Guidance to Microsoft Fabric Security Best Practices



Microsoft Fabric Security Best Practices

Prepared for

Contoso

1/2/2025

Version 0.9 Draft

Prepared by

**Baihas Issa**

Principal Consultant

Baihas.issa@microsoft.com

Revision and Signoff Sheet

Change Record

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Author | Version | Change Reference |
| December 11, 2024 | Baihas Issa | 0.9 Draft | Draft Version for Review and feedback |
|  |  |  |  |
|  |  |  |  |

Reviewers

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Version Approved | Position | Date |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

[1 Introduction 7](#_Toc185330229)

[1.1 Microsoft Fabric concepts and licenses 7](#_Toc185330230)

[*1.2* *Contoso Fabric POC Architecture Introduction* 8](#_Toc185330231)

[2 Microsoft Fabric Security 8](#_Toc185330232)

[2.1 Microsoft Fabric security fundamentals 9](#_Toc185330233)

[2.1.1 Fabric platform 9](#_Toc185330234)

[2.1.2 Microsoft Fabric Architectural diagram 10](#_Toc185330235)

[2.1.3 Authentication 11](#_Toc185330236)

[2.1.4 Authorization 11](#_Toc185330237)

[2.1.5 Data residency 12](#_Toc185330238)

[2.2 Data handling 12](#_Toc185330239)

[2.2.1 Data at rest 12](#_Toc185330240)

[2.2.2 Data at transit 12](#_Toc185330241)

[2.2.3 Telemetry 13](#_Toc185330242)

[2.3 OneLake 13](#_Toc185330243)

[2.3.1 Workspace security 13](#_Toc185330244)

[2.3.2 Item security 13](#_Toc185330245)

[3 Permission model 13](#_Toc185330246)

[3.1 Workspace roles 14](#_Toc185330247)

[4 Power BI security 16](#_Toc185330248)

[4.1 Data at rest 16](#_Toc185330249)

[4.2 Row-level security (RLS) with Power BI 17](#_Toc185330250)

[4.2.1 Define roles and rules in Power BI Desktop 18](#_Toc185330251)

[4.2.2 Manage security on your semantic model 21](#_Toc185330252)

[4.2.3 Working with members 22](#_Toc185330253)

[4.3 Object-level security (OLS) 26](#_Toc185330254)

[4.3.1 Create a report that uses OLS 27](#_Toc185330255)

[5 Network Security 31](#_Toc185330256)

[5.1 Protect Inbound Traffic 31](#_Toc185330257)

[5.1.1 Entra Conditional Access 33](#_Toc185330258)

[5.1.2 Trusted access 35](#_Toc185330259)

[5.1.3 Private links 35](#_Toc185330260)

[5.2 Outbound 35](#_Toc185330261)

[5.2.1 What are managed virtual networks? 35](#_Toc185330262)

[6 Collaborate, Share, and manage security 37](#_Toc185330263)

[6.1 Security for data warehousing in Microsoft Fabric 37](#_Toc185330264)

[6.2 Warehouse access model 37](#_Toc185330265)

[6.3 Workspace roles 37](#_Toc185330266)

[6.4 Item permissions 38](#_Toc185330267)

[6.5 Granular security 38](#_Toc185330268)

[6.5.1 Object-Level Security 38](#_Toc185330269)

[6.5.2 Row-level security 39](#_Toc185330270)

[6.5.3 Column-level security 39](#_Toc185330271)

[6.5.4 Dynamic data masking 39](#_Toc185330272)

[6.5.5 User audit logs 43](#_Toc185330273)

[7 Microsoft Purview to govern Microsoft Fabric 44](#_Toc185330274)

[7.1 Live view in Microsoft Purview (preview) 44](#_Toc185330275)

[7.2 Microsoft Purview hub for administrators (preview) 48](#_Toc185330276)

[7.2.1 Access to Purview hub for administrators requires the Fabric administrator role or higher 49](#_Toc185330277)

[7.3 Information protection in Microsoft Fabric 49](#_Toc185330278)

[7.3.1 Considerations and limitations 50](#_Toc185330279)

[7.4 Protected sensitivity labels in Fabric and Power BI 52](#_Toc185330280)

[7.4.1 How protected labels work in Fabric and Power BI 52](#_Toc185330281)

[7.4.2 Relaxations to accommodate automatic labeling scenarios 54](#_Toc185330282)

[7.5 Default label policy for Power BI 54](#_Toc185330283)

[7.6 Domain-level default sensitivity labels in Microsoft Fabric 55](#_Toc185330284)

[7.7 Mandatory label policy for Fabric and Power BI 55](#_Toc185330285)

[7.7.1 What happens when a mandatory label policy is in effect? 56](#_Toc185330286)

[7.8 Sensitivity label support for paginated reports 57](#_Toc185330287)

[7.9 Programmatic labeling 57](#_Toc185330288)

[7.10 Sensitivity label downstream inheritance 58](#_Toc185330289)

[7.11 Sensitivity label inheritance from data sources 59](#_Toc185330290)

[7.12 Audit schema for sensitivity labels in Fabric 60](#_Toc185330291)

[8 Appendix A (References) 61](#_Toc185330292)

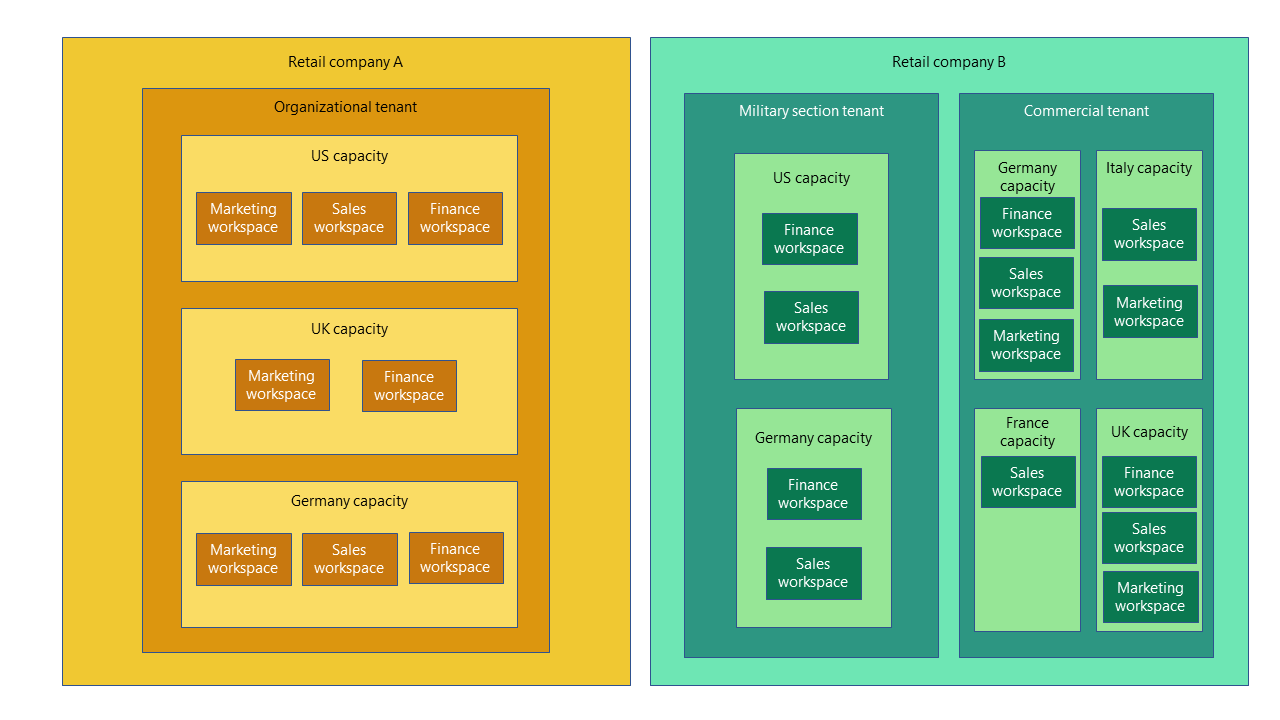
1. Introduction

This document outlines Microsoft Security best practices for Contoso, focusing on Microsoft Fabric Security and Microsoft Purview's data protection, classification, and governance features.

* 1. Microsoft Fabric concepts and licenses

Microsoft Fabric is a platform that allows users to get, create, share, and visualize data using an array of tools. To share content and collaborate in Microsoft Fabric, your organization needs to have an F or P capacity, and at least one per-user license.

A Microsoft Fabric deployment can be organized in various ways according to your organizational needs. This illustration shows two different ways of deploying Fabric in an organization. Retail company A has a single Microsoft Entra tenant for the entire company. Retail company B has two Microsoft Entra tenants, which have complete separation between them, one for military products and another for commercial products. Both companies deployed Fabric capacities according to their geographical location.



**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. Different items consume different capacity at a certain time. 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 right people to have access to the right 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.

* 1. *Contoso Fabric POC Architecture Introduction*

1. Microsoft Fabric Security

Microsoft Fabric is a software as a service (SaaS) platform that lets users get, create, share, and visualize data.

As a SaaS service, Fabric offers a complete security package for the entire platform. Fabric removes the cost and responsibility of maintaining your security solution, and transfers it to the cloud. With Fabric, you can use the expertise and resources of Microsoft to keep your data secure, patch vulnerabilities, monitor threats, and comply with regulations. Fabric also allows you to manage, control and audit your security settings, in line with your changing needs and demands.

As you bring your data to the cloud and use it with various analytic experiences such as Power BI, Data Factory, and the next generation of Synapse, Microsoft ensures that built-in security and reliability features secure your data at rest and in transit. Microsoft also makes sure that your data is recoverable in cases of infrastructure failures or disasters.

**Fabric security is:**

* **Always on** - Every interaction with Fabric is encrypted by default and authenticated using Microsoft Entra ID. All communication between Fabric experiences travels through the Microsoft backbone internet. Data at rest is automatically stored encrypted. To regulate access to Fabric, you can add extra security features such as Private Links or Entra Conditional Access . Fabric can also connect to data protected by a firewall or a private network using trusted access.
* **Compliant** – Fabric has data sovereignty out of the box with multi geo capacities. Fabric also supports a wide range of compliance standards.
* **Governable -** Fabric comes with a set of governance tools such data lineage, information protection labels, data loss prevention and purview integration.
* **Configurable** - You can configure Fabric security in accordance with your organizational policies.
* **Evolving** - Microsoft is constantly improving Fabric security, by adding new features and controls.
  1. Microsoft Fabric security fundamentals

This document presents a big-picture perspective of the Microsoft Fabric security architecture by describing how the main security flows in the system work. It also describes how users authenticate with Fabric, how data connections are established, and how Fabric stores and moves data through the service.

The document is primarily targeted at Fabric administrators, who are responsible for overseeing Fabric in the organization. It's also relevant to enterprise security stakeholders, including security administrators, network administrators, Azure administrators, workspace administrators, and database administrators.

* + 1. Fabric platform

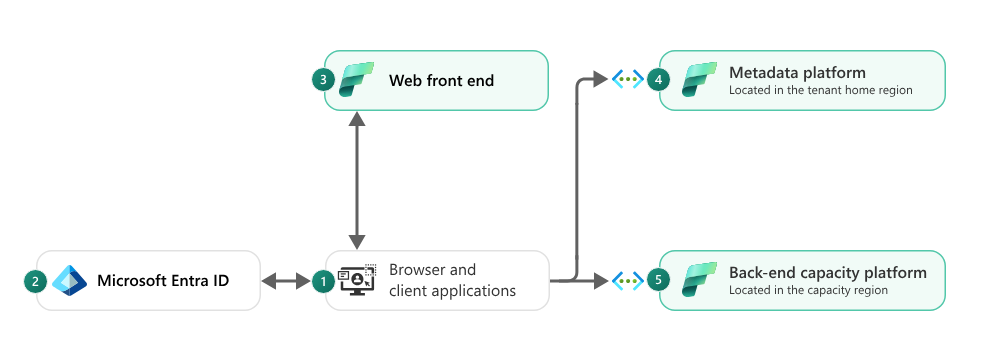
Microsoft Fabric is an all-in-one analytics solution for enterprises that covers everything from data movement to data science, real-time analytics, and business intelligence (BI). The Fabric platform comprises a series of services and infrastructure components that support the common functionality for all Fabric experiences. Collectively, they offer a comprehensive set of analytics experiences designed to work together seamlessly. Experiences include Lakehouse, Data Factory, Fabric Data Engineering, Fabric Data Warehouse, Power BI, and others.

With Fabric, you don't need to piece together different services from multiple vendors. Instead, you benefit from a highly integrated, end-to-end, and easy-to-use product that's designed to simplify your analytics needs. Fabric was designed from the outset to protect sensitive assets.

The Fabric platform is built on a foundation of software as a service (SaaS), which delivers reliability, simplicity, and scalability. It's built on Azure, which is Microsoft's public cloud computing platform. Traditionally, many data products have been platform as a service (PaaS), requiring an administrator of the service to set up security, compliance, and governance for each service. Because Fabric is a SaaS service, many of these features are built into the SaaS platform and require no setup or minimal setup.

* + 1. Microsoft Fabric Architectural diagram

The architectural diagram below shows a high-level representation of the Fabric security architecture.



The architectural diagram depicts the following concepts.

1. **A user uses a browser or a client application**, like Power BI Desktop, to connect to the Fabric service.
2. **Authentication is handled by Microsoft Entra ID**, previously known as Azure Active Directory, which is the cloud-based identity and access management service that authenticates the user or service principal and manages access to Fabric.
3. **The web front** end receives user requests and facilitates sign-in. It also routes requests and serves front-end content to the user.
4. **The metadata platform stores tenant metadata**, which can include customer data. Fabric services query this platform on demand in order to retrieve authorization information and to authorize and validate user requests. It's located in the tenant home region.

Fabric platform infrastructure services are multitenant. There is logical isolation between tenants. These services don't process complex user input and are all written in managed code. Platform services never run any user-written code.

The metadata platform and the back-end capacity platform each run in secured virtual networks. These networks expose a series of secure endpoints to the internet so that they can receive requests from customers and other services. Apart from these endpoints, services are protected by network security rules that block access from the public internet. Communication within virtual networks is also restricted based on the privilege of each internal service.

The application layer ensures that tenants are only able to access data from within their own tenant.

* + 1. Authentication

Fabric relies on Microsoft Entra ID to authenticate users (or service principals). When authenticated, users receive access tokens from **Microsoft Entra ID**. Fabric uses these tokens to perform operations in the context of the user.

A key feature of Microsoft Entra ID is conditional access. **Conditional access** ensures that tenants are secure by enforcing multifactor authentication, allowing only Microsoft Intune enrolled devices to access specific services. Conditional access also restricts user locations and IP ranges

* + 1. Authorization

All Fabric permissions are stored centrally by the metadata platform. Fabric services query the metadata platform on demand in order to retrieve authorization information and to authorize and validate user requests.

For performance reasons, Fabric sometimes encapsulates authorization information into signed tokens. Signed tokens are only issued by the back-end capacity platform, and they include the access token, authorization information, and other metadata.

* + 1. Data residency

In Fabric, a tenant is assigned to a home metadata platform cluster, which is located in a single region that meets the data residency requirements of that region's geography. Tenant metadata, which can include organizations data, is stored in this cluster.

Organizations can control where their workspaces are located. They can choose to locate their workspaces in the same geography as their metadata platform cluster, either explicitly by assigning their workspaces on capacities in that region or implicitly by using Fabric Trial, Power BI Pro, or Power BI Premium Per User license mode. In the latter case, all customer data is stored and processed in this single geography. For more information, see Microsoft Fabric concepts and licenses.

Organizations can also create Multi-Geo capacities located in geographies (geos) other than their home region. In this case, compute and storage (including OneLake and experience-specific storage) is located in the multi-geo region, however the tenant metadata remains in the home region. Customer data will only be stored and processed in these two geographies. For more information, see [**Configure Multi-Geo support for Fabric**](https://learn.microsoft.com/en-us/fabric/admin/service-admin-premium-multi-geo?tabs=power-bi-premium).

* 1. Data handling

This section provides an overview of how data handling works in Fabric. It describes storage, processing, and the movement of customer data.

* + 1. Data at rest

All Fabric data stores are encrypted at rest by using Microsoft-managed keys. Fabric data includes customer data as well as system data and metadata.

While data can be processed in memory in an unencrypted state, it's never persisted to permanent storage while in an unencrypted state.

* + 1. Data at transit

Data in transit between Microsoft services is always encrypted with at least TLS 1.2. Fabric negotiates to TLS 1.3 whenever possible. Traffic between Microsoft services always routes over the Microsoft global network.

Inbound Fabric communication also enforces TLS 1.2 and negotiates to TLS 1.3, whenever possible. Outbound Fabric communication to customer-owned infrastructure prefers secure protocols but might fall back to older, insecure protocols (including TLS 1.0) when newer protocols aren't supported.

* + 1. Telemetry

Telemetry is used to maintain performance and reliability of the Fabric platform. The Fabric platform telemetry store is designed to be compliant with data and privacy regulations for customers in all regions where Fabric is available, including the European Union (EU).

* 1. OneLake

OneLake is a single, unified, logical data lake for the entire organization, and it's automatically provisioned for every Fabric tenant. It's built on Azure and it can store any type of file, structured or unstructured. Also, all Fabric items, like warehouses and lakehouses, automatically store their data in OneLake. OneLake supports the same Azure Data Lake Storage Gen2 (ADLS Gen2) APIs and SDKs, therefore it's compatible with existing ADLS Gen2 applications, including Azure Databricks.

* + 1. Workspace security

Workspaces represent the primary security boundary for data stored in OneLake. Each workspace represents a single domain or project area where teams can collaborate on data. You manage security in the workspace by assigning users to workspace roles.

* + 1. Item security

Within a workspace, you can assign permissions directly to Fabric items, like warehouses and lakehouses. Item security provides the flexibility to grant access to an individual Fabric item without granting access to the entire workspace. Users can set up per item permissions either by sharing an item or by managing the permissions of an item.

1. Permission model

Microsoft Fabric has a flexible permission model that allows you to control access to data in your organization.

A workspace is a logical entity for grouping items in Fabric. Workspace roles define access permissions for workspaces. Although items are stored in one workspace, they can be shared with other users across Fabric. When you share Fabric items, you can decide which permissions to grant the user you're sharing the item with. Certain items such as Power BI reports, allow even more granular control of data. Reports can be set up so that depending on their permissions, users who view them only see a portion of the data they hold.

* 1. Workspace roles

Workspace roles are used to control access to workspaces and the content within them. A Fabric administrator can assign workspace roles to individual users or groups. Workspace roles are confined to a specific workspace and don't apply to other workspaces, the capacity the workspace is in, or the tenant.

There are four Workspace roles and they apply to all items within the workspace. Users that don't have any of these roles, can't access the workspace. The roles are:

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

Below table shows

a small set of the capabilities each role has.

|  |
| --- |
|  |
| **Capability** | **Admin** | **Member** | **Contributor** | **Viewer** |
| Update and delete the workspace. | ✅ |  |  |  |
| Add or remove people, including other admins. | ✅ |  |  |  |
| Add members or others with lower permissions. | ✅ | ✅ |  |  |
| Allow others to reshare items.1 | ✅ | ✅ |  |  |
| Create or modify database mirroring items. | ✅ | ✅ |  |  |
| Create or modify warehouse items. | ✅ | ✅ |  |  |
| Create or modify SQL database items. | ✅ | ✅ |  |  |
| View and read content of data pipelines, notebooks, Spark job definitions, ML models and experiments, and eventstreams. | ✅ | ✅ | ✅ | ✅ |
| View and read content of KQL databases, KQL query-sets, and real-time dashboards. | ✅ | ✅ | ✅ | ✅ |
| Connect to SQL analytics endpoint of Lakehouse or the Warehouse | ✅ | ✅ | ✅ | ✅ |
| Read Lakehouse and Data warehouse data and shortcuts2 with T-SQL through TDS endpoint. | ✅ | ✅ | ✅ | ✅ |
| Read Lakehouse and Data warehouse data and shortcuts2 through OneLake APIs and Spark. | ✅ | ✅ | ✅ |  |
| Read Lakehouse data through Lakehouse explorer. | ✅ | ✅ | ✅ |  |
| Write or delete data pipelines, notebooks, Spark job definitions, ML models, and experiments, and eventstreams. | ✅ | ✅ | ✅ |  |
| Write or delete Eventhouses3, KQL Querysets, Real-Time Dashboards, and schema and data of KQL Databases, Lakehouses, data warehouses, and shortcuts. | ✅ | ✅ | ✅ |  |
| Execute or cancel execution of notebooks, Spark job definitions, ML models, and experiments. | ✅ | ✅ | ✅ |  |
| Execute or cancel execution of data pipelines. | ✅ | ✅ | ✅ |  |
| View execution output of data pipelines, notebooks, ML models and experiments. | ✅ | ✅ | ✅ | ✅ |
| Schedule data refreshes via the on-premises gateway.4 | ✅ | ✅ | ✅ |  |
| Modify gateway connection settings.4 | ✅ | ✅ | ✅ |  |

**Notes:**

* 1 Contributors and Viewers can also share items in a workspace, if they have Reshare permissions.
* 2 Other permissions are needed to read data from shortcut destination. Learn more about [shortcut security model.](https://learn.microsoft.com/en-us/fabric/onelake/onelake-shortcuts?#types-of-shortcuts)
* 3 Other permissions are needed to perform certain operations on data in an Eventhouse. Learn more about the [hybrid role-based access control model](https://learn.microsoft.com/en-us/kusto/access-control/role-based-access-control?view=microsoft-fabric&preserve-view=true).
* 4 Keep in mind that you also need permissions on the gateway. Those permissions are managed elsewhere, independent of workspace roles and permissions.
* Other permissions are needed to perform certain operations on data in an Eventhouse. Learn more about the [hybrid role-based access control model](https://learn.microsoft.com/en-us/kusto/access-control/role-based-access-control?view=microsoft-fabric&preserve-view=true).
* Keep in mind that you also need permissions on the gateway. Those permissions are managed elsewhere, independent of workspace roles and permissions.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Workspace ID** | **Admin** | **Member** | **Contributor** | **Viewer** | **Notes** |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

1. Power BI security

Power BI is an online software service (SaaS, or Software as a Service) offering as part of Microsoft Fabric that lets you easily and quickly create self-service Business Intelligence dashboards, reports, semantic models, and visualizations. With Power BI, you can connect to many different data sources, combine, and shape data from those connections, then create reports and dashboards that can be shared with others.

.

* 1. Data at rest

Power BI uses two primary data storage resource types:

* Azure Storage
* Azure SQL Databases

In most scenarios, Azure Storage is utilized to persist the data of Power BI artifacts, while Azure SQL Databases are used to persist artifact metadata.

All data persisted by Power BI is encrypted by default using Microsoft-managed keys. Customer data stored in Azure SQL Databases is fully encrypted using Azure SQL's Transparent Data Encryption (TDE) technology. Customer data stored in Azure storage is encrypted using Azure Storage Encryption.

Optionally, organizations can utilize Power BI Premium to use their own keys to encrypt data at rest that is imported into a semantic model. This approach is often described as bring your own key (BYOK). Utilizing BYOK helps ensure that even in case of a service operator error, customer data won't be exposed – something that can't easily be achieved using transparent service-side encryption. See Bring your own encryption keys for Power BI for more information.

Power BI semantic models allow for various data source connection modes that determine whether the data source data is persisted in the service or not.

|  |  |
| --- | --- |
| **Semantic Model Mode (Kind)** | **Data Persisted in Power BI** |
| Import | Yes |
| DirectQuery | No |
| Live Connect | No |
| DirectLake | No (stored in OneLake) |
| Composite | If contains an Import data source |
| Streaming | If configured to persist |

Regardless of the semantic model mode utilized, Power BI may temporarily cache any retrieved data to optimize query and report load performance.

* 1. Row-level security (RLS) with Power BI

Row-level security (RLS) with Power BI can be used to restrict data access for given users. Filters restrict data access at the row level, and you can define filters within roles. In the Power BI service, users with access to a workspace have access to semantic models in that workspace. RLS only restricts data access for users with Viewer permissions. It doesn't apply to Admins, Members, or Contributors.

You can configure RLS for data models imported into Power BI with Power BI. You can also configure RLS on semantic models that are using DirectQuery, such as SQL Server. For Analysis Services or Azure Analysis Services live connections, you configure row-level security in the model, not in Power BI. The security option doesn't show up for live connection semantic models.

* + 1. Define roles and rules in Power BI Desktop

You can define roles and rules within Power BI Desktop. With this editor, you can toggle between using the default drop-down interface and a DAX interface. When you publish to Power BI, you also publish the role definitions.

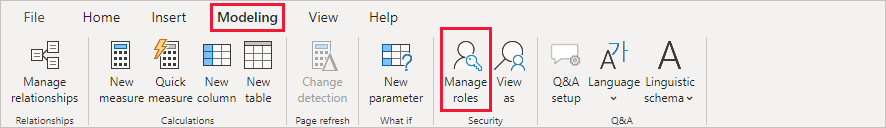
To define security roles:

1. Import data into your Power BI Desktop report or configure a DirectQuery connection.

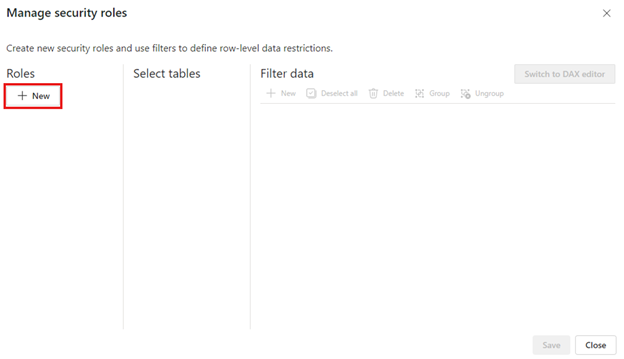
**Note:**

You can't define roles within Power BI Desktop for Analysis Services live connections. You need to do that within the Analysis Services model.

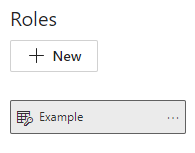
1. From the Modeling tab, select Manage Roles.



1. From the Manage roles window, select New to create a new role.



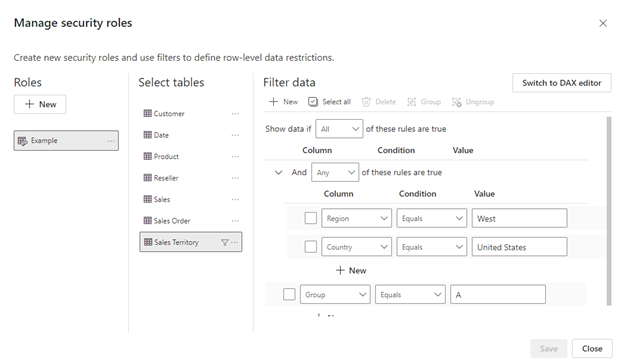
1. Under Roles, provide a name for the role and select enter.



**Note:**

You can't define a role with a comma, for example *London,ParisRole*.

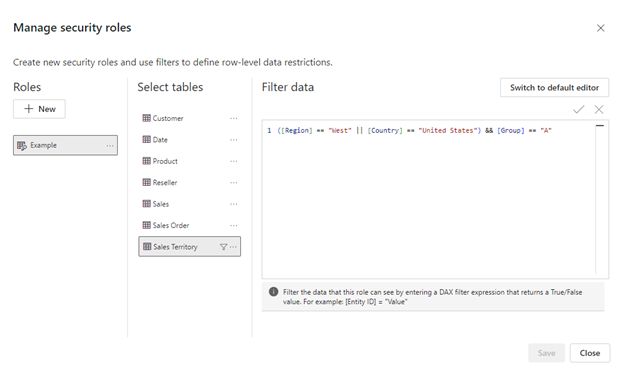
1. Under Select tables, select the table to which you want to apply a row-level security filter.
2. Under Filter data, use the default editor to define your roles. The expressions created return a true or false value.



**Note:**

Not all row-level security filters supported in Power BI can be defined using the default editor. Limitations include expressions that today can only be defined using DAX including dynamic rules such as *username()* or *userprincipalname()*. To define roles using these filters switch to use the DAX editor.

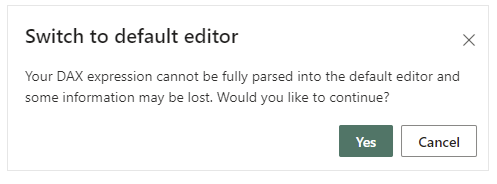
1. Optionally select Switch to DAX editor to switch to using the DAX editor to define your role. DAX expressions return a value of true or false. For example: [Entity ID] = “Value”. The DAX editor is complete with autocomplete for formulas (intellisense). You can select the checkmark above the expression box to validate the expression and the X button above the expression box to revert changes.



**Note:**

You can use username() within this expression. Be aware that username() has the format of DOMAIN\username within Power BI Desktop. Within the Power BI service and Power BI Report Server, it's in the format of the user's User Principal Name (UPN). Additionally, in this expression box, use commas to separate DAX function arguments even if you're using a locale that normally uses semicolon separators, such as French or German.

1. You can switch back to the default editor by selecting Switch to default editor. All changes made in either editor interface persist when switching interfaces when possible. When defining a role using the DAX editor that can't be defined in the default editor, if you attempt to switch to the default editor you'll be prompted with a warning that switching editors may result in some information being lost. To keep this information, select Cancel and continue only editing this role in the DAX editor.



**Note:**

In this expression box, use commas to separate DAX function arguments even if you're using a locale that normally uses semicolon separators, such as French or German.

1. Select **Save.**

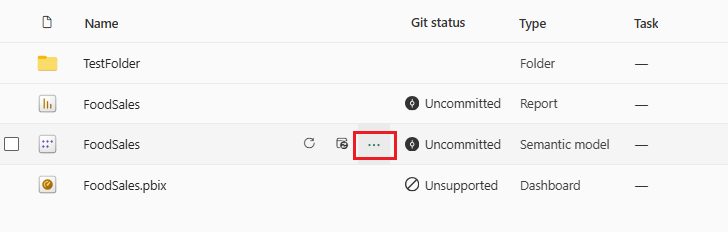
You can't assign users to a role within Power BI Desktop. You assign them in the Power BI service. You can enable dynamic security within Power BI Desktop by making use of the *username()* or *userprincipalname()* DAX functions and having the proper relationships configured.

By default, row-level security filtering uses single-directional filters, whether the relationships are set to single direction or bi-directional. You can manually enable bi-directional cross-filtering with row-level security by selecting the relationship and checking the **Apply security filter in both directions’** checkbox. Note that if a table takes part in multiple bi-directional relationships, you can only select this option for one of those relationships. Select this option when you've also implemented dynamic row-level security at the server level, where row-level security is based on username or login ID.

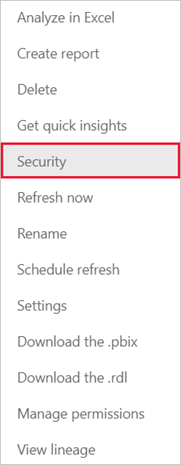
* + 1. Manage security on your semantic model

To manage security on your semantic model, open the workspace where you saved your semantic model in Fabric and do the following steps:

1. In Fabric, select the More options menu for a semantic model. This menu appears when you hover on a semantic model name.



1. Select Security.



Security takes you to the Role-Level Security page where you add members to a role you created. Contributor (and higher workspace roles) will see Security and can assign users to a role.

* + 1. Working with members
       1. Add members

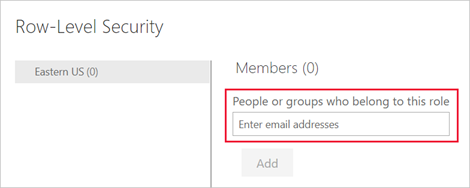
In the Power BI service, you can add a member to the role by typing in the email address or name of the user or security group. You can't add Groups created in Power BI. You can add members external to your organization.

You can use the following groups to set up row-level security.

* **Distribution Group**
* **Mail-enabled Group**
* **Microsoft Entra Security Group**

**Note:**

Microsoft 365 groups aren't supported and can't be added to any roles.

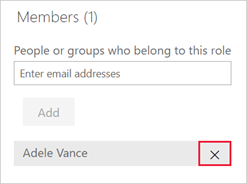


You can also see how many members are part of the role by the number in parentheses next to the role name, or next to Members.



* + - 1. Remove members

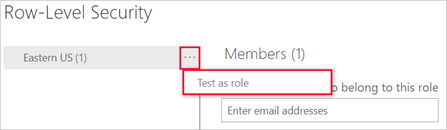
You can remove members by selecting the X next to their name.



* + - 1. Validating the role within the Power BI service

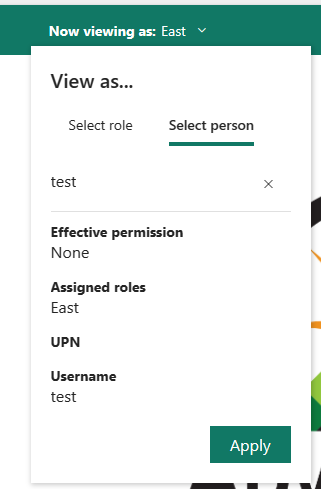
You can validate that the role you defined is working correctly in the Power BI service by testing the role.

1. Select **More options** (...) next to the role.
2. Select **Test as role**.

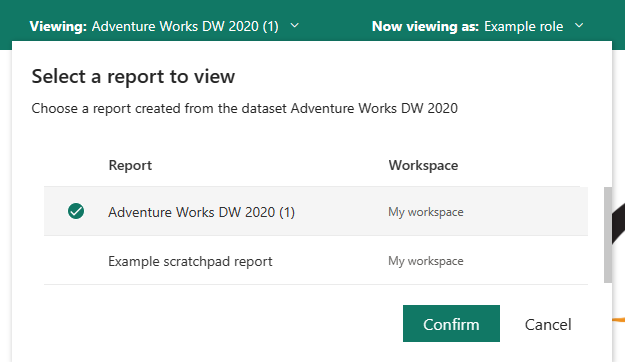


You're redirected to the report that was published from Power BI Desktop with this semantic model, if it exists. Dashboards aren't available for testing using the Test as role option.

In the page header, the role being applied is shown. Test other roles, a combination of roles, or a specific person by selecting Now viewing as. Here you see important permissions details pertaining to the individual or role being tested. For more information about how permissions interact with RLS, see RLS user experience.



Test other reports connected to the semantic model by selecting Viewing in the page header. You can only test reports located in the same workspace as your semantic model.



To return to normal viewing, select Back to Row-Level Security.

**Note:**

The Test as role feature doesn't work for DirectQuery models with Single Sign-On (SSO) enabled. Additionally, not all aspects of a report can be validated in the Test as role feature including Q&A visualizations, Quick insights visualizations, and Copilot.

* + - 1. Using the username() or userprincipalname() DAX function

You can take advantage of the DAX functions *username()* or *userprincipalname()* within your dataset. You can use them within expressions in Power BI Desktop. When you publish your model, it will be used within the Power BI service.

Within Power BI Desktop, *username()* will return a user in the format of *DOMAIN\User* and *userprincipalname()* will return a user in the format of *user@contoso.com*.

Within the Power BI service, *username()* and *userprincipalname()* will both return the user's User Principal Name (UPN). This looks similar to an email address.

* + - 1. Using RLS with workspaces in Power BI

If you publish your Power BI Desktop report to a workspace in the Power BI service, the RLS roles are applied to members who are assigned to the Viewer role in the workspace. Even if Viewers are given Build permissions to the semantic model, RLS still applies. For example, if Viewers with Build permissions use Analyze in Excel, their view of the data is restricted by RLS. Workspace members assigned Admin, Member, or Contributor have edit permission for the semantic model and, therefore, RLS doesn’t apply to them. If you want RLS to apply to people in a workspace, you can only assign them the Viewer role. Read more about roles in workspaces.

* + - 1. Considerations and limitations

You can see the current limitations for row-level security on cloud models here:

* If you previously defined roles and rules in the Power BI service, you must re-create them in Power BI Desktop.
* You can define RLS only on the semantic models created with Power BI Desktop. If you want to enable RLS for semantic models created with Excel, you must convert your files into Power BI Desktop (PBIX) files first. [Learn more](https://learn.microsoft.com/en-us/power-bi/connect-data/desktop-import-excel-workbooks).
* Service principals can't be added to an RLS role. Accordingly, RLS isn't applied for apps using a service principal as the final effective identity.
* Only Import and DirectQuery connections are supported. Live connections to Analysis Services are handled in the on-premises model.
* The Test as role/View as role feature doesn't work for DirectQuery models with single sign-on (SSO) enabled.
* The Test as role/view as role feature shows only reports from semantic models workspace.
* The Test as role/View as role feature doesn't work for paginated reports.

Keep in mind that if a Power BI report references a row with RLS configured then the same message displays as for a deleted or non-existing field. To these users, it looks like the report is broken.

* 1. Object-level security (OLS)

Object-level security (OLS) enables model authors to secure specific tables or columns from report viewers. For example, a column that includes personal data can be restricted so that only certain viewers can see and interact with it. In addition, you can also restrict object names and metadata. This added layer of security prevents users without the appropriate access levels from discovering business critical or sensitive personal information like employee or financial records. For viewers that don’t have the required permission, it's as if the secured tables or columns don't exist.

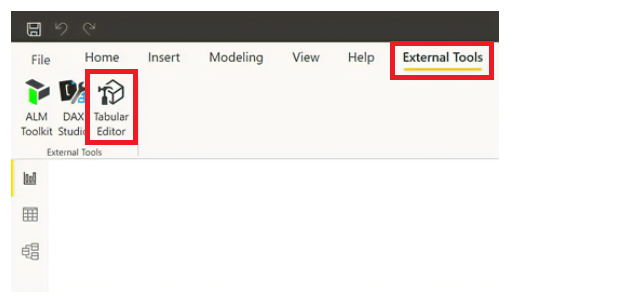
* + 1. Create a report that uses OLS

Like RLS, OLS is also defined within model roles. Currently, you can't create OLS definitions natively in Power BI Desktop.

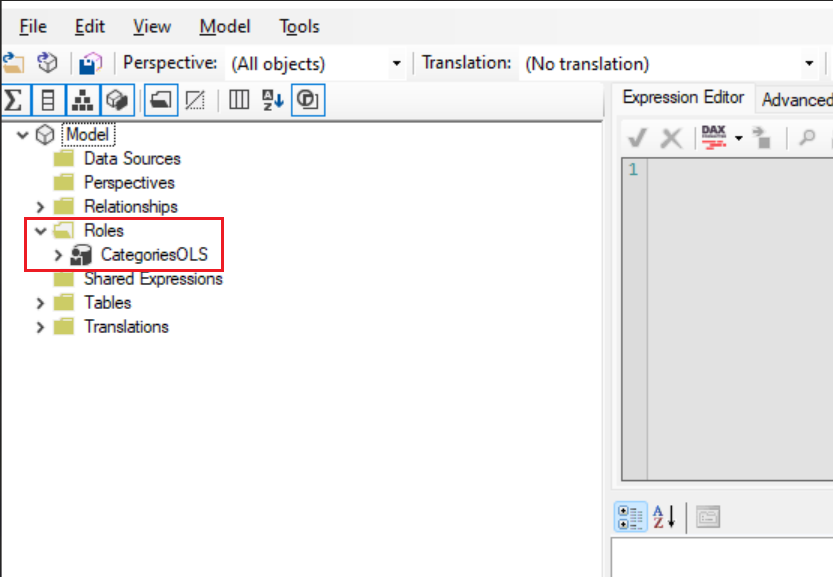
To create roles on Power BI Desktop semantic models, use external tools such as Tabular Editor.

* + - 1. Configure object-level security using tabular editor

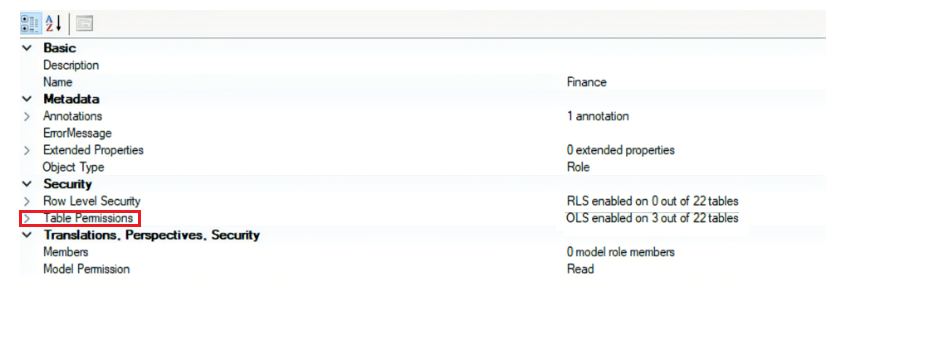
1. In Power BI Desktop, create the model and roles that will define your OLS rules.
2. On the External Tools ribbon, select Tabular Editor. If you don’t see the Tabular Editor button, install the program. When open, Tabular Editor will automatically connect to your model.



1. In the Model view, select the drop-down menu under Roles. The roles you created in step one will appear.



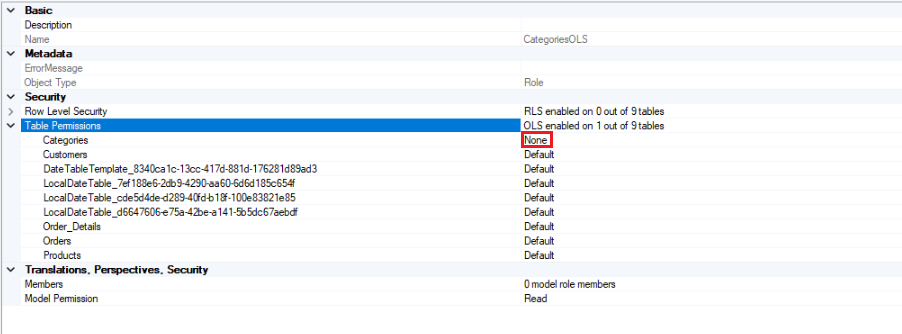
1. Select the role you want to enable an OLS definition for, and expand the Table Permissions.



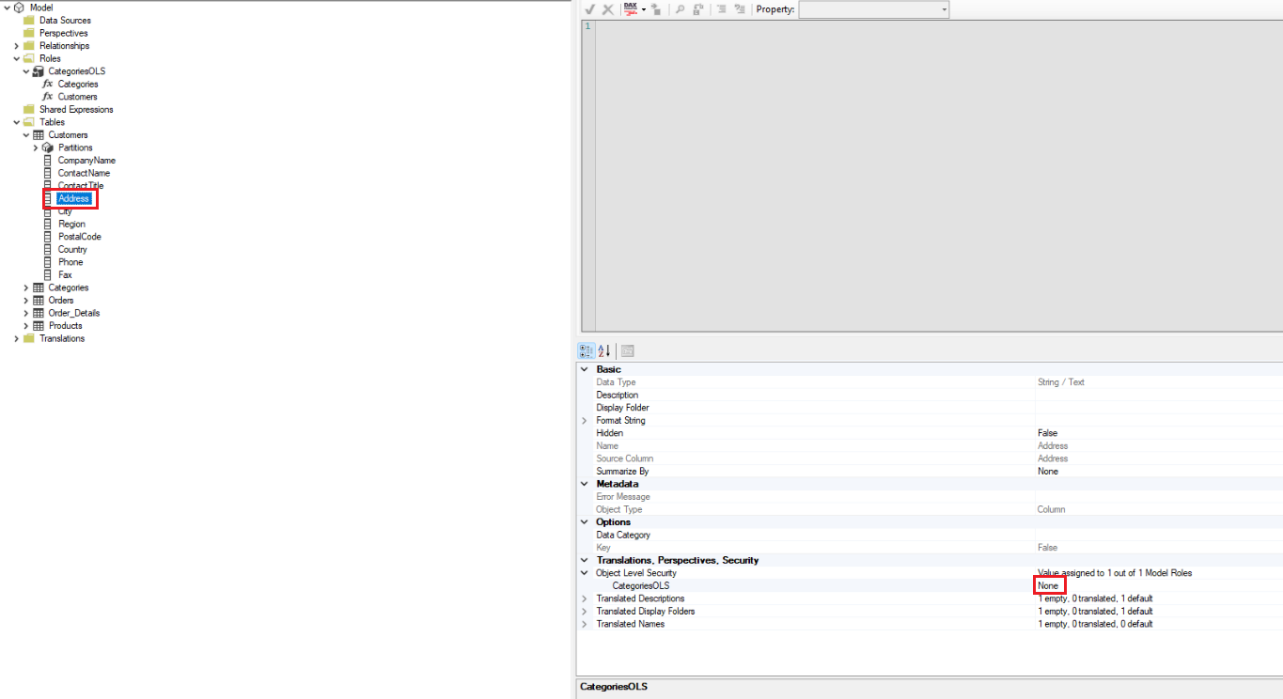
1. Set the permissions for the table or column to *None* or *Read*.

**None**: OLS is enforced and the table or column will be hidden from that role  
**Read**: The table or column will be visible to that role

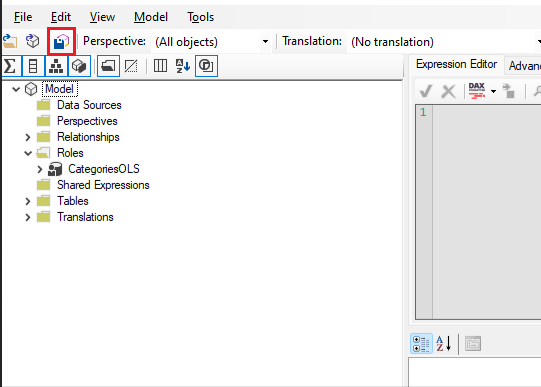
* To secure the whole table
  + Set categories under Table *permissions to None*



* To secure specific columns
  + Select the category and set the *Object Level Security to None*

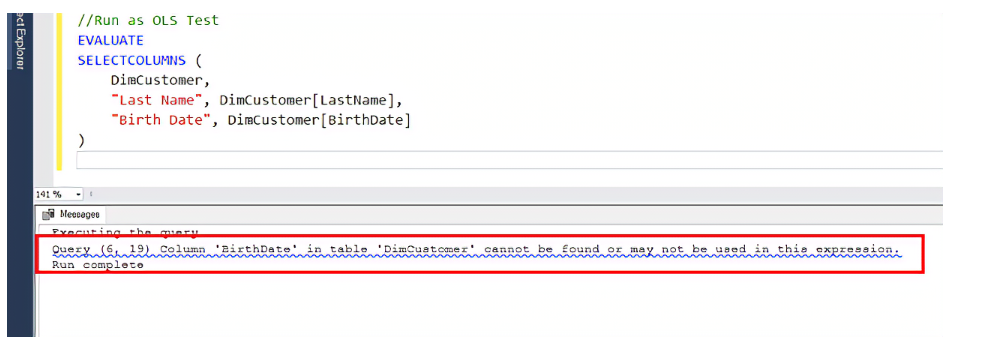


1. After you define object-level security for the roles, save your changes.



1. In Power BI Desktop, publish your semantic model to the Power BI Service.
2. In Power BI Desktop, publish your semantic model to the Power BI Service.

The OLS rules are now defined. Users without the required permission will receive a message that the field can't be found for all report visuals using that field.

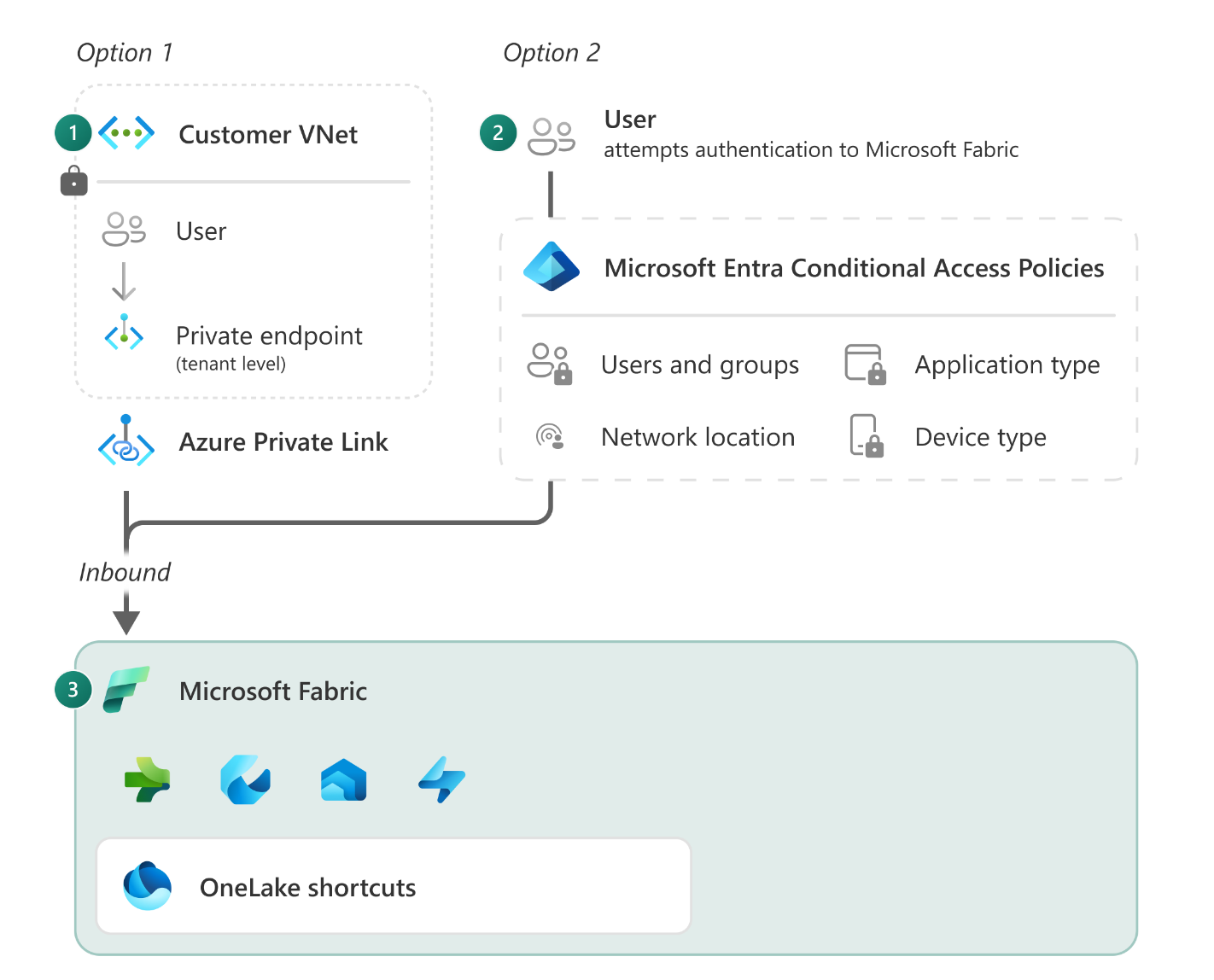


* + - 1. Considerations and limitations
* OLS only applies to Viewers in a workspace. Workspace members assigned Admin, Member, or Contributor have edit permission for the semantic model and, therefore, OLS doesn’t apply to them. Read more about [roles in workspaces](https://learn.microsoft.com/en-us/power-bi/collaborate-share/service-roles-new-workspaces).
* Semantic models with OLS configured for one or more table or column objects aren't supported with these Power BI features:
  + Quick insights visualizations
  + Smart narrative visualizations
  + Excel Data Types gallery
* [See other OLS restrictions](https://learn.microsoft.com/en-us/analysis-services/tabular-models/object-level-security#restrictions)

1. Network Security
   1. Protect Inbound Traffic

Inbound traffic is traffic coming into Fabric from the internet. This document explains the differences between the two ways to protect inbound traffic in Microsoft Fabric, *private links* and *Entra Conditional Access*. **Private links** (Option 1, Customer Vnet) - Fabric uses a private IP address from your virtual network. The endpoint allows users in your network to communicate with Fabric over the private IP address using private links.

* **Entra Conditional Access** - (Option 2, User) - When a user authenticates access is determined based on a set of policies that might include IP address, location, and managed devices.



Once traffic enters Fabric, it gets authenticated by Microsoft Entra ID, which is the same authentication method used by Microsoft 365, OneDrive, and Dynamics 365. Microsoft Entra ID authentication allows users to securely connect to cloud applications from any device and any network, whether they're at home, remote, or in their corporate office.

The Fabric backend platform is protected by a virtual network and isn't directly accessible from the public internet other than through secure endpoints.

By default, Fabric communicates between experiences using the internal Microsoft backbone network. When a Power BI report loads data from OneLake, the data goes through the internal Microsoft network. This configuration is different from having to set up multiple Platform as a Service (PaaS) services to connect to each other over a private network. Inbound communication between clients such as your browser or SQL Server Management Studio (SSMS) and Fabric, uses the TLS 1.2 protocol and negotiates to TLS 1.3 when possible.

* Fabric's default security settings include:
* Microsoft Entra ID which is used to authenticate every request.
* Upon successful authentication, requests are routed to the appropriate backend service through secure Microsoft managed endpoints.
* Internal traffic between experiences in Fabric is routed over the Microsoft backbone.
* Traffic between clients and Fabric is encrypted using at least the Transport Layer Security (TLS) 1.2 protocol.
  + 1. Entra Conditional Access

Every interaction with Fabric is authenticated with Microsoft Entra ID. Microsoft Entra ID is based upon the Zero Trust security model, which assumes that you're not fully protected within your organization's network perimeter. Instead of looking at your network as a security boundary, Zero Trust looks at identity as the primary perimeter for security.

To determine access at the time of authentication you can define and enforce conditional access policies based on your users' identity, device context, location, network, and application sensitivity. For example, you can require multifactor authentication, device compliance, or approved apps for accessing your data and resources in Fabric. You can also block or limit access from risky locations, devices, or networks.

Conditional access policies help you protect your data and applications without compromising user productivity and experience. Here are a few examples of access restrictions you can enforce using conditional access.

* Define a list of IPs for inbound connectivity to Fabric.
* Use Multifactor Authentication (MFA).
* Restrict traffic based on parameters such as country of origin or device type.

**Note:**

Fabric doesn't support other authentication methods such as account keys or SQL authentication, which rely on usernames and passwords.

To ensure that conditional access for Fabric works as intended and expected, it's recommended to adhere to the following best practices:

Configure a single, common, conditional access policy for the Power BI Service, Azure Data Explorer, Azure SQL Database, and Azure Storage. Having a single, common policy significantly reduces unexpected prompts that might arise from different policies being applied to downstream services, and the consistent security posture provides the best user experience in Microsoft Fabric and its related products.

The products to include in the policy are the following:

**Product**

* Power BI Service
* Azure Data Explorer
* Azure SQL Database
* Azure Storage

**Note:**

If you create a restrictive policy (such as one that blocks access for all apps except Power BI), certain features, such as dataflows, won't work.

The following steps show how to configure a conditional access policy for Microsoft Fabric:

1. Sign in to the Azure portal as at least a [Conditional Access Administrator](https://learn.microsoft.com/en-us/entra/identity/role-based-access-control/permissions-reference#conditional-access-administrator).
2. Select **Microsoft Entra ID**.
3. On the Overview page, choose **Security** from the menu.
4. On the Security | Getting started page, choose **Conditional Access**.
5. On the Conditional Access | Overview page, select **+Create new policy**.
6. Provide a name for the policy.
7. Under **Assignments**, select the **Users** field. Then, on the Include tab, choose **Select users and groups**, and then check the **Users and groups** checkbox. The **Select users and groups** pane opens, and you can search for and select a Microsoft Entra user or group for conditional access. When done, click **Select**.
8. Place your cursor in the **Target resources** field and choose **Cloud apps** from the drop-down menu. Then, on the Include tab, choose **Select apps** and place your cursor in the **Select** field. In the **Select** side pane that appears, find and select **Power BI Service**, **Azure Data Explorer**, **Azure SQL Database**, and **Azure Storage**. When you've selected all four items, close the side pane by clicking **Select**.
9. Under **Access controls**, put your cursor in the **Grant** field. In the **Grant** side pane that appears, configure the policy you want to apply, and then click **Select**.
10. Set the **Enable policy** toggle to **On**, then select **Create**.
    * 1. Trusted access

Fabric doesn't need to reside in your private network, even when you have your data stored inside one. With PaaS services, it's common to put the compute in the same private network as the storage account. However, with Fabric this isn't needed. To enable trusted access into Fabric, you can use features such as on-premises Data gateways, Trusted workspace access and managed private endpoints.

* + 1. Private links

With private endpoints your service is assigned a private IP address from your virtual network. The endpoint allows other resources in the network to communicate with the service over the private IP address.

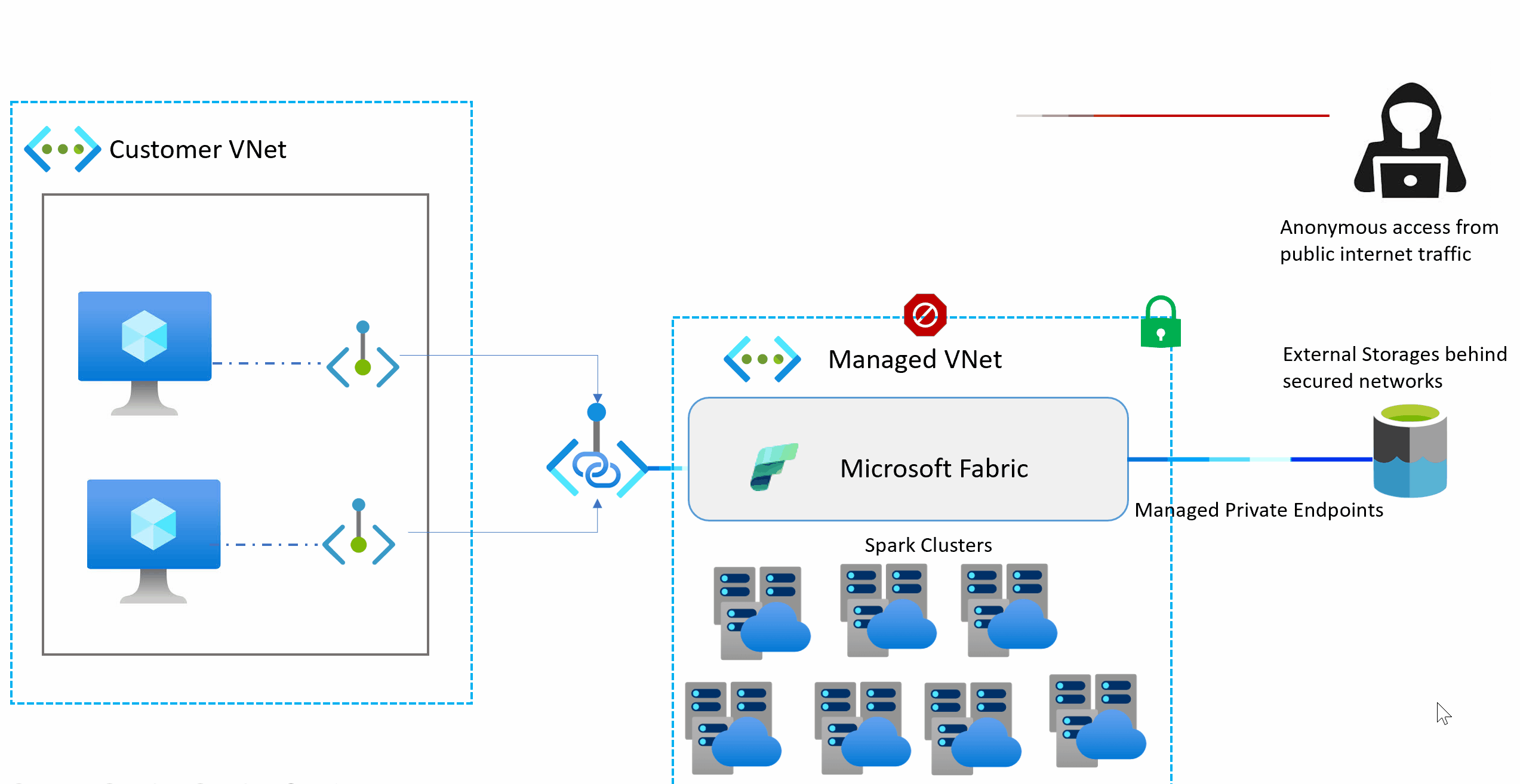
Using Private links, a tunnel from the service into one of your subnets creates a private channel. Communication from external devices travels from their IP address, to a private endpoint in that subnet, through the tunnel and into the service.

After implementing private links, Fabric is no longer accessed through the public internet. To access Fabric, all users have to connect through the private network. The private network is required for all communications with Fabric, including viewing a Power BI report in the browser and using SQL Server Management Studio (SSMS) to connect to a SQL connections string like <guid\_unique\_your\_item>.datawarehouse.fabric.microsoft.com.

* 1. Outbound
     1. What are managed virtual networks?

Managed virtual networks are virtual networks that are created and managed by Microsoft Fabric for each Fabric workspace. Managed virtual networks provide network isolation for Fabric Spark workloads, meaning that the compute clusters are deployed in a dedicated network and are no longer part of the shared virtual network.

Managed virtual networks also enable network security features such as managed private endpoints, and private link support for Data Engineering and Data Science items in Microsoft Fabric that use Apache Spark.



Fabric workspaces that are provisioned with a dedicated virtual network provide you with value in three ways:

* With a managed virtual network you get complete network isolation for the Spark clusters running your Spark jobs (which allow users to run arbitrary user code) while offloading the burden of managing the virtual network to Microsoft Fabric.
* You don't need to create a subnet for the Spark clusters based on peak load, as this is managed for you by Microsoft Fabric.
* A managed virtual network for your workspace, along with managed private endpoints, allows you to access data sources that are behind firewalls or otherwise blocked from public access.

**Note:**

Managed virtual networks are currently not supported in the Switzerland West and West Central US regions.

Outbound: Managed Private Endpoints are not available in Fabric workspaces attached to capacities in Switzerland West and West Central US regions.

Inbound: If workspaces are attached to Fabric capacities in this region within tenants where the Private Link setting is enabled, Data Engineering jobs originating from notebooks, Spark job definitions, and lakehouse operations will result in errors.

1. Collaborate, Share, and manage security
   1. Security for data warehousing in Microsoft Fabric

Applies to: ✅ SQL analytics endpoint and Warehouse in Microsoft Fabric

This document covers security topics for securing the SQL analytics endpoint of the lakehouse and the Warehouse in Microsoft Fabric.

**Note:**

For information on Microsoft Fabric security, see [Security in Microsoft Fabric](https://learn.microsoft.com/en-us/fabric/security/security-overview).

For information on connecting to the SQL analytics endpoint and Warehouse, see [Connectivity](https://learn.microsoft.com/en-us/fabric/data-warehouse/connectivity).

* 1. Warehouse access model

Microsoft Fabric permissions and granular SQL permissions work together to govern Warehouse access and the user permissions once connected.

* Warehouse connectivity is dependent on being granted the Microsoft Fabric Read permission, at a minimum, for the Warehouse.
* Microsoft Fabric item permissions enable the ability to provide a user with SQL permissions, without needing to grant those permissions within SQL.
* Microsoft Fabric workspace roles provide Microsoft Fabric permissions for all warehouses within a workspace.
* Granular user permissions can be further managed via T-SQL.
  1. Workspace roles

Workspace roles are used for development team collaboration within a workspace. Role assignment determines the actions available to the user and applies to all items within the workspace.

* For an overview of Microsoft Fabric workspace roles, see [Roles in workspaces](https://learn.microsoft.com/en-us/fabric/get-started/roles-workspaces). (Check section 3.1 Workspace roles, in this document)
* For instructions on assigning workspace roles, see [Give workspace access](https://learn.microsoft.com/en-us/fabric/get-started/give-access-workspaces). (Check section 4 and sub sections 4.x in this document).
  1. Item permissions

In contrast to workspace roles, which apply to all items within a workspace, item permissions can be assigned directly to individual Warehouses. The user will receive the assigned permission on that single warehouse. The primary purpose for these permissions is to enable sharing for downstream consumption of the Warehouse.

**Note:**

Refer toShare your data and manage permissions:[**Share your warehouse and manage permissions - Microsoft Fabric | Microsoft Learn**](https://learn.microsoft.com/en-us/fabric/data-warehouse/share-warehouse-manage-permissions)

* 1. Granular security

Workspace roles and item permissions provide an easy way to assign coarse permissions to a user for the entire warehouse. However, in some cases, more granular permissions are needed for a user. To achieve this, standard T-SQL constructs can be used to provide specific permissions to users.

Microsoft Fabric data warehousing supports several data protection technologies that administrators can use to protect sensitive data from unauthorized access. By securing or obfuscating data from unauthorized users or roles, these security features can provide data protection in both a Warehouse and SQL analytics endpoint without application changes.

* [Object-level security](https://learn.microsoft.com/en-us/fabric/data-warehouse/security#object-level-security) controls access to specific database objects.
* [Column-level security](https://learn.microsoft.com/en-us/fabric/data-warehouse/security#column-level-security) prevents unauthorized viewing of columns in tables.
* [Row-level security](https://learn.microsoft.com/en-us/fabric/data-warehouse/security#row-level-security) prevents unauthorized viewing of rows in tables, using familiar WHERE clause filter predicates.
* [Dynamic data masking](https://learn.microsoft.com/en-us/fabric/data-warehouse/security#dynamic-data-masking) prevents unauthorized viewing of sensitive data by using masks to prevent access to complete, such as email addresses or numbers.
  + 1. Object-Level Security

Object-level security is a security mechanism that controls access to specific database objects, such as tables, views, or procedures, based on user privileges or roles. It ensures that users or roles can only interact with and manipulate the objects they have been granted permissions for, protecting the integrity and confidentiality of the database schema and its associated resources.

**Note:**

For managing granular permissions in SQL refer to [SQL granular permissions](https://learn.microsoft.com/en-us/fabric/data-warehouse/sql-granular-permissions).

* + 1. Row-level security

Row-level security is a database security feature that restricts access to individual rows or records within a database table based on specified criteria, such as user roles or attributes. It ensures that users can only view or manipulate data that is explicitly authorized for their access, enhancing data privacy and control.

**Note:**

For row-level security details refer to [Row-level security in Fabric data warehousing](https://learn.microsoft.com/en-us/fabric/data-warehouse/row-level-security), covered in in section 4.2 in this document.

* + 1. Column-level security

Column-level security is a database security measure that limits access to specific columns or fields within a database table, allowing users to see and interact with only the authorized columns while concealing sensitive or restricted information. It offers fine-grained control over data access, safeguarding confidential data within a database.

**Note:**

Further details of column-level security refer to: [Column-level security in Fabric data warehousing](https://learn.microsoft.com/en-us/fabric/data-warehouse/column-level-security)

* + 1. Dynamic data masking

Dynamic data masking helps prevent unauthorized viewing of sensitive data by enabling administrators to specify how much sensitive data to reveal, with minimal effect on the application layer. Dynamic data masking can be configured on designated database fields to hide sensitive data in the result sets of queries. With dynamic data masking, the data in the database isn't changed, so it can be used with existing applications since masking rules are applied to query results. Many applications can mask sensitive data without modifying existing queries.

* + - 1. Dynamic data masking in Fabric data warehousing

Applies to: ✅ SQL analytics endpoint and Warehouse in Microsoft Fabric

Dynamic data masking limits sensitive data exposure by masking it to nonprivileged users. It can be used to greatly simplify the design and coding of security in your application.

Dynamic data masking helps prevent unauthorized viewing of sensitive data by enabling administrators to specify how much sensitive data to reveal, with minimal effect on the application layer. Dynamic data masking can be configured on designated database fields to hide sensitive data in the result sets of queries. With dynamic data masking, the data in the database isn't changed, so it can be used with existing applications since masking rules are applied to query results. Many applications can mask sensitive data without modifying existing queries.

* A central data masking policy acts directly on sensitive fields in the database.
* Designate privileged users or roles that do have access to the sensitive data.
* Dynamic data masking features full masking and partial masking functions, and a random mask for numeric data.
* Simple Transact-SQL commands define and manage masks.

The purpose of dynamic data masking is to limit exposure of sensitive data, preventing users who shouldn't have access to the data from viewing it. Dynamic data masking doesn't aim to prevent database users from connecting directly to the database and running exhaustive queries that expose pieces of the sensitive data.

Dynamic data masking is complementary to other Fabric security features like [column-level security](https://learn.microsoft.com/en-us/fabric/data-warehouse/column-level-security) and [row-level security](https://learn.microsoft.com/en-us/fabric/data-warehouse/row-level-security). It's highly recommended to use these data protection features together in order to protect the sensitive data in the database.

* + - 1. Define a dynamic data mask

A masking rule can be defined on a column in a table, in order to obfuscate the data in that column. There are four types of masks available.

|  |  |  |
| --- | --- | --- |
| **Function** | **Description** | **Examples** |
| Default | Full masking according to the data types of the designated fields.  For string data types, use XXXX (or fewer) if the size of the field is fewer than 4 characters (**char**, **nchar**, **varchar**, **nvarchar**, **text**, **ntext**).  For numeric data types use a zero value (**bigint**, **bit**, **decimal**, **int**, **money**, **numeric**, **smallint**, **smallmoney**, **tinyint**, **float**, **real**).  For date and time data types, use 1900-01-01 00:00:00.0000000 (**date**, **datetime2**, **datetime**, **datetimeoffset**, **smalldatetime**, **time**).  For binary data types use a single byte of ASCII value 0 (**binary**, **varbinary**, **image**). | Example column definition syntax: Phone# varchar(12) MASKED WITH (FUNCTION = 'default()') NULL  Example of alter syntax: ALTER COLUMN Gender ADD MASKED WITH (FUNCTION = 'default()') |
| Email | Masking method that exposes the first letter of an email address and the constant suffix ".com", in the form of an email address. aXXX@XXXX.com. | Example definition syntax: Email varchar(100) MASKED WITH (FUNCTION = 'email()') NULL  Example of alter syntax: ALTER COLUMN Email ADD MASKED WITH (FUNCTION = 'email()') |
| Random | A random masking function for use on any numeric type to mask the original value with a random value within a specified range. | Example definition syntax: Account\_Number bigint MASKED WITH (FUNCTION = 'random([start range], [end range])')  Example of alter syntax: ALTER COLUMN [Month] ADD MASKED WITH (FUNCTION = 'random(1, 12)') |
| Custom String | Masking method that exposes the first and last letters and adds a custom padding string in the middle. prefix,[padding],suffix  If the original value is too short to complete the entire mask, part of the prefix or suffix isn't exposed. | Example definition syntax: FirstName varchar(100) MASKED WITH (FUNCTION = 'partial(prefix,[padding],suffix)') NULL  Example of alter syntax: ALTER COLUMN [Phone Number] ADD MASKED WITH (FUNCTION = 'partial(1,"XXXXXXX",0)')  This turns a phone number like 555.123.1234 into 5XXXXXXX.  Additional example:  ALTER COLUMN [Phone Number] ADD MASKED WITH (FUNCTION = 'partial(5,"XXXXXXX",0)')  This turns a phone number like 555.123.1234 into 555.1XXXXXXX. |

**Note:**

Refer to Examples of How to implement dynamic data masking in Fabric Data Warehouse:

[How to implement dynamic data masking in Fabric Data Warehouse - Microsoft Fabric | Microsoft Learn](https://learn.microsoft.com/en-us/fabric/data-warehouse/howto-dynamic-data-masking)

**Permissions:**

Users without the Administrator, Member, or Contributor rights on the workspace, and without elevated permissions on the Warehouse, will see masked data.

You don't need any special permission to create a table with a dynamic data mask, only the standard CREATE TABLE and ALTER on schema permissions.

Adding, replacing, or removing the mask of a column, requires the ALTER ANY MASK permission and ALTER permission on the table. It's appropriate to grant ALTER ANY MASK to a security officer.

Users with SELECT permission on a table can view the table data. Columns that are defined as masked will display masked data. Grant the UNMASK permission to a user to enable them to retrieve unmasked data from the columns for which masking is defined.

The CONTROL permission on the database includes both the ALTER ANY MASK and UNMASK permission that enables the user to view unmasked data. Administrative users or roles such as Admin, Member, or Contributor have CONTROL permission on the database by design and can view unmasked data by default. Elevated permissions on the Warehouse include CONTROL permission.

**Security consideration: bypassing masking using inference or brute-force techniques**

Dynamic data masking is designed to simplify application development by limiting data exposure in a set of predefined queries used by the application. While Dynamic Data Masking can also be useful to prevent accidental exposure of sensitive data when accessing data directly, it's important to note that unprivileged users with query permissions can apply techniques to gain access to the actual data.

As an example, consider a user that has sufficient privileges to run queries on the Warehouse, and tries to 'guess' the underlying data and ultimately infer the actual values. Assume that we have a mask defined on the [Employee].[Salary] column, and this user connects directly to the database and starts guessing values, eventually inferring the [Salary] value in the Employees table:

**SQL**

SELECT ID, Name, Salary FROM Employees

WHERE Salary > 99999 and Salary < 100001;

**Results in:**

|  |  |  |
| --- | --- | --- |
| **ID** | **Name** | **Salary** |
| 62543 | Jane Doe | 0 |
| 91245 | John Smith | 0 |

This demonstrates that dynamic data masking shouldn't be used alone to fully secure sensitive data from users with query access to the Warehouse or SQL analytics endpoint. It's appropriate for preventing sensitive data exposure, but doesn't protect against malicious intent to infer the underlying data.

It's important to properly manage object-level security with [SQL granular permissions](https://learn.microsoft.com/en-us/fabric/data-warehouse/sql-granular-permissions), and to always follow the minimal required permissions principle.

* + 1. User audit logs

To track user activity in warehouse and SQL analytics endpoint for meeting regulatory compliance and records managements requirements, a set of audit activities are accessible via Microsoft Purview and PowerShell. You can use user audit logs to identify who is taking what action on your Fabric items.

**Important:**

* Track user activities in Microsoft Fabric: <https://learn.microsoft.com/en-us/fabric/admin/track-user-activities>
* Operation list: <https://learn.microsoft.com/en-us/fabric/admin/operation-list>

1. Microsoft Purview to govern Microsoft Fabric

Microsoft Purview and Microsoft Fabric are part of the Microsoft Intelligent data platform that allows you to store, analyze, and govern your data. With Microsoft Fabric and Microsoft Purview together you're able to govern your entire estate and lineage of data. From data source down to the Power BI report, Microsoft Purview and Fabric work together seamlessly so you can store, analyze, and govern your data without piecing together services from multiple vendors.

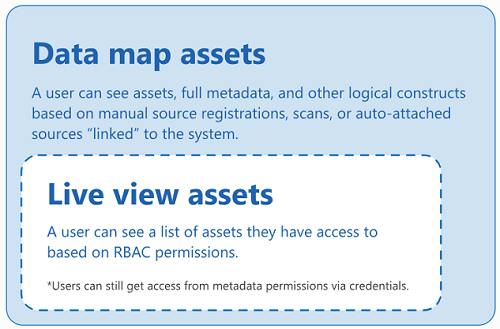
Microsoft Purview works with Microsoft Fabric so users can discover and manage Microsoft Fabric items in Microsoft Purview applications. The integration currently allows you to take advantage of these applications:

* **Microsoft Purview Data Catalog** - automatically view metadata about your Microsoft Fabric items in the Microsoft Purview Data Catalog with live view in Microsoft Purview. Or, connect your data catalog to Microsoft Fabric in the same tenant or across tenants.
* **Microsoft Purview Information Protection** - allows you to discover, classify, and protect Fabric data using sensitivity labels from Microsoft Purview Information Protection. Sensitivity labels can be set on all Fabric items. Data remains protected when it's exported via supported export paths. Compliance admins can monitor activities on sensitivity labels in Microsoft Purview Audit For more information, see Information Protection in Microsoft Purview.
* **Microsoft Purview Data Loss Prevention (DLP)** - DLP policies are currently supported in Fabric for Power BI semantic models only. DLP policies detect upload of sensitive data into semantic models. They can detect sensitivity labels and sensitive info types, such as credit card and social security numbers. They can be configured to generate policy tips for semantic model owners and alerts for security admins. DLP policies can also be configured to allow data owners to override them. For more information, see data loss prevention policies.
* **Microsoft Purview Audit** - all Microsoft Fabric user activities are logged and available in the Microsoft Purview audit log. For more information, see track user activities for Microsoft Fabric and track user activities in Power BI.
  1. Live view in Microsoft Purview (preview)

Some Azure resources are available to be seen in live view. Live view means that users with data access permissions can find these resources and their data assets in the catalog without setup or scanning. These resources and their metadata are immediately available when you access the free or enterprise versions of Microsoft Purview, so you can start your governance journey quickly.

**Important:**

This feature is currently in preview. The Supplemental Terms of Use for Microsoft Azure Previews include additional legal terms that apply to Azure features that are in beta, in preview, or otherwise not yet released into general availability



**Limitations**

* Live view for Microsoft Fabric limitations:
  + Personal workspaces aren't supported yet in live view.
  + Only Microsoft Fabric item level metadata are available in live view.
  + Power BI Datasets don’t support showing schema in live view if the item hasn’t been scanned.
  + Dataflow Gen2 (Preview) items are displayed as Dataflow items.

**What resources have live view available?**

Live view resources in the same tenant as your Microsoft Purview account are automatically accessible for users in your catalog. These resources have live view available:

|  |  |
| --- | --- |
| **Resource** | **Level of view available** |
| Azure Blob Storage | Storage account, Azure Blob Service, Container |
| Azure Data Lake Storage Gen 2 | Storage account, Azure Data Lake Storage Gen2 Service, File system |
| Azure SQL Database | Server, Database, Schema, Tables, Columns |
| Azure Subscriptions | Live view assets in the subscription |
| Microsoft Fabric | Workspace, Item |

Users with permissions can browse and search for these assets in the catalog without needing to set up a scan.

**Metadata available in live view**

Resources in live view in the catalog automatically have this metadata available:

* Name
* Properties
* Schema
* Lineage

**Note:**

Live view resources won't automatically have classifications applied. To automatically apply classifications to these resources, you need to upgrade to the enterprise version of Microsoft Purview, register and scan the resource.

Data owners can also update the metadata on live view resources to add more information. Including manually adding description, contacts, etc.

**Note:**

* Metadata curation in Microsoft Purview: <https://learn.microsoft.com/en-us/purview/metadata-curation>
* Data Owners: <https://learn.microsoft.com/en-us/purview/live-view#data-owners>

**Live view permissions**

Users that have permissions on the available resources are automatically able to access these resources in the catalog as either data readers or data owners. No other permissions in Microsoft Purview are required.

|  |  |
| --- | --- |
| **Data source** | **Reader permission** |
| Azure SQL Database | [Reader](https://learn.microsoft.com/en-us/azure/role-based-access-control/built-in-roles), or [these actions.](https://learn.microsoft.com/en-us/purview/roles-permissions#data-access-permissions) |
| Azure Blob Storage | [Reader](https://learn.microsoft.com/en-us/azure/role-based-access-control/built-in-roles), or [these actions.](https://learn.microsoft.com/en-us/purview/roles-permissions#data-access-permissions) |
| Azure Data Lake Storage Gen2 | [Reader](https://learn.microsoft.com/en-us/azure/role-based-access-control/built-in-roles), or [these actions.](https://learn.microsoft.com/en-us/purview/roles-permissions#data-access-permissions) |
| Azure Subscription | Read permission on subscription, or [these actions.](https://learn.microsoft.com/en-us/purview/roles-permissions#data-access-permissions) |

**Data owners**

These are the permissions needed on the resources for a user to be able to see it as an owner and curate its metadata in live view in Microsoft Purview.

|  |  |
| --- | --- |
| **Data source** | **Owner permission** |
| Azure SQL Database | "Microsoft.Sql/servers/write", "Microsoft.Sql/servers/databases/write", "Microsoft.Authorization/roleAssignments/write" |
| Azure Blob Storage | "Microsoft.Storage/storageAccounts/write", "Microsoft.Authorization/roleAssignments/write" |
| Azure Data Lake Storage Gen2 | "Microsoft.Storage/storageAccounts/write", "Microsoft.Authorization/roleAssignments/write" |

**Extend live view permissions**

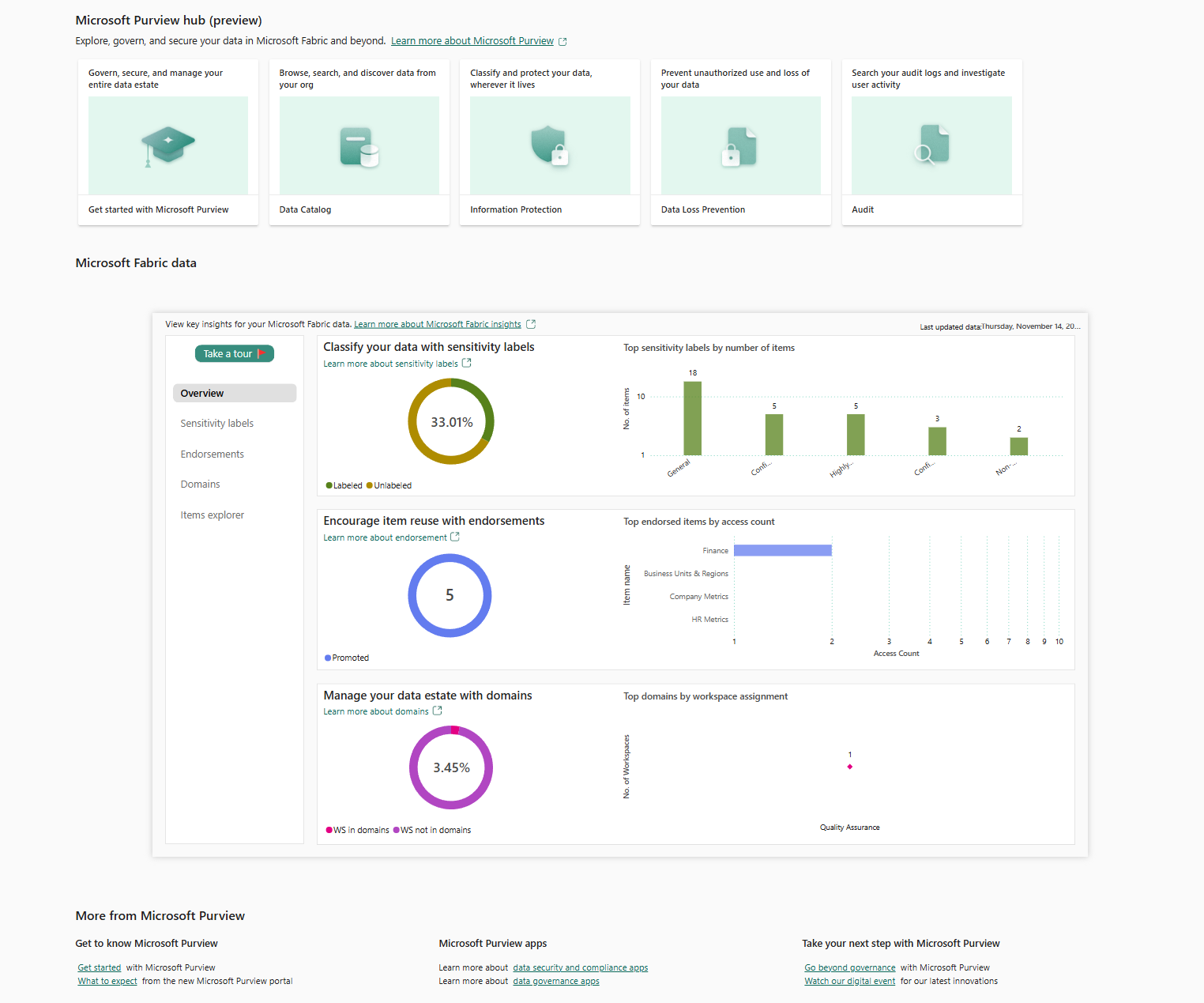
You can expand discoverability of these resources to users that don't have permissions already by enabling live view for all users. This can only be enabled with an enterprise version of Microsoft Purview. Extending live view also deepens the level of information available.

|  |  |
| --- | --- |
| **Resource** | **Level of view available** |
| Azure Blob Storage | Storage account, Azure Blob Service, Container, Folders, Files |
| Azure Data Lake Storage Gen 2 | Storage account, Azure Data Lake Storage Gen2 Service, File system, Folders, Files |
| Azure SQL Database | Server, Database, Schema, Tables, Columns, Views, Stored procedures |

**Note**

* For Azure Blob storage and Azure Data Lake Storage Gen 2, file schema is provided for parquet files.
* You can't extend live view permissions for Microsoft Fabric sources.
* To enable these permissions, the user must be a Data Source Administrator and Reader on the collection where they're going to host the resource: <https://learn.microsoft.com/en-us/purview/classic-data-governance-permissions#roles>
  1. Microsoft Purview hub for administrators (preview)

Microsoft Purview hub for administrators is a centralized place in Fabric that helps Fabric administrators manage and govern their 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.



**Requirements**

Access to Purview hub for administrators requires the Fabric administrator role or higher

* + 1. Access to Purview hub for administrators requires the Fabric administrator role or higher

To be a Microsoft Fabric admin for your organization, you must be in one of the following roles:

* Power Platform administrator
* Fabric administrator

Microsoft 365 user admins assign users to the Fabric administrator or Power Platform administrator roles in the Microsoft 365 admin portal, or by using a PowerShell script. For more information, see Assign roles to user accounts with PowerShell.

Users in Fabric administrator and Power Platform administrator roles have full control over org-wide Microsoft Fabric settings and admin features, except for licensing. Once a user is assigned an admin role, they can access the admin portal. There, they have access to org-wide usage metrics and can control org-wide usage of Microsoft Fabric features. These admin roles are ideal for users who need access to the Fabric admin portal without also granting those users full Microsoft 365 administrative access.

* 1. Information protection in Microsoft Fabric

The following table summarizes the information protection capabilities in Fabric that help you achieve maximum coverage of the sensitive information in your organization. Fabric support is indicated in the third column. See the sections under Considerations and limitations for details.

|  |  |  |
| --- | --- | --- |
| **Capability** | **Scenario** | **Support status** |
| Manual labeling | Users can manually apply sensitive labels to Fabric items | [Supported for all Fabric items](https://learn.microsoft.com/en-us/fabric/governance/information-protection#manual-labeling). |
| Default labeling | When an item is created or edited, it gets a default sensitivity label unless a label is applied through other means. | [Supported for all Fabric items, with limitations](https://learn.microsoft.com/en-us/fabric/governance/information-protection#default-labeling). |
| Mandatory labeling | Users can't save items unless a sensitivity label is applied to the item. This means they can't remove a label either. | [Currently fully supported for Power BI items only. Supported for some non-Power BI Fabric items, with limitations](https://learn.microsoft.com/en-us/fabric/governance/information-protection#mandatory-labeling). |
| Programmatic labeling | Sensitivity labels can be added, changed, or deleted programmatically via Power BI admin REST APIs. | [Supported for all Fabric items](https://learn.microsoft.com/en-us/fabric/governance/information-protection#programmatic-labeling). |
| Downstream inheritance | When a sensitivity label is applied to an item, the label propagates downstream to all dependent items. | [Supported for all Fabric items, with limitations](https://learn.microsoft.com/en-us/fabric/governance/information-protection#downstream-inheritance). |
| Inheritance upon creation | When you create a new item from an existing item, the new item inherits the label of the existing item. | [Supported for all Power BI Fabric items. Supported for some non-Power BI Fabric items as described in the considerations and limitations](https://learn.microsoft.com/en-us/fabric/governance/information-protection#inheritance-upon-creation). |
| Inheritance from data sources | When a Fabric item ingests data from a data source that has a sensitivity label, that label is applied to the Fabric item. The label then propagates downstream to the child items of that Fabric item via downstream inheritance. | [Currently supported for Power BI semantic models only](https://learn.microsoft.com/en-us/fabric/governance/information-protection#inheritance-from-data-sources). |
| Export | When a user exports data from an item that has a sensitivity label, the sensitivity label moves with it to the exported format. | [Currently supported for Power BI items in supported export paths](https://learn.microsoft.com/en-us/fabric/governance/information-protection#export). |

* + 1. Considerations and limitations

**Manual labeling**

When you enable sensitivity labels on your tenant, you specify which users can apply sensitivity labels. While the other information protection capabilities described in this document can ensure that most items get labeled without someone having to manually apply a label, manual labeling makes it possible for users to change labels on items.

**Note:**

For a user to be able to apply sensitivity labels to Fabric items, it's not enough just to include the user in the list of specified users. The sensitivity label must also be published to the user as part of the label's policy definitions in the Microsoft Purview compliance center.

**Mandatory labeling**

Mandatory labeling is currently supported for Power BI items only. Mandatory labeling isn’t enforced if changes are made via the flyout menu.

For lakehouses, pipelines, and data warehouses: Assuming that information protection is enabled, if mandatory labeling is on and default labeling is off, it will be possible for the user to select a label. However, mandatory labeling logic isn't enforced. That means that the user can save the item without a label, unless the experience itself requires that a label be set.

**Programmatic labeling**

Programmatic labeling is supported for all Fabric items. For more information, see [Set or remove sensitivity labels using Power BI REST admin APIs](https://learn.microsoft.com/en-us/fabric/governance/service-security-sensitivity-label-inheritance-set-remove-api).

**Downstream inheritance**

Downstream inheritance is on by default. It's supported in Fabric as follows:

Supported:

* Power BI item to Power BI item
* Fabric item to Fabric item
* Fabric item to Power BI item

**Not supported:**

* Power BI item to Fabric item

Autogenerated items from a lakehouse or data warehouse take their sensitivity label from their parent lakehouse or data warehouse. They don't inherit the label from items further upstream.

**Inheritance upon creation**

Inheritance upon creation is supported for Power BI Fabric items and in other scenarios with non-Power BI items where one item is created from another item:

* A Pipeline created from a Lakehouse inherits the sensitivity label of the Lakehouse.
* A Notebook created from a Lakehouse inherits the sensitivity label of the Lakehouse.
* A Lakehouse shortcut created from a Lakehouse inherits the sensitivity label of the Lakehouse.
* A Pipeline created from a Notebook inherits the sensitivity label of the Notebook.
* A KQL Queryset created from a KQL Database inherits the sensitivity label of KQL Database.
* A Pipeline created from a KQL Database inherits the sensitivity label of KQL Database.

**Inheritance from data sources**

Inheritance from data sources is currently supported for Power BI semantic models only. For more information, see [Sensitivity label inheritance from data sources](https://learn.microsoft.com/en-us/fabric/governance/service-security-sensitivity-label-inheritance-from-data-sources).

**Export**

Sensitivity label inheritance upon export is supported for Power BI items only in supported export paths. Currently no other Fabric experience uses an export method that transfers the sensitivity label to the exported output. However, if they do export an item that has a sensitivity label, a warning is issued.

* 1. Protected sensitivity labels in Fabric and Power BI

Protected labels are sensitivity labels that have file protection policies associated with them and can be used to protect files and data. A file protection policy for a label grants usage rights to users, such as the ability to open a file, edit a file, copy from a file, etc. When a protected label is applied to a file or item, users who are included in the policy can perform the actions they have the usage rights for under the label policy. A file protection policy can grant different sets of users different sets of usage rights. For example, under the policy, one set of users might be granted full control over the file, while another set of users might only be permitted to open and view the file.

Having a set of sensitivity labels and policies in place is a prerequisite for using sensitivity labels in Fabric and Power BI. The labels and their file protection policies are defined by security admins in the Microsoft Purview compliance portal. Typically, the same set of protected labels is used across an organization for Office apps such as Word, Excel, and PowerPoint as well as for Fabric and Power BI.

* + 1. How protected labels work in Fabric and Power BI

In Fabric and the Power BI service, protected labels only control the ability to change or remove labels on items. They don't control access to content. In order for a user to be able to change or remove a protected label from an item, the user must either be the user who applied the sensitivity label (the RMS owner), or have at least one of the following usage rights requirements for the label.

* OWNER
* EXPORT
* EDIT and EDITRIGHTSDATA

If the label on the item was set via an automated process, such as inheritance from data sources or downstream inheritance, then the third option, EDIT and EDITRIGHTSDATA, is reduced to just EDIT.

In Power BI Desktop, protected labels control not only the ability to change or remove the protected label, but also access to content (viewing, editing, exporting, etc.). As a result, collaboration scenarios with protected PBIX files might be blocked, since it's unlikely that most users will have sufficient usage rights under the label to open and edit the file.

For example, imagine that you create a report in Power BI Desktop, apply a protected label to it, and then share the PBIX file with another user. It's quite likely that the user won't have sufficient permissions to open the file.

To prevent this situation and enable more users to work with protected PBIX files, the Fabric administrator should enable the **Increase the number of users who can edit and republish encrypted PBIX files (preview)** tenant setting. When this setting is enabled, more users (see note) will able to open, edit, and publish/republish protected PBIX files, with the following restrictions:

* They can't export to formats that don't support sensitivity labels, such as CSV files.
* They can't change the label on the PBIX file.
* They can only republish the PBIX file to the original workspace the file came from.

**Note:**

The file must have been published at least once for other users to be able to publish it back to that specific workspace. If the file hasn't yet been published, then the latest label issuer (the one who most recently set the protected label) or a user with sufficient usage rights must publish it and then share the file with the other editors.)

These restrictions ensure that the security of the content remains under the control of those who have high enough permissions to set the label.

**Note:**

Users must have all of the following usage rights under the label policy:

* View Content (VIEW)
* Edit Content (DOCEDIT)
* Save (EDIT)
* Copy and extract content (EXTRACT)
* Allow Macros (OBJMODEL)

These usage rights are a subset of the Co-Author permissions preset in the Microsoft Purview compliance portal.

In addition, the Less elevated user support preview feature switch in Power BI Desktop must be selected.

* + 1. Relaxations to accommodate automatic labeling scenarios

Fabric and Power BI support several capabilities, such as label inheritance from data sources and downstream inheritance, which automatically apply sensitivity labels to content. These automated scenarios can result in situations where no user has been set as the RMS label issuer for a label on an item. This means that there's no user who is guaranteed to be able to change or remove the label.

In such cases, the usage rights requirements for changing or removing the label are relaxed - a user needs just one of the following usage rights to be able to change or remove the label:

* OWNER
* EXPORT
* EDIT

If no user has even these usage rights, nobody will be able to change or remove the label from the item, and access to the item is potentially endangered.

To avoid this situation, the Fabric admin can enable the **Allow workspace admins to override automatically applied sensitivity labels** tenant setting. This makes it possible for workspace admins to override automatically applied sensitivity labels without regard to label change enforcement rules.

To enable this setting, go to: **Admin portal > Tenant settings > Information protection**, and enable the toggle on the **Allow workspace admins to override automatically applied sensitivity labels** setting.

* 1. Default label policy for Power BI

What happens when a default label policy is in effect?

* In Power BI Desktop, when a user to whom the policy applies opens a new *.pbix* file or an existing unlabeled *.pbix* file, the default label is applied to the file. If the user is working offline, the label is applied when the user signs in.
* In the Power BI service, when a user to whom the policy applies creates a new semantic model, report, dashboard, dataflow or scorecard, the default label is applied to that item.

**Note:**

**Enabling a default label policy for Power BI:** <https://learn.microsoft.com/en-us/fabric/governance/service-security-sensitivity-label-default-label-policy>

* 1. Domain-level default sensitivity labels in Microsoft Fabric

Domain admins can set a default sensitivity label for their domains. This label will be applied by default to items in workspaces that belong to the domain in the following instances:

* When a new item is created and saved in the workspace.
* Whan an unlabeled item that already exists in the workspace is updated and saved.

When an existing labeled item in the workspace is updated and saved, the existing label is either retained or overwritten according to the following logic:

|  |  |
| --- | --- |
| **Existing label** | **Override with domain-level default label** |
| Manually applied, any priority | No |
| Automatically applied, lower priority | Yes |
| Automatically applied, higher priority | No |
| Default label from policy, lower priority | Yes |
| Default label from policy, higher priority | No |

**Requirements**

The tenant setting Domain admins can set default sensitivity labels for their domains (preview) must be enabled.

**Limitations**

Reports, semantic models, dataflows, dashboards, scorecards, and some additional item types aren't currently supported.

* 1. Mandatory label policy for Fabric and Power BI

To help ensure comprehensive protection and governance of sensitive data, you can require your organization's Fabric and Power BI users to apply sensitivity labels to items they create or edit. You do this by enabling, in their sensitivity label policies, a special setting for mandatory labeling in Fabric and Power BI.

**Note:**

The mandatory label policy setting for Fabric and Power BI is independent of the mandatory label policy setting for files and email.

Mandatory labeling in Fabric and Power BI isn't supported for service principals and APIs. Service principals and APIs aren't subject to mandatory label policies.

* + 1. What happens when a mandatory label policy is in effect?

**In Fabric and the Power BI service**:

* Users must apply a sensitivity label before they can save new items (if the item is a [supported item type](https://learn.microsoft.com/en-us/fabric/governance/mandatory-label-policy#supported-item-types)).
* Users must apply a sensitivity label before they can save changes to the settings or content of existing, unlabeled items (if the item is a [supported item type](https://learn.microsoft.com/en-us/fabric/governance/mandatory-label-policy#supported-item-types)).
* If users try to import data from an unlabeled *.pbix* file, a prompt requires them to select a label before the import can continue. The label they select is applied to the resulting semantic model and report in the service. **It's not applied to the *.pbix* file itself**.

**In Power BI Desktop**:

* Users must apply sensitivity labels to unlabeled *.pbix* files before they can save or publish to the service.

**Supported item types**

Mandatory labeling in Fabric and Power BI is supported for all item types except:

* RDL report
* Scorecard
* Dataflow Gen 1
* Dataflow Gen 2
* Streaming semantic model
* Streaming dataflow

**Note:**

**Enabling a mandatory label policy for Fabric and Power BI:** <https://learn.microsoft.com/en-us/fabric/governance/mandatory-label-policy>

* 1. Sensitivity label support for paginated reports

Sensitivity labels can be applied to paginated reports hosted in the Power BI service. After uploading a paginated report to the service, apply the label to the report just as you would to a regular Power BI report.

When you export data from a labeled paginated report to a supported file type (Excel, PDF, PPTX, and Word), the sensitivity label on the paginated report is applied to the exported file.

Sensitivity labels on paginated reports are included in protection metrics (as part of the Report count), and can be audited (label-change audits only) and modified by public APIs, just like labels on regular Power BI reports.

**Considerations and limitations:**

* [Downstream inheritance](https://learn.microsoft.com/en-us/fabric/governance/service-security-sensitivity-label-downstream-inheritance) isn't supported. The label of an upstream model won't propagate down to its downstream paginated reports. Likewise, the label of a paginated report won't propagate down to the report’s downstream content.
* [Mandatory labeling](https://learn.microsoft.com/en-us/fabric/governance/mandatory-label-policy) doesn't apply to paginated reports

**Paginated Report visuals**

A Paginated Report visual is a special type of visual that you can include in a regular Power BI report. It renders a selected paginated report inside the regular Power BI report.

When a supported file type is exported from a Paginated Report visual that is included in a Power BI report, and the original paginated report being rendered in the visual has a sensitivity label, the exported file inherits the sensitivity label of the original paginated report. If the original paginated report doesn't have a label, the exported file inherits the label of the Power BI report, if it has one.

* 1. Programmatic labeling

**Set or remove sensitivity labels using Power BI REST admin APIs:**

To meet compliance requirements, organizations are often required to classify and label all sensitive data in Power BI. This task can be challenging for tenants that have large volumes of data in Power BI. To make the task easier and more effective, you can use Power BI admin REST APIs to programatically setLabels and removeLabels on large numbers of Power BI artifacts.

The APIs set or remove labels from artifacts by artifact ID.

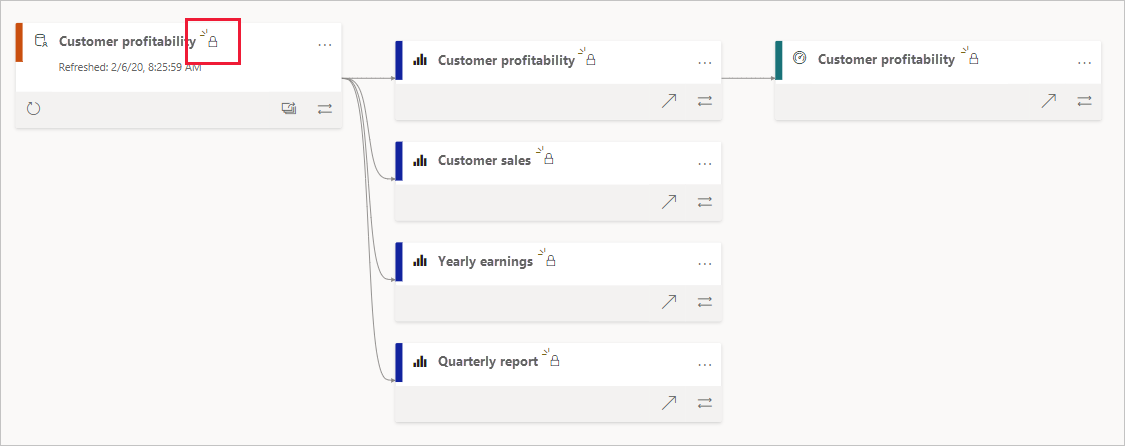
**Requirements and considerations**

* Users must be Fabric administrators to call these APIs.
* The admin user (and the delegated user, if provided) must have sufficient [usage rights](https://learn.microsoft.com/en-us/azure/information-protection/configure-usage-rights) to set or remove labels.
* To set a sensitivity label using the setLabels API, the admin user (or the delegated user, if provided) must have the label included in their label policy.
* The APIs allow a maximum of 25 requests per hour. Each request can update up to 2000 artifacts.
* Required scope: Tenant.ReadWrite.All
  1. Sensitivity label downstream inheritance

When a sensitivity label is applied to a semantic model or report in the Power BI service, it's possible to have the label trickle down and be applied to content that's built from that semantic model or report. For semantic models, this means other semantic models, reports, and dashboards. For reports, this means dashboards.

Downstream inheritance is a critical link in Power BI’s end-to-end information protection solution. Together with inheritance from data sources, inheritance upon creation of new content, inheritance upon export to file, and other capabilities for applying sensitivity labels, downstream inheritance helps ensure that sensitive data remains protected throughout its journey through Power BI, from data source to point of consumption.

Downstream inheritance is illustrated using lineage view. When a label is applied to the semantic model Customer profitability, that label filters down and gets applied to the semantic model's downstream content: the reports that are built using that semantic model, and, in this case, a dashboard that's built from visuals from one of those reports.



**Important**

• Downstream inheritance never overwrites labels that were applied manually.

• Downstream inheritance never overwrites a label with a less restrictive label.

**Downstream inheritance modes**

Downstream inheritance operates in one of two modes. The Power BI admin decides via a [tenant setting](https://learn.microsoft.com/en-us/fabric/governance/service-security-sensitivity-label-downstream-inheritance#enabling-fully-automated-downstream-inheritance) which mode is operable on the tenant.

* Downstream inheritance with user consent (default): In this mode, when users apply sensitivity labels on semantic models or reports, they can choose whether to also apply that label downstream. They make their choice by selecting the box that appears with the sensitivity label selector.
* Fully automated downstream inheritance (when enabled by a Power BI admin): In this mode, downstream inheritance happens automatically whenever a label is applied to a semantic model or report. There's no checkbox provided for user consent.
  1. Sensitivity label inheritance from data sources

Power BI semantic models that connect to sensitivity-labeled data in supported data sources can inherit those labels, so that the data remains classified and secure when brought into Power BI.

Currently supported data sources:

* Excel files stored on OneDrive or SharePoint Online\*
* Azure Synapse Analytics (formerly SQL Data Warehouse)
* Azure SQL Database

**Note:**

Inheritance from Excel files requires specific configuration and isn't supported for Excel files stored behind a gateway, such as files stored locally. See [Sensitivity label inheritance from Excel files](https://learn.microsoft.com/en-us/fabric/governance/service-security-sensitivity-label-inheritance-from-data-sources#sensitivity-label-inheritance-from-excel-files) for more detail.

To be operative, [sensitivity label inheritance from data sources must be enabled on the tenant](https://learn.microsoft.com/en-us/fabric/admin/service-admin-portal-information-protection#apply-sensitivity-labels-from-data-sources-to-their-data-in-power-bi).

**Requirements:** <https://learn.microsoft.com/en-us/fabric/governance/service-security-sensitivity-label-inheritance-from-data-sources#requirements>

* 1. Audit schema for sensitivity labels in Fabric

Whenever a sensitivity label on a Fabric item is applied, changed, or removed, that activity is recorded in the audit log for Fabric, where you can track it. For information about tracking activities in the audit log, see [Track user activities in Microsoft Fabric](https://learn.microsoft.com/en-us/fabric/admin/track-user-activities).

Nelo information in the Fabric auditing schema that's specific to sensitivity labels. It covers the following activity keys:

* SensitivityLabelApplied
* SensitivityLabelChanged
* SensitivityLabelRemoved

Audit schema for sensitivity labels in Fabric: <https://learn.microsoft.com/en-us/fabric/governance/sensitivity-label-audit-schema>

1. Appendix A (References)

|  |  |  |
| --- | --- | --- |
| **#** | **Topic** | **URL** |
|  | Location of Customer Data for EU Data Boundary Services | <https://www.microsoft.com/licensing/terms/product/PrivacyandSecurityTerms/MPSA#EUDataBoundaryServices> |
|  | OneLake | <https://learn.microsoft.com/en-us/fabric/onelake/onelake-overview> |
|  | Introduction to Azure Data Lake Storage | <https://learn.microsoft.com/en-us/azure/storage/blobs/data-lake-storage-introduction> |
|  | What is Azure Databricks?: | <https://learn.microsoft.com/en-us/azure/databricks/introduction/> |
|  | Workspaces in Microsoft Fabric and Power BI | <https://learn.microsoft.com/en-us/fabric/get-started/workspaces> |
|  | Secure Data with Fabric, Compute Engines, and OneLake | <https://learn.microsoft.com/en-us/fabric/onelake/security/fabric-onelake-security> |
|  | Share items in Microsoft Fabric | <https://learn.microsoft.com/en-us/fabric/get-started/share-items> |
|  | Share items in Microsoft Fabric: | <https://learn.microsoft.com/en-us/fabric/get-started/share-items> |
|  | Microsoft Fabric concepts and licenses | <https://learn.microsoft.com/en-us/fabric/get-started/share-items> |
|  | Microsoft Fabric concepts and licenses | <https://learn.microsoft.com/en-us/fabric/enterprise/licenses> |
|  | Multi-Geo support for Fabric | <https://learn.microsoft.com/en-us/fabric/admin/service-admin-premium-multi-geo?tabs=power-bi-premium> |
|  | Transparent data encryption for SQL Database, SQL Managed Instance, and Azure Synapse Analytics | <https://learn.microsoft.com/en-us/azure/azure-sql/database/transparent-data-encryption-tde-overview?view=azuresql&tabs=azure-portal> |
|  | Azure Storage encryption for data at rest | <https://learn.microsoft.com/en-us/azure/storage/common/storage-service-encryption> |
|  | Bring your own encryption keys for Power BI | <https://learn.microsoft.com/en-us/power-bi/enterprise/service-encryption-byok> |
|  | Default Power BI semantic models in Microsoft Fabric | <https://learn.microsoft.com/en-us/fabric/data-warehouse/semantic-models> |
|  | Dynamic data masking in Fabric data warehousing | <https://learn.microsoft.com/en-us/fabric/data-warehouse/dynamic-data-masking> |
|  | User audit logs | <https://learn.microsoft.com/en-us/fabric/data-warehouse/security#user-audit-logs> |
|  | Track user activities in Microsoft Fabric | <https://learn.microsoft.com/en-us/fabric/admin/track-user-activities> |
|  | Access control in the classic Microsoft Purview governance portal | <https://learn.microsoft.com/en-us/purview/classic-data-governance-permissions#roles> |
|  | Understand Microsoft Fabric admin roles | <https://learn.microsoft.com/en-us/fabric/admin/roles> |
|  | Mandatory label policy for Fabric and Power BI | <https://learn.microsoft.com/en-us/fabric/governance/mandatory-label-policy> |
|  | Domain-level default sensitivity labels in Microsoft Fabric | <https://learn.microsoft.com/en-us/fabric/governance/domain-default-sensitivity-label> |
|  | Programmatic labeling | <https://learn.microsoft.com/en-us/fabric/governance/service-security-sensitivity-label-inheritance-set-remove-api> |
|  | Sensitivity labels troubleshooting | <https://learn.microsoft.com/en-us/fabric/governance/service-security-sensitivity-label-troubleshooting> |