Exploring The Possibilities of Azure Fabric Abuses

Viktor Gazdag | Cloud Village | DefCon 33

WhoAmI

- Principal Security Consultant NCC Group
- Cloud and CI/CD Security
- Azure Certs: AZ-104, AZ-400, AZ-500, SC-100, CARTP, CAWASP
- Black Hat USA 2022, DefCon 31/32/33 Cloud Village, DoD CyberDT XSWG #15, DefCamp

What Is Fabric?

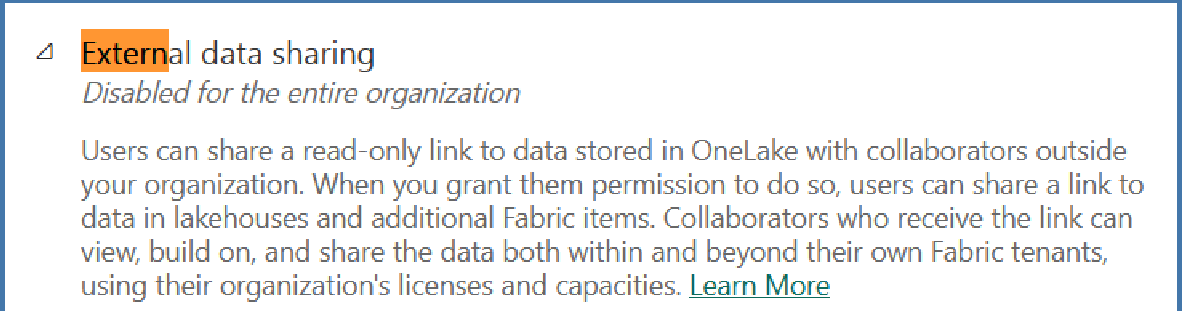
- “An enterprise-ready, end-to-end analytics platform. It unifies data movement, data processing, ingestion, transformation, real-time event routing, and report building. It supports these capabilities with integrated services like Data Engineering, Data Factory, Data Science, Real-Time Intelligence, Data Warehouse, and Databases.”
- SaaS analytics platform including Data Engineering, Data Warehouse, Synapse, Database etc.

Tenant Level Settings

- The Top 3 Settings To Review Are Related To:
- Sharing
- Service Principal
- User Function

Tenant Level

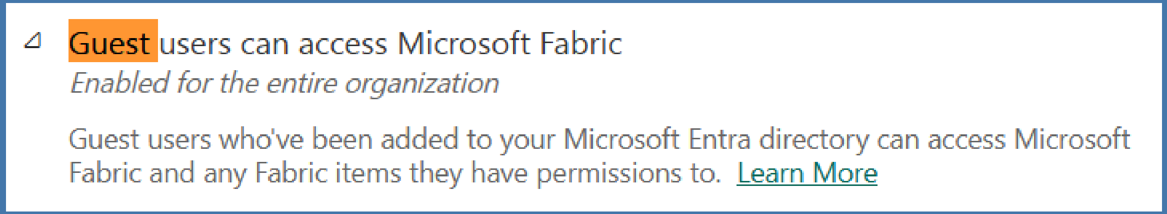
- Export and Sharing
  - External Data Sharing
  - Guest Users Can Access Microsoft Fabric



External data sharing

Disabled for the entire organization

Users can share a read-only link to data stored in OneLake with collaborators outside your organization. When you grant them permission to do so, users can share a link to data in lakehouses and additional Fabric items. Collaborators who receive the link can view, build on, and share the data both within and beyond their own Fabric tenants, using their organization's licenses and capacities. Learn More



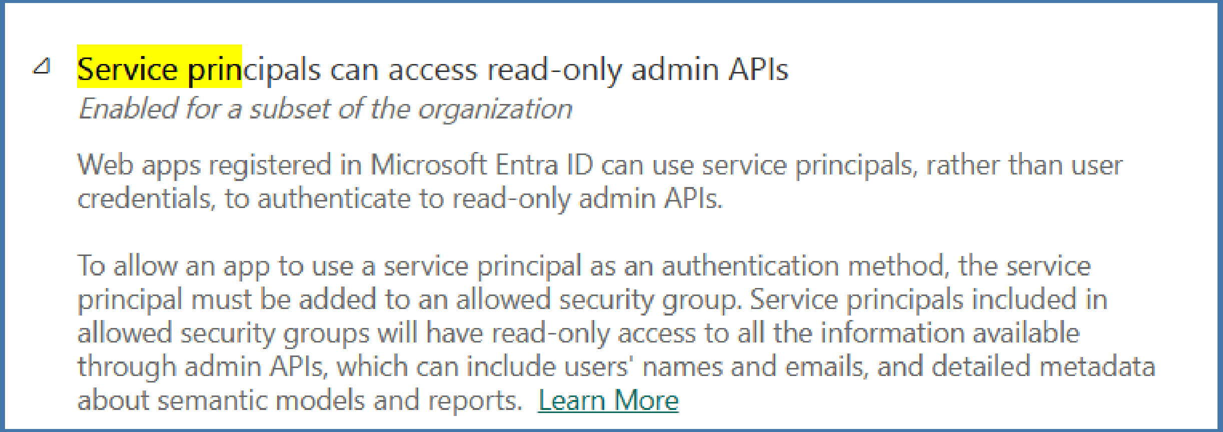
Guest users can access Microsoft Fabric

Enabled for the entire organization

Guest users who've been added to your Microsoft Entra directory can access Microsoft Fabric and any Fabric items they have permissions to. Learn More

Tenant Level

- Service Principal Can Access Read-only Admin APIs
- Service Principal Can Call Fabric Public APIs

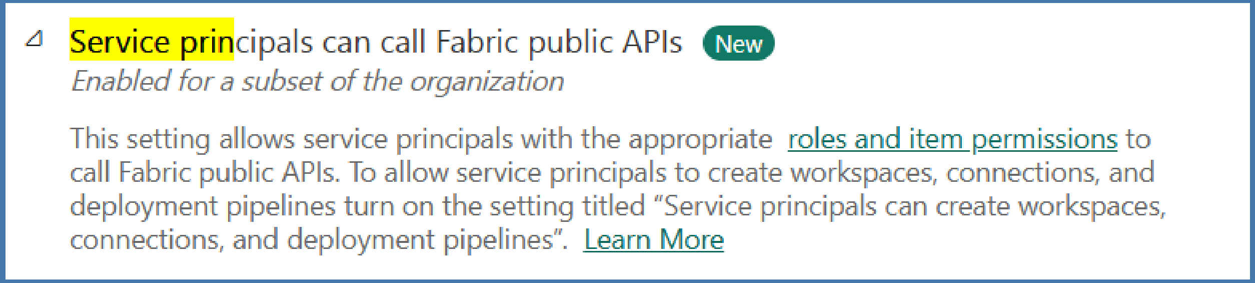


Service principals can access read-only admin APIs

Enabled for a subset of the organization

Web apps registered in Microsoft Entra ID can use service principals, rather than user credentials, to authenticate to read-only admin APIs.

To allow an app to use a service principal as an authentication method, the service principal must be added to an allowed security group. Service principals included in allowed security groups will have read-only access to all the information available through admin APIs, which can include users' names and emails, and detailed metadata about semantic models and reports. Learn More



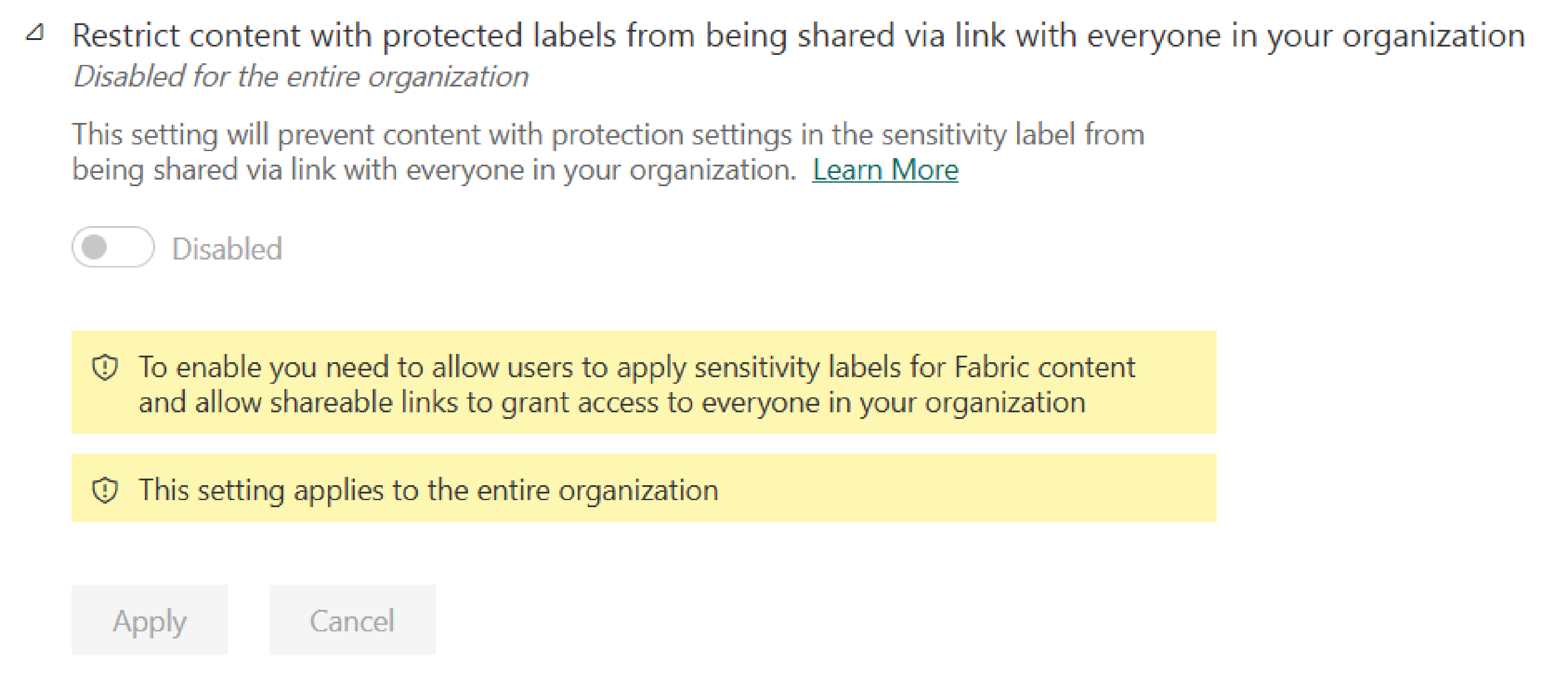
Service principals can call Fabric public APIs New

Enabled for a subset of the organization

This setting allows service principals with the appropriate roles and item permissions to call Fabric public APIs. To allow service principals to create workspaces, connections, and deployment pipelines turn on the setting titled “Service principals can create workspaces, connections, and deployment pipelines”. Learn More

Tenant Level

- Restrict Content With Protected Labels From Being Shared Via Link With Everyone In Your Organization



◿ Restrict content with protected labels from being shared via link with everyone in your organization

Disabled for the entire organization

This setting will prevent content with protection settings in the sensitivity label from being shared via link with everyone in your organization. Learn More

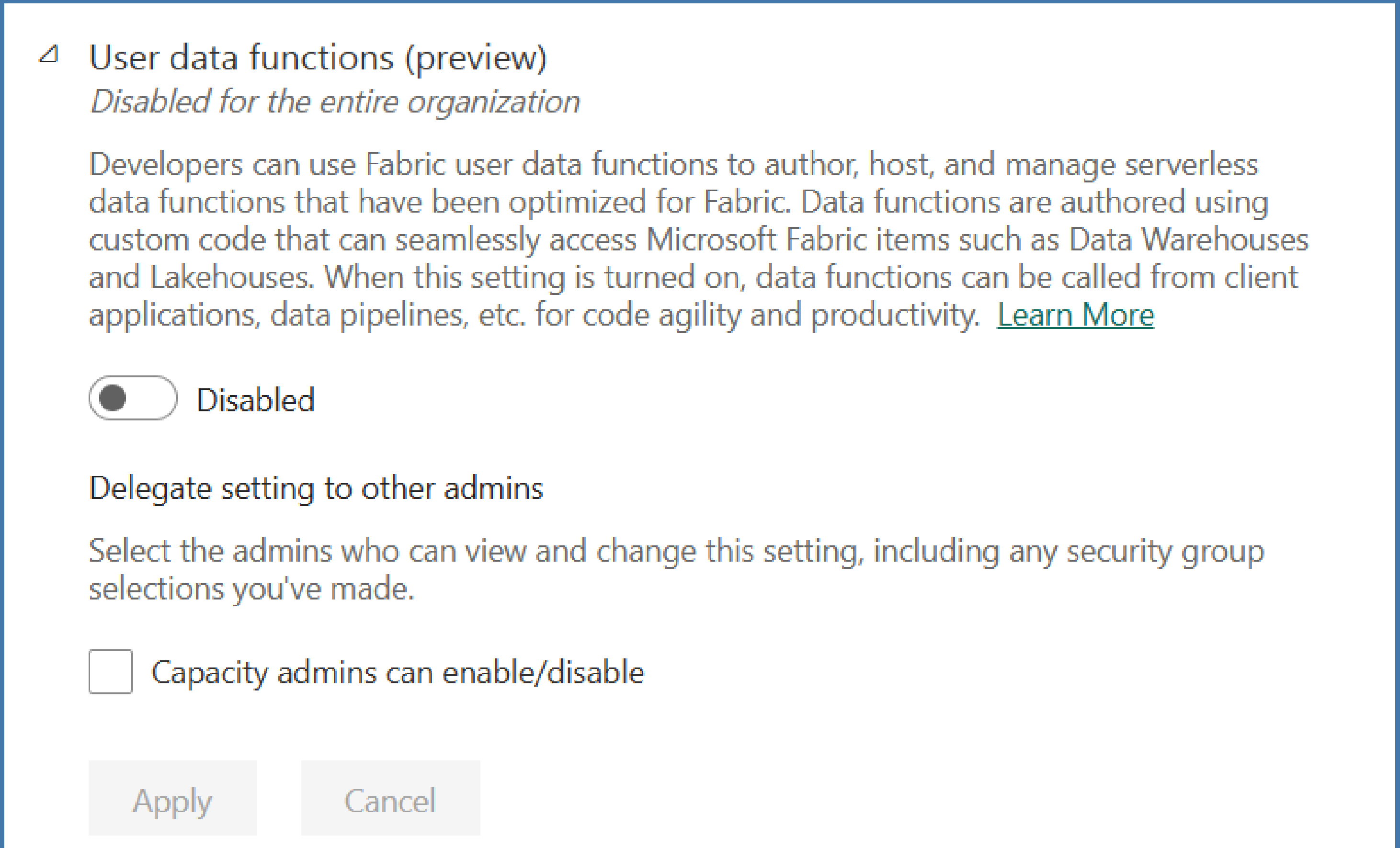
○ Disabled

To enable you need to allow users to apply sensitivity labels for Fabric content and allow shareable links to grant access to everyone in your organization

This setting applies to the entire organization

Tenant Level

- User Data Functions (Only To Specific Groups)



◿ User data functions (preview)

Disabled for the entire organization

Developers can use Fabric user data functions to author, host, and manage serverless data functions that have been optimized for Fabric. Data functions are authored using custom code that can seamlessly access Microsoft Fabric items such as Data Warehouses and Lakehouses. When this setting is turned on, data functions can be called from client applications, data pipelines, etc. for code agility and productivity. Learn More

○ Disabled

Delegate setting to other admins

Select the admins who can view and change this setting, including any security group selections you've made.

□ Capacity admins can enable/disable

Tenant Level

- There Are More Than 3 Settings
- These Were Only The Top 3 Settings I Pick

- Conditional Access Policy
- Private Links For Secure Access To Fabric

Exploring Possibilities

- Backdoor
- Data Exfiltration

Backdoor

- Activator (Can Call Below)
- Notebook (Python SDK, 3rd Party Library)
- Power Automate (Low Code No Code, Entra ID Steps)
- Scheduled Spark Job (Python Code, Library Management Via Environment)

Backdoor

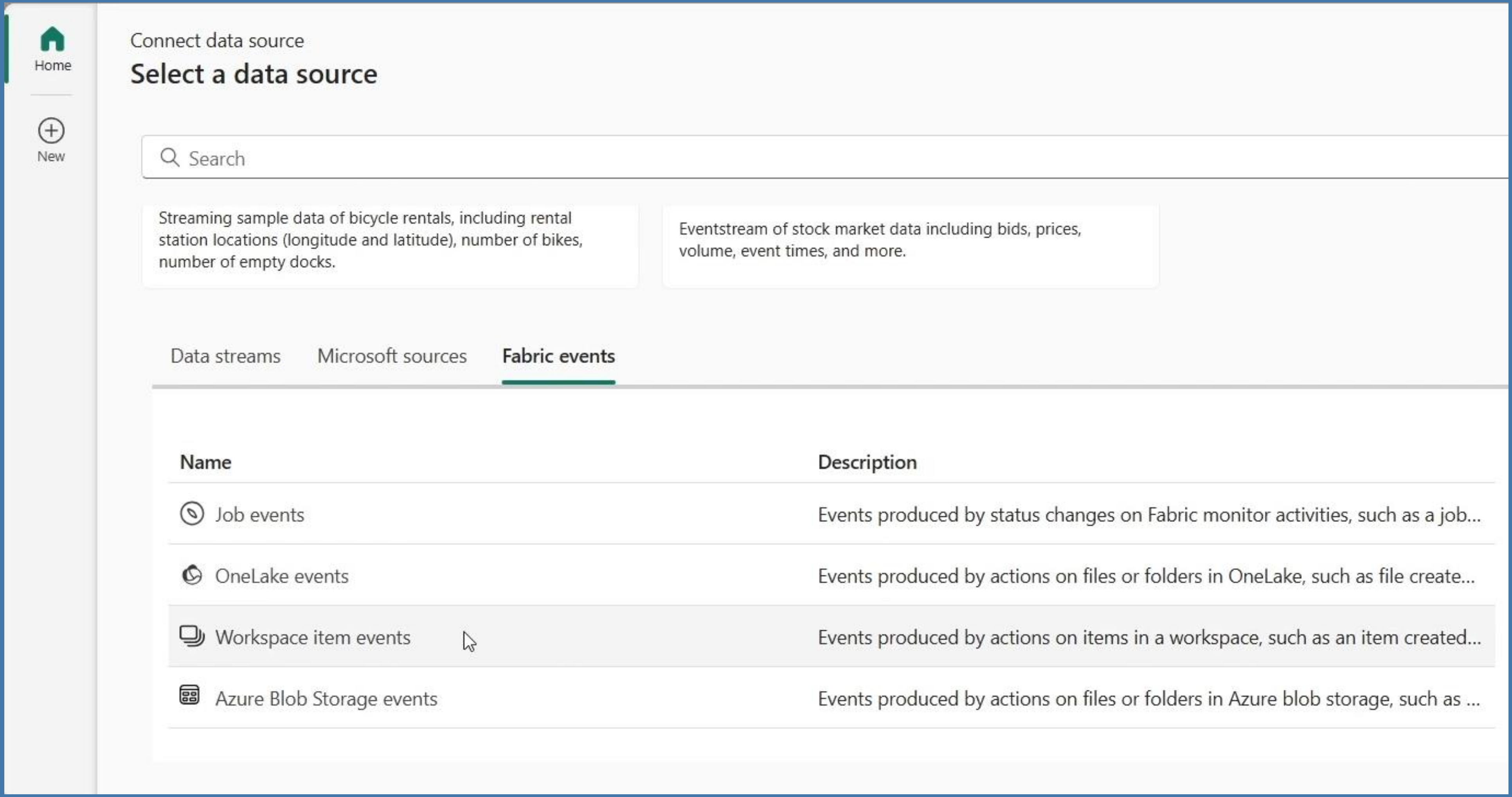
- Activator
  - Event Trigger
- Notebook
  - Azure Python SDK
  - Library Management
- User Or Identity
  - User
  - Enable Identity At Tenant Level Settings

Backdoor

- What Is Activator
  - Event Triggers For Fabric
    - Job Events
    - OneLake Events
    - Workspace Item Events
    - Azure Blog Storage Events

- Create/Update/Delete
- Succeeded/Failed
- Delay

Backdoor



Connect data source

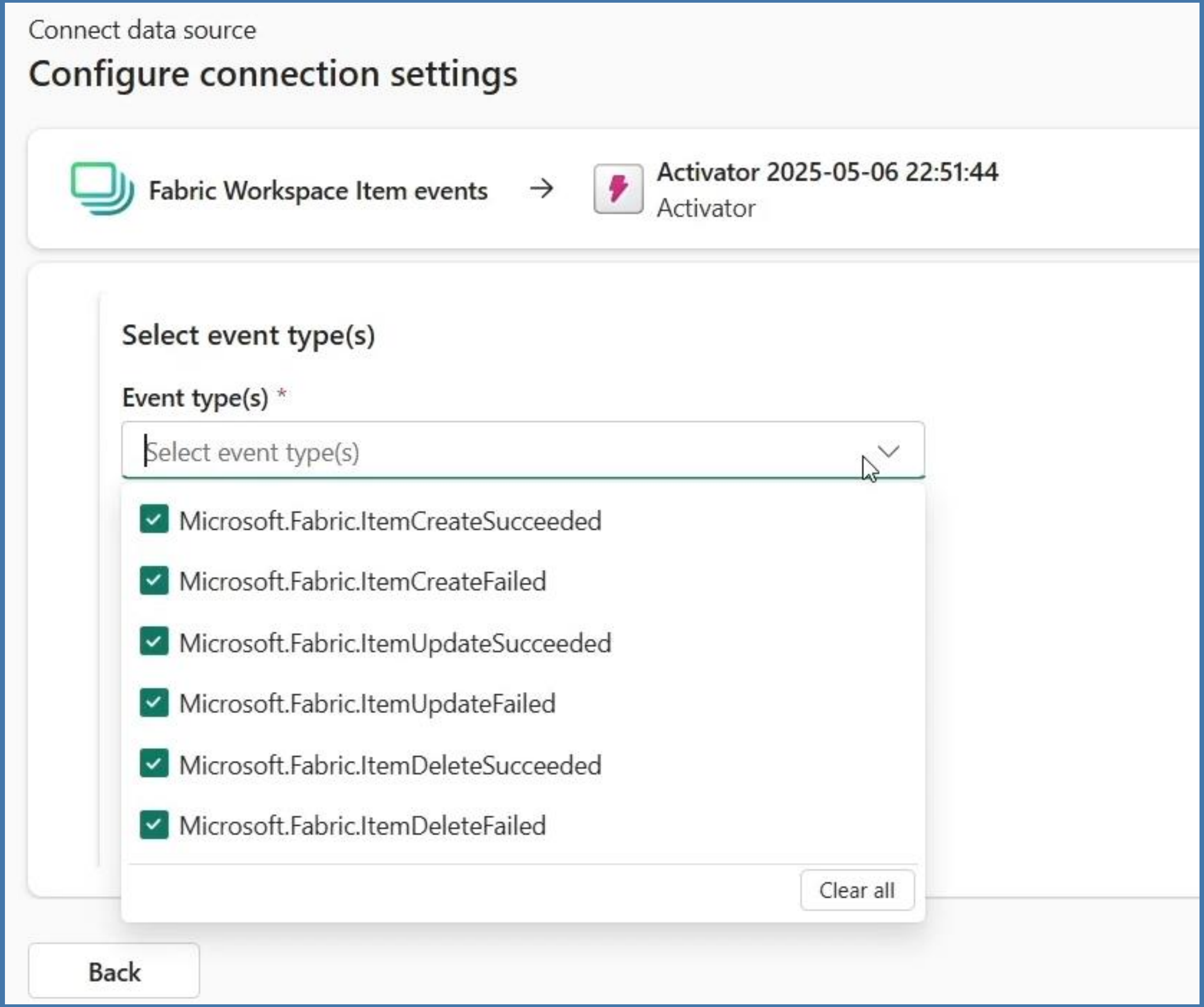
Select a data source

Streaming sample data of bicycle rentals, including rental station locations (longitude and latitude), number of bikes, number of empty docks.

Eventstream of stock market data including bids, prices, volume, event times, and more.

Name	Description
Job events	Events produced by status changes on Fabric monitor activities, such as a job...
OneLake events	Events produced by actions on files or folders in OneLake, such as file create...
Workspace item events	Events produced by actions on items in a workspace, such as an item created...
Azure Blob Storage events	Events produced by actions on files or folders in Azure blob storage, such as ...

Backdoor



Connect data source

Configure connection settings

Fabric Workspace Item events → Activator 2025-05-06 22:51:44
Activator

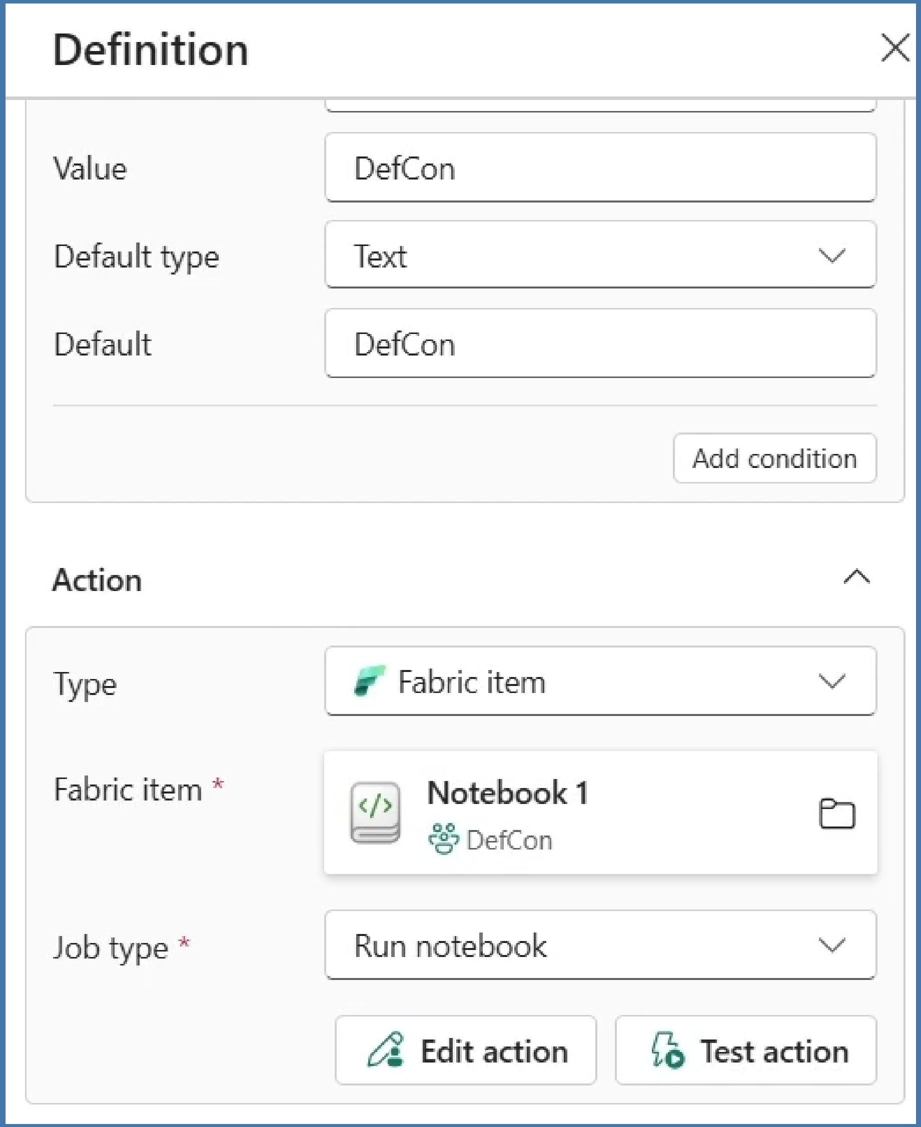
Select event type(s)

Event type(s) *

Select event type(s)

- ☑ Microsoft.Fabric.ItemCreateSucceeded
- ☑ Microsoft.Fabric.ItemCreateFailed
- ☑ Microsoft.Fabric.ItemUpdateSucceeded
- ☑ Microsoft.Fabric.ItemUpdateFailed
- ☑ Microsoft.Fabric.ItemDeleteSucceeded
- ☑ Microsoft.Fabric.ItemDeleteFailed

Backdoor



Definition

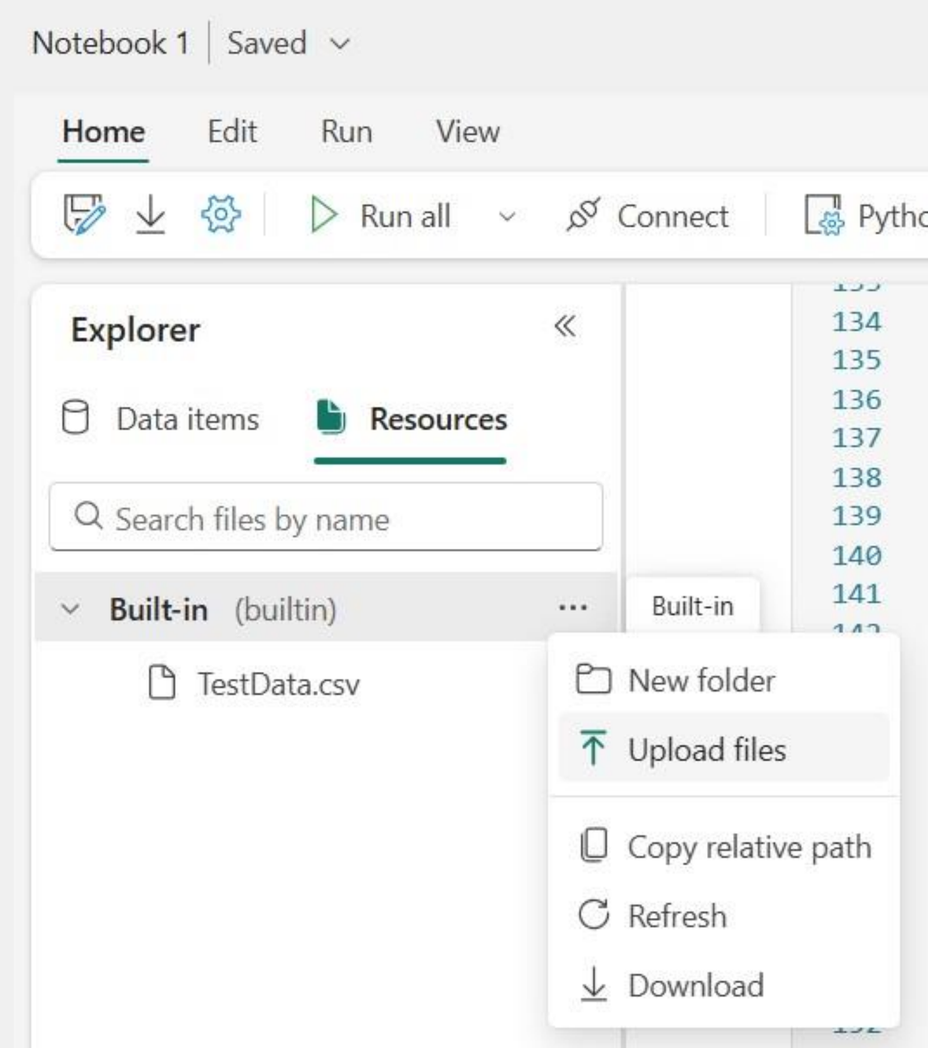
Value	DefCon
Default type	Text
Default	DefCon

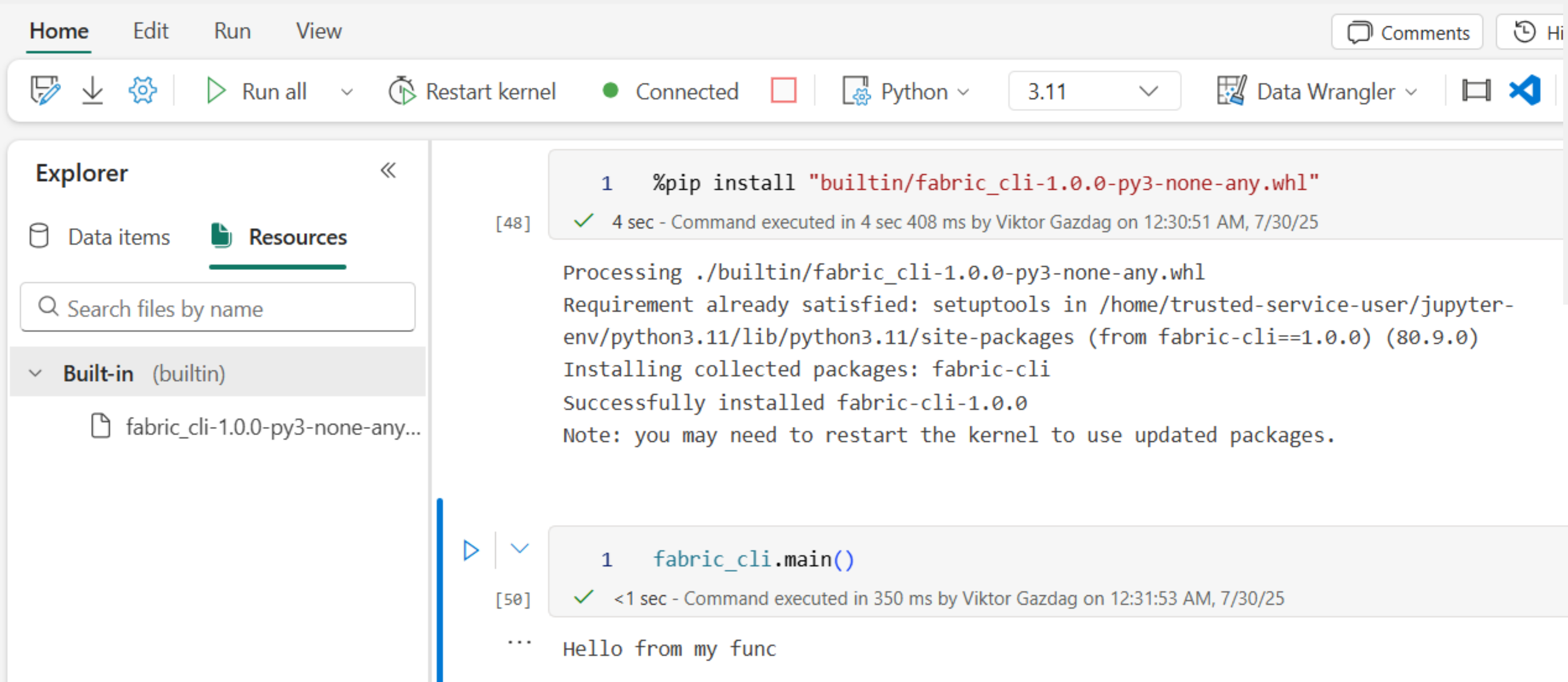
Action

Type	Fabric item
Fabric item *	Notebook 1 DefCon
Job type *	Run notebook

Backdoor

- Notebook
- Azure Python SDK
- 3rd Party Library





```
[48]
%pip install "builtin/fabric_cli-1.0.0-py3-none-any.whl"
```

✓ 4 sec - Command executed in 4 sec 408 ms by Viktor Gazdag on 12:30:51 AM, 7/30/25

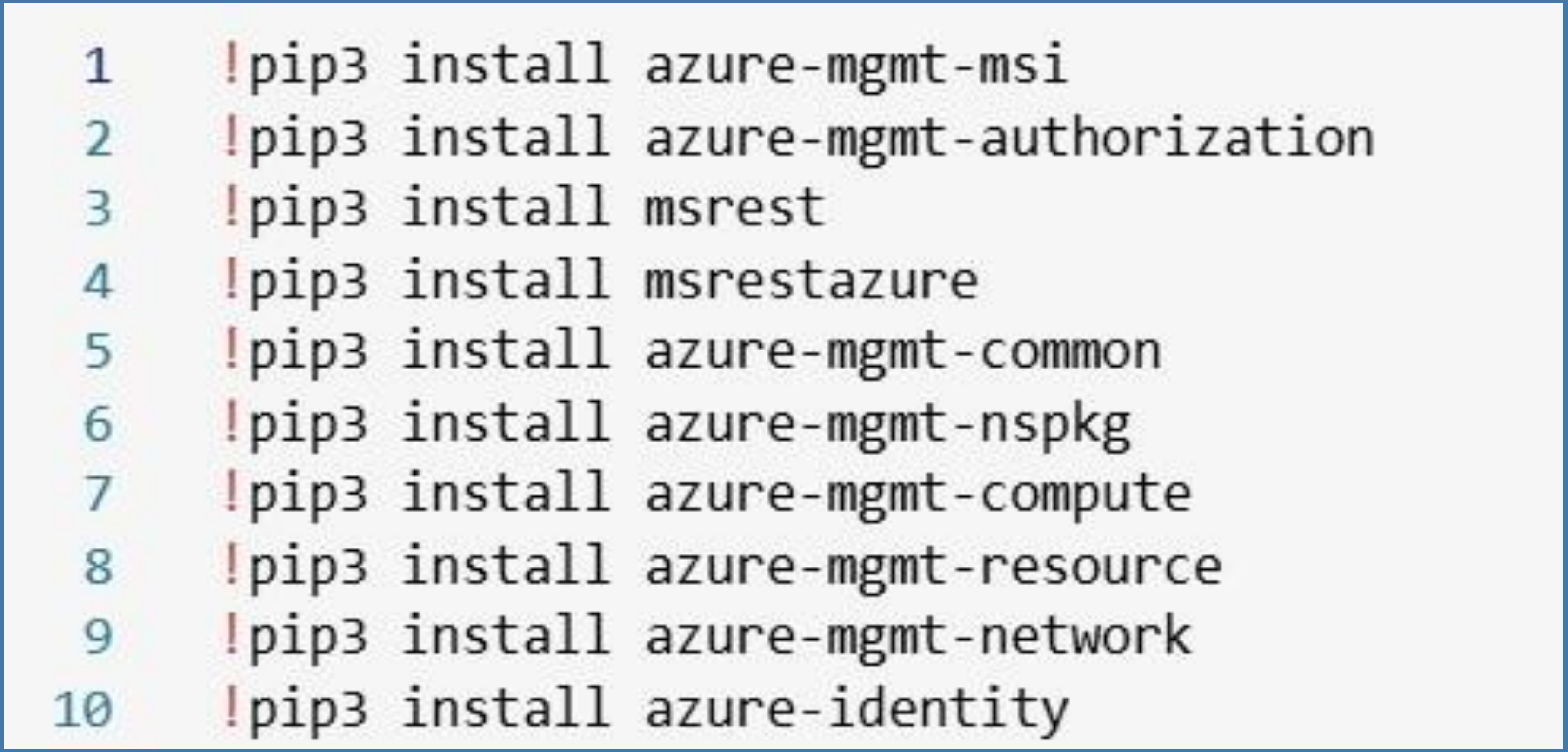
```
Processing ./builtin/fabric_cli-1.0.0-py3-none-any.whl
Requirement already satisfied: setuptools in /home/trusted-service-user/jupyter-
env/python3.11/lib/python3.11/site-packages (from fabric-cli==1.0.0) (80.9.0)
Installing collected packages: fabric-cli
Successfully installed fabric-cli-1.0.0
Note: you may need to restart the kernel to use updated packages.
```

```
[50]
fabric_cli.main()
```

✓ <1 sec - Command executed in 350 ms by Viktor Gazdag on 12:31:53 AM, 7/30/25

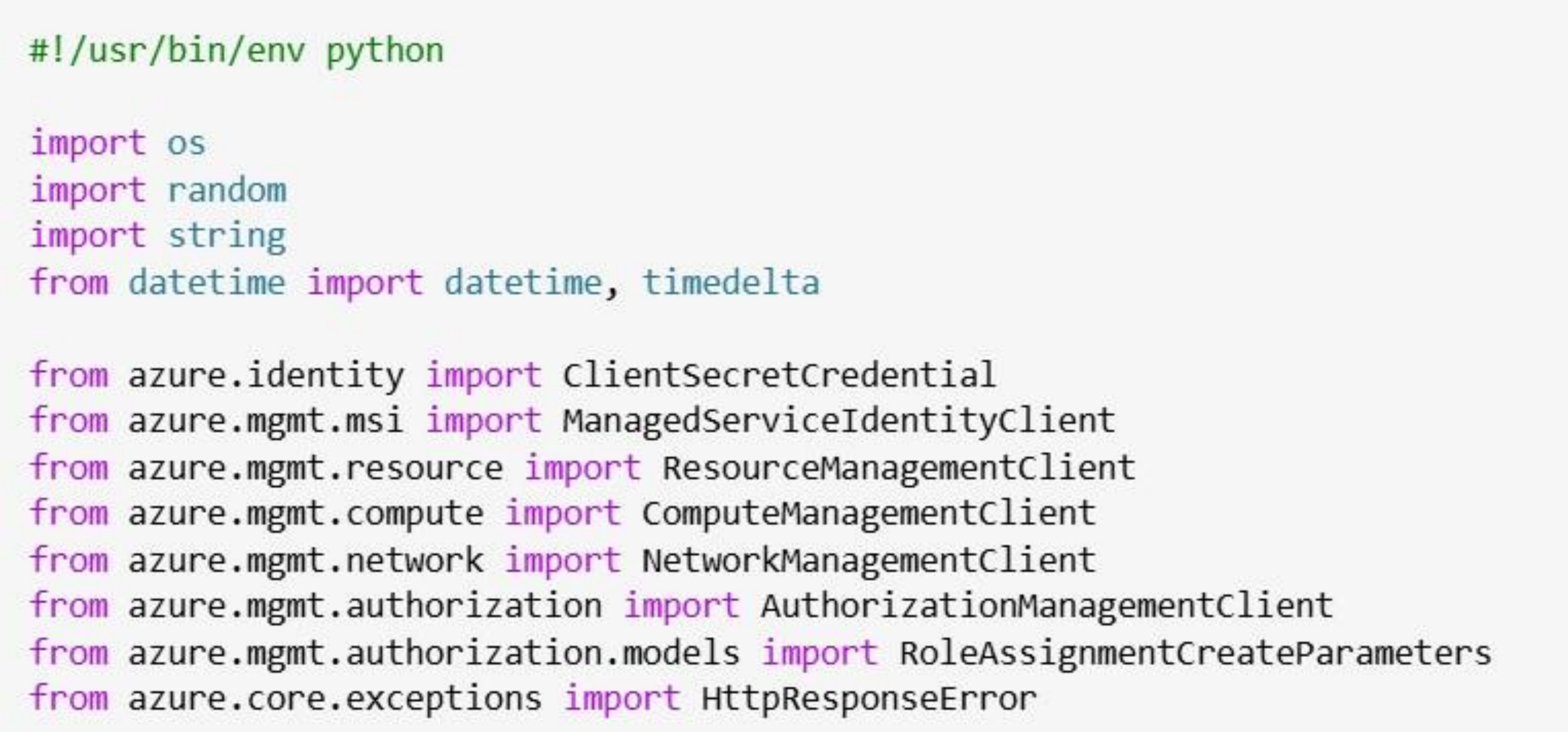
```
Hello from my func
```

Azure Python SDK – Packages Install



```
!pip3 install azure-mgmt-msi
!pip3 install azure-mgmt-authorization
!pip3 install msrest
!pip3 install msrestazure
!pip3 install azure-mgmt-common
!pip3 install azure-mgmt-nspkg
!pip3 install azure-mgmt-compute
!pip3 install azure-mgmt-resource
!pip3 install azure-mgmt-network
!pip3 install azure-identity
```

Azure Python SDK - Import

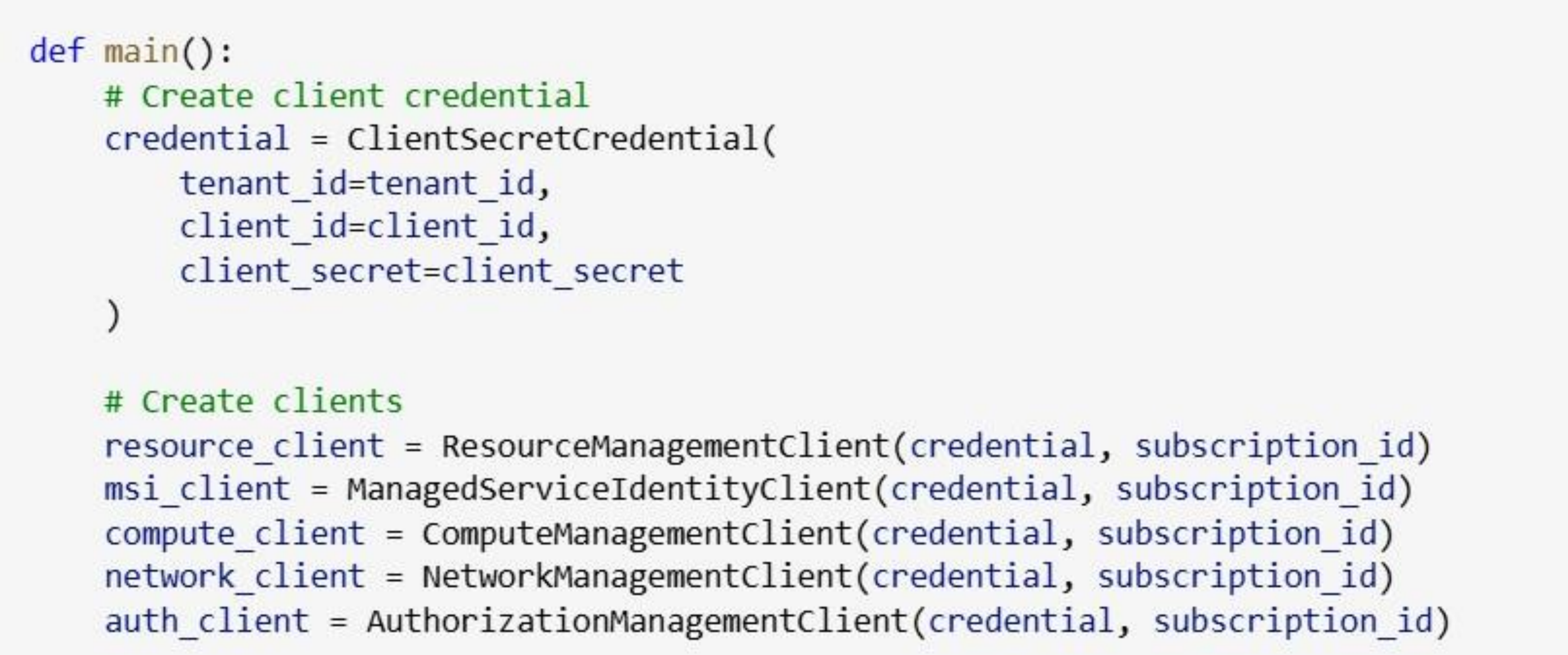


```
#!/usr/bin/env python

import os
import random
import string
from datetime import datetime, timedelta

from azure.identity import ClientSecretCredential
from azure.mgmt.msi import ManagedServiceIdentityClient
from azure.mgmt.resource import ResourceManagementClient
from azure.mgmt.compute import ComputeManagementClient
from azure.mgmt.network import NetworkManagementClient
from azure.mgmt.authorization import AuthorizationManagementClient
from azure.mgmt.authorization.models import RoleAssignmentCreateParameters
from azure.core.exceptions import HttpResponseError
```

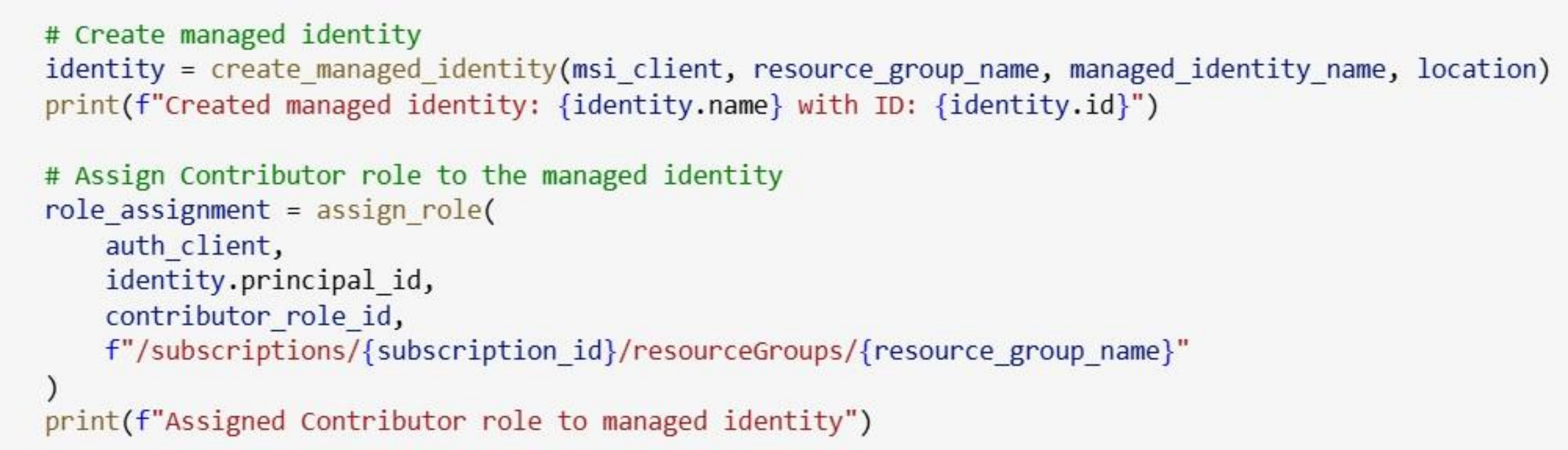
Azure Python SDK - Login



```python
def main():
    # Create client credential
    credential = ClientSecretCredential(
        tenant_id=tenant_id,
        client_id=client_id,
        client_secret=client_secret
    )

    # Create clients
    resource_client = ResourceManagementClient(credential, subscription_id)
    msi_client = ManagedServiceIdentityClient(credential, subscription_id)
    compute_client = ComputeManagementClient(credential, subscription_id)
    network_client = NetworkManagementClient(credential, subscription_id)
    auth_client = AuthorizationManagementClient(credential, subscription_id)
```

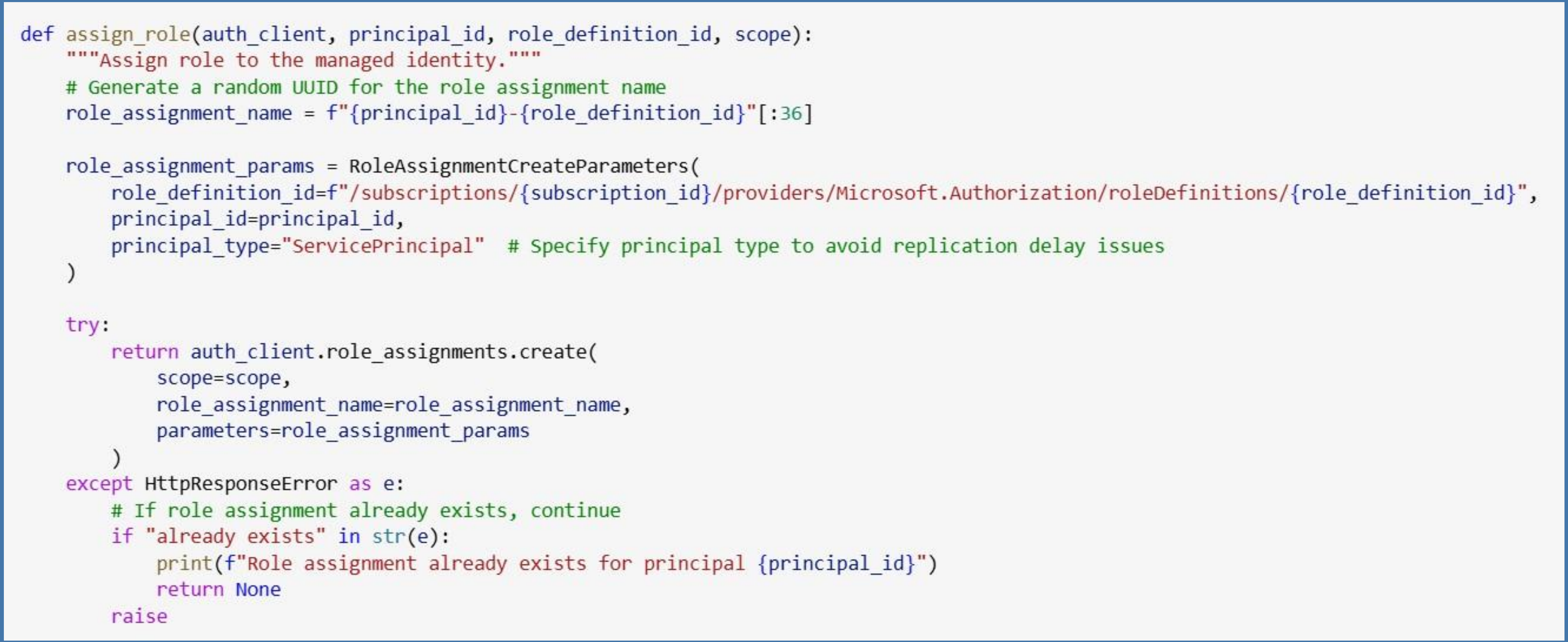
Azure Python SDK – Role Assignment



```python
# Create managed identity
identity = create_managed_identity(msi_client, resource_group_name, managed_identity_name, location)
print(f"Created managed identity: {identity.name} with ID: {identity.id}")

# Assign Contributor role to the managed identity
role_assignment = assign_role(
    auth_client,
    identity.principal_id,
    contributor_role_id,
    f"/subscriptions/{subscription_id}/resourceGroups/{resource_group_name}"
)
print(f"Assigned Contributor role to managed identity")
```

Azure Python SDK – Role Assignment 2

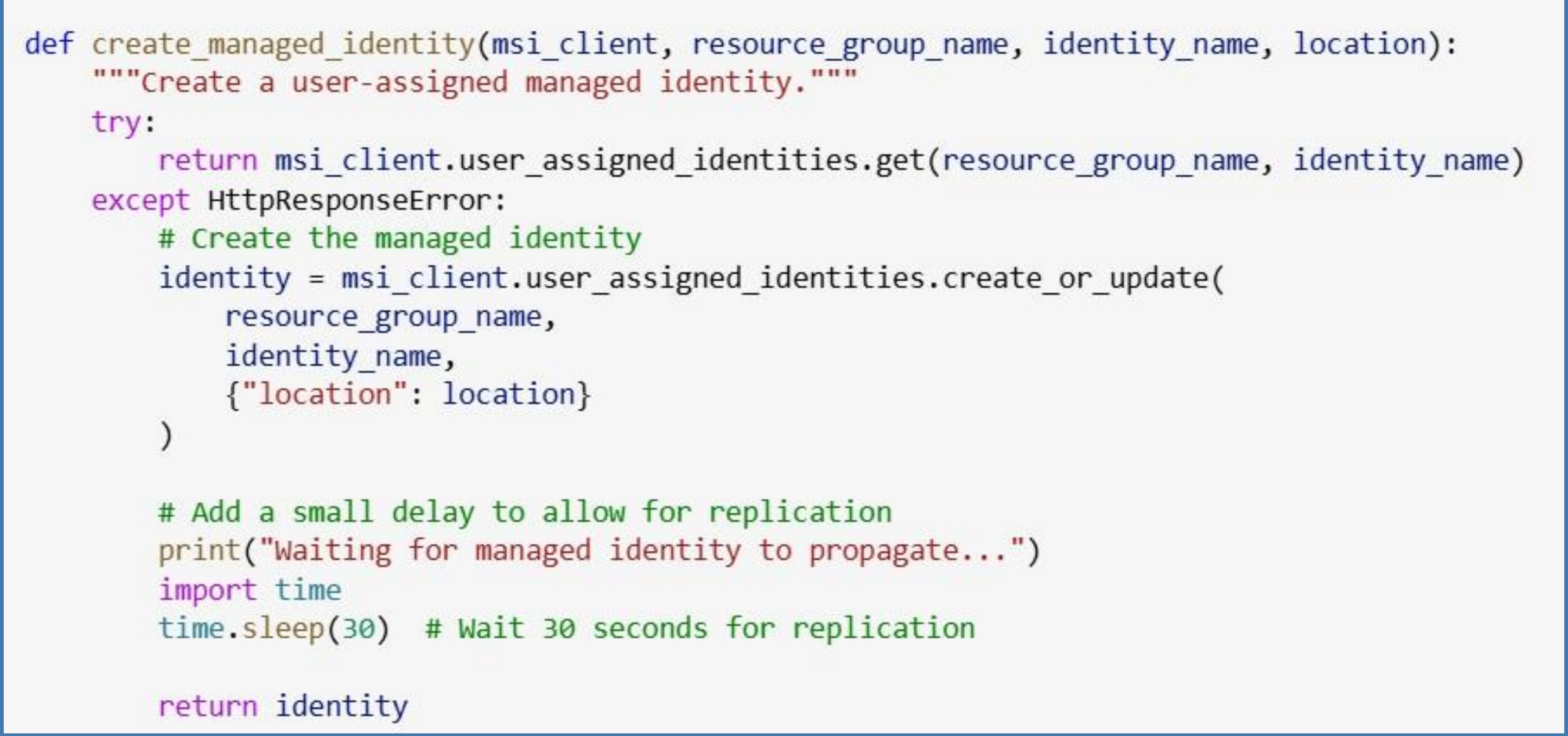


```python
def assign_role(auth_client, principal_id, role_definition_id, scope):
    """Assign role to the managed identity."""
    # Generate a random UUID for the role assignment name
    role_assignment_name = f"{principal_id}-{role_definition_id}"[:36]

    role_assignment_params = RoleAssignmentCreateParameters(
        role_definition_id=f"/subscriptions/{subscription_id}/providers/Microsoft.Authorization/roleDefinitions/{role_definition_id}",
        principal_id=principal_id,
        principal_type="ServicePrincipal"  # Specify principal type to avoid replication delay issues
    )

    try:
        return auth_client.role_assignments.create(
            scope=scope,
            role_assignment_name=role_assignment_name,
            parameters=role_assignment_params
        )
    except HttpResponseError as e:
        # If role assignment already exists, continue
        if "already exists" in str(e):
            print(f"Role assignment already exists for principal {principal_id}")
            return None
        raise
```

Azure Python SDK – Managed Identity

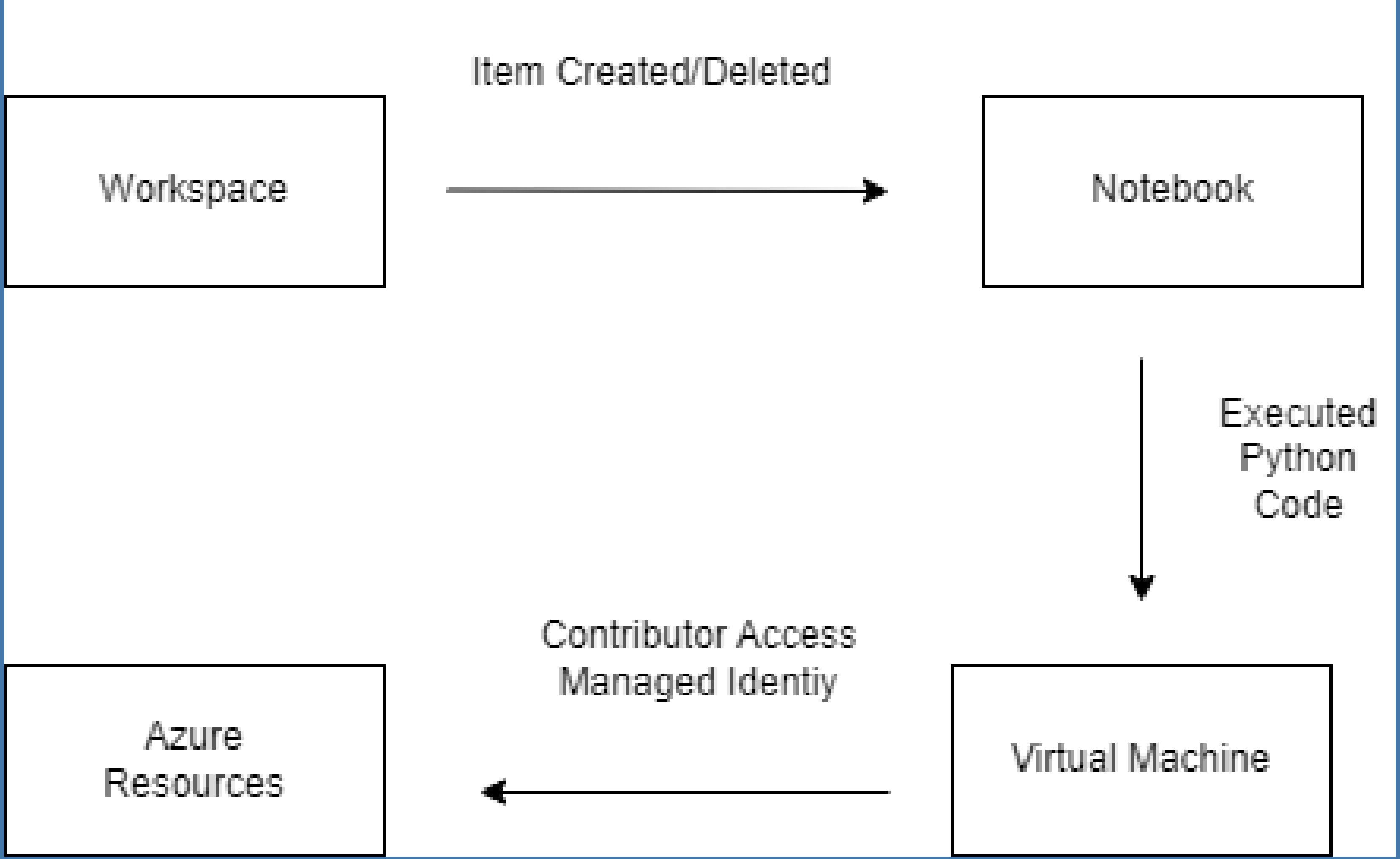


```python
def create_managed_identity(msi_client, resource_group_name, identity_name, location):
    """Create a user-assigned managed identity."""
    try:
        return msi_client.user_assigned_identities.get(resource_group_name, identity_name)
    except HttpResponseError:
        # Create the managed identity
        identity = msi_client.user_assigned_identities.create_or_update(
            resource_group_name,
            identity_name,
            {"location": location}
        )

        # Add a small delay to allow for replication
        print("Waiting for managed identity to propagate...")
        import time
        time.sleep(30)  # Wait 30 seconds for replication

        return identity
```

Backdoor


Item Created/Deleted
Workspace
Notebook
Executed Python Code
Contributor Access Managed Identiy
Azure Resources
Virtual Machine

Backdoor

- Demo video – 7:42 mins

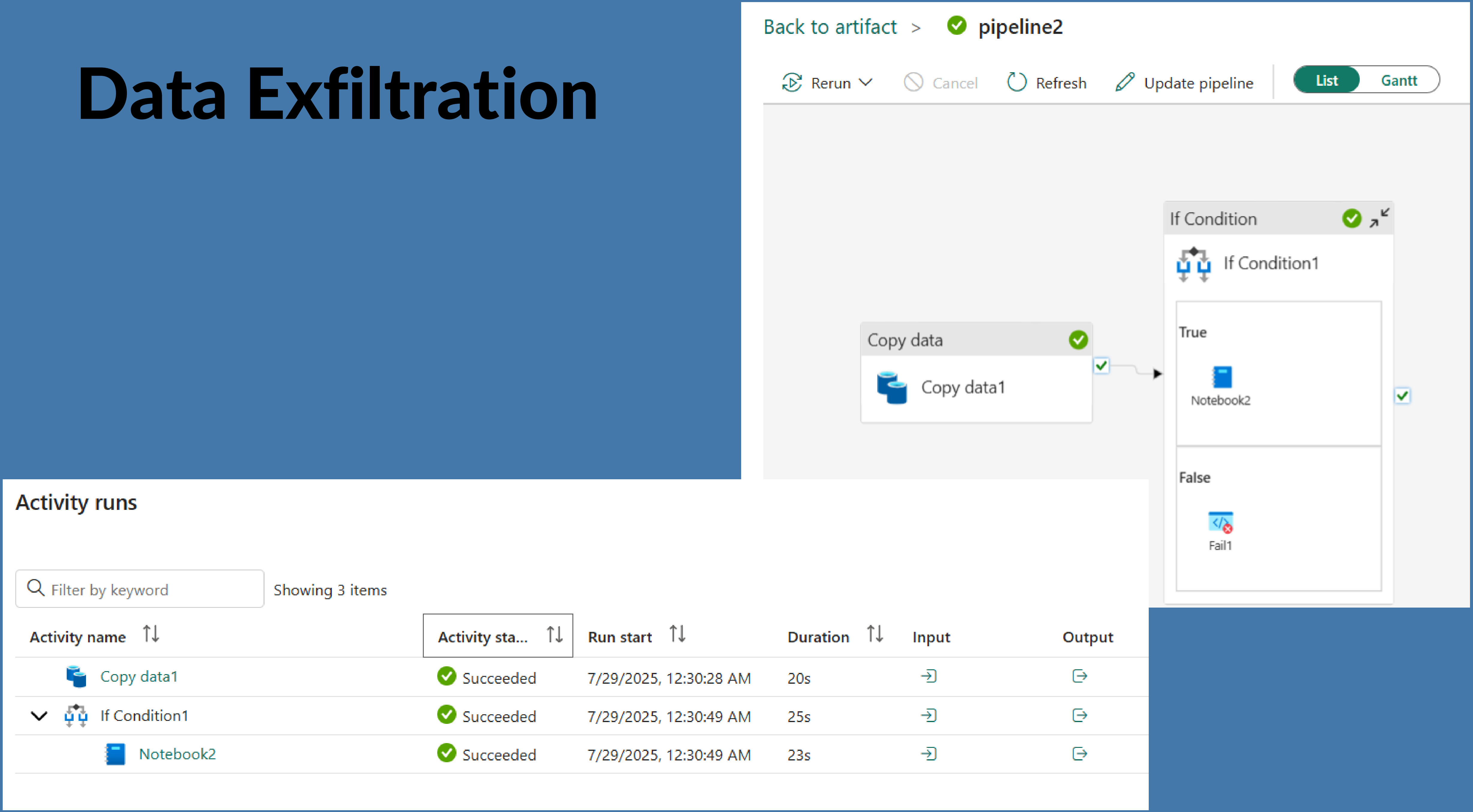
Data Exfiltration

- Data Pipeline
- Notebook
- Shortcut
- SQL Endpoint
- Mirrored DB

Data Exfiltration

- Data Pipeline
- Add A Step To Copy Data To Somewhere Else
- Use Notebook With Hidden Code

Data Exfiltration

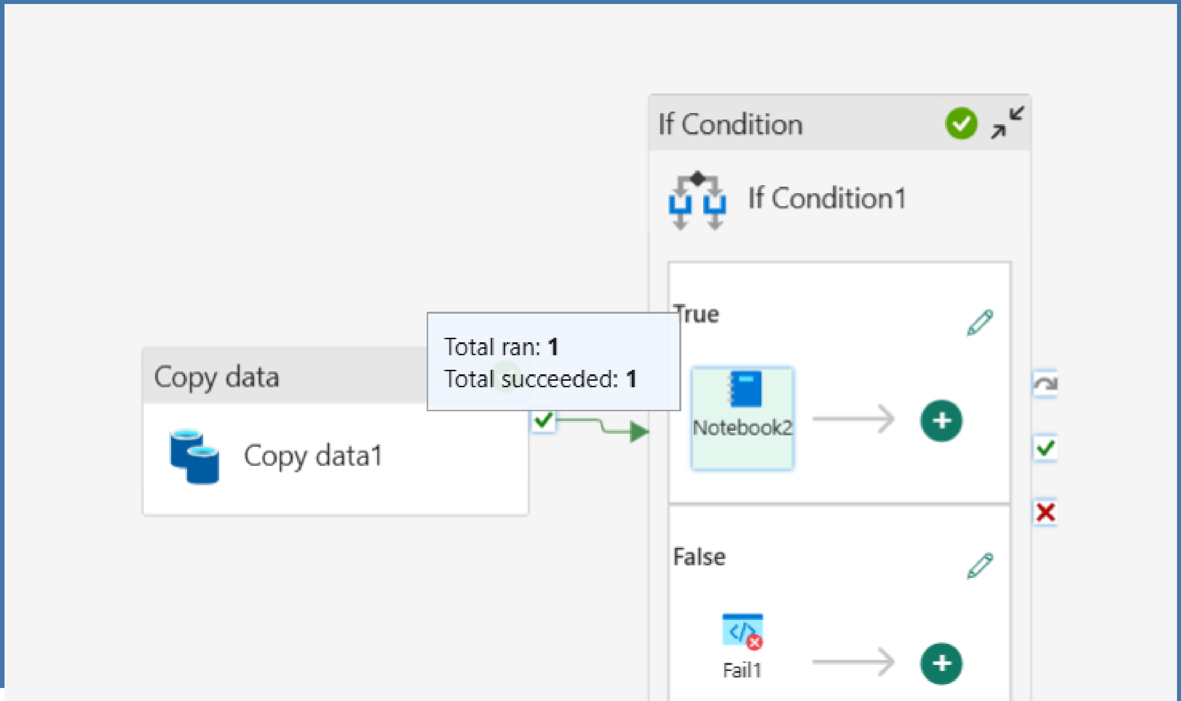


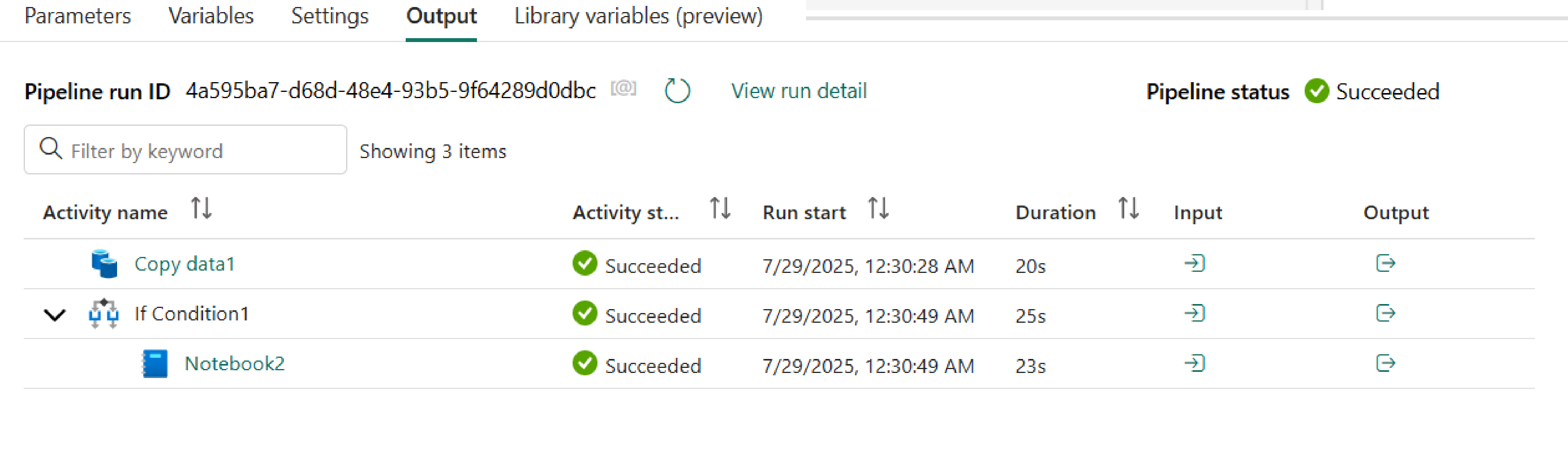
Activity runs

Showing 3 items

Activity name	Activity sta...	Run start	Duration	Input	Output
Copy data1	✓ Succeeded	7/29/2025, 12:30:28 AM	20s		
If Condition1	✓ Succeeded	7/29/2025, 12:30:49 AM	25s		
Notebook2	✓ Succeeded	7/29/2025, 12:30:49 AM	23s		

Data Exfiltration



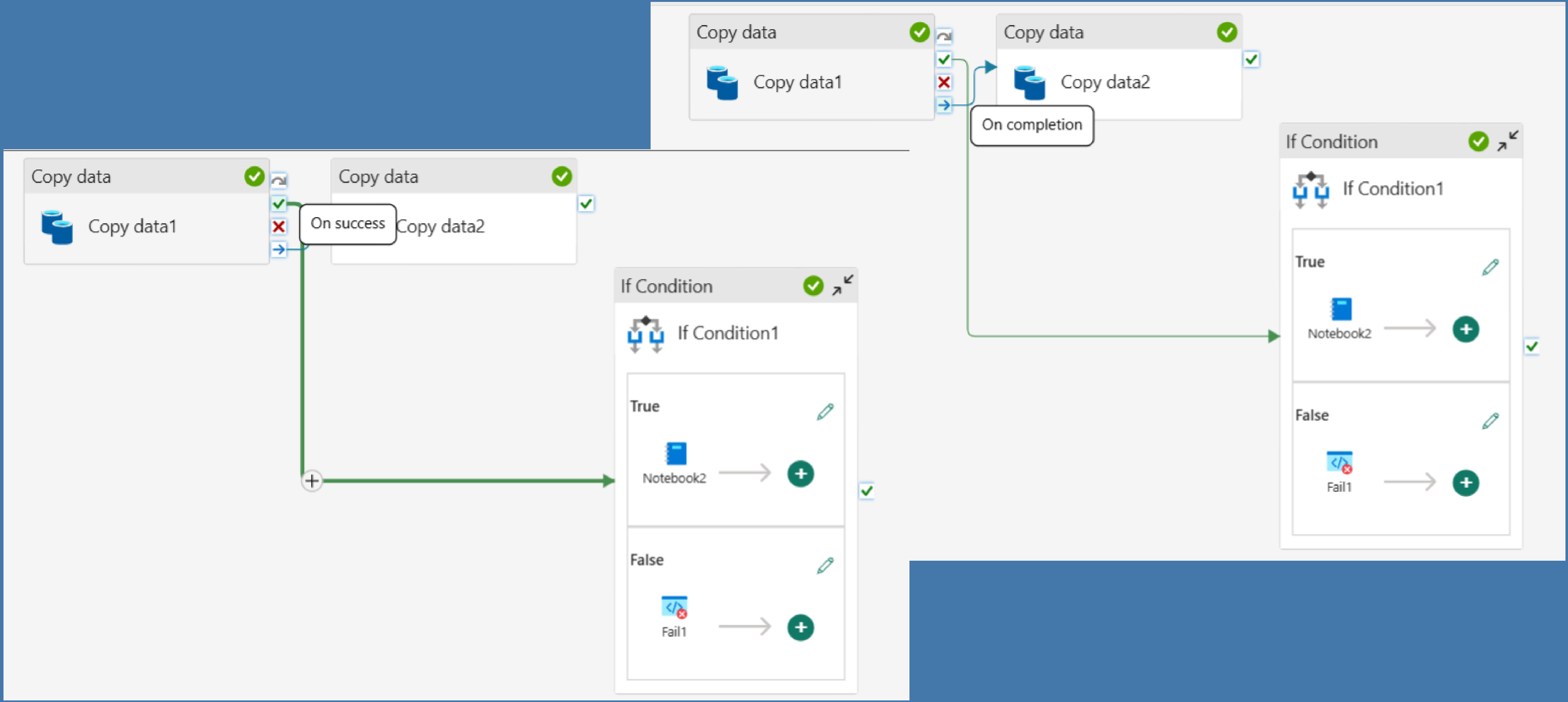


Pipeline run ID 4a595ba7-d68d-48e4-93b5-9f64289d0dbc View run detail **Pipeline status** Succeeded

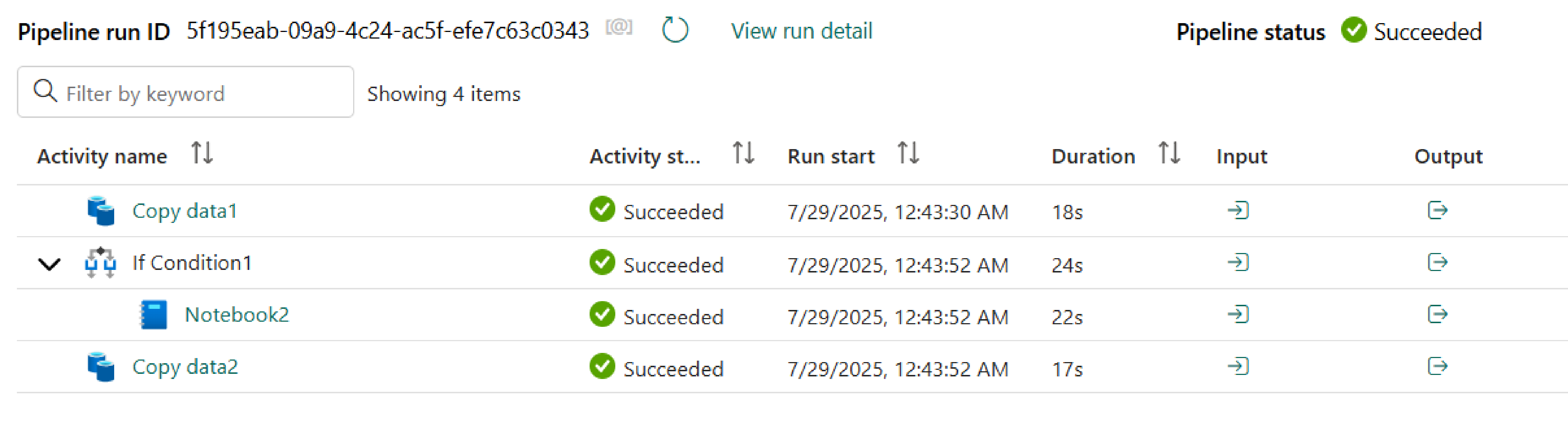
Filter by keyword Showing 3 items

Activity name	Activity st...	Run start	Duration	Input	Output
Copy data1	Succeeded	7/29/2025, 12:30:28 AM	20s		
If Condition1	Succeeded	7/29/2025, 12:30:49 AM	25s		
Notebook2	Succeeded	7/29/2025, 12:30:49 AM	23s		

Data Exfiltration



Data Exfiltration



Pipeline run ID 5f195eab-09a9-4c24-ac5f-efe7c63c0343 View run detail **Pipeline status** ✓ Succeeded

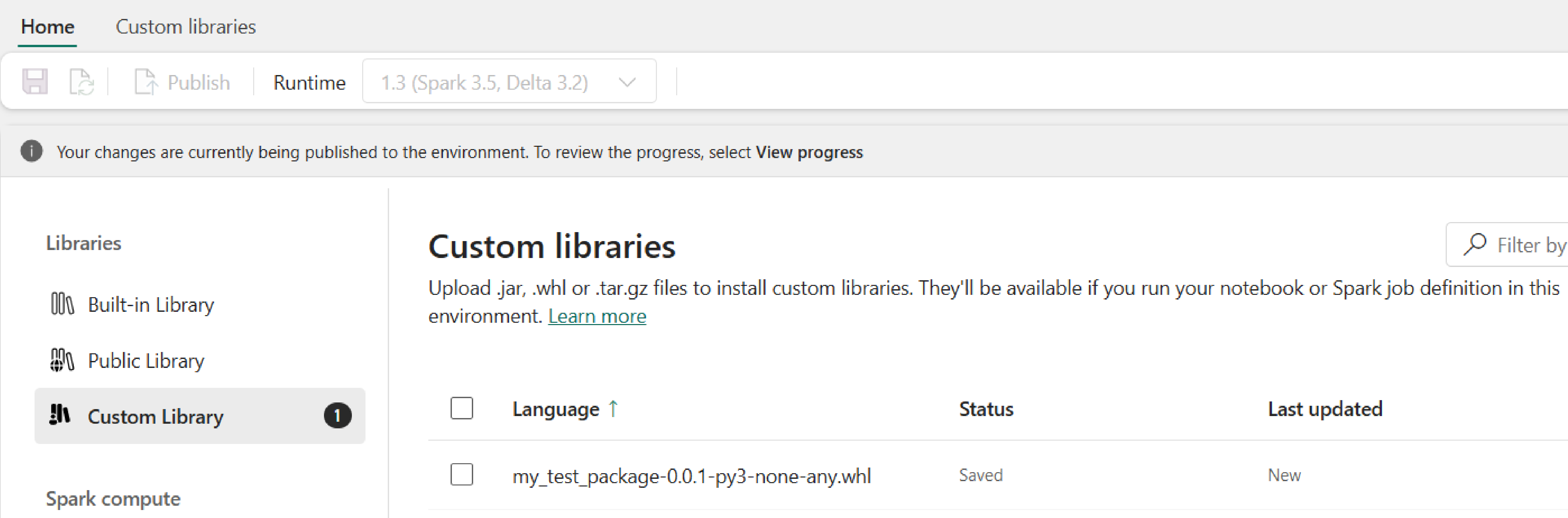
Filter by keyword Showing 4 items

Activity name	Activity st...	Run start	Duration	Input	Output
Copy data1	✓ Succeeded	7/29/2025, 12:43:30 AM	18s	→	→
If Condition1	✓ Succeeded	7/29/2025, 12:43:52 AM	24s	→	→
Notebook2	✓ Succeeded	7/29/2025, 12:43:52 AM	22s	→	→
Copy data2	✓ Succeeded	7/29/2025, 12:43:52 AM	17s	→	→

Data Exfiltration

- Notebook
- Python Runtime
- %pip% Install Custom Libraries
  - Built-in Resource
  - Spark Environment Custom Library
- Internet Access Unless – Private Links For Secure Access To Fabric (Can Break Things)
- Scheduled Notebook Running

Data Exfiltration



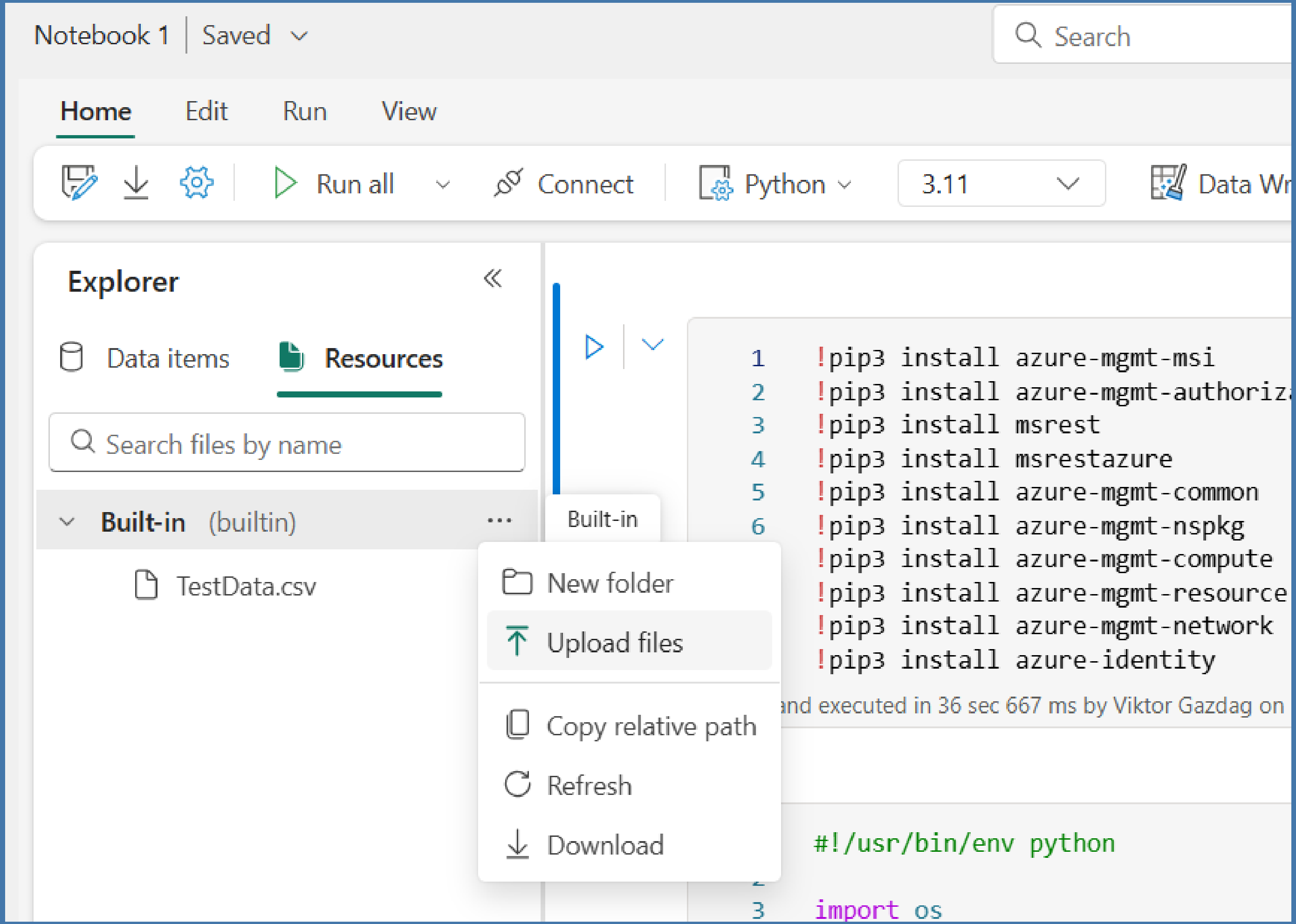
Your changes are currently being published to the environment. To review the progress, select **View progress**

Custom libraries

Upload .jar, .whl or .tar.gz files to install custom libraries. They'll be available if you run your notebook or Spark job definition in this environment. Learn more

□	Language ↑	Status	Last updated
□	my_test_package-0.0.1-py3-none-any.whl	Saved	New

Data Exfiltration



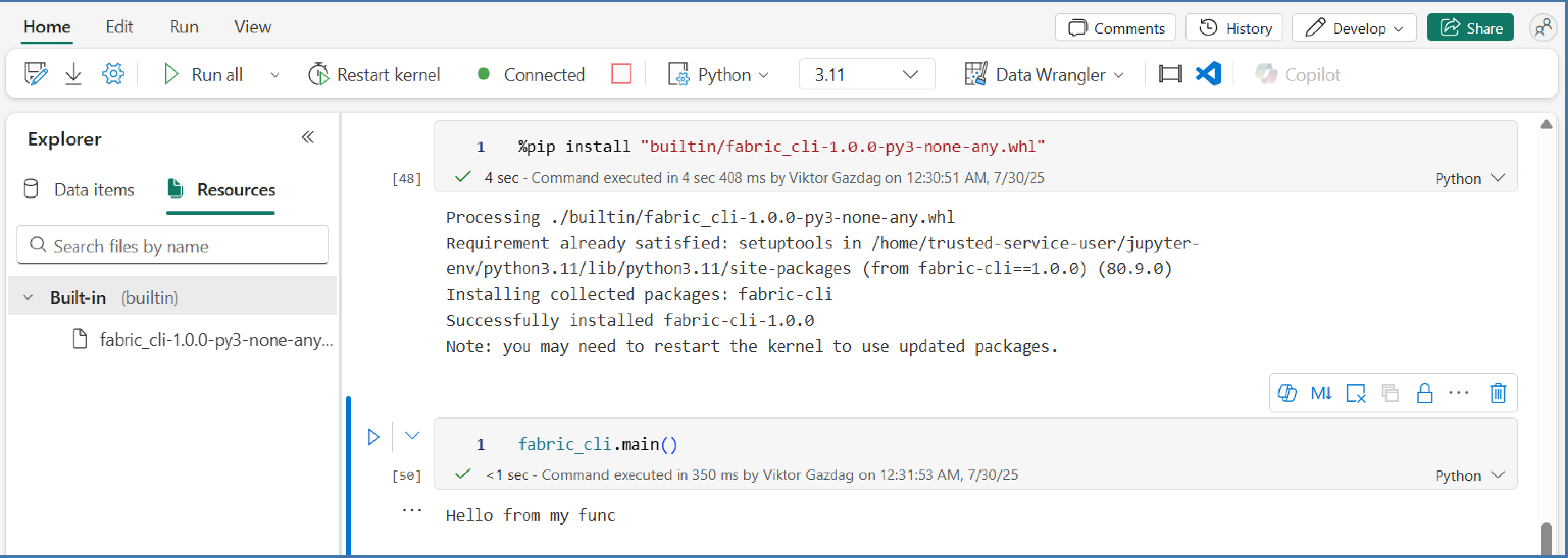
```
!pip3 install azure-mgmt-msi
!pip3 install azure-mgmt-authoriz
!pip3 install msrest
!pip3 install msrestazure
!pip3 install azure-mgmt-common
!pip3 install azure-mgmt-nspkg
!pip3 install azure-mgmt-compute
!pip3 install azure-mgmt-resource
!pip3 install azure-mgmt-network
!pip3 install azure-identity
```

and executed in 36 sec 667 ms by Viktor Gazdag on

```
#!/usr/bin/env python

import os
```

Data Exfiltration



```
%pip install "builtin/fabric_cli-1.0.0-py3-none-any.whl"
```

[48] ✓ 4 sec - Command executed in 4 sec 408 ms by Viktor Gazdag on 12:30:51 AM, 7/30/25

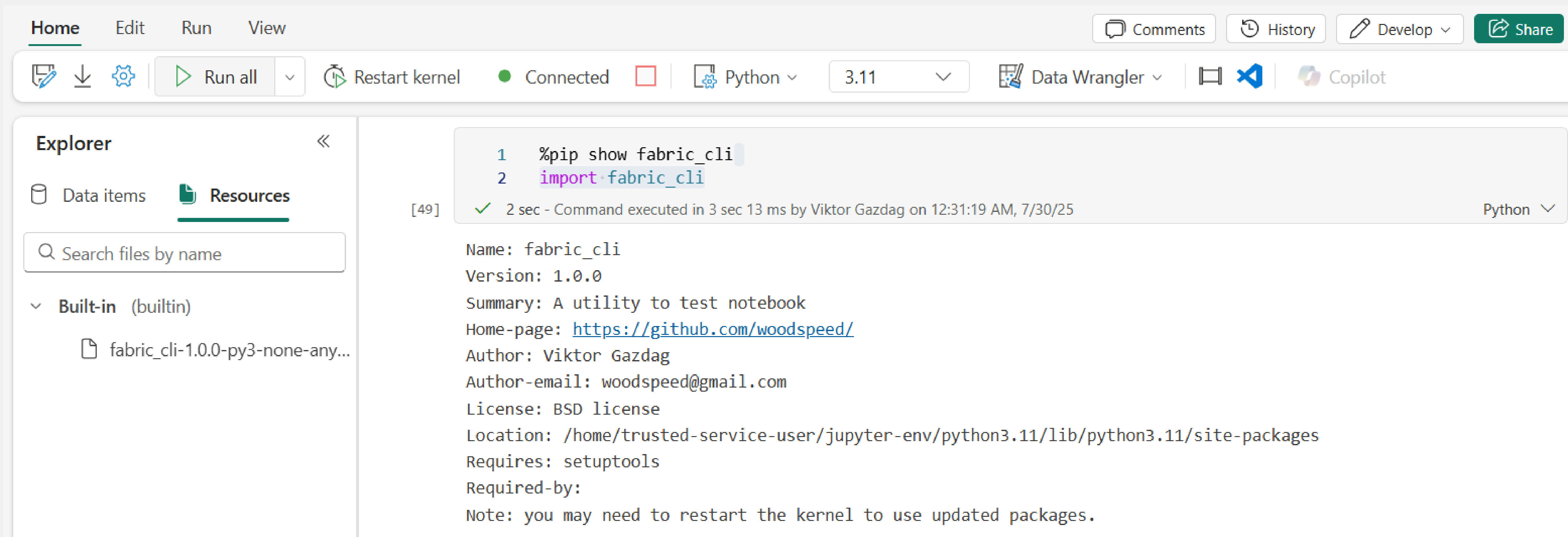
```
Processing ./builtin/fabric_cli-1.0.0-py3-none-any.whl
Requirement already satisfied: setuptools in /home/trusted-service-user/jupyter-env/python3.11/lib/python3.11/site-packages (from fabric-cli==1.0.0) (80.9.0)
Installing collected packages: fabric-cli
Successfully installed fabric-cli-1.0.0
Note: you may need to restart the kernel to use updated packages.
```

```
fabric_cli.main()
```

[50] ✓ <1 sec - Command executed in 350 ms by Viktor Gazdag on 12:31:53 AM, 7/30/25

```
Hello from my func
```

Data Exfiltration



```
%pip show fabric_cli
import fabric_cli
```

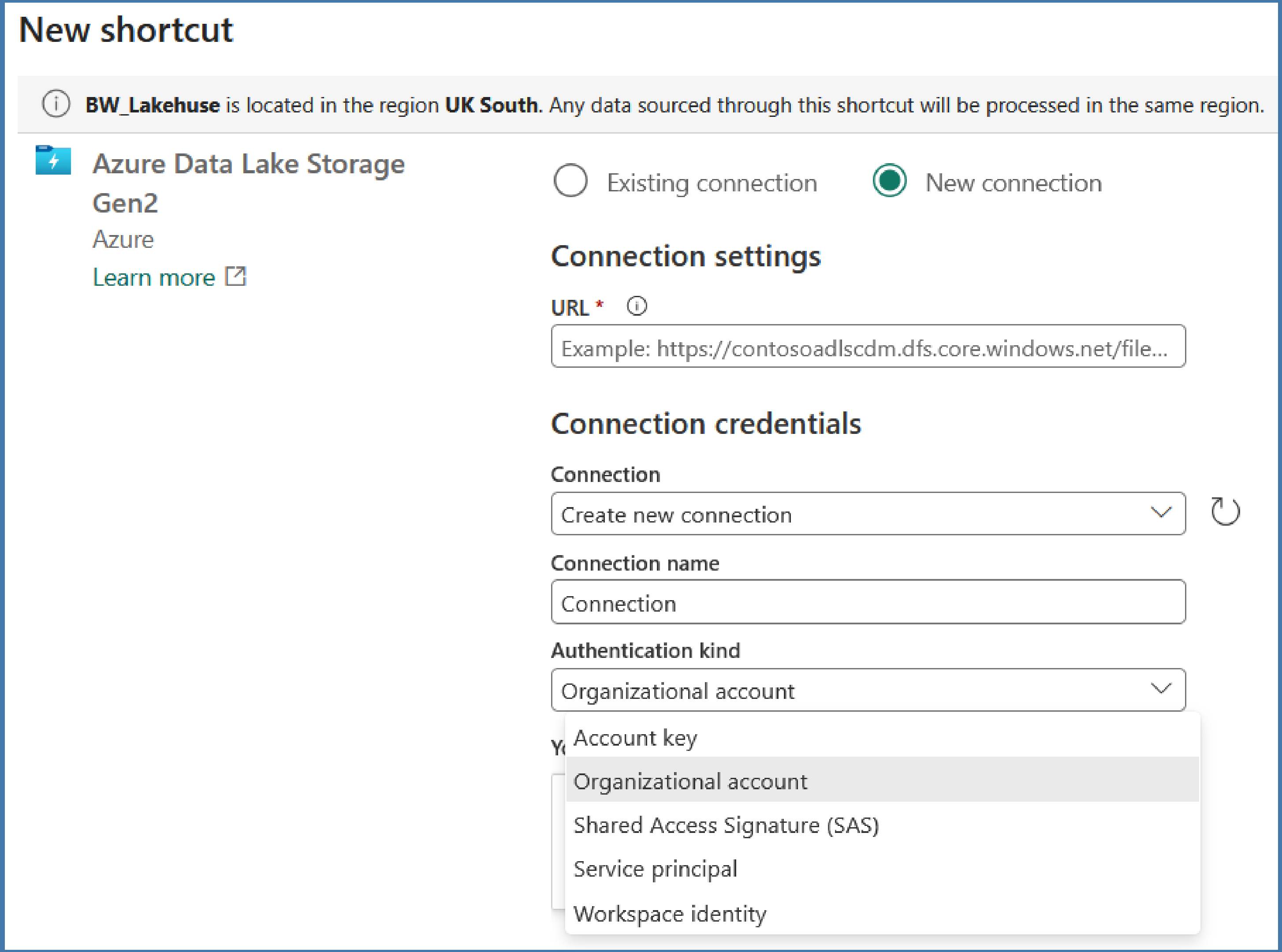
[49] ✓ 2 sec - Command executed in 3 sec 13 ms by Viktor Gazdag on 12:31:19 AM, 7/30/25

```
Name: fabric_cli
Version: 1.0.0
Summary: A utility to test notebook
Home-page: https://github.com/woodspeed/
Author: Viktor Gazdag
Author-email: woodspeed@gmail.com
License: BSD license
Location: /home/trusted-service-user/jupyter-env/python3.11/lib/python3.11/site-packages
Requires: setuptools
Required-by: 
Note: you may need to restart the kernel to use updated packages.
```

Data Exfiltration

- Shortcut (Symbolic Link)
  - Point To Target
  - Work On Target DB
- Internal To Fabric
  - Lakehouse
  - Warehouse
- External To Fabric
  - Datalake
  - S3
- Require ReadAll On Source And Write+ReadAll On Target For Reading Data

Data Exfiltration



New shortcut

ⓘ **BW_Lakehuse** is located in the region **UK South**. Any data sourced through this shortcut will be processed in the same region.

Azure Data Lake Storage Gen2

Azure

Learn more

○ Existing connection ● New connection

Connection settings

URL * ⓘ

Example: https://contosoadlscdm.dfs.core.windows.net/file...

Connection credentials

Connection

Create new connection

Connection name

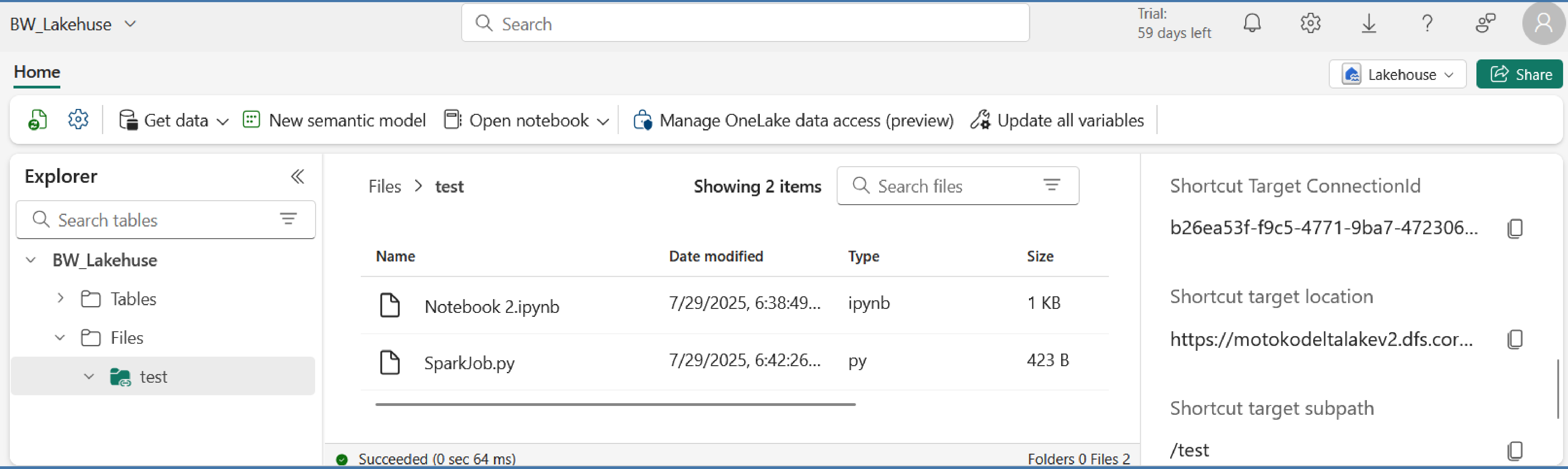
Connection

Authentication kind

Organizational account

Y

Data Exfiltration



Name	Date modified	Type	Size
Notebook 2.ipynb	7/29/2025, 6:38:49...	ipynb	1 KB
SparkJob.py	7/29/2025, 6:42:26...	py	423 B

Succeeded (0 sec 64 ms)

Folders 0 Files 2

Shortcut Target ConnectionId

b26ea53f-f9c5-4771-9ba7-472306...

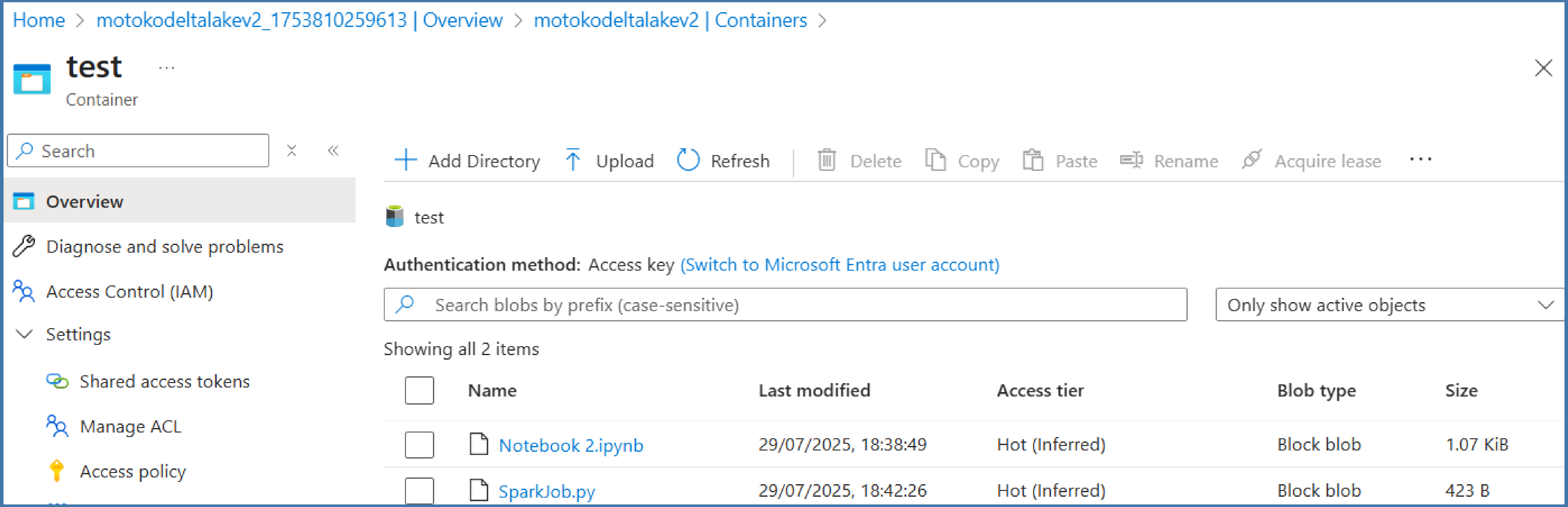
Shortcut target location

https://motokodeltalakev2.dfs.cor...

Shortcut target subpath

/test

Data Exfiltration



Home > motokodeltalakev2_1753810259613 | Overview > motokodeltalakev2 | Containers >

test

Authentication method: Access key (Switch to Microsoft Entra user account)

Search blobs by prefix (case-sensitive)

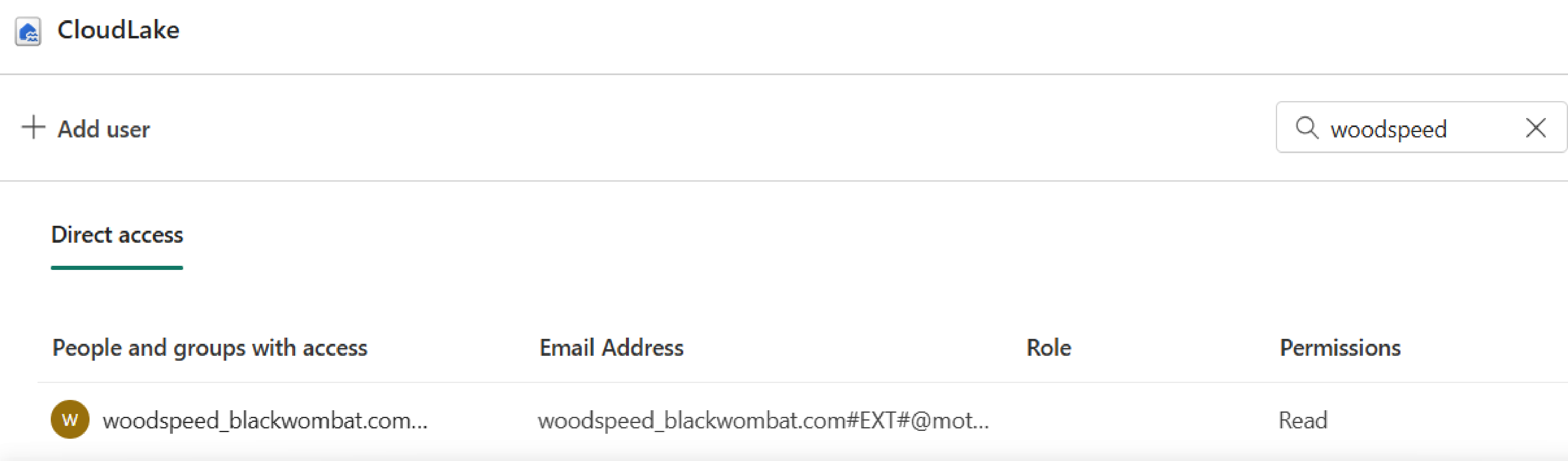
Only show active objects

Showing all 2 items

□	Name	Last modified	Access tier	Blob type	Size
□	Notebook 2.ipynb	29/07/2025, 18:38:49	Hot (Inferred)	Block blob	1.07 KiB
□	SparkJob.py	29/07/2025, 18:42:26	Hot (Inferred)	Block blob	423 B

Data Exfiltration

- SQL Endpoint
- Automatically Created With Lakehouse
- Read-only
- Share (Read (able to connect) + Permission At Database Level = “Hidden Access”
- Share With Entra ID Guest Account And Grant Select On Database



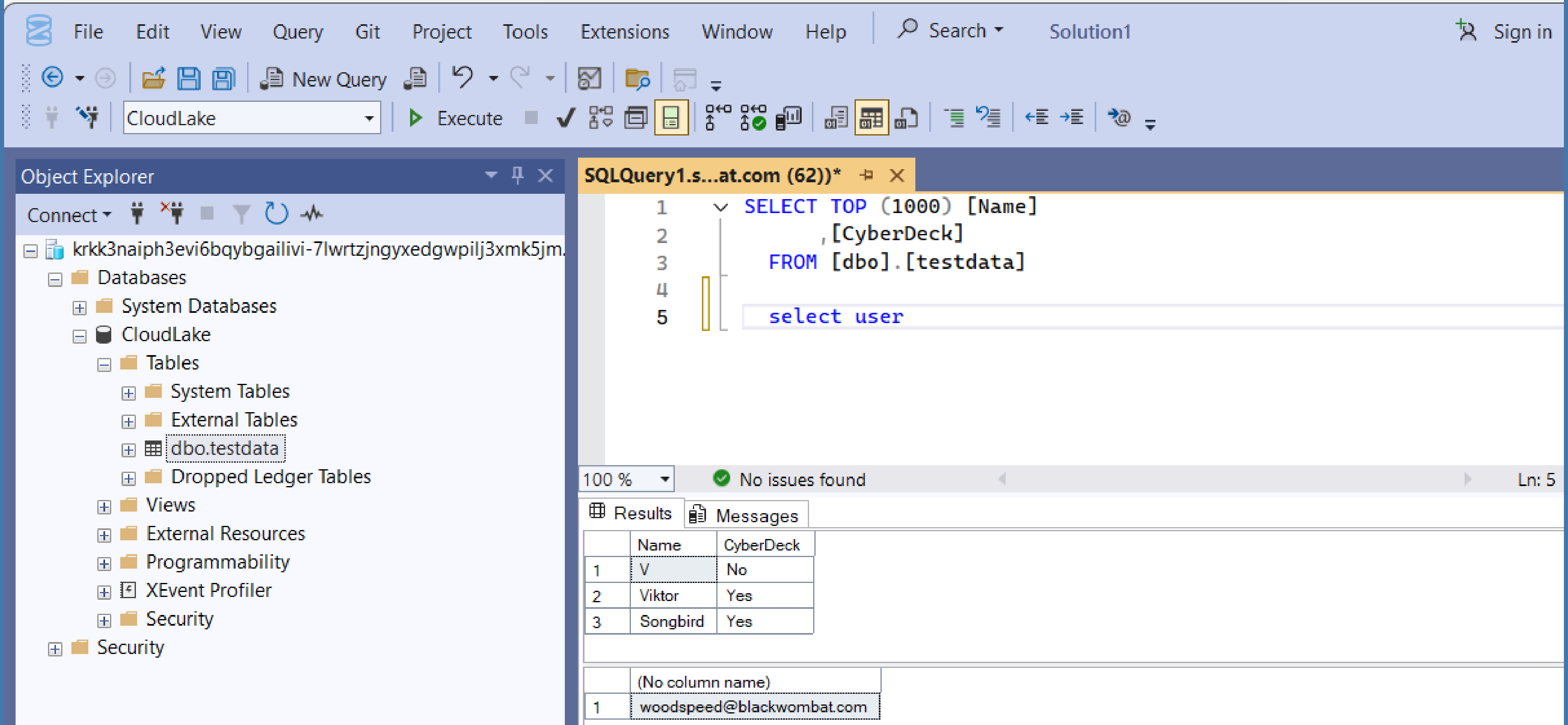
CloudLake

+ Add user

woodspeed

Direct access

People and groups with access	Email Address	Role	Permissions
W woodspeed_blackwombat.com...	woodspeed_blackwombat.com#EXT#@mot...		Read



krkk3naiph3evi6bqybgailivi-7lwrtzjngyxedgwpilj3xmk5jm.
- Databases
  - System Databases
  - CloudLake
    - Tables
      - System Tables
      - External Tables
      - dbo.testdata
      - Dropped Ledger Tables
    - Views
    - External Resources
    - Programmability
    - XEvent Profiler
    - Security
- Security

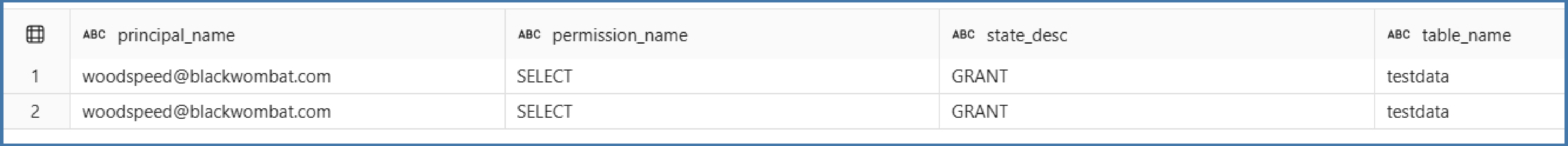
```
SELECT TOP (1000) [Name]
      ,[CyberDeck]
  FROM [dbo].[testdata]

  select user
```

	Name	CyberDeck
1	V	No
2	Viktor	Yes
3	Songbird	Yes

	(No column name)
1	woodspeed@blackwombat.com

Data Exfiltration



	principal_name	permission_name	state_desc	table_name
1	woodspeed@blackwombat.com	SELECT	GRANT	testdata
2	woodspeed@blackwombat.com	SELECT	GRANT	testdata

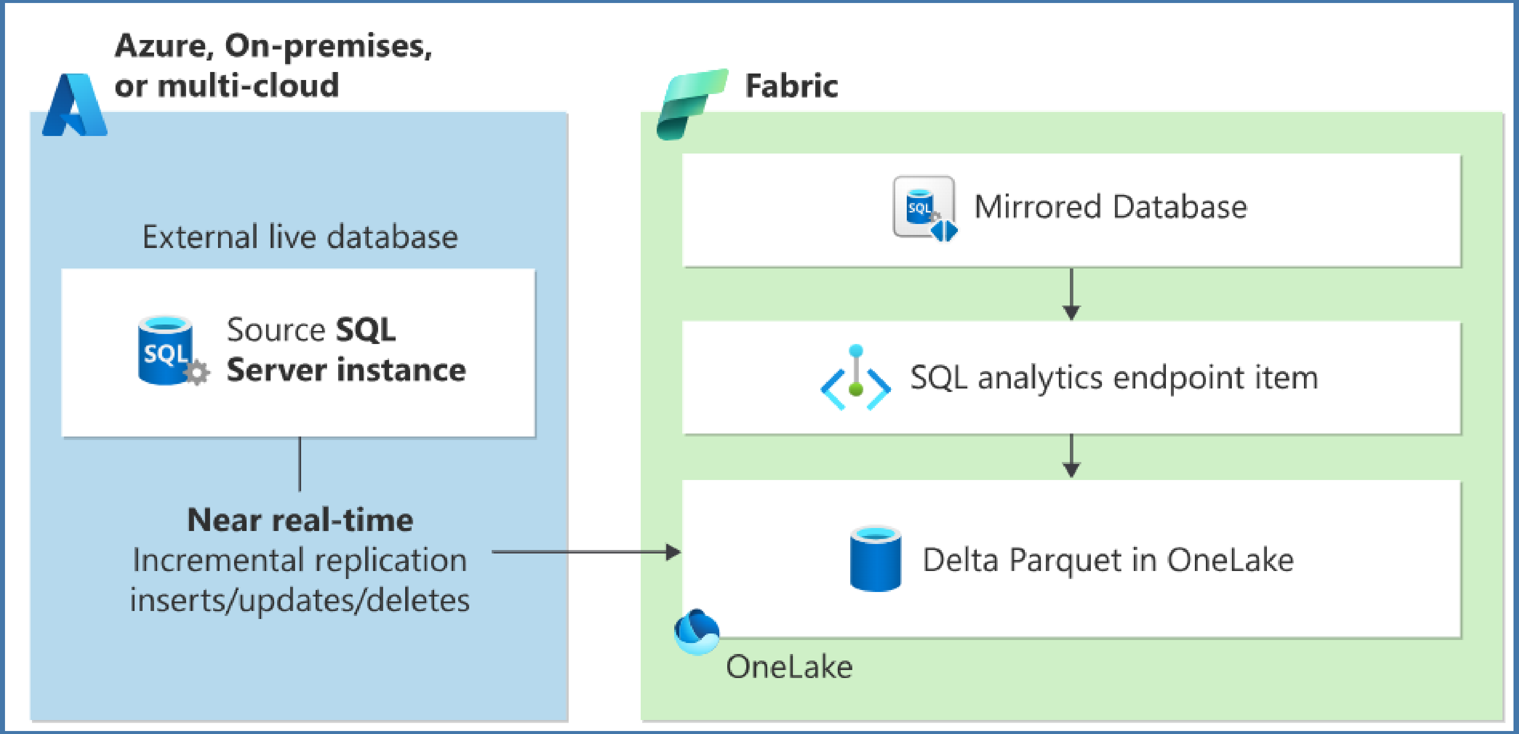
GRANT select ([Name], [CyberDeck])
ON [CloudLake].[dbo].[testdata]
to “username"

Data Exfiltration

- Mirrored DB (Transaction Log)
- Replicate DB to Fabric
  - From Target Azure DB To Attacker Fabric OneLake
  - Work On Delta Parquet In Fabric
- Drops The Mirrored Table Data From Fabric OneLake

Data Exfiltration

- Replicate DB To Fabric
  - From Target Azure DB To Attacker Fabric OneLake
  - Work On Delta Parquet In Fabric



Summary

- Tenant Level Settings
  - Sharing
  - Service Principal
  - User Function

- Backdoor
  - Notebook

- Data Exfiltration
  - Pipeline
  - Shortcut
  - Notebook
  - SQL Endpoint
  - Mirrored DB

Links

- Azure Python SDK
    - https://learn.microsoft.com/en-gb/python/api/overview/azure/compute?view=azure-python
- Fabric Settings Tool
    - https://github.com/dcnsakthi/blogs/blob/main/fabric/admin/TenantSettings.md
- Fabric Tenant Settings Descriptions
    - https://learn.microsoft.com/en-us/fabric/admin/tenant-settings-index
- Microsoft Fabric Security Whitepaper
    - https://learn.microsoft.com/en-us/fabric/security/white-paper-landing-page

Questions & Contacts

- X (Twitter)
    - @wucpi

- Email
    - viktor.gazdag@nccgroup.com

- Conference Slides
    - https://github.com/woodspeed/conferences