduction

<https://learn.microsoft.com/en-us/training/modules/secure-data-warehouse-in-microsoft-fabric/1-introduction>

Introduction

Completed

Microsoft Fabric Data Warehouse is a complete platform for data, analytics, and AI (Artificial Intelligence). It refers to the process of storing, organizing, and managing large volumes of structured and semi-structured data.

In a warehouse, administrators have access to a suite of technologies aimed at safeguarding sensitive information. These security measures are capable of securing or masking data from users or roles without proper authorization, ensuring data protection across both Warehouse and SQL analytics endpoints. This ensures a smooth and secure user experience, with no need for alterations to the existing applications.

Understand security features

Data engineers who are often well-versed in the SQL engine and adept at using T-SQL, will find warehouses in Microsoft Fabric easy to use.

This is because warehouses are powered by the same SQL engine they're familiar with, enabling them to perform complex queries and data manipulations. The SQL engine's wide range of security features further allows for sophisticated security mechanism at the warehouse level.

- **Workspaces roles** – Designed to provide different levels of access and control within the workspace. You can assign users to the various workspace roles such as **Admin**, **Member**, **Contributor**, and **Viewer**. These roles are crucial for maintaining the security and efficiency of data warehousing operations within an organization.
- **Item permissions** – Individual warehouses can have item permissions assigned directly to them. The main intent behind assigning such permissions is to facilitate the sharing of the Warehouse for downstream use.

- **Data protection security** – For more precise control, you can use T-SQL to grant specific permissions to users. Warehouse supports a range of data protection features that enable administrators to shield sensitive data from unauthorized access. This includes object-level security for database objects, column-level security for table columns, row-level security for table rows using `WHERE` clause filters, and dynamic data masking to obscure sensitive data like email addresses. These features ensure data protection across Warehouses and SQL analytics endpoints without necessitating changes to applications.

In the next units, we'll explore various ways of enabling security in a warehouse, and how they can facilitate the tasks of keeping your data warehouse workload protected.

2. Explore dynamic data masking

<https://learn.microsoft.com/en-us/training/modules/secure-data-warehouse-in-microsoft-fabric/2-explore-dynamic-data-masking>

Explore dynamic data masking

Completed

Dynamic Data Masking (DDM) is a security feature that limits data exposure to nonprivileged users by obscuring sensitive information.

Dynamic data masking offers several key benefits that enhance the security and manageability of your data. One of the primary advantages is its real-time masking feature. When querying sensitive data, DDM applies dynamic masking to it in real time. This process means that the actual data is never exposed to unauthorized users, thus enhancing the security of your data. Furthermore, DDM is straightforward to implement. It doesn't require complex coding, making it accessible for users of all skill levels.

Another benefit of DDM is that the data in the database isn't changed when DDM is applied. Instead, the masking rules are applied to the query results. This benefit means that the actual data remains intact and secure, while nonprivileged users only see a masked version of the data.

Define masking rule

Dynamic data masking, which is set up at the column level, offers a suite of features including comprehensive and partial masking capabilities, along with a random masking function designed for numeric data.

Masking Type	Description	Use Case	Limitations	Masking Rule
Default	Full masking according to the data types of the designated fields.	Useful when you want to completely hide the actual data.	Completely mask the data. No information is visible.	<code>default()</code>
Email	Exposes the first letter of an email address and the constant suffix ".com"	Useful when you want to show that the data field contains an email without revealing the actual email.	Only applicable to email fields.	<code>email()</code>
Custom Text	Exposes the first and last 'n' characters and adds a custom padding string in the middle.	Useful when you want to partially hide the actual data.	Not suitable for numeric, date, and time data types.	<code>partial(prefix_padding, padding_string, suffix_padding)</code>
Random	Replaces any numeric or binary value with a random number within a specified range.	Useful when you want to hide the actual numeric or binary data.	Only applicable to numeric and binary data types.	<code>random(low, high)</code>

The `prefix_padding` and `suffix_padding` parameters in the `partial()` function specify the number of characters to expose at the beginning and end of the string, and the `padding_string` parameter specifies the string to use for masking the remaining characters.

The `low` and `high` parameters in the `random()` function specify the range of random numbers to generate.

These masking types help prevent unauthorized viewing of sensitive data by enabling administrators to specify how much sensitive data to reveal, with minimal effect on the application layer. They're applied to query results, so the data in the database isn't changed. This approach allows many applications to mask sensitive data without modifying existing queries.

Configure data masking

Let's consider an example of a warehouse that stores customer information. The warehouse contains a `Customer` table with fields such as `CustomerName`, `Email`, `PhoneNumber`, and `CreditCardNumber`.

To apply data masking on the `CustomerName`, `Email`, `PhoneNumber`, and `CreditCardNumber` columns, run the following command:

```
-- For Email
ALTER TABLE Customers
ALTER COLUMN Email ADD MASKED WITH (FUNCTION = 'email()');

-- For PhoneNumber
ALTER TABLE Customers
ALTER COLUMN PhoneNumber ADD MASKED WITH (FUNCTION = 'partial(0,"XXX-XXX-",4)');

-- For CreditCardNumber
ALTER TABLE Customers
ALTER COLUMN CreditCardNumber ADD MASKED WITH (FUNCTION = 'partial(0,"XXXX-XXXX-XX
```

View masked results

Without Dynamic Data Masking, if a nonprivileged user runs a query to fetch customer details, they might see something like this:

```
CustomerName: John Doe
Email: johndoe@contoso.com
PhoneNumber: 123-456-7890
CreditCardNumber: 1234-5678-9012-3456
```

However, with DDM applied to the `Email`, `PhoneNumber`, and `CreditCardNumber` fields, the same query would return:

```
CustomerName: John Doe
Email: j*****@contoso.com
PhoneNumber: XXX-XXX-7890
CreditCardNumber: XXXX-XXXX-XXXX-3456
```

As you can see, the sensitive data is hidden from the nonprivileged user, enhancing the security of your data. This scenario is a basic example of how Dynamic Data Masking works. It helps to ensure that sensitive data isn't exposed to users who don't have the necessary privileges to view it.

Note

Unprivileged users with query permissions can infer the actual data since the data isn't physically obfuscated.

DDM should be used as part of a comprehensive data security strategy that includes proper management of object-level security with SQL granular permissions and adherence to the principle of minimal required permissions.

3. Implement row-level security

Implement row-level security

Completed

Row-Level Security (RLS) is a feature that provides granular control over access to rows in a table based on group membership or execution context.

For example, in an e-commerce platform, you can ensure that sellers only have access to order rows that are related to their own products. This way, each seller can manage their orders independently, while maintaining the privacy of other sellers' order information.

If you have experience with SQL Server, you find that row-level security shares similar characteristics and features.

Protect your data

Row-Level Security (RLS) works by associating a function, known as a security predicate, with a table. This function is defined to return *true* or *false* based on certain conditions, typically involving the values of one or more columns in the table. When a user attempts to access data in the table, the security predicate function is invoked. If the function returns *true*, the row is accessible to the user; if it returns *false*, the row doesn't show up in the query results.

Depending on the business requirements, an RLS predicate can be as simple as `WHERE CustomerId = 29` or as complex as required.

This process is transparent to the user and is enforced automatically by SQL Server, ensuring consistent application of security rules.

Row-level security is implemented in two main steps:

- **Filter predicates** - It's an inline table-valued function that filters the results based on the predicate defined.

Access	Definition
SELECT	Can't view rows that are filtered.
UPDATE	Can't update rows that are filtered.
DELETE	Can't delete rows that are filtered.
INSERT	Not applicable.

- **Security policy** - It's a security policy that invokes an inline table-valued function to protect access to the rows in a table.

Because access control is configured and applied at the warehouse level, application changes are minimal - if any. Also, users can directly have access to the tables and can query their own data.

Configure row-level security

The T-SQL commands below demonstrate how to use RLS in a scenario where user access is segregated by tenant:

```
-- Create supporting objects for this example
CREATE TABLE [Sales] (SalesID INT,
    ProductID INT,
    TenantName NVARCHAR(50),
    OrderQty INT,
    UnitPrice MONEY)
GO

INSERT INTO [Sales] VALUES (1, 3, 'tenant1@contoso.com', 5, 10.00);
INSERT INTO [Sales] VALUES (2, 4, 'tenant2@contoso.com', 2, 57.00);
INSERT INTO [Sales] VALUES (3, 7, 'tenant3@contoso.com', 4, 23.00);
INSERT INTO [Sales] VALUES (4, 2, 'tenant4@contoso.com', 2, 91.00);
INSERT INTO [Sales] VALUES (5, 9, 'tenant5@contoso.com', 5, 80.00);

-- View all the rows in the table
SELECT * FROM Sales;
```

Next, we create a new schema, an inline table-valued function, and grant user access to the new function. The `WHERE @TenantName = USER_NAME() OR USER_NAME() = 'TenantAdmin'` predicate evaluates if the user name executing the query matches the *TenantName* column values.

```
--Create a schema
CREATE SCHEMA [Sec];
GO

--Create the filter predicate
CREATE FUNCTION sec.tvf_SecurityPredicatebyTenant(@TenantName AS NVARCHAR(10))
    RETURNS TABLE
WITH SCHEMABINDING
AS
    RETURN      SELECT 1 AS result
                WHERE @TenantName = USER_NAME() OR USER_NAME() = 'tenantAdmin'
GO

--Create security policy and add the filter predicate
CREATE SECURITY POLICY sec.SalesPolicy
ADD FILTER PREDICATE sec.tvf_SecurityPredicatebyTenant(TenantName) ON [dbo].[Sales]
WITH (STATE = ON);
GO
```

The `tenantAdmin@contoso.com` user should see all the rows. The `tenant1@contoso.com` to `tenant5@contoso.com` users should only see their own rows.

If you alter the security policy with `WITH (STATE = OFF);`, you notice that users see all the rows.

Note

There is a risk of information leakage if an attacker writes a query with a specially crafted `WHERE` clause and, for example, a divide-by-zero error, to force an exception if the `WHERE` condition is true. This is known as a *side-channel attack*.

Explore use cases

Row-level security is ideal for many scenarios, including:

- When you need to isolate departmental access at the row level.
- When you need to restrict customers' data access to only the data relevant to their company.
- When you need to restrict access for compliance purposes.

Apply best practices

Here are a few best practices to consider when implementing RLS:

- It's recommended to create a separate schema for predicate functions, and security policies.
- Whenever possible, avoid type conversions in predicate functions.
- To maximize performance, avoid using excessive table joins and recursion in predicate functions.

4. Implement column-level security

<https://learn.microsoft.com/en-us/training/modules/secure-data-warehouse-in-microsoft-fabric/4-implement-column-level-security>

Implement column-level security

Completed

Column-level security (CLS) allows you to restrict column access in order to protect sensitive data. It provides granular control over who can access specific pieces of data, enhancing the overall security of your data warehouse.

Secure sensitive data

Let's consider a practical example of column-level security (CLS) in the healthcare industry. Suppose we have a table named `Patients` with the following columns: `PatientID`, `Name`, `Address`, `DateOfBirth`, and `MedicalHistory`.

The `MedicalHistory` column contains sensitive health information about the patients. As per the healthcare regulations and privacy laws, this information should only be accessible to authorized medical personnel such as doctors or nurses.

Here's how you might implement column-level security in this scenario:

1. **Identify the sensitive columns:** In this case, the `MedicalHistory` column is identified as containing sensitive data.
2. **Define access roles:** Define roles such as `Doctor` and `Nurse` who are allowed to access the `MedicalHistory` column. Other roles such as `Receptionist` or `Patient` might be restricted from accessing this column.
3. **Assign roles to users:** Assign the appropriate roles to each user in the warehouse. For example, user `DrSmith` might be assigned the `Doctor` role, while user `JohnDoe` might be assigned the `Patient` role.
4. **Implement access control:** Restrict access to the `MedicalHistory` column based on the user's role.

Column-level security can help ensure that sensitive health information is only accessible to those who are authorized to see it, though protecting patient privacy and complying with healthcare regulations.

Configure column-level security

In the scenario we've recently explored, the syntax to use for implementing column-level security might look as follows:

```
-- Create roles
CREATE ROLE Doctor AUTHORIZATION dbo;
CREATE ROLE Nurse AUTHORIZATION dbo;
CREATE ROLE Receptionist AUTHORIZATION dbo;
CREATE ROLE Patient AUTHORIZATION dbo;
GO

-- Grant SELECT on all columns to all roles
GRANT SELECT ON dbo.Patients TO Doctor;
GRANT SELECT ON dbo.Patients TO Nurse;
GRANT SELECT ON dbo.Patients TO Receptionist;
GRANT SELECT ON dbo.Patients TO Patient;
GO
```

```
-- Deny SELECT on the MedicalHistory column to the Receptionist and Patient roles
DENY SELECT ON dbo.Patients (MedicalHistory) TO Receptionist;
DENY SELECT ON dbo.Patients (MedicalHistory) TO Patient;
GO
```

In this example, we first create the roles `Doctor`, `Nurse`, `Receptionist`, and `Patient`. We then grant `SELECT` permissions on all columns in the `Patients` table to all roles. Finally, we deny `SELECT` permissions on the `MedicalHistory` column to the `Receptionist` and `Patient` roles. This ensures that only users with the `Doctor` or `Nurse` role can access the `MedicalHistory` column.

Understand the benefits

In the realm of warehouse security, two commonly used techniques are column-level security and views. Both methods serve to restrict access to sensitive data, but they do so in different ways and offer different advantages. The following table provides a comparative analysis of these two techniques across various aspects such as granularity of access control, maintenance, performance, transparency, and flexibility.

This comparison can help you understand the strengths and weaknesses of each method and guide you in choosing the most suitable approach for your specific application requirements.

Aspect	Column-Level Security	Views
Granularity of Access Control	Allows control at a more granular level. Can specify different access rights for different users or roles on different columns within the same table.	Would need to create different views for different sets of permissions.
Maintenance	Permissions are tied to the columns themselves, so they automatically adapt to changes in table structure.	If the underlying table structure changes, views may need to be updated to reflect these changes.
Performance	Generally more efficient because it operates directly on the table data.	Can introduce performance overhead, especially if they are complex or if the underlying tables are large.
Transparency	The restrictions are transparent to the user. The user queries the table as usual, and the database engine takes care of applying the security rules.	The user needs to query a different object (the view instead of the table).
Flexibility	Less flexible than views.	Very flexible and can provide row-level security (by including a <code>WHERE</code> clause in the view definition) in addition to column-level security. They can also transform the data (for example, by calculating

Aspect	Column-Level Security	Views
		derived columns) which is not possible with column-level security.

The choice between using column-level security or views will depend on the specific requirements of your application. Always make sure to test any security changes in a safe environment before applying them to a production warehouse.

5. Configure SQL granular permissions using T-SQL

<https://learn.microsoft.com/en-us/training/modules/secure-data-warehouse-in-microsoft-fabric/5-configure-sql-granular-permissions>

Configure SQL granular permissions using T-SQL

Completed

If you're familiar with relational databases and enterprise warehouses, it's common knowledge that there are four fundamental permissions governing [Data Manipulation Language \(DML\)](#) operations. These permissions, namely `SELECT`, `INSERT`, `UPDATE`, and `DELETE`, are universally applicable across all database platforms.

All of these permissions can be granted, revoked or denied on tables and views. If a permission is granted using the `GRANT` statement, then the permission is given to the user or role referenced in the `GRANT` statement. Users can also be denied permissions using the `DENY` command. If a user is granted a permission and denied the same permission, the `DENY` will always supersede the grant, and the user will be denied access to the specific object.

Table and view permissions

Tables and views represent the objects on which permissions can be granted within a warehouse. Within those tables and views, you can additionally restrict the columns that are accessible to a given security principal.

Permission	Definition
<code>SELECT</code>	Allows the user to view the data within the object (table or view). When denied, the user will be prevented from viewing the data within the object.

Permission	Definition
INSERT	Allows the user to insert data into the object. When denied, the user will be prevented from inserting data into the object.
UPDATE	Allows the user the update data within the object. When denied, the user will be prevented from updating data in the object.
DELETE	Allows the user to delete data within the object. When denied, the user will be prevented from deleting data from the object.

Function and stored procedure permissions

Like tables and views, functions and stored procedures have several permissions, which can be granted or denied.

Permission	Definition
ALTER	Grants the user the ability to change the definition of the object.
CONTROL	Grants the user all rights to the object.

Principle of least privilege

The basic idea of the principle of least privilege is that users and applications should only be given the permissions needed in order for them to complete the task. Applications should only have permissions that they need to do in order to complete the task at hand.

As an example, if an application accesses all data through stored procedures, then the application should only have the permission to execute the stored procedures, with no access to the tables.

Dynamic SQL

Dynamic SQL is a concept where a query is built programmatically. Dynamic SQL allows T-SQL statements to be generated within a stored procedure or a query itself. A simple example is shown below.

```
CREATE PROCEDURE sp_TopTenRows @tableName NVARCHAR(128)
AS
BEGIN
    DECLARE @query NVARCHAR(MAX);
    SET @query = N'SELECT TOP 10 * FROM ' + QUOTENAME(@tableName);
    EXEC sp_executesql @query;
END;
```

In this example, `@tableName` is the parameter that you can replace with the name of the table you want to inspect. The `QUOTENAME` function is used to safely quote the table name, preventing SQL

injection attacks. The `sp_executesql` stored procedure is then used to execute the dynamically built query.

Please note that this is a simple example and real-world data warehouse tasks might require more complex dynamic SQL queries. Always be cautious when using dynamic SQL due to the risk of SQL injection attacks. Always use parameterization methods like `sp_executesql` or `QUOTENAME` to sanitize inputs.

6. Exercise: Secure a warehouse in Microsoft Fabric

<https://learn.microsoft.com/en-us/training/modules/secure-data-warehouse-in-microsoft-fabric/6-exercise-secure-data-warehouse>

Exercise: Secure a warehouse in Microsoft Fabric

Completed

Now it's your chance to secure a warehouse in Microsoft Fabric.

In this exercise, you'll learn how to secure a warehouse using the concepts explored in this module.

Note

- You need a Microsoft Fabric trial license with the Fabric preview enabled in your tenant. See [Getting started with Fabric](#) to enable your Fabric trial license.
- This exercise includes optional steps that require a second user account to view the applied security measures. If you're unable to create a second account, you can still complete the exercise using your own account. In that case, please refer to the screenshots provided at the end of each section to see the expected results.

How to create a second user for this exercise

If you're part of an organization that has an Entra or Microsoft 365 tenant:

Work with your identity administrator to create the second user either in [Entra](#) or the [Microsoft 365 admin center](#).

If you're *not* a member of an organization with an Entra or Microsoft 365 tenant:

1. You're unable to sign up for a Fabric trial with your personal email address. You can sign up for a **Microsoft 365 trial** and a new organizational tenant will be created and you'll become the User and Billing administrator of the tenant and can then create users.
2. Create the second user in the **Microsoft 365 admin center** See: **Add users**
3. Enable the Fabric trial using **Getting started with Fabric**.

Launch the exercise and follow the instructions.

Launch Exercise

7. Module assessment

<https://learn.microsoft.com/en-us/training/modules/secure-data-warehouse-in-microsoft-fabric/7-knowledge-check>

Module assessment

Completed

8. Summary

<https://learn.microsoft.com/en-us/training/modules/secure-data-warehouse-in-microsoft-fabric/8-summary>

Summary

Completed

In this module, you learned about Dynamic Data Masking (DDM), Row-Level Security (RLS), Column-level security (CLS), and granular SQL permissions in Fabric warehouses.

The main takeaways from this module include understanding how DDM, RLS, and CLS work and their use cases. DDM operates at the column level, offering full, email, custom text, and random masking types. RLS works by associating a security predicate function with a table. CLS can be implemented by identifying sensitive columns, defining access roles, assigning roles to users, and implementing access control. In addition, you learned about the principle of least privilege, which suggests that users and applications should only be given the permissions necessary to complete their tasks.

For additional reading, you can refer to the following URLs:

- [Create a Warehouse in Microsoft Fabric](#)
- [Security for data warehousing in Microsoft Fabric](#)
- [Share your warehouse and manage permissions](#)

Govern data in Microsoft Fabric with Purview

1. Govern data in the Microsoft Purview hub

<https://learn.microsoft.com/en-us/training/modules/fabric-data-governance-purview/govern-data-purview-hub>

Govern data in the Microsoft Purview hub

Completed

To govern Microsoft Fabric items using Microsoft Purview, you first need to establish a connection between Fabric and Purview. Once connected, the Microsoft Purview hub becomes available within Fabric, offering in-depth analysis and insights into your Fabric items. The Purview Hub acts as a gateway between Fabric and Purview.

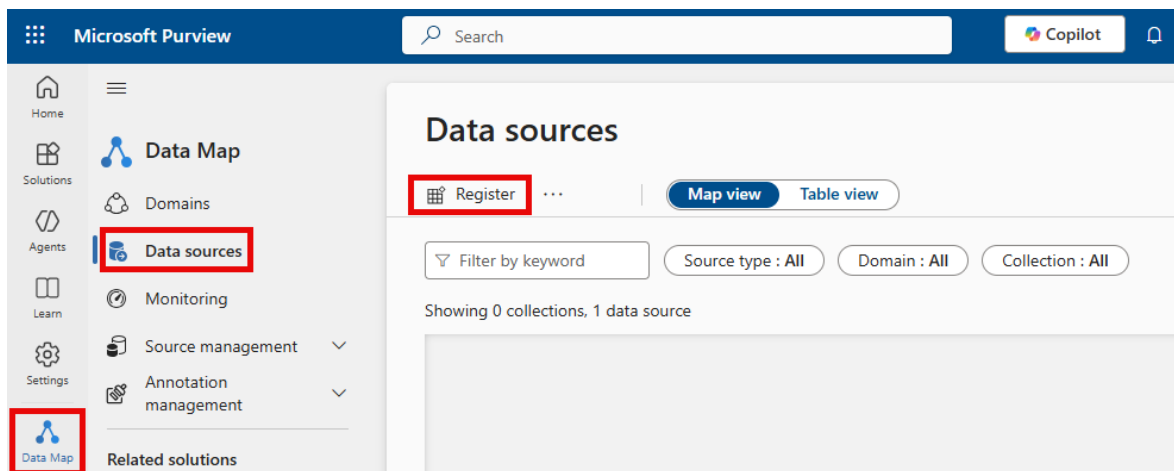
Imagine you're part of a healthcare organization that recently adopted Microsoft Purview. Now, you're ready to connect it to Fabric and explore the insights Purview provides on your Fabric data in the Microsoft Purview Hub.

In this unit, you'll explore the Purview hub.

Connect Purview to Fabric

The stages you must complete to connect Purview to Fabric are:

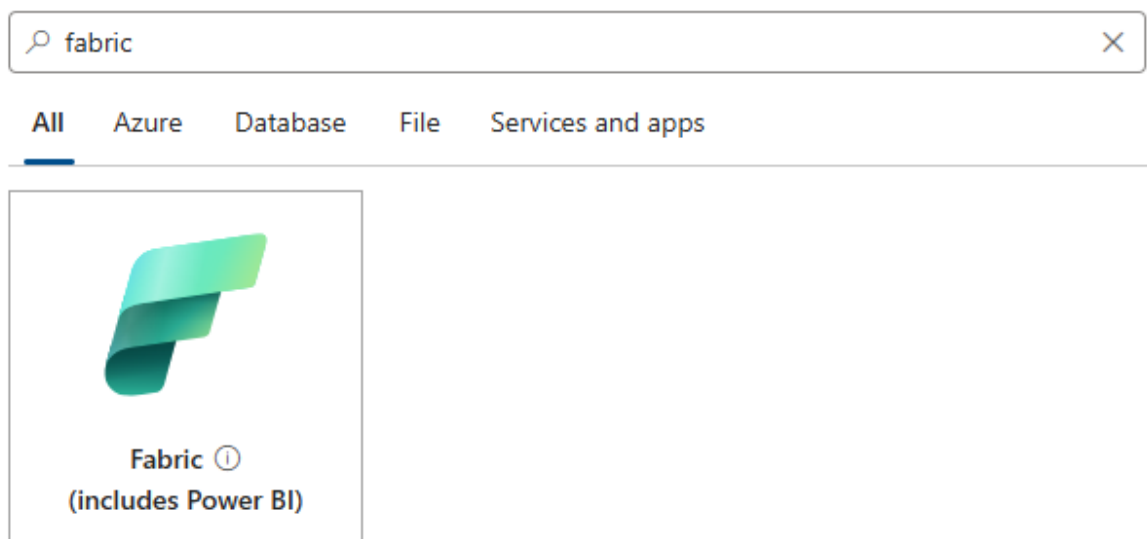
1. In Purview, register your Fabric tenant as a data source from the **Data Map** icon in the left navigation pane:



2. Select Fabric as a data source.

Register data source

Select the type of data source to register. [What data sources are supported?](#)



3. To scan Fabric, Purview needs to authenticate with it. In **Microsoft Entra ID**, configure authentication, using for example, the managed identity of your Purview account.

4. Create a scan in Purview that uses the Fabric data source.

Note

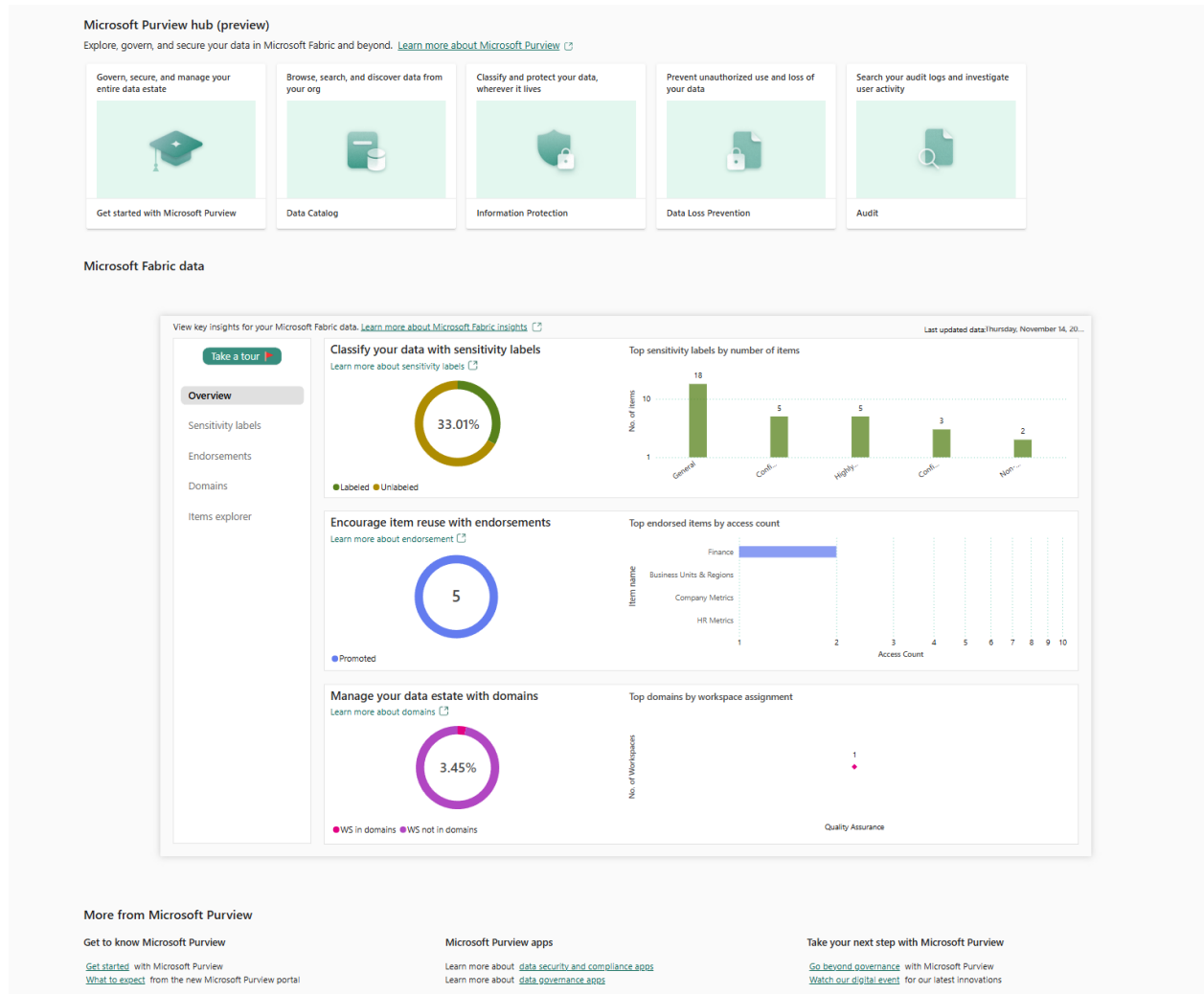
To configure authentication to Fabric from Purview, see

- [Connect to your Microsoft Fabric tenant in the same tenant as Microsoft Purview](#)
- [Connect to your Microsoft Fabric tenant from Microsoft Purview in a different tenant](#)

What is the Microsoft Purview hub in Microsoft Fabric?

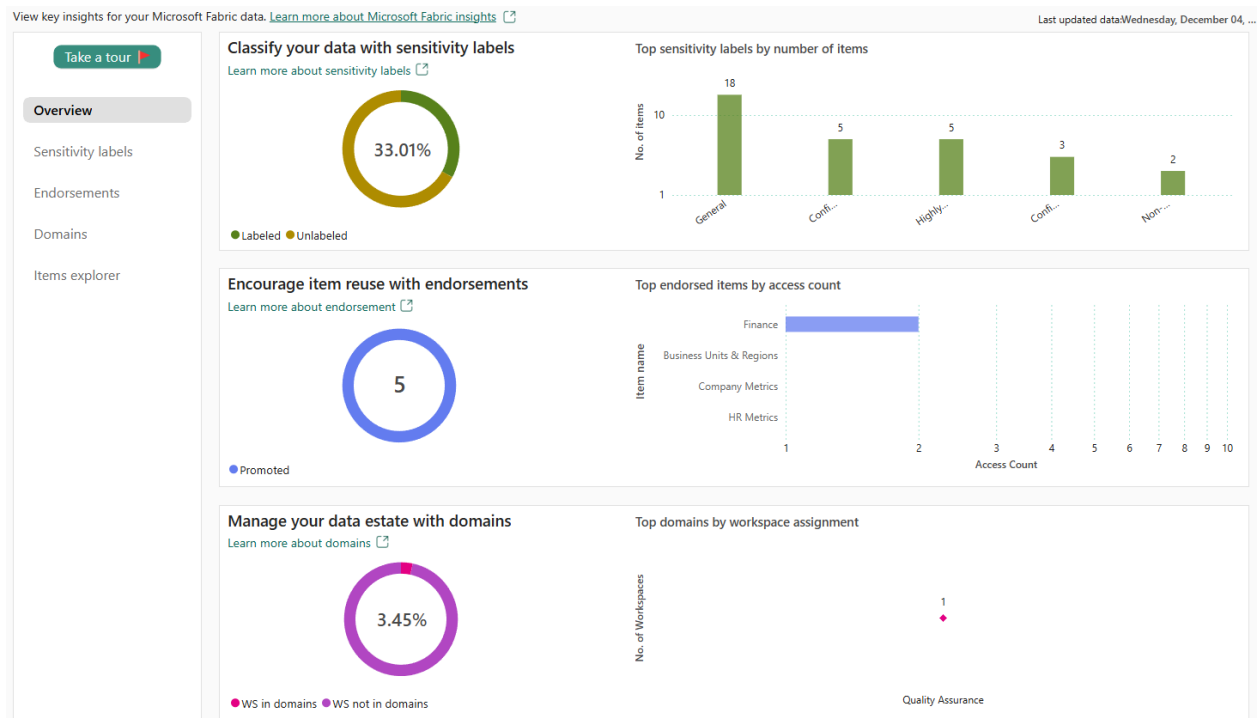
Once Purview and Fabric are connected, the Purview hub in Fabric displays reports and insights about your Fabric items. It acts as a centralized location to begin data governance and acts as a gateway to access more advanced features in Microsoft Purview.

To see the Purview hub, in Microsoft Fabric, open the Fabric settings, and then select **Microsoft Purview hub**.



The tiles at the top of the Purview hub enable you to access advanced governance and compliance capabilities in Purview. The Purview hub report provides dashboards that reflect the security posture of your organization's data estate. Select the following tabs for descriptions of the report pages.

- [Overview](#)
- [Sensitivity labels](#)
- [Endorsement](#)
- [Domains](#)
- [Items explorer](#)



The overview page provides high-level insights about your tenant's data estate. Select the **Take a tour** button in the navigation pane for a quick introduction to the main features of the report.

2. Module assessment

<https://learn.microsoft.com/en-us/training/modules/fabric-data-governance-purview/knowledge-check>

Module assessment

Completed

3. Summary

<https://learn.microsoft.com/en-us/training/modules/fabric-data-governance-purview/summary>

Summary

Completed

In this module, you explored how Microsoft Purview enhances the built-in governance capabilities of Microsoft Fabric to streamline data governance. With tools like the Purview Data Map, Unified Catalog, Information Protection, Data Loss Prevention, and Purview Audit, you can gain comprehensive visibility into your organization's data, safeguard it throughout its lifecycle, manage regulatory and risk requirements, and establish end-to-end governance.

Learn more

- [Microsoft Purview documentation](#)
- [Microsoft Fabric governance documentation](#)
- [Use Microsoft Purview to govern Microsoft Fabric](#)

4. Introduction

<https://learn.microsoft.com/en-us/training/modules/fabric-data-governance-purview/introduction>

Introduction

Completed

Microsoft Fabric is an end-to-end, analytics and data platform that provides a unified solution for data storage, movement, transformation, processing, real-time event routing, and reporting. When working with a large volume of data in Fabric, some of which is highly sensitive, you must ensure that it's accurate, trustworthy, and protected throughout the data lifecycle. Data governance is the set of techniques and tools that you use to implement these requirements.

Microsoft Purview has a data governance solution designed to help you discover, catalog, and manage data assets across your organization. It provides a unified view of your data estate by scanning and cataloging metadata from various data sources, including Microsoft Fabric. Purview has three pillars: data governance, data security, and risk and compliance. In this module, we'll focus on data governance.

Imagine you're in a data stewardship role at a healthcare organization, responsible for managing and safeguarding sensitive patient information. You rely on Microsoft Fabric to store, analyze, manipulate, and report on data, all while ensuring compliance with health data regulations such as HIPAA. Fabric offers built-in data governance capabilities to support these responsibilities. Now, you're looking to explore how integrating Microsoft Purview with Fabric can further enhance your ability to protect sensitive data and maintain regulatory compliance.

In this module, you'll learn how Purview data governance features work with Fabric and how to use them for compliance, security, and discoverability.

5. Govern data in Microsoft Fabric

<https://learn.microsoft.com/en-us/training/modules/fabric-data-governance-purview/govern-data>

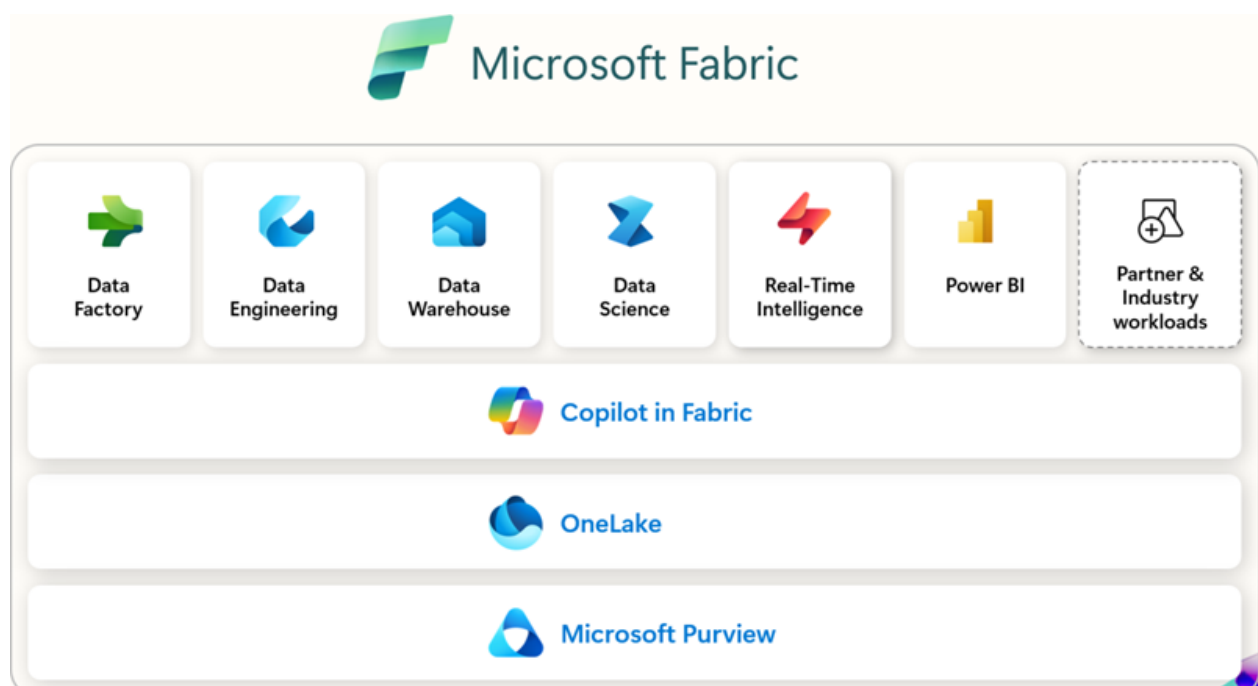
Govern data in Microsoft Fabric

Completed

Microsoft Fabric includes a set of governance and compliance capabilities that might be sufficient for the needs of some organizations. Other governance capabilities can be accessed in Microsoft Purview.

If, for example, you work at a health care company, you have specific data governance needs because of regulatory requirements. In this unit, we'll explore Fabric's governance features so you can assess if they meet your requirements.

What is Microsoft Fabric?



Fabric is Microsoft's end-to-end analytics and data platform. The platform can store, move, process, ingest, transform, and analyze your data regardless of its size and current format. Once you have data in the location and format you need it, advanced business intelligence features can be used to analyze data, and generate insights to support your decision making.

OneLake

In Fabric, all information is stored in a data lake, which can hold both structured and unstructured data. Microsoft's implementation of the data lake is called OneLake and is built on Azure Data Lake Storage (ADLS) generation 2. With all data residing in OneLake, there's a single copy of data that doesn't need to be moved or duplicated. This single underlying storage mechanism helps ensure that policies and security controls are applied universally.

What is data governance?

The data that an organization works with is often business-critical and highly sensitive. Sensitive data must be: held securely, easily available, and processed in compliance with legislation in all the locations where an organization operates.

Data governance is the practice of managing data to ensure the quality, consistency, security, and usability of data. It involves setting up a framework that includes roles, responsibilities, processes, policies, and standards that are used to manage data throughout its lifecycle.

A data governance program usually includes:

- **Auditing:** Recording where data originates and who modifies it.
- **Evaluation:** Assessing the usefulness and accuracy of data.
- **Documentation:** Providing clear descriptions of data to support informed use.
- **Management:** Correcting inaccurate data, fulfilling access requests, and maintaining compliance with data legislation.
- **Protection:** Securing data from unauthorized access, ransomware threats, and other malicious attacks.

Well-governed data is reliable and readily accessible to those who need it.

Tip

In large organizations, data is often stored and managed separately across different business units, creating what are known as data silos. These silos can hinder effective data governance due to inconsistent standards and policies across teams. Breaking down these silos by integrating data—while preserving security and compliance—is essential for establishing a unified and governed data environment.

Implementing data governance can lead to benefits such as:

- A single source of truth that reduces confusion and supports decision making.
- Improved data quality.
- Faster compliance with access requests.
- Reduced data storage and management costs.

Data governance features in Microsoft Fabric

Many data governance tasks can be performed using your Microsoft Fabric license, without the need for an additional Microsoft Purview license. Let's explore some of the built-in data governance capabilities available directly within Fabric.

Managing the data estate

Your organization's data estate encompasses all its data assets. Managing a large data estate can be complex and resource-intensive. Microsoft Fabric simplifies managing your data estate with these tools:

- **The Fabric Admin portal:** control tenant settings, capacities, domains, and other objects, typically reserved for administrators.
- **Tenants, domains, and workspaces:** logical containers that you can use to control access to data and capabilities. Fabric administrators, for example, should have access to all settings in the tenant whereas team level data controllers might only have control of settings on their domain or workspace.
 - Domains group data that is relevant to a single business area or subject field.
 - Workspaces group Fabric items used by a single team or department.
- **Capacities:** These objects limit compute resource usage for all Fabric workloads.
- **Metadata scanning:** Scanning extracts values such as names, identities, sensitivities, endorsements, and so on, from data lakes. You can use this metadata to analyze and set governance policies.

Securing and protecting data

Secure data is safeguarded from unauthorized access and malicious attacks. It's also compliant with data storage regulations applicable in your region. Fabric includes the following tools to secure and protect data:

- **Data tags:** Use tags to identify the sensitivity of data and apply data retentions and protection policies.
- **Workspace roles:** Use roles to define the users who are authorized to access the data in a workspace.
- **Data-level controls:** Use controls at the level of Fabric items such as tables, rows, and columns to impose granular restrictions.
- **Certifications:** Fabric is compliant with many data management certifications, including HIPAA BAA, ISO/IEC 27017, ISO/IEC 27018, ISO/IEC 27001, and ISO/IEC 27701.

Encouraging data discovery and use

Data only becomes valuable when users can find and analyze it. These Microsoft Fabric features help promote data discovery and encourage meaningful use:

- **OneLake data hub:** This tool makes it easy for users to find and explore the data in your estate.
- **Endorsement:** Users endorse a Fabric item to identify it as of high quality. Endorsements help other users to trust the data that the item contains.
- **Data lineage:** This feature helps users to understand the flow of data between items in a workspace and the impact that a change would have.

Monitoring data usage

Fabric helps users track how data is being used through built-in monitoring tools:

- **Monitoring Hub:** Provides a centralized view of Microsoft Fabric activities. Users can only see activity related to Fabric items they have permission to access.
- **Capacity Metrics App:** Offers insights into Fabric usage and resource consumption, helping teams manage capacity effectively.

Learn more

- [What is Microsoft Fabric?](#)
- [What is data governance?](#)
- [Microsoft Fabric governance documentation](#)

6. Why use Microsoft Purview with Microsoft Fabric?

<https://learn.microsoft.com/en-us/training/modules/fabric-data-governance-purview/why-purview-fabric>

Why use Microsoft Purview with Microsoft Fabric?

Completed

When you need data governance features beyond those built into Microsoft Fabric, Microsoft Purview provides comprehensive data governance capabilities.

In the last unit, we reviewed the built-in governance features of Microsoft Fabric. Let's assume you work for a health care provider and new regulations mean you need stricter governance for data stored in Microsoft Fabric. Now, you want to investigate what features Microsoft Purview includes that can help you comply with new regulations.

In this unit, you'll learn what capabilities Microsoft Purview adds to the built-in data governance and compliance capabilities available in Microsoft Fabric.

What is Microsoft Purview?

Microsoft Purview is a set of solutions that you can use to govern, protect, and manage data wherever it resides. It has three pillars: data governance, data security, and risk and compliance. Purview includes data governance tools for data discovery, classification, and cataloging and can be used to gain a clear understanding of your data landscape. Purview automates the identification of sensitive information and provides a centralized repository for metadata. Use it to find, manage, and govern data across various environments, both on-premises and in the cloud.

Purview also supports data security and risk and compliance management with features that monitor regulatory adherence and assess data vulnerabilities. It integrates with other Microsoft services and third-party tools. By streamlining data access controls, enforcing policies, and delivering insights into data lineage, Purview can help you maintain data integrity, comply with regulations, and use data effectively for strategic decision-making.

The Purview solutions can be accessed via the Microsoft Purview Portal.



What are some of the Purview tools?

Microsoft Purview enables you to discover, manage, and protect Fabric data using Purview tools. Purview can be configured to scan your Fabric item and present data governance findings in the Purview Hub located in Fabric or in Purview itself. To unlock the following capabilities, register your Fabric tenant in Microsoft Purview.

Microsoft Purview Data Map

Data Map can be used to scan all of your data assets, to capture metadata about them and to identify sensitive data. It captures metadata about data in existing analytics, software-as-a-service, and operational systems in hybrid, on-premises, and multicloud environments. There's a built-in scanning and classification system that keeps the Data Map updated.

Microsoft Purview Unified Catalog

The Purview Unified Catalog is a searchable catalog of your scanned data where you curate, grant access to and, improve the health of your data.

In the Purview Unified Catalog, you can inventory of all your data assets, their metadata, and their lineage so you can understand the topography of your data estate. You can search for and browse datasets, and view metadata, data lineage, classification, and sensitivity labels. The Unified Catalog promotes collaboration because users can annotate datasets with tags to improve discoverability.

Suppose you work at a health care provider and regulations require you to control who can access patient records. In the Unified Catalog, users and administrators can:

- Discover where patient records are held by searching for keywords.
- Label documents and items as patient records to differentiate them from other, less sensitive information.
- Use access policies to manage self-service access requests to patient records.

Microsoft Purview Information Protection

Information Protection is used to classify, label, and protect sensitive data throughout your organization. By applying customizable sensitivity labels, you can classify records. Then policies can define access controls and enforce encryption. These labels follow the data wherever it goes, whether in emails, documents, or cloud storage, and integrate with other tools like Data Loss Prevention (DLP) to prevent unauthorized sharing. This helps organizations meet compliance requirements while safeguarding data against accidental exposure or malicious threats.

If, for example, you worked as a data steward at a health care provider, you could use Information Protection to:

- Identify data items that contain patient information, even when item titles don't identify them as patient records.
- Train classifiers to spot patient records automatically.
- Protect records with policies to encrypt data and impose Information Rights Management (IRM).

Microsoft Purview Data Loss Prevention (DLP)

DLP helps protect sensitive information with policies that automatically detect, monitor, and control the sharing or movement of sensitive data. Administrators can customize rules to block, restrict, or

alert when sensitive data is transferred to prevent accidental or malicious data leaks.

If you worked at a health care provider, you would need to ensure that users don't share patient records with external parties, either accidentally or maliciously. DLP can prevent such transfers on information labeled as patient records.

Microsoft Purview Audit

Auditing solutions in Purview are used to search for activities performed in Microsoft services by users and admins. User activities such as creating files or accessing Fabric items are automatically logged and appear in the Purview audit log.

Thorough auditing is an important protection for the policies you apply to patient records. For example, if a malicious administrator changes DLP policies so that they can copy a patient record to a USB device, this change is recorded and can be identified during later investigations.

Learn more

- [Use Microsoft Purview to govern Microsoft Fabric](#)
- [Learn about Microsoft Purview](#)