# Expose multiple Azure Function apps as a consistent API by using Azure API Management

## Azure API Management

Azure API Management is a fully managed cloud service that you can use to publish, secure, transform, maintain, and monitor APIs. It helps organizations publish APIs to external, partner, and internal developers to unlock the potential of their data and services. API Management handles all the tasks involved in mediating API calls, including request authentication and authorization, rate limit and quota enforcement, request and response transformation, logging and tracing, and API version management. API Management enables you to create and manage modern API gateways for existing backend services no matter where they're hosted.

### Microservices architecture challenges

Client apps are coupled to microservices. If you want to change the location or definition of the microservice, you may have to reconfigure or update the client app.

Each microservice may be presented under different domain names or IP addresses. This presentation can give an impression of inconsistency to users and can negatively affect your branding.

It can be difficult to enforce consistent API rules and standards across all microservices. For example, one team may prefer to respond with XML and another may prefer JSON.

You're reliant on individual teams to implement security in their microservice correctly. It's difficult to impose these requirements centrally.

### API Management also includes helpful tools

You can test each microservice and its operations to ensure that they behave in accordance with your requirements. You can also monitor the behavior and performance of deployed services.

Azure API Management supports importing Azure Function Apps as new APIs or appending them to existing APIs. The process automatically generates a host key in the Azure Function App, which is then assigned to a named value in Azure API Management.

## Create a new API in API Management from a function app

### Create functions

[Create an Azure Function App - ProductFunction](#_Create_a_function)

[Create ProductDetails function in your ProductFunction app](#_Create_a_function_1)

[Test ProductDetails function](#_Test_your_Azure)

### Expose function app as an API using Azure API Management

Azure Portal -> All resources -> ProductFunction function app

-> API -> API Management -> Create new

-> Install API Management gateway pane appears

“Subscription” = Concierge Subscription,

“Resource group” = [sandbox resource group name]

“Region” = select any that supports the Consumption Plan

“Administrator email” = Enter an email address.

“Pricing tier” = Consumption (99.95% SLA)

-> Next: Monitoring and clear the Application Insights option

-> Review + Create -> Create -> Link API

-> Import Azure Functions API Management service pane appears

-> Select “ProductDetails” function and Select to continue

-> “API URL suffix” = “products”

-> Create.

### Test the OnlineStore products endpoint

API Management pane of your function app -> Test -> GET ProductDetails

-> Under Query parameters, select Add parameter

-> Enter name in the NAME field and value and in the VALUE field

-> Send

-> The **HTTP response** section of the console has two tabs, Message and Trace. The Message tab is populated with the HTTP response. The product details appear in JSON format at the end of the response.

-> Scroll up to the HTTP request section and notice the format of the request. The request was sent to a destination in the azure-api.net domain. This location is different from the azurewebsites.net domain where the function app is hosted.

## Add another Azure Functions app to an existing API

### Create another function

[Create OrderDetails function in your ProductFunction app](#_Create_a_function_1)

[Test the OrderDetails function](#_Test_your_Azure)

### Add a function to an existing API

Azure Portal -> All resources -> OnlineStore API Management service

-> APIs -> APIs -> Create from Azure resource -> Function App

-> Browse -> Import Azure Functions pane appears

-> Select -> Select Azure Function App pane appears

-> Select OrderFunction\*\*\*\*\*\*\* App -> Select -> Select OrderDetails function -> Select

-> “API URL suffix” = “orders” -> Create

### Test the OnlineStore orders endpoint in the portal

[Test orders endpoint](#_Test_the_OnlineStore_1)

Notice that both the functions can now be called through endpoints within the **azure-api.net** domain (as defined by the GATEWAY\_URL), which is the domain used by Azure API Management.

In other Learn modules**, you can learn how to**

* + - **apply policies,**
    - **security settings,**
    - **external caches,**
    - **and other features to the functions in an API Management Gateway.**

**APIM gateway provides you with a central control point, where you can manage multiple microservices without altering their code.**