NASUNI

Technical white paper

How to use Nasuni data for Microsoft 365 Copilot using the File Share Graph connector

Table of contents

Value overview

The integration of Nasuni file shares into Microsoft's 365 semantic index through the Graph connector offers significant benefits:

Unified search experience

Users can discover content across all connected data sources, including external file shares, through a single search interface.

Intelligent content discovery

AI-powered search capabilities, such as natural language processing and semantic understanding, can be applied to external file share content.

Enhanced knowledge mining

Machine learning algorithms can extract insights, identify patterns, and surface relationships within the indexed Nasuni content, regardless of its original location.

Improved compliance and governance

AI-driven classification and sensitivity labeling can be extended to Nasuni file share content, enhancing data governance practices.

Personalized experiences

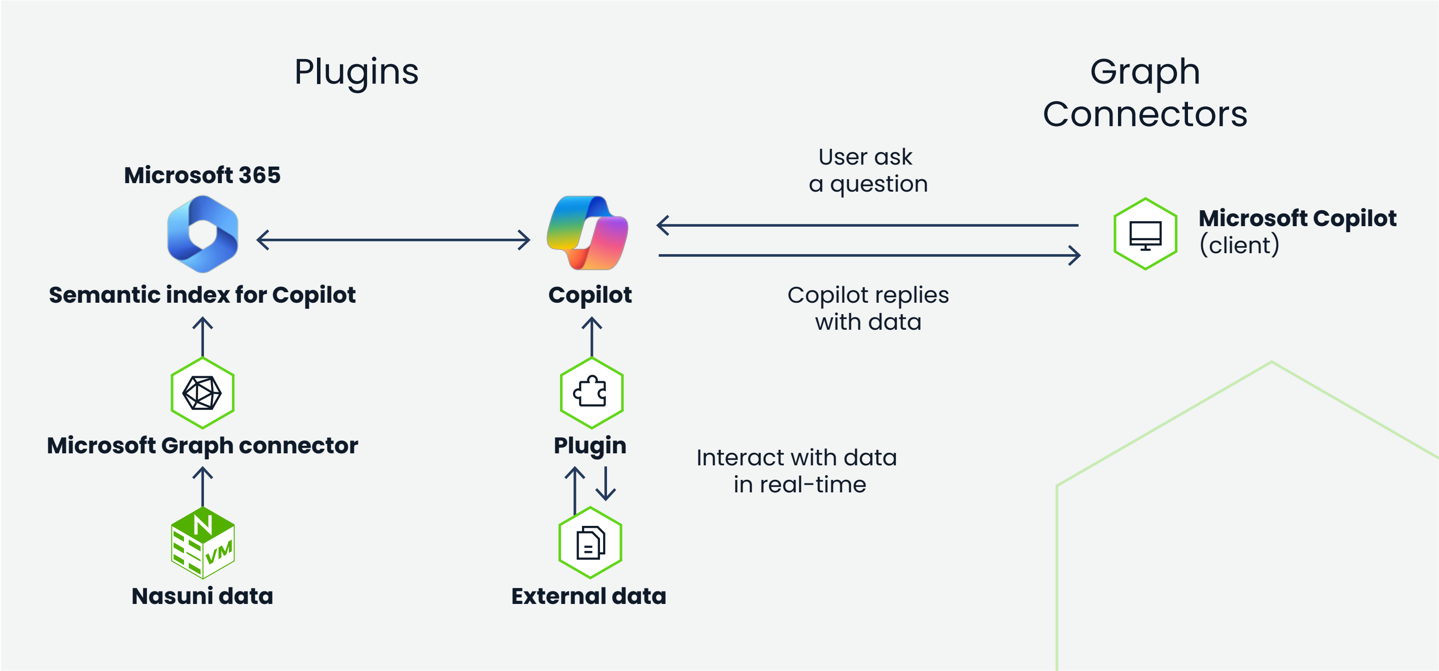
AI can leverage the comprehensive index to deliver more relevant and personalized content recommendations across Microsoft 365 services.

Advanced analytics

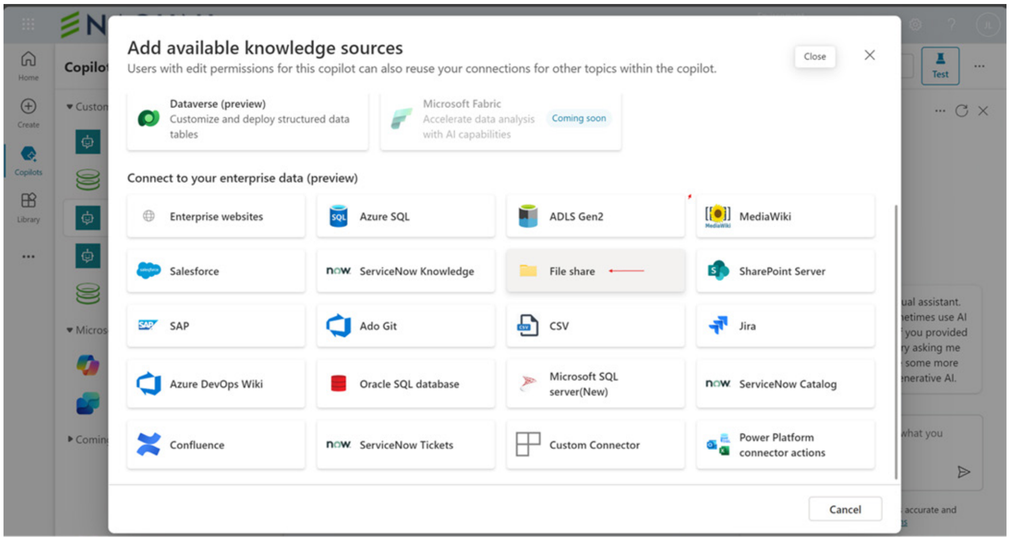
The inclusion of Nasuni file share data in the semantic index enables more comprehensive analytics and reporting capabilities, powered by AI and machine learning.

Recommended reading

- File share Microsoft Graph connector, Microsoft Learn
- Microsoft Graph connector agent, Microsoft Learn
- Microsoft Graph connectors license requirements and pricing, Microsoft Learn
- GitHub Microsoft File share Graph Connector
- List of file types supported by the Microsoft Graph connector


Plugins
Graph Connectors
Microsoft 365
Semantic index for Copilot
Microsoft Graph connector
Nasuni data
Copilot
Plugin
External data
User ask a question
Copilot replies with data
Interact with data in real-time
Microsoft Copilot (client)

Additionally, the File Share Graph connector can also be used in Copilot Studio to enable easier ingestion of Nasuni data sets on-demand.


Add available knowledge sources

Users with edit permissions for this copilot can also reuse your connections for other topics within the copilot.

By implementing the File Share Graph connector, organizations can unlock the full potential of their distributed content, applying Microsoft's advanced AI and search capabilities to drive productivity, innovation, and data-driven decision-making.

Setting up the Microsoft Graph connector agent

Requirements

- Windows 10 Desktop or Windows Server 2019 and higher are supported
- The Connector agent requires Microsoft .NET 7 Desktop or higher
- Azure tenant application registration creation ermissions and authentication
  - Either an Application ID client/secret (demonstrated) or certificate
- Nasuni Edge Appliance Joined to AD or Microsoft Entra DS

Configuration steps

The following steps can be used to quickly configure the integration.

1 Create the app registration in the Azure tenant.

a. Launch the Azure portal

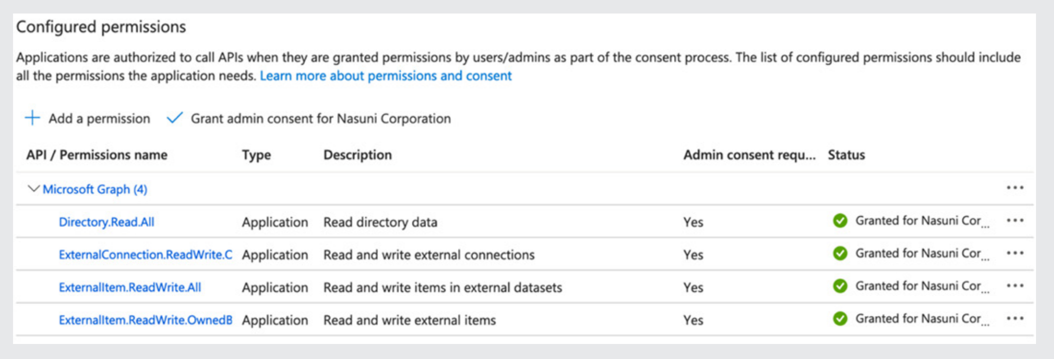
b. Select 'Microsoft Azure Entra ID'

c. Select 'App Registrations'

d. Create a new registration

e. Assign the following API permissions

- Directory.Read.All
- ExternalConnection.ReadWrite.OwnedBy
- ExternalItem.ReadWrite.All
- ExternalItem.ReadWrite.OwnedBy

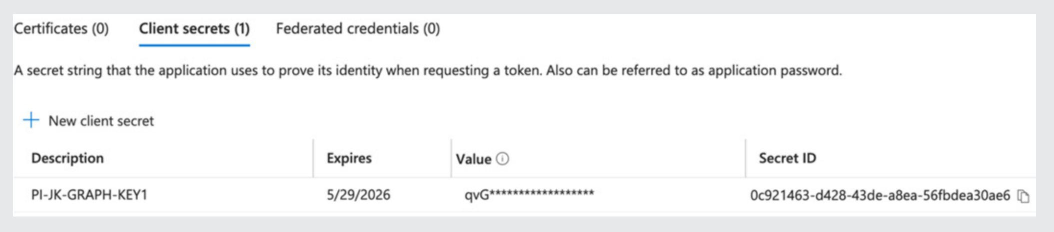


Configured permissions

Applications are authorized to call APIs when they are granted permissions by users/admins as part of the consent process. The list of configured permissions should include all the permissions the application needs. Learn more about permissions and consent

API / Permissions name	Type	Description	Admin consent requ...	Status
Microsoft Graph (4)				
Directory.Read.All	Application	Read directory data	Yes	✓ Granted for Nasuni Cor...
ExternalConnection.ReadWrite.C	Application	Read and write external connections	Yes	✓ Granted for Nasuni Cor...
ExternalItem.ReadWrite.All	Application	Read and write items in external datasets	Yes	✓ Granted for Nasuni Cor...
ExternalItem.ReadWrite.OwnedB	Application	Read and write external items	Yes	✓ Granted for Nasuni Cor...

f. Create a new client secret and record it for the Graph connector setup.



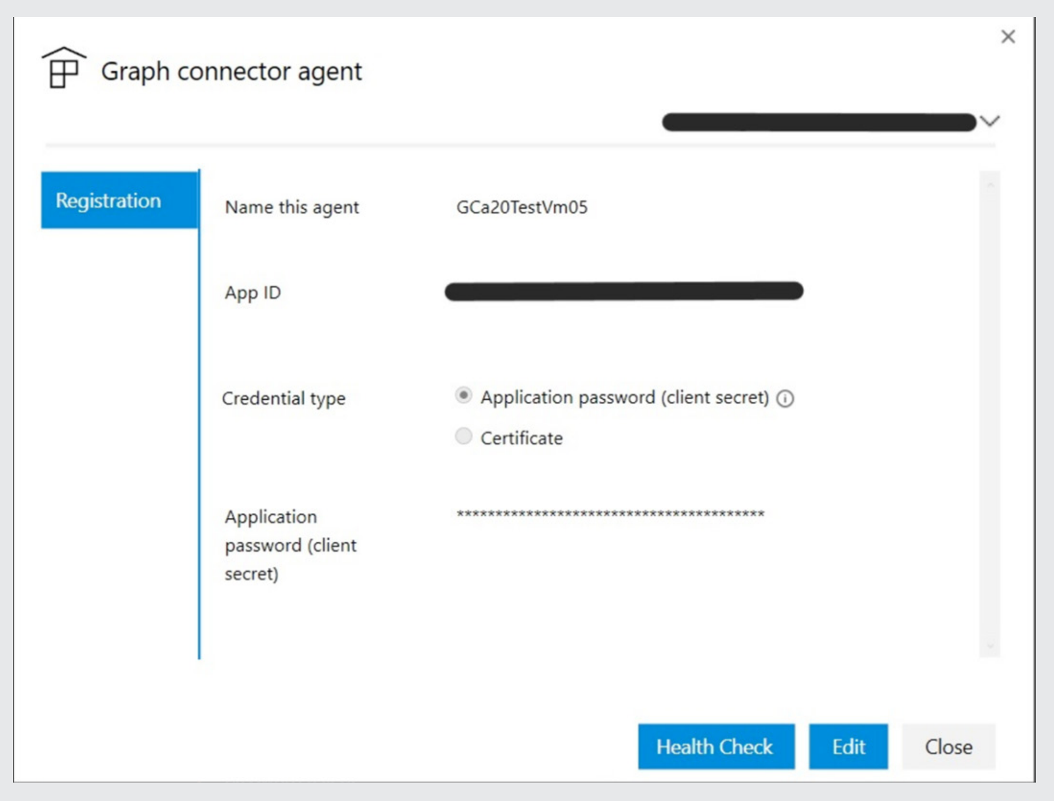
A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.

Description	Expires	Value	Secret ID
PI-JK-GRAPH-KEY1	5/29/2026	qvG******************	0c921463-d428-43de-a8ea-56fbdea30ae6

2 On the Windows VM, download the agent installer.

3 Launch the installer on Windows.

4 Configure the connector by authenticating and providing the app registration.



Graph connector agent

[redacted]

Registration

Name this agent GCa20TestVm05

App ID [redacted]

Credential type

● Application password (client secret)

○ Certificate

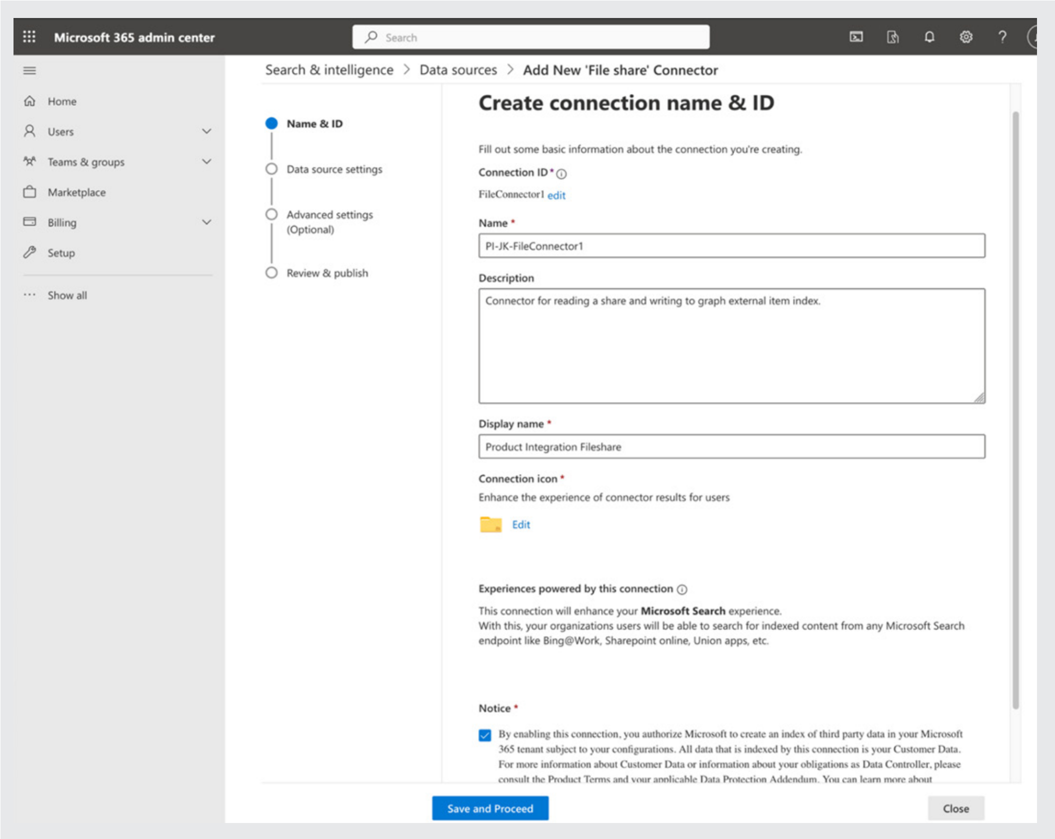
Application password (client secret) **

5 Visit admin.microsoft.com and search ‘Connectors’ page to add the File Share connector:

g. Under ‘Connections’ select ‘Add’.

h. Under ‘Other data sources’ select File Share.

i. Assign a friendly name, description and display name, then at minimum select the consent to allowing Microsoft to index. *Note the Connection ID (e.g. FileConnector1) assigned by Microsoft that is later used in Graph Explorer or custom API requests. Save and proceed.


Microsoft 365 admin center
Search
Home
Users
Teams & groups
Marketplace
Billing
Setup
Show all
Search & intelligence > Data sources > Add New 'File share' Connector
● Name & ID
○ Data source settings
○ Advanced settings (Optional)
○ Review & publish

Create connection name & ID

Fill out some basic information about the connection you're creating.

Connection ID *
FileConnector1 edit

Name *
PI-JK-FileConnector1

Description
Connector for reading a share and writing to graph external item index.

Display name *
Product Integration Fileshare

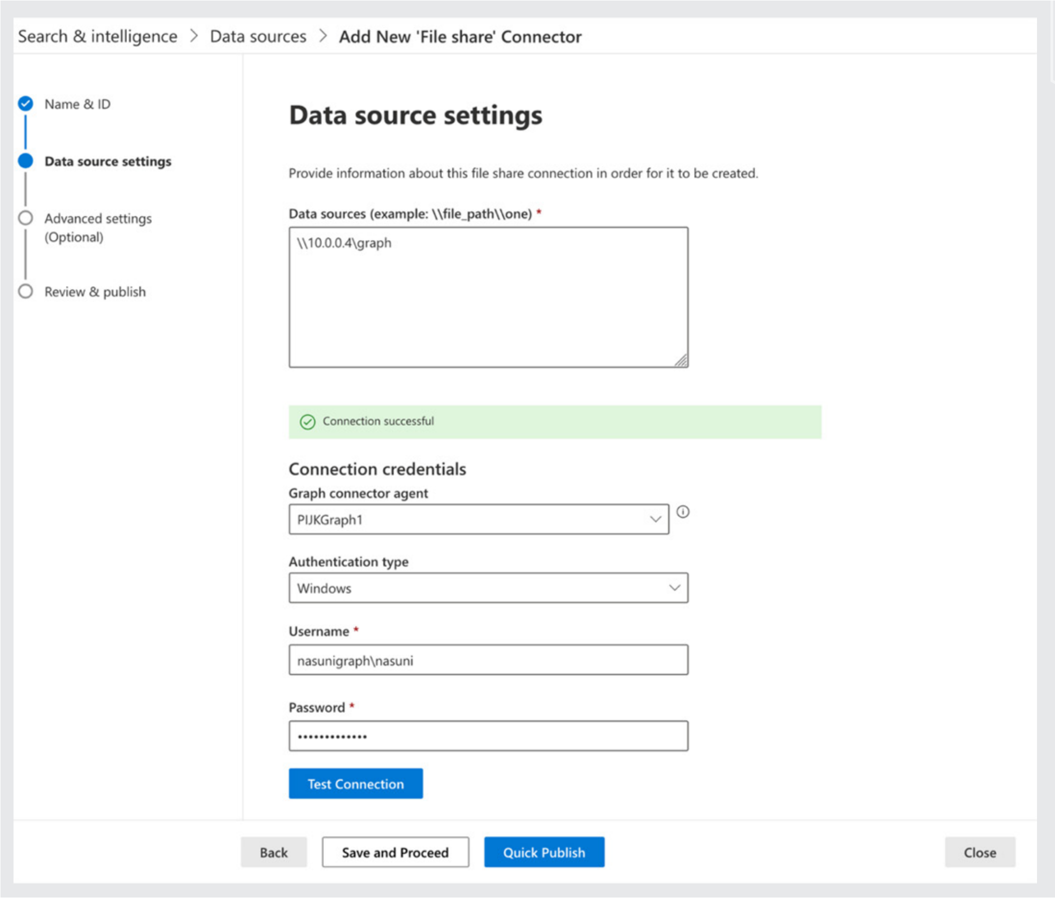
Connection icon *
Enhance the experience of connector results for users
Edit

Experiences powered by this connection
This connection will enhance your **Microsoft Search** experience.
With this, your organizations users will be able to search for indexed content from any Microsoft Search endpoint like Bing@Work, Sharepoint online, Union apps, etc.

Notice *
☑ By enabling this connection, you authorize Microsoft to create an index of third party data in your Microsoft 365 tenant subject to your configurations. All data that is indexed by this connection is your Customer Data. For more information about Customer Data or information about your obligations as Data Controller, please consult the Product Terms and your applicable Data Protection Addendum. You can learn more about

j. Specify the 'Data sources' as UNC path(s) to where on Nasuni the root share paths to scan under.

k. Select the 'Windows' Authentication type and provide the mandatory credentials. Save and proceed.



Data source settings

Provide information about this file share connection in order for it to be created.

Data sources (example: \\file_path\\one) *: \\10.0.0.4\graph

Connection successful

Connection credentials

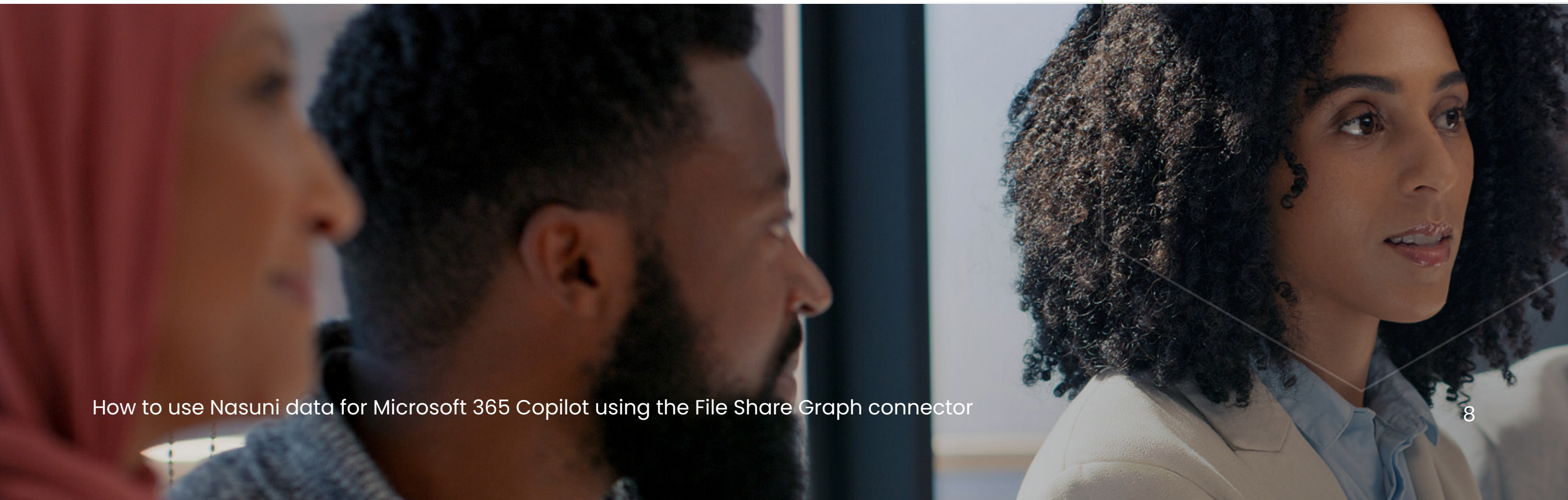
Graph connector agent: PIJKGraph1

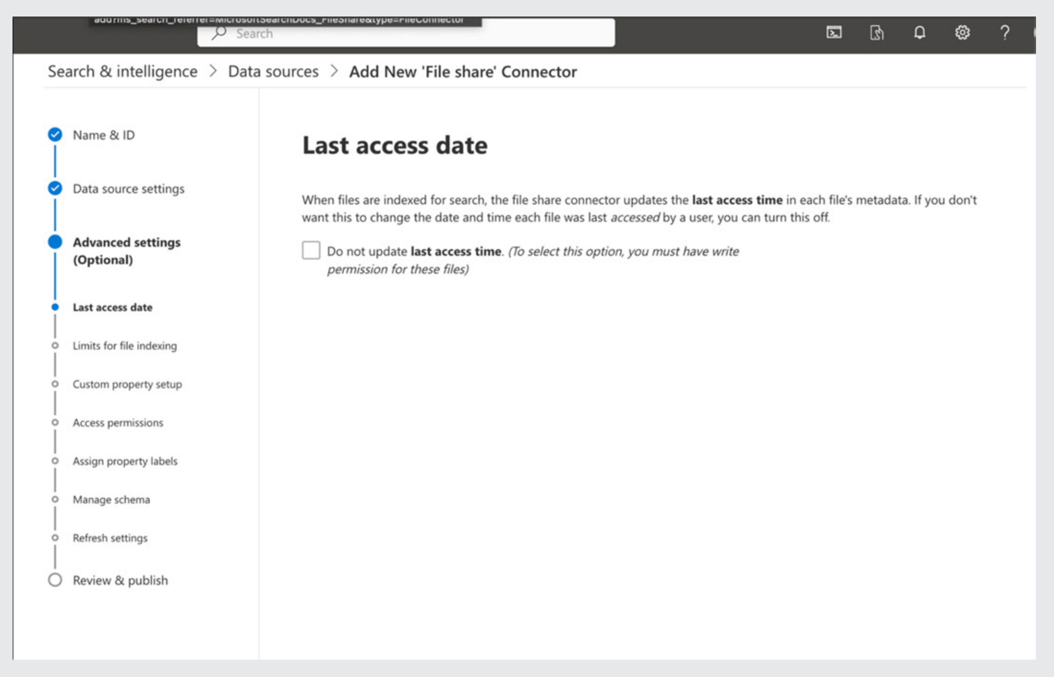
Authentication type: Windows

Username *: nasunigraph\nasuni

Password *: •••••••••••••

l. Ignore the ‘Last access date’ option for the connector. Nasuni does not automatically update time information during simple reads.





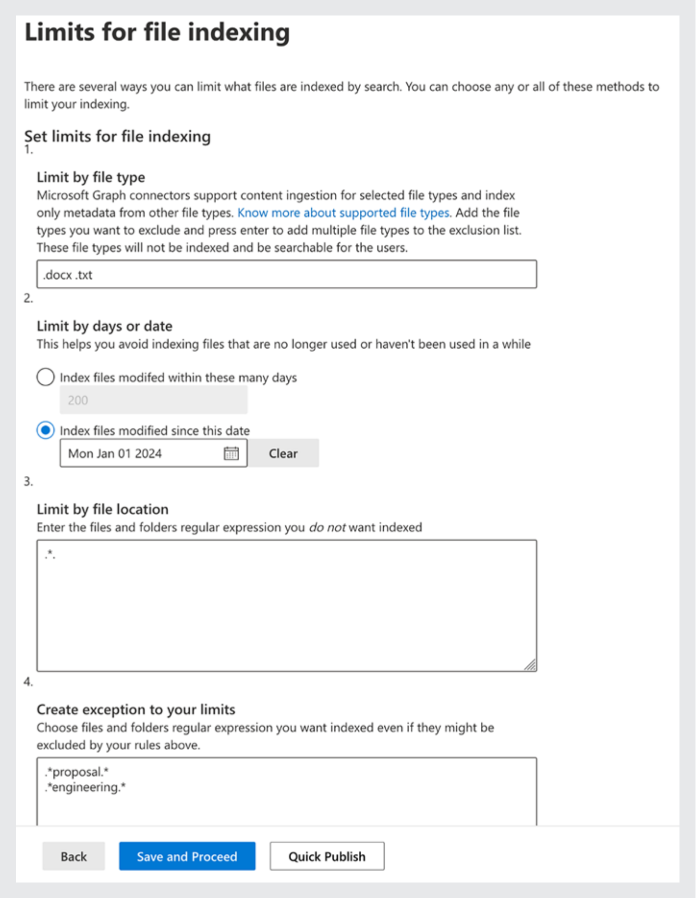
Last access date

When files are indexed for search, the file share connector updates the **last access time** in each file's metadata. If you don't want this to change the date and time each file was last *accessed* by a user, you can turn this off.

□ Do not update **last access time**. *(To select this option, you must have write permission for these files)*

m. Optionally specify limits for indexing:

- Specify files by extensions.
- Limit by days withing or date since last modified.
- Limit by file location to exclude:
  - The restrictive example below limits indexing to files that only contain certain keywords in the filename (regex type filter).
  - The expression .*. excludes everything from indexing.
  - The expressions .*proposal.* and .*engineering.* are exceptions that will index files names that contain the expression.
- Save and proceed.
- Assign the search permissions that will limit the result scope for search and other services like copilot to content that the user has ACL permissions to.



Limits for file indexing

There are several ways you can limit what files are indexed by search. You can choose any or all of these methods to limit your indexing.

Set limits for file indexing

1.

Limit by file type
Microsoft Graph connectors support content ingestion for selected file types and index only metadata from other file types. Know more about supported file types. Add the file types you want to exclude and press enter to add multiple file types to the exclusion list. These file types will not be indexed and be searchable for the users.

.docx .txt

2.

Limit by days or date
This helps you avoid indexing files that are no longer used or haven't been used in a while

○ Index files modifed within these many days
200

● Index files modified since this date
Mon Jan 01 2024

3.

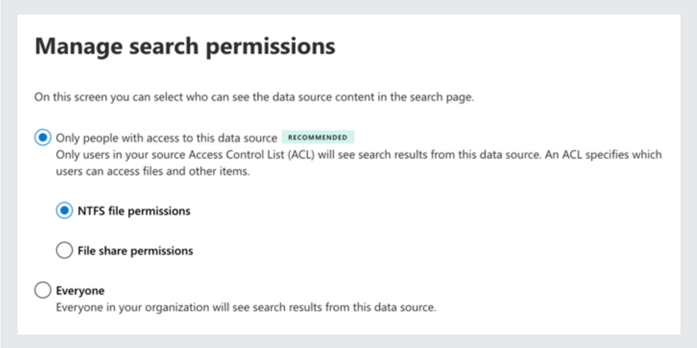
Limit by file location
Enter the files and folders regular expression you *do not* want indexed

.*.

4.

Create exception to your limits
Choose files and folders regular expression you want indexed even if they might be excluded by your rules above.

.*proposal.*
.*engineering.*



Manage search permissions

On this screen you can select who can see the data source content in the search page.

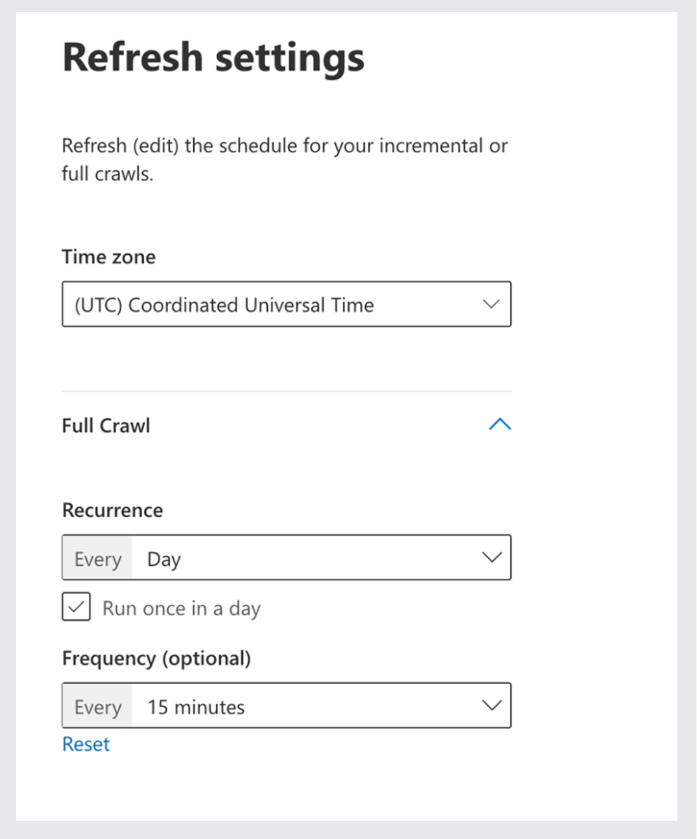
● Only people with access to this data source RECOMMENDED
Only users in your source Access Control List (ACL) will see search results from this data source. An ACL specifies which users can access files and other items.

● NTFS file permissions

○ File share permissions

○ Everyone
Everyone in your organization will see search results from this data source.

- Specify the Refresh interval for 'Full Crawl' (MS File Share is Full Crawl only).
  - The shortest interval is 15 minutes.



Refresh settings

Refresh (edit) the schedule for your incremental or full crawls.

Time zone: (UTC) Coordinated Universal Time

Full Crawl

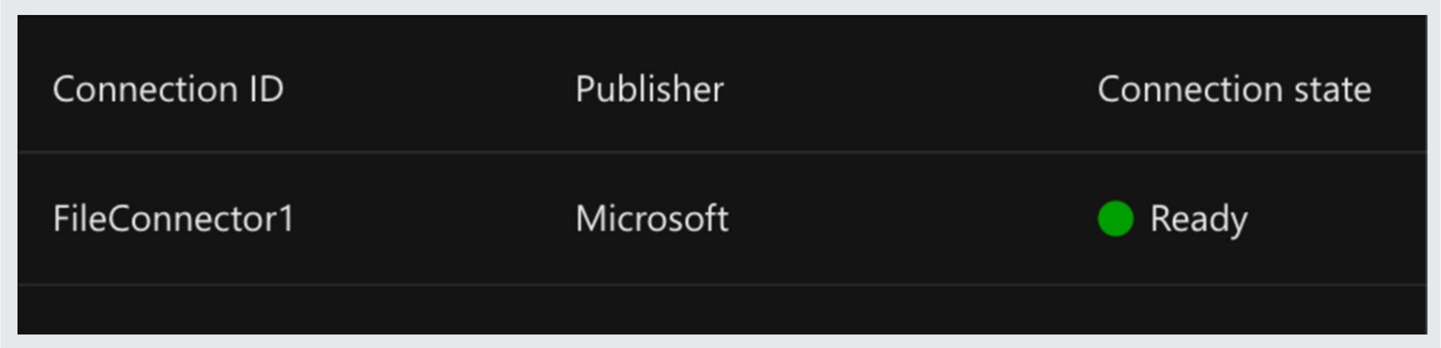
Recurrence: Every Day

☑ Run once in a day

Frequency (optional): Every 15 minutes

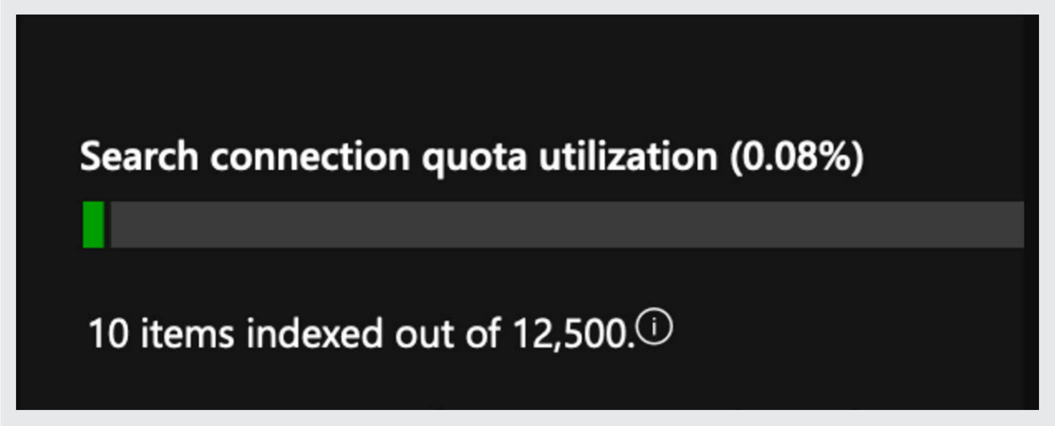
6 Monitor the file indexing process

n. The connector state should display as 'Ready'.



Connection ID	Publisher	Connection state
FileConnector1	Microsoft	● Ready

o. Shortly after the scanning interval has begun, the index should begin to grow.



Search connection quota utilization (0.08%)

10 items indexed out of 12,500.ⓘ

The Graph index can be queried to confirm the ingest of searchable text by using an API tool such as Graph Explorer. In addition to Graph API application consent permissions, the API user's ACL permissions against the file index are checked. If the connector was configured with either NTFS or File share permissions, only records that the API user has access to will be returned.

Troubleshooting / useful information

- General agent configuration errors can be viewed in: C:\Users\{User Account}\AppData\Local\Microsoft\GraphConnectorAgent\ConfigApp\logs
- Errors pertaining to individual file index import can be viewed in summary from the Search Connector page of Microsoft 365 admin center, and fully downloaded using PowerShell.
- See Microsoft Graph connectors view details and errors for more detail.
- By default, Microsoft Graph connectors support up to 50 million items per tenant, which includes any built-in quota bundled into Microsoft 365 copilot, Microsoft 365 or Office 365 E5 licenses.
- Higher item count per connection limit requests can be sent to Microsoft.
- A single File Share connector instance can be configured to use a maximum of 20 paths. The paths can belong to different edge appliances, allowing a fan-out of connections.
- Multiple connector instances each running the Graph connector can be registered and configured uniquely, allowing different schedules, paths and search verticals.

- The maximum supported file size that the Graph connector supports is 100 MB. Files that exceed 100 MB aren't indexed.
- While the connector performs reads on larger files, ultimately the Graph data limit for inserting body text when ingesting and indexing an item is 4 MB.
  - Note: The 4 MB item size limit refers to the total size of parsed text content that is typically 10% of the original file size for common formats (for example, docx, ppt, and PDF). To contextualize, 4 MB equals 4,000,000 bytes that translates to approximately 600K-700K words or 1,400 pages (averaging 500 words per page).
- It's possible for Graph API throttling to provide back pressure of new API requests. In single node testing, none were experienced, however the application is designed to automatically respond and recover to throttling events which could impact ingest performance.
  - Microsoft Graph throttling guidance - Microsoft Graph
- When a file gets added to the index as a Graph "external item", it consumes one unit of index quota. Should that file no longer exist the connector following a refresh, its entry in the index will be removed. Therefore, the allotted capacity is based on the number of actively available (now) files that are configured for scanning.

Nasuni filesystem interaction

The File Share portion of the Graph connector agent running on a Windows instance leverages the default SMB client library to read the Nasuni Edge Appliance share/path. A refresh undergoes two main stages:

Full directory walk

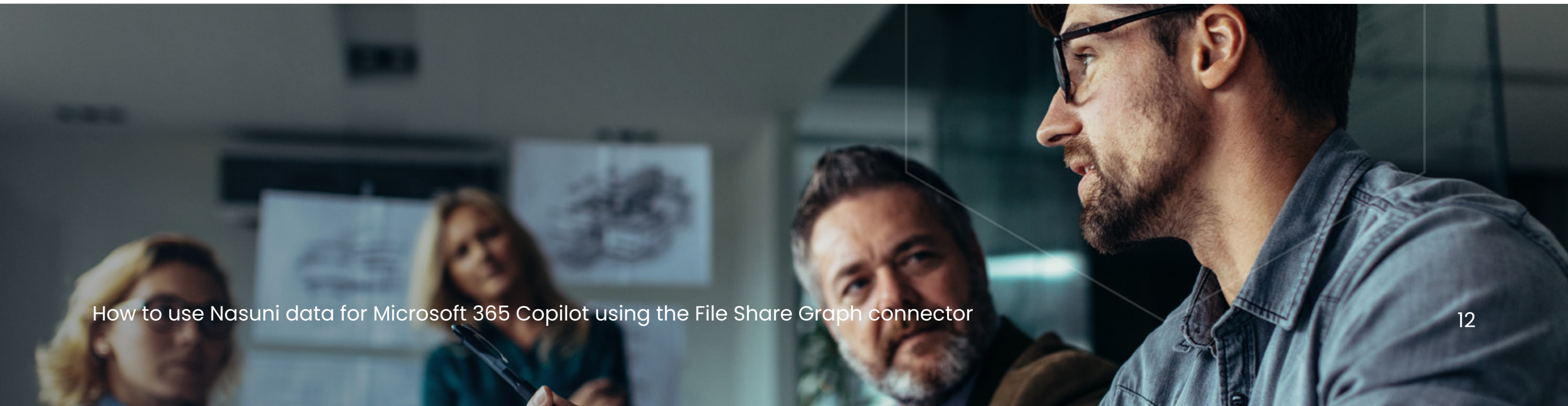
- The connector discovers directories and files.
- It reads file metadata and ACLs.
- This full walk occurs during every refresh interval.

Read new files for indexing into Graph

- Read the content of new files for adding to the Graph index.
- Remove previous index entries for any removed files.
- Update (re-read metadata) and/or re-read changed file contents.

The two stages are observed to run simultaneously. Each refresh iteration runs similarly. The full directory structure is walked, then previously un-indexed files are read.

Note: Anytime the renaming of a parent directory has occurred, the next scan interval will force all files under the parent directory to again undergo reading, as if they were newly discovered.



File ACL permissions

Each Graph connector added to a tenant can be configured with the following security setting options:

Everyone: Allow all indexed items to be searchable by any Graph API authenticated user.

File Share: Anyone with AD/Entra file share read permissions are allowed to search the same indexed items that were readable by the connector through that share permission.

NTFS ACLs: The most secure option that reads AD/Entra NTFS ACLs assigned to the files, then filters search results based on access to the relevant files. Matched indexes to files that are not accessible to the user are quietly omitted from the results.

Nasuni File Share connector best practices

- Pinning metadata cache can reduce the cumulative return time of scans that include metadata reading, especially in a mode where all scanning occurs on a dedicated single edge appliance with access to one or more volumes. When pinning metadata cache for frequent scanning, Nasuni recommends a dedicated appliance to account for the increase in snapshot sync time that could impact other processes.
- If the Graph API permission application registration resides in an Azure AD/Entra domain, ensure federated domains/domain trusts are in place so that ACLs discovered on Nasuni are accessible by Azure AD/Entra users.
- Evaluate the potential impact of using Global Locking, particularly for initial ingest/reading.
- To control index capacity costs, limit the indexing to any or all of paths, file types, date modified, or path/filename expression matching.
- Ignore the 'Last access date' option for the connector. Nasuni does not automatically update time information during simple reads.

Let’s talk

Want to find out more about how Nasuni can provide your business with a fluid data infrastructure designed for the hybrid cloud world?

Nasuni’s hybrid cloud platform unifies file and object data storage to deliver effortless scale and control at the network edge.

Learn more

Nasuni is a scalable data platform for enterprises facing an explosion of unstructured data in an AI world, eliminating the choice between expensive tinkering or an overwhelming transformation of your entire data infrastructure.

The Nasuni File Data Platform delivers effortless scale in hybrid cloud environments, enables control at the network edge, and meets the modern enterprise expectation for protected, insight- and AI-ready data. It simplifies file data management while increasing access and performance.

Consolidate data, cut costs, and empower users – all while transforming your data from obstacle into opportunity.

NASUNI