



Automate Microsoft Defender for Cloud Alerts using Logic Apps

This project demonstrates how to automate security recommendations from Microsoft Defender for Cloud using Logic Apps (Standard), the modern approach to security orchestration in Azure.

Modern Portal Walkthrough – 2025 Edition (Workflow Standard Plan)



#PerparimLabs | Azure Security Automation



Objective

Goal: Set up a Logic App to automatically send email notifications when new **Defender for Cloud recommendations** are triggered — demonstrating automation in modern Azure environments.

Logic Apps enable security teams to respond automatically when new risks are detected, reducing mean time to respond (MTTR) and ensuring consistent alert handling.



Create Logic App (Workflow Standard Plan)

- Created a new Logic App using the **Workflow Standard** plan, with integrated storage and Application Insights monitoring enabled.

The Standard plan hosts the Logic App on an App Service Plan for enterprise scalability and hybrid connectivity. It replaces the legacy Consumption model.

 Note: The new portal defaults to the **Standard hosting plan**, replacing the older Consumption model.

Details

Subscription	LogicAppWorkflow
Resource Group	logicappworkflowdemo
Name	

Hosting

Storage (New)

Storage account	logicappworkflowae4f
-----------------	----------------------

Plan (New)

Hosting options and plans	Workflow Standard
Name	ASP-LogicAppWorkflow-bc04
Operating System	Windows
Region	Canada Central
SKU	Workflow Standard
Size	Small
ACU	210 total ACU
Memory	3.5 GB memory

Monitoring (New)

Application Insights	Enabled
Name	logicappworkflowdemo
Region	Canada Central

Deployment

Basic authentication	Disabled
Continuous deployment	Not enabled / Set up after app creation

Authentication

Host storage (AzureWebJobsStorage)	
Name	logicappworkflowae4f
Role	Not applicable when using secrets

Azure Files

Name	logicappworkflowae4f
Role	Not applicable when using secrets

Application Insights

Name	logicappworkflowdemo
Role	Not applicable when using secrets

Deployment Complete

- Deployment succeeded — confirming Logic App, Storage Account, and App Service Plan resources created successfully.

App Service, Storage Account, and Application Insights are provisioned automatically to provide persistence, logging, and monitoring for each workflow run.

The screenshot shows the Microsoft Azure Deployment Overview page for a deployment named 'Microsoft.Web-LogicApp-Portal-58430aac-960b'. The deployment is marked as complete. The page displays deployment details, resource status, and a summary table of resources and their status.

Deployment Details:

- Deployment name: Microsoft.Web-LogicApp-Portal-58430aac-960b
- Subscription: Azure subscription 1
- Resource group: LogicAppWorkflow

Resource Status:

Type	Status	Operation details
Microsoft.Web/sites/basicPublis	OK	Operation details
Microsoft.Web/sites/basicPublis	OK	Operation details
Microsoft.Web/sites	OK	Operation details
Application Insights	OK	Operation details
Application Insights	OK	Operation details
Storage account	OK	Operation details
Storage account	OK	Operation details
Microsoft.Web/serverfarms	OK	Operation details
Deployment	OK	Operation details

Summary Table:

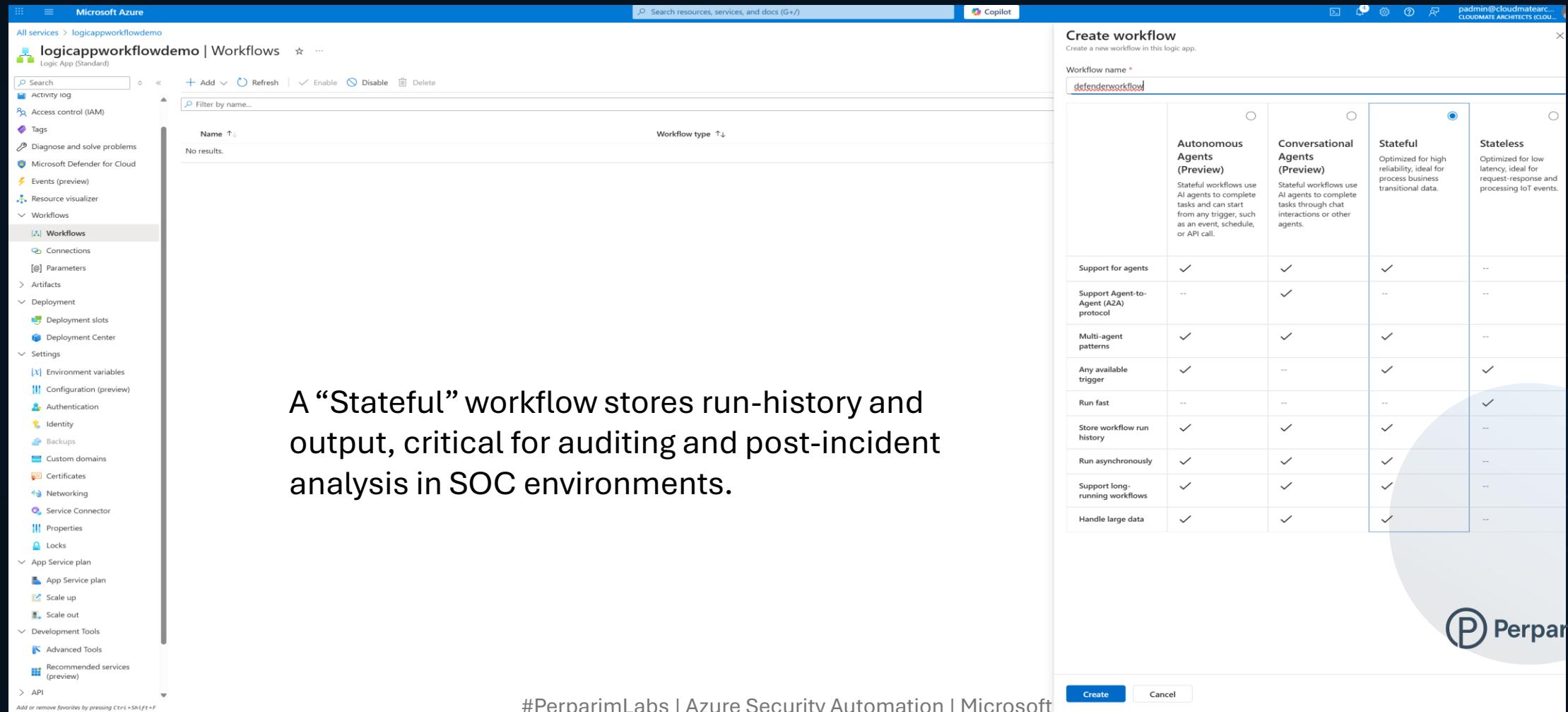
Resource	Type	Status	Operation details
logicappworkflowdemo/ftp	Microsoft.Web/sites/basicPublis	OK	Operation details
logicappworkflowdemo/scm	Microsoft.Web/sites/basicPublis	OK	Operation details
logicappworkflowdemo	Microsoft.Web/sites	OK	Operation details
logicappworkflowdemo	Application Insights	OK	Operation details
logicappworkflowdemo	Application Insights	OK	Operation details
logicappworkflowae4f	Storage account	OK	Operation details
logicappworkflowae4f	Storage account	OK	Operation details
ASP-LogicAppWorkflow-bc04	Microsoft.Web/serverfarms	OK	Operation details
newWorkspaceTemplate	Deployment	OK	Operation details

Next Steps:

- Go to resource

Create Workflow

- Inside the Logic App, added a new workflow named **defenderworkflow** using the **Stateful** mode — optimized for long-running, reliable processes.



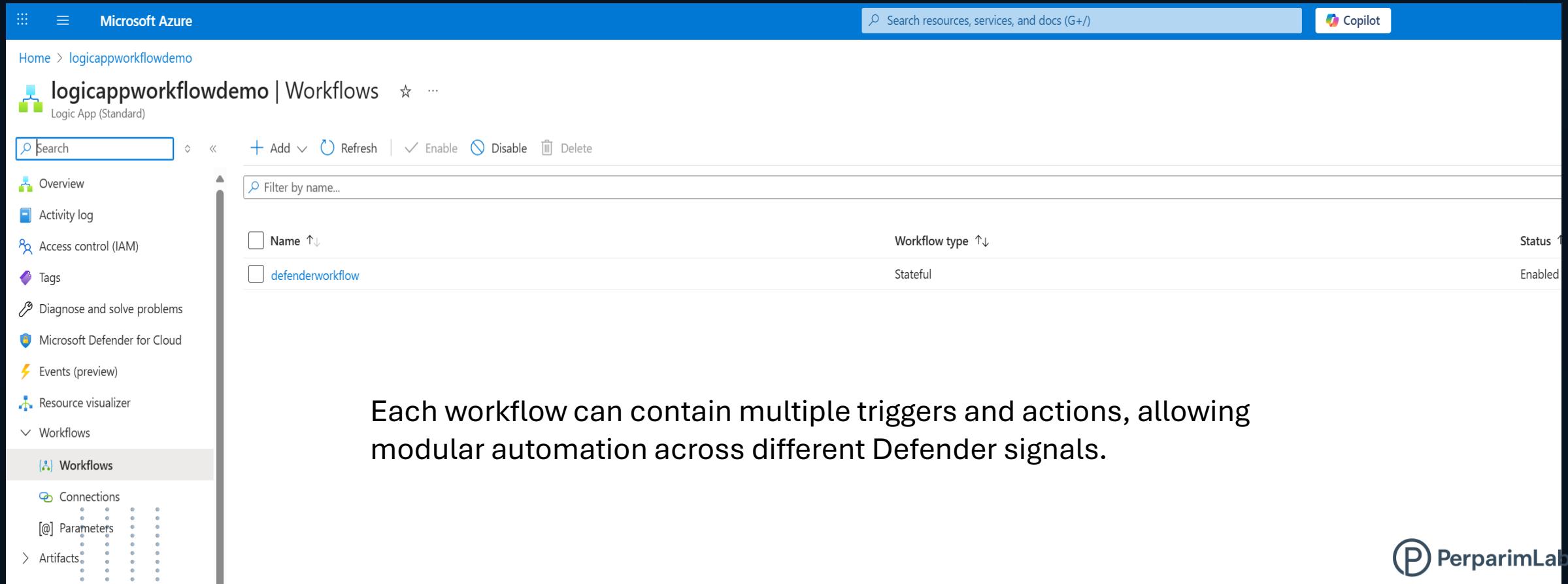
The screenshot shows the Microsoft Azure Logic App interface. On the left, the 'Workflows' blade is open, showing a list of available workflows. In the center, the 'Workflows' list is empty, showing 'No results.' On the right, a 'Create workflow' dialog box is open, titled 'Create workflow'. It asks for a 'Workflow name' and has 'defenderworkflow' entered. Below this, a table compares five workflow modes: Autonomous Agents (Preview), Conversational Agents (Preview), Stateful, and Stateless. The 'Stateful' mode is selected and highlighted with a blue border. The table details the features supported by each mode. At the bottom of the dialog are 'Create' and 'Cancel' buttons.

	Autonomous Agents (Preview)	Conversational Agents (Preview)	Stateful	Stateless
Support for agents	✓	✓	✓	--
Support Agent-to-Agent (A2A) protocol	--	✓	--	--
Multi-agent patterns	✓	✓	✓	--
Any available trigger	✓	--	✓	✓
Run fast	--	--	--	✓
Store workflow run history	✓	✓	✓	--
Run asynchronously	✓	✓	✓	--
Support long-running workflows	✓	✓	✓	--
Handle large data	✓	✓	✓	--

A “Stateful” workflow stores run-history and output, critical for auditing and post-incident analysis in SOC environments.

Workflow Created

- Workflow successfully created and enabled — ready to define triggers and actions.



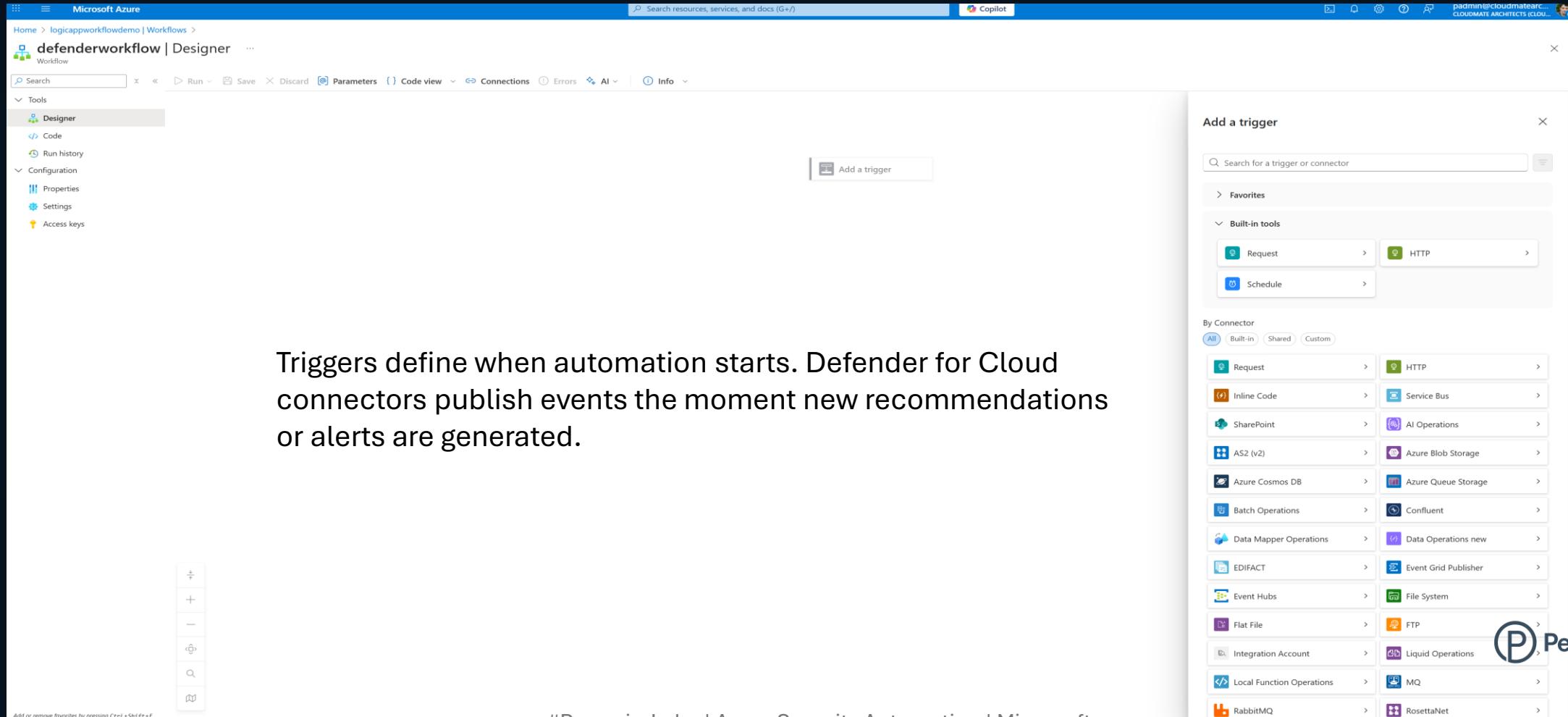
The screenshot shows the Microsoft Azure Logic App (Standard) Workflows blade. The left sidebar includes links for Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Microsoft Defender for Cloud, Events (preview), Resource visualizer, Workflows (selected), Workflows (highlighted), Connections, Parameters, Artifacts, and Dev Center. The main content area displays a table with one workflow entry:

Name	Workflow type	Status
defenderworkflow	Stateful	Enabled

A text overlay in the center of the blade states: "Each workflow can contain multiple triggers and actions, allowing modular automation across different Defender signals."

Add Trigger

- Opened the **Designer** and selected the trigger source. Logic Apps Designer now provides built-in connectors for Defender, Azure, and third-party integrations.



The screenshot shows the Microsoft Azure Logic Apps Designer interface. The left sidebar has 'defenderworkflow | Designer' selected. The main area shows a large 'Add a trigger' button. A modal window titled 'Add a trigger' is open on the right, containing a search bar and a list of trigger connectors. The 'By Connector' section is expanded, showing various categories and specific triggers like 'Request', 'HTTP', 'Schedule', 'Service Bus', etc. A watermark for 'PerparimLabs' is visible in the bottom right corner.

Triggers define when automation starts. Defender for Cloud connectors publish events the moment new recommendations or alerts are generated.

Defender Trigger

- Selected “**When a Microsoft Defender for Cloud recommendation is created or triggered.**”
This ensures automation activates instantly when new security recommendations are generated.

The screenshot shows the Microsoft Azure Logic Apps Designer interface. The workflow is titled "defenderworkflow". On the left, the "Tools" sidebar is open, showing "Designer" is selected. In the center, there is a large empty workspace with a "Add a trigger" button. On the right, a modal dialog titled "Add a trigger" is displayed, showing a search bar with "Microsoft Defender for Cloud" and a list of triggers. The list includes:

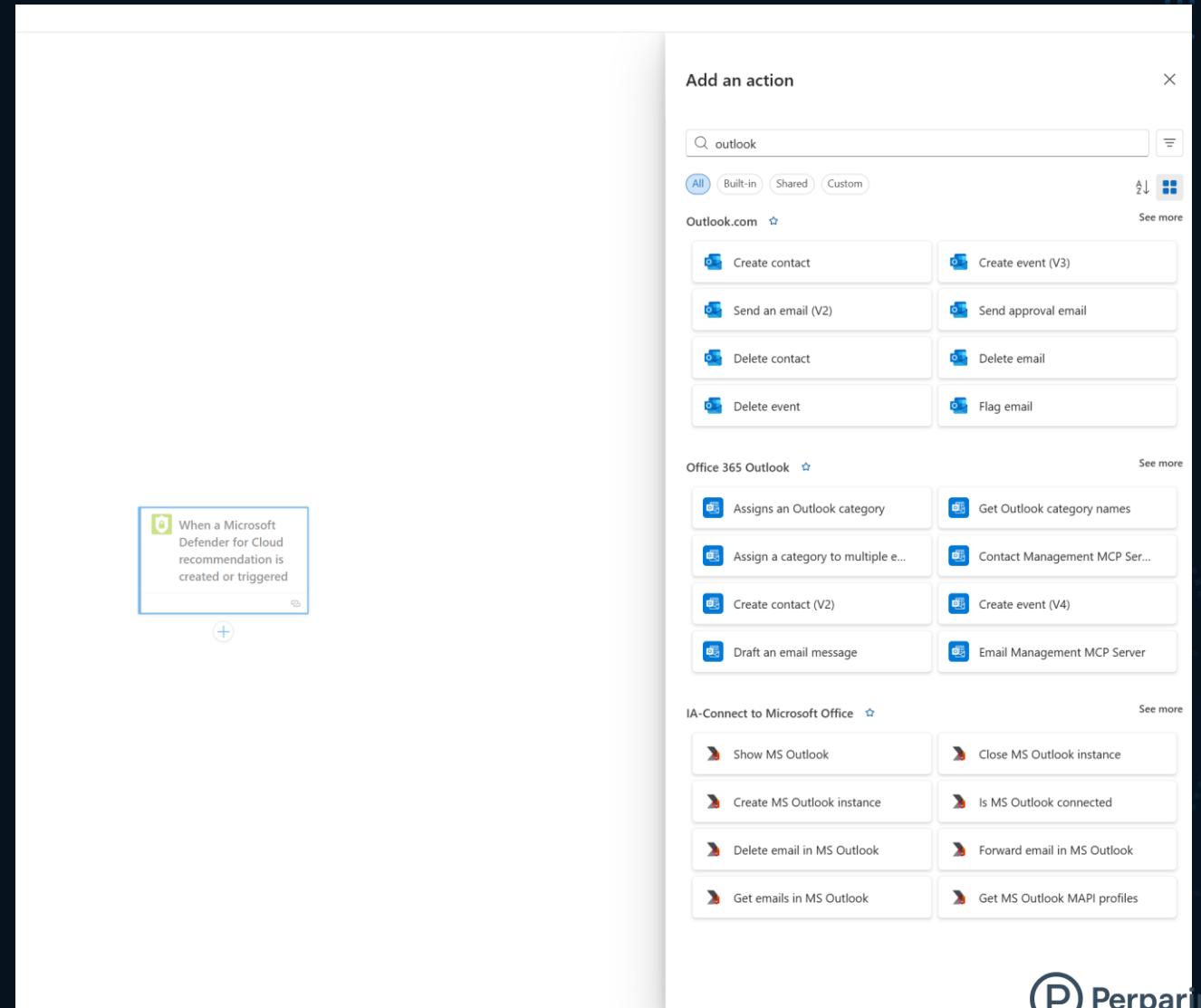
- Microsoft Defender for Cloud Alert**
When a Microsoft Defender for ...
- Microsoft Defender for Cloud Recommendation**
When a Microsoft Defender for ...
- Microsoft Defender for Cloud Regulatory Compliance**
When a Defender for Cloud reg...
- Microsoft Defender ATP**
Triggers when a new remediati...
Triggers - Trigger when new W...

Selecting this trigger binds the workflow directly to Defender's recommendation API—no need for manual polling or scripts.

Add Email Action

- Added an **Outlook (V2) Send Email** action to automatically alert the admin mailbox.

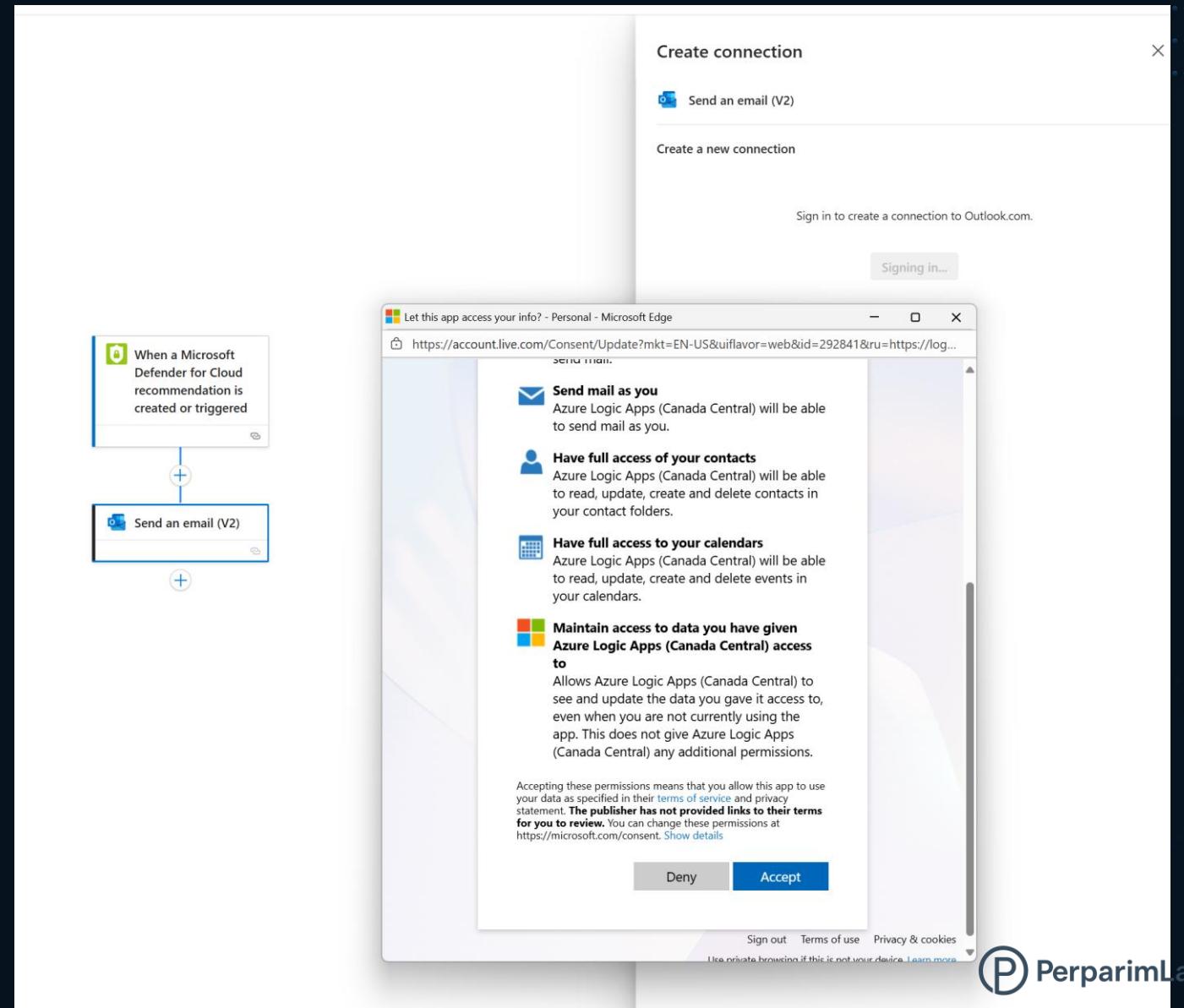
The Outlook connector is a quick notification channel; in production, this could integrate with Teams, ITSM, or Sentinel Playbooks.



Connect Outlook

- Authorized Azure Logic Apps (Canada Central) to send email using the connected Outlook account.

Uses secure OAuth 2.0 delegated permissions—no stored credentials—ensuring compliance with least-privilege principles.



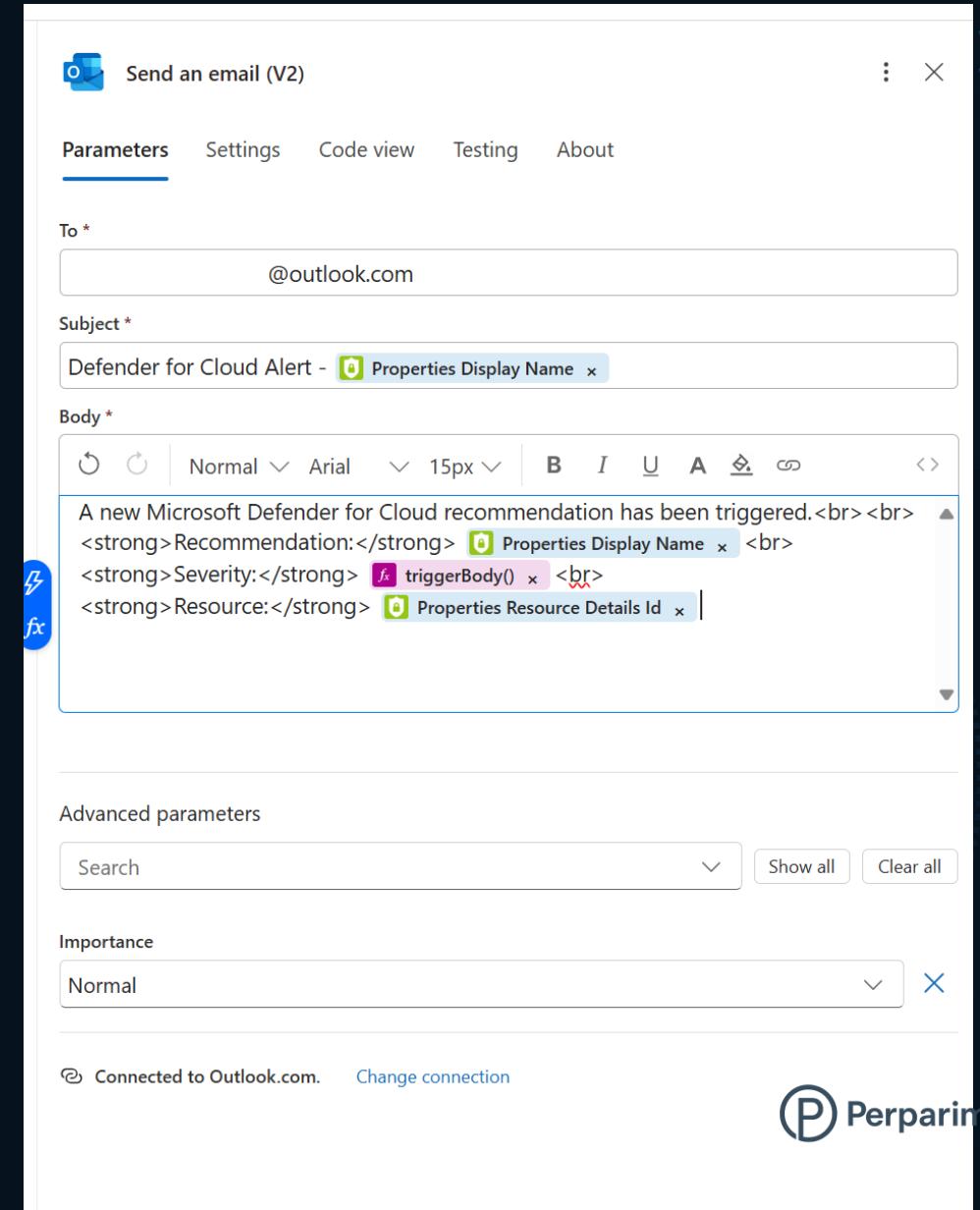
Configure Email Content

- Customized the email body with dynamic content:

- **Recommendation name**
- **Severity**
- **Resource details**

Dynamic content tokens pull real-time data (recommendation name, severity, resource ID) directly from Defender payloads, making alerts actionable.

 *Tip:* You can format the body with HTML for rich, readable alerts.

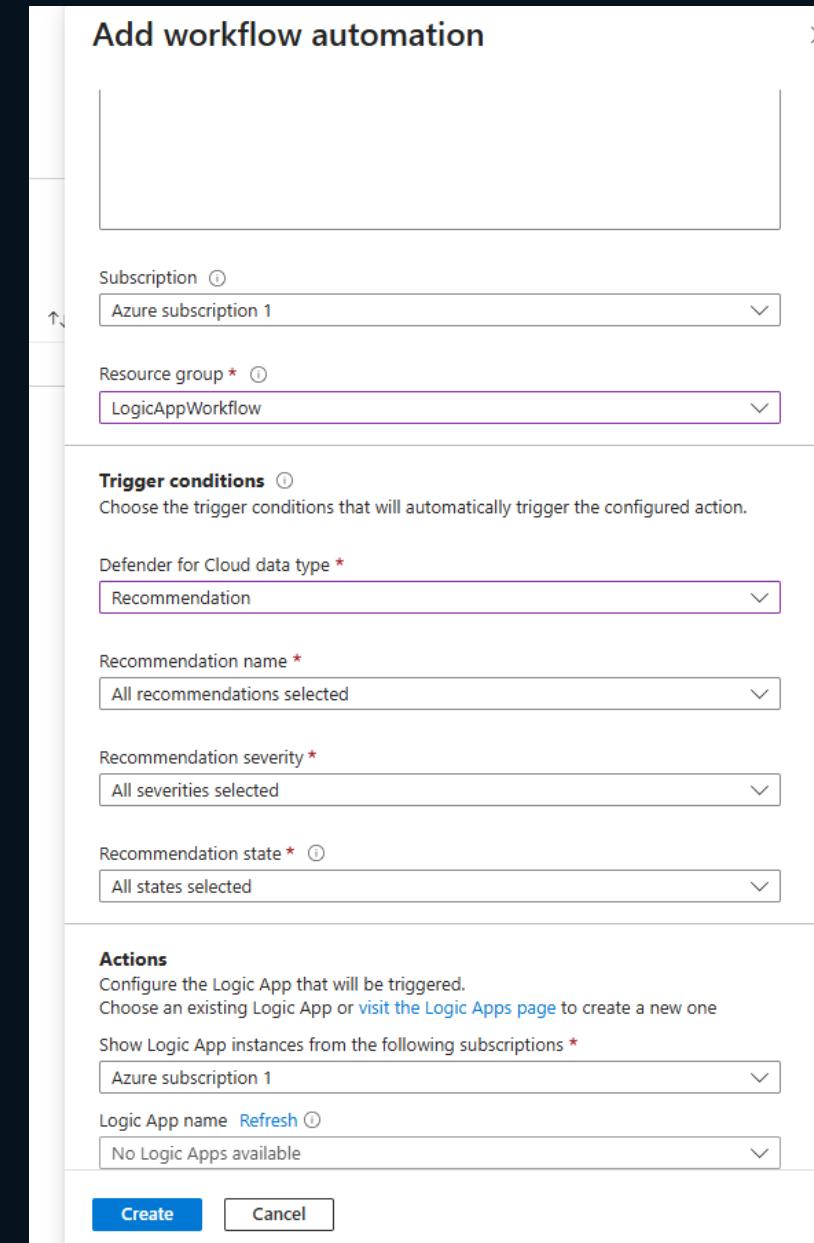




Defender Workflow Automation Setup

- Opened **Defender for Cloud** → **Workflow Automation** to connect the Logic App. However, modern environments currently show:
- “No Logic Apps available”
-  **Reason:** Logic App (Standard) resources are not yet discoverable from Defender for Cloud automation panel — only **Consumption plans** appear there.

As of 2025, Defender for Cloud’s automation pane lists only Consumption Logic Apps. Standard-plan Logic Apps integrate through native triggers instead.



Summary

Lab Completed:

- Built Logic App with Defender trigger and Outlook alert.
- Observed modern portal limitations (Standard vs. Consumption).
- Demonstrated full automation workflow concept.

This architecture establishes a reusable pattern: detect → notify → remediate. Future enhancements can add playbooks, ticket creation, or auto-remediation.

Key Takeaway:

Microsoft Defender for Cloud automation now integrates with **Logic Apps Standard**, but configuration visibility in Defender UI will be updated in future releases.

Cleanup

✍️ Delete the resource group **LogicAppWorkflow** to stop charges.

💡 Always remove unused Standard Logic Apps — they run under App Service Plans and can accumulate small costs.

Microsoft Azure

All services > Logic apps > logicappworkflowdemo

LogicAppWorkflow Resource group

Search

Create Manage view Delete resource group Refresh Export to CSV Open query Assign tags Move Group by none JSON View

Overview

Activity log Access control (IAM) Tags Resource visualizer Events Settings Cost Management Monitoring Automation Help

Essentials

Subscription (move) : Azure subscription 1 Deployment : 1 Failed, 1 Succeeded Subscription ID : Location : East US Tags (edit) : Add tags

Resources Recommendations

Name Type Location

Name	Type	Location
Application Insights Smart Detection	Action group	Global
ascessment	API Connection	Canada Central
ASP-LogicAppWorkflow-bc04	App Service plan	Canada Central
logicappworkflowae4f	Storage account	Canada Central
logicappworkflowdemo	Logic App (Standard)	Canada Central
logicappworkflowdemo	Application Insights	Canada Central
outlook	API Connection	Canada Central

Standard Logic Apps run persistently—deleting the resource group immediately halts compute billing.