

Azure Cognitive Services documentation

Learn how to use ready-made AI services to build intelligent apps, websites, and bots. Develop software that can see, hear, speak, and interpret your user's needs.



OVERVIEW
[What are Cognitive Services?](#)



CONCEPT
[Cognitive Services on-premises containers](#)



CONCEPT
[Responsible use of AI](#)



CONCEPT
[Development options](#)



CONCEPT
[Cognitive Services and machine learning](#)



CONCEPT
[Cognitive Services with databases](#)



REFERENCE
[Language support](#)



REFERENCE
[Support and help options](#)

Vision

Recognize, identify, caption, index, and moderate your pictures, videos, and digital ink content.

- [Computer Vision](#)
- [Custom Vision](#)
- [Face](#)

Language

Allow your apps to process natural language with pre-built scripts, evaluate sentiment and learn how to recognize what users want.

- [Azure Cognitive Service for Language](#)
- [Language Understanding \(LUIS\)](#)
- [QnA Maker](#)
- [Translator](#)

Speech

Convert speech into text and text into natural-sounding speech. Translate from one language to another and enable speaker verification and recognition.

 [Speech service](#)

 [Customize with Speech Studio](#) ↗

Decision

Build apps that surface recommendations for informed and efficient decision-making.

 [Anomaly Detector](#)

 [Content Moderator](#)

 [Personalizer](#)

Azure OpenAI

Apply advanced language models to variety of use cases with the Azure OpenAI service

 [Azure OpenAI](#)

 [Azure OpenAI Models](#)

 [Get started generating text using Azure OpenAI](#)

 [Generate embeddings with Azure OpenAI](#)

 [Customize a model for your application](#)

 [Apply for access to Azure OpenAI](#) ↗

Cognitive Service Containers

Container support in Azure Cognitive Services allows developers to use the same rich APIs that are available in Azure, and enables flexibility in where to deploy and host the services that come with...

 [Cognitive Services containers documentation](#)

 [What are Cognitive Services containers?](#)

 [Deploy and run container on Azure Container Instance](#)

 [Use Docker Compose to deploy multiple containers](#)

 [Use Docker containers disconnected from the internet](#)

 [Cognitive Services containers FAQ](#)

Cognitive Services Use Cases

Real-world scenarios for using Cognitive Services in business solutions

 [Captioning](#)

 [Call Center Transcription](#)

Additional resources

Portals for customization

[Custom Translator](#) ↗

[Custom Vision](#) ↗

[Language Understanding \(LUIS\)](#) ↗

SDKs & REST APIs

[C#](#)

[Go](#) ↗

[Java](#)

[QnA Maker ↗](#)

[Speech Studio ↗](#)

[JavaScript](#)

[Python](#)

[R ↗](#)

[Apache Spark and Databases](#)

[Cognitive Services Management REST API](#)

Explore other Azure AI Services

[Azure Applied AI Services](#)

[Azure Machine Learning](#)

Tools

[Azure CLI](#)

[PowerShell](#)

What are Azure Cognitive Services?

Article • 11/07/2022 • 3 minutes to read

Azure Cognitive Services are cloud-based artificial intelligence (AI) services that help developers build cognitive intelligence into applications without having direct AI or data science skills or knowledge. They are available through REST APIs and client library SDKs in popular development languages. Azure Cognitive Services enables developers to easily add cognitive features into their applications with cognitive solutions that can see, hear, speak, and analyze.

Categories of Cognitive Services

Cognitive Services can be categorized into four main pillars:

- Vision
- Speech
- Language
- Decision

See the tables below to learn about the services offered within those categories.

Vision APIs

Service Name	Service Description	Quickstart
Computer Vision	The Computer Vision service provides you with access to advanced cognitive algorithms for processing images and returning information.	Computer Vision quickstart
Custom Vision	The Custom Vision Service lets you build, deploy, and improve your own image classifiers. An image classifier is an AI service that applies labels to images, based on their visual characteristics.	Custom Vision quickstart
Face	The Face service provides access to advanced face algorithms, enabling face attribute detection and recognition.	Face quickstart

Speech APIs

Service Name	Service Description	Quickstart

Service Name	Service Description	Quickstart
Speech service	Speech service adds speech-enabled features to applications. Speech service includes various capabilities like speech-to-text, text-to-speech, speech translation, and many more.	Go to the Speech documentation to choose a subservice quickstart.

Language APIs

Service Name	Service Description	Quickstart
Language service	Azure Language service provides several Natural Language Processing (NLP) features to understand and analyze text.	Go to the Language documentation to choose a subservice quickstart.
Translator	Translator provides machine-based text translation in near real time.	Translator quickstart
Language Understanding LUIS	Language Understanding (LUIS) is a cloud-based conversational AI service that applies custom machine-learning intelligence to a user's conversational or natural language text to predict overall meaning and pull out relevant information.	LUIS quickstart
QnA Maker	QnA Maker allows you to build a question and answer service from your semi-structured content.	QnA Maker quickstart

Decision APIs

Service Name	Service Description	Quickstart
Anomaly Detector	Anomaly Detector allows you to monitor and detect abnormalities in your time series data.	Anomaly Detector quickstart
Content Moderator	Content Moderator provides monitoring for possible offensive, undesirable, and risky content.	Content Moderator quickstart
Personalizer	Personalizer allows you to choose the best experience to show to your users, learning from their real-time behavior.	Personalizer quickstart

Create a Cognitive Services resource

You can create a Cognitive Services resource with hands-on quickstarts using any of the following methods:

- [Azure portal](#)
- [Azure CLI](#)
- [Azure SDK client libraries](#)
- [Azure Resource Manager \(ARM template\)](#)

Use Cognitive Services in different development environments

With Azure and Cognitive Services, you have access to several development options, such as:

- Automation and integration tools like Logic Apps and Power Automate.
- Deployment options such as Azure Functions and the App Service.
- Cognitive Services Docker containers for secure access.
- Tools like Apache Spark, Azure Databricks, Azure Synapse Analytics, and Azure Kubernetes Service for big data scenarios.

To learn more, see [Cognitive Services development options](#).

Containers for Cognitive Services

Azure Cognitive Services also provides several Docker containers that let you use the same APIs that are available from Azure, on-premises. These containers give you the flexibility to bring Cognitive Services closer to your data for compliance, security, or other operational reasons. For more information, see [Cognitive Services Containers](#).

Regional availability

The APIs in Cognitive Services are hosted on a growing network of Microsoft-managed data centers. You can find the regional availability for each API in [Azure region list](#).

Looking for a region we don't support yet? Let us know by filing a feature request on our [UserVoice forum](#).

Language support

Cognitive Services supports a wide range of cultural languages at the service level. You can find the language availability for each API in the [supported languages list](#).

Security

Azure Cognitive Services provides a layered security model, including [authentication](#) with Azure Active Directory credentials, a valid resource key, and [Azure Virtual Networks](#).

Certifications and compliance

Cognitive Services has been awarded certifications such as CSA STAR Certification, FedRAMP Moderate, and HIPAA BAA. You can [download ↗](#) certifications for your own audits and security reviews.

To understand privacy and data management, go to the [Trust Center ↗](#).

Help and support

Cognitive Services provides several support options to help you move forward with creating intelligent applications. Cognitive Services also has a strong community of developers that can help answer your specific questions. For a full list of support options available to you, see [Cognitive Services support and help options](#).

Next steps

- Select a service from the tables above and learn how it can help you meet your development goals.
- [Create a Cognitive Services resource using the Azure portal](#)
- [Plan and manage costs for Cognitive Services](#)

Natural language support for Azure Cognitive Services

Article • 01/11/2023 • 2 minutes to read

Azure Cognitive Services enable you to build applications that see, hear, speak with, and understand your users. Between these services, more than three dozen languages are supported, allowing users to communicate with your application in natural ways. Use the links below to view language availability by service.

These Cognitive Services are language agnostic and don't have limitations based on human language.

- [Anomaly Detector](#)
- [Custom Vision](#)
- [Face](#)
- [Personalizer](#)

Vision

- [Computer Vision](#)
- [Ink Recognizer \(Preview\)](#)
- [Video Indexer](#)

Language

- [Language Understanding \(LUIS\)](#)
- [QnA Maker](#)
- [Language service](#)
- [Translator](#)

Speech

- [Speech Service: Speech-to-Text](#)
- [Speech Service:Text-to-Speech](#)
- [Speech Service: Speech Translation](#)

Decision

- Content Moderator

See also

- [What are the Cognitive Services?](#)
- [Create an account](#)

Additional resources

Documentation

[Post-call transcription and analytics quickstart - Speech service - Azure Cognitive Services](#)

In this quickstart, you perform sentiment analysis and conversation summarization of call center transcriptions.

[Sentiment Analysis and Opinion Mining language support - Azure Cognitive Services](#)

This article explains which languages are supported by the Sentiment Analysis and Opinion Mining features of Azure Cognitive Service for Language.

[Azure Cognitive Services for Call Center Overview - Azure Cognitive Services](#)

Azure Cognitive Services for Language and Speech can help you realize partial or full automation of telephony-based customer interactions, and provide accessibility across multiple channels.

[Quickstart: Use the NER client library - Azure Cognitive Services](#)

Use this quickstart to start using the Named Entity Recognition (NER) API.

[Text Analysis Runtime - Analyze Text - REST API \(Azure Cognitive Services - Language\)](#)

Request text analysis over a collection of documents. Submit a collection of text documents for analysis. Specify a single unique task to be executed immediate

[What is custom named entity recognition in Azure Cognitive Service for Language \(preview\) - Azure Cognitive Services](#)

Learn how use custom named entity recognition.

[Quickstart: Use the Key Phrase Extraction client library - Azure Cognitive Services](#)

Use this quickstart to start using the Key Phrase Extraction API.

[Language Detection language support - Azure Cognitive Services](#)

This article explains which natural languages are supported by the Language Detection API.

[Show 5 more](#)

Quickstart: Create a Cognitive Services resource using the Azure portal

Article • 09/11/2022 • 14 minutes to read

Use this quickstart to create a Cognitive Services resource. After you create a Cognitive Service resource in the Azure portal, you'll get an endpoint and a key for authenticating your applications.

Azure Cognitive Services are cloud-based artificial intelligence (AI) services that help developers build cognitive intelligence into applications without having direct AI or data science skills or knowledge. They are available through REST APIs and client library SDKs in popular development languages. Azure Cognitive Services enables developers to easily add cognitive features into their applications with cognitive solutions that can see, hear, speak, and analyze.

Types of Cognitive Services resources

You can access Azure Cognitive Services through two different resources: A multi-service resource, or a single-service one.

- Multi-service resource:
 - Access multiple Azure Cognitive Services with a single key and endpoint.
 - Consolidates billing from the services you use.
- Single-service resource:
 - Access a single Azure Cognitive Service with a unique key and endpoint for each service created.
 - Use the free tier to try out the service.

Prerequisites

- A valid Azure subscription - [Create one for free](#).
- Your Azure account must have a Cognitive Services Contributor role assigned in order for you to agree to the responsible AI terms and create a resource. To get this role assigned to your account, follow the steps in the [Assign roles](#) documentation, or contact your administrator.

Create a new Azure Cognitive Services resource

The multi-service resource is named **Cognitive Services** in the portal. The multi-service resource enables access to the following Cognitive Services:

- **Decision** - Content Moderator
- **Language** - Language, Translator
- **Speech** - Speech
- **Vision** - Computer Vision, Custom Vision, Face

1. You can select this link to create an Azure Cognitive multi-service resource:

[Create a Cognitive Services resource](#).

2. On the **Create** page, provide the following information:

Project details	Description
Subscription	Select one of your available Azure subscriptions.
Resource group	The Azure resource group that will contain your Cognitive Services resource. You can create a new group or add it to a pre-existing group.
Region	The location of your cognitive service instance. Different locations may introduce latency, but have no impact on the runtime availability of your resource.
Name	A descriptive name for your cognitive services resource. For example, <i>MyCognitiveServicesResource</i> .
Pricing tier	The cost of your Cognitive Services account depends on the options you choose and your usage. For more information, see the API pricing details .

Create Cognitive Services

X

Basics Tags Review + create

Get access to Vision, Language, Search, and Speech Cognitive Services with a single API key. Quickly connect services together to achieve more insights into your content and easily integrate with other services like Azure Search. [Learn more](#)

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ

Resource group * ⓘ

[Create new](#)

Instance details

Region * ⓘ



Location specifies the region only for included regional services. This does not specify a region for included non-regional services. [Click here for more details.](#)

Name * ⓘ



Pricing tier * ⓘ



[View full pricing details](#)

By checking this box, I certify that use of this service is not by or for a police department in the United States.

I confirm I have read and understood the notice below.

[Review + create](#)

< Previous

Next : Tags >

3. Configure other settings for your resource as needed, read and accept the conditions (as applicable), and then select **Review + create**.

Tip

If your subscription doesn't allow you to create a Cognitive Service resource, you may need to enable the privilege of that **Azure resource provider** using the **Azure portal**, **PowerShell command** or an **Azure CLI command**. If you are not the subscription owner, ask the *Subscription Owner* or someone with a role of *admin* to complete the registration for you or ask for the */register/action* privileges to be granted to your account.

Get the keys for your resource

1. After your resource is successfully deployed, select **Next Steps > Go to resource**.

The screenshot shows the 'Deployment details' section of the Azure portal. It includes a table with one row for 'DocsCognitiveServicesResource'. The table columns are 'Resource', 'Type', 'Status', and 'Operation details'. The 'Resource' column shows a green checkmark icon and the name 'DocsCognitiveServicesResource'. The 'Type' column shows 'Microsoft.CognitiveServices/accounts'. The 'Status' column shows 'Created'. The 'Operation details' column contains a blue link labeled 'Operation details'. Below the table is a section titled 'Next steps' with a blue button labeled 'Go to resource'.

2. From the quickstart pane that opens, you can access the resource endpoint and manage keys.

Configure an environment variable for authentication

Your application must be authenticated to access Cognitive Services resources. For production, use a secure way of storing and accessing your credentials. For example, after you have a key for your resource, write it to a new environment variable on the local machine running the application.

Tip

Don't include the key directly in your code, and never post it publicly. See the Cognitive Services **security** article for more authentication options like **Azure Key Vault**.

To set the environment variable for your resource key, open a console window, and follow the instructions for your operating system and development environment. To set the `COGNITIVE_SERVICE_KEY` environment variable, replace `your-key` with one of the keys for your resource.

Windows

Console

```
setx COGNITIVE_SERVICE_KEY your-key
```

Note

If you only need to access the environment variable in the current running console, you can set the environment variable with `set` instead of `setx`.

After you add the environment variable, you may need to restart any running consoles or other programs that will need to read the environment variable. For example, if you are using Visual Studio as your editor, restart Visual Studio before running the example.

To set the environment variable for your Speech resource region, follow the same steps. Set `COGNITIVE_SERVICE_REGION` to the region of your resource. For example, `westus`.

Clean up resources

If you want to clean up and remove a Cognitive Services subscription, you can delete the resource or resource group. Deleting the resource group also deletes any other resources contained in the group.

1. In the Azure portal, expand the menu on the left side to open the menu of services, and choose **Resource Groups** to display the list of your resource groups.
2. Locate the resource group containing the resource to be deleted.
3. If you want to delete the entire resource group, select the resource group name. On the next page, Select **Delete resource group**, and confirm.
4. If you want to delete only the Cognitive Service resource, select the resource group to see all the resources within it. On the next page, select the resource that you want to delete, select the ellipsis menu for that row, and select **Delete**.

If you need to recover a deleted resource, see [Recover deleted Cognitive Services resources](#).

See also

- See [Authenticate requests to Azure Cognitive Services](#) on how to securely work with Cognitive Services.
- See [What are Azure Cognitive Services?](#) to get a list of different categories within Cognitive Services.
- See [Natural language support](#) to see the list of natural languages that Cognitive Services supports.
- See [Use Cognitive Services as containers](#) to understand how to use Cognitive Services on-premises.

- See [Plan and manage costs for Cognitive Services](#) to estimate cost of using Cognitive Services.

Quickstart: Create a Cognitive Services resource using the Azure CLI

Article • 09/11/2022 • 8 minutes to read

Use this quickstart to create a Cognitive Services resource using [Azure Command-Line Interface \(CLI\)](#) commands. After you create the resource, use the keys and endpoint generated for you to authenticate your applications.

Azure Cognitive Services are cloud-based artificial intelligence (AI) services that help developers build cognitive intelligence into applications without having direct AI or data science skills or knowledge. They are available through REST APIs and client library SDKs in popular development languages. Azure Cognitive Services enables developers to easily add cognitive features into their applications with cognitive solutions that can see, hear, speak, and analyze.

Types of Cognitive Services resources

You can access Azure Cognitive Services through two different resources: A multi-service resource, or a single-service one.

- Multi-service resource:
 - Access multiple Azure Cognitive Services with a single key and endpoint.
 - Consolidates billing from the services you use.
- Single-service resource:
 - Access a single Azure Cognitive Service with a unique key and endpoint for each service created.
 - Use the free tier to try out the service.

Prerequisites

- A valid Azure subscription - [Create one](#) for free.
- The [Azure CLI](#)
- Your Azure account must have a Cognitive Services Contributor role assigned in order for you to agree to the responsible AI terms and create a resource. To get this role assigned to your account, follow the steps in the [Assign roles](#) documentation, or contact your administrator.
- You must create your first Face, Language service, or Computer Vision resources from the Azure portal to review and acknowledge the terms and conditions. You can do so here: [Face](#), [Language service](#), [Computer Vision](#). After that, you can

create subsequent resources using any deployment tool (SDK, CLI, or ARM template, etc) under the same Azure subscription.

Install the Azure CLI and sign in

Install the [Azure CLI](#). To sign into your local installation of the CLI, run the `az login` command:

```
Azure CLI  
az login
```

You can also use the green Try It button to run these commands in your browser.

Create a new Azure Cognitive Services resource group

Before you create a Cognitive Services resource, you must have an Azure resource group to contain the resource. When you create a new resource, you can either create a new resource group, or use an existing one. This article shows how to create a new resource group.

Choose your resource group location

To create a resource, you'll need one of the Azure locations available for your subscription. You can retrieve a list of available locations with the [az account list-locations](#) command. Most Cognitive Services can be accessed from several locations. Choose the one closest to you, or see which locations are available for the service.

ⓘ Important

- Remember your Azure location, as you will need it when calling the Azure Cognitive Services resources.
- The availability of some Cognitive Services can vary by region. For more information, see [Azure products by region](#).

```
Azure CLI  
az account list-locations --query "[].{Region:name}" --out table
```

After you have your Azure location, create a new resource group in the Azure CLI using the [az group create](#) command. In the example below, replace the Azure location `westus2` with one of the Azure locations available for your subscription.

Azure CLI

```
az group create --name cognitive-services-resource-group --location westus2
```

Create a Cognitive Services resource

Choose a service and pricing tier

When you create a new resource, you'll need to know the kind of service you want to use, along with the [pricing tier](#) (or SKU) you want. You'll use this and other information as parameters when you create the resource.

You can access Azure Cognitive Services through two different resources: A multi-service resource, or a single-service one.

- Multi-service resource:
 - Access multiple Azure Cognitive Services with a single key and endpoint.
 - Consolidates billing from the services you use.
- Single-service resource:
 - Access a single Azure Cognitive Service with a unique key and endpoint for each service created.
 - Use the free tier to try out the service.

See the list of SKUs and pricing information below.

Multi-service

Service	Kind
Multiple services. For more information, see the pricing page.	CognitiveServices

Vision

Service	Kind
Computer Vision	ComputerVision

Service	Kind
Custom Vision - Prediction	CustomVision.Prediction
Custom Vision - Training	CustomVision.Training
Face	Face
Form Recognizer	FormRecognizer

Speech

Service	Kind
Speech Services	SpeechServices

Language

Service	Kind
LUIS	Luis
QnA Maker	QnAMaker
Language service	TextAnalytics
Text Translation	TextTranslation

Decision

Service	Kind
Anomaly Detector	AnomalyDetector
Content Moderator	ContentModerator
Personalizer	Personalizer

Pricing tiers and billing

Pricing tiers (and the amount you get billed) are based on the number of transactions you send using your authentication information. Each pricing tier specifies the:

- maximum number of allowed transactions per second (TPS).

- service features enabled within the pricing tier.
- cost for a predefined number of transactions. Going above this number will cause an extra charge as specified in the [pricing details](#) for your service.

ⓘ Note

Many of the Cognitive Services have a free tier you can use to try the service. To use the free tier, use `F0` as the SKU for your resource.

You can find a list of available Cognitive Service "kinds" with the [az cognitiveservices account list-kinds](#) command:

Azure CLI

```
az cognitiveservices account list-kinds
```

Add a new resource to your resource group

To create and subscribe to a new Cognitive Services resource, use the [az cognitiveservices account create](#) command. This command adds a new billable resource to the resource group you created earlier. When you create your new resource, you'll need to know the "kind" of service you want to use, along with its pricing tier (or SKU) and an Azure location:

You can create an F0 (free) resource for Anomaly Detector, named `anomaly-detector-resource` with the command below.

Azure CLI

```
az cognitiveservices account create --name anomaly-detector-resource --resource-group cognitive-services-resource-group --kind AnomalyDetector --sku F0 --location westus2 --yes
```

💡 Tip

If your subscription doesn't allow you to create a Cognitive Service resource, you may need to enable the privilege of that **Azure resource provider** using the **Azure portal**, **PowerShell command** or an **Azure CLI command**. If you are not the subscription owner, ask the *Subscription Owner* or someone with a role of *admin* to complete the registration for you or ask for the `/register/action` privileges to be granted to your account.

Get the keys for your resource

To log into your local installation of the Command-Line Interface(CLI), use the [az login](#) command.

```
Azure CLI
```

```
az login
```

Use the [az cognitiveservices account keys list](#) command to get the keys for your Cognitive Service resource.

```
Azure CLI
```

```
az cognitiveservices account keys list --name anomaly-detector-resource  
--resource-group cognitive-services-resource-group
```

Configure an environment variable for authentication

Your application must be authenticated to access Cognitive Services resources. For production, use a secure way of storing and accessing your credentials. For example, after you have a key for your resource, write it to a new environment variable on the local machine running the application.

Tip

Don't include the key directly in your code, and never post it publicly. See the Cognitive Services [security](#) article for more authentication options like [Azure Key Vault](#).

To set the environment variable for your resource key, open a console window, and follow the instructions for your operating system and development environment. To set the `COGNITIVE_SERVICE_KEY` environment variable, replace `your-key` with one of the keys for your resource.

Windows

Console

```
setx COGNITIVE_SERVICE_KEY your-key
```

➊ Note

If you only need to access the environment variable in the current running console, you can set the environment variable with `set` instead of `setx`.

After you add the environment variable, you may need to restart any running consoles or other programs that will need to read the environment variable. For example, if you are using Visual Studio as your editor, restart Visual Studio before running the example.

To set the environment variable for your Speech resource region, follow the same steps. Set `COGNITIVE_SERVICE_REGION` to the region of your resource. For example, `westus`.

Pricing tiers and billing

Pricing tiers (and the amount you get billed) are based on the number of transactions you send using your authentication information. Each pricing tier specifies the:

- maximum number of allowed transactions per second (TPS).
- service features enabled within the pricing tier.
- The cost for a predefined number of transactions. Going above this amount will cause an extra charge as specified in the [pricing details](#) for your service.

Get current quota usage for your resource

Use the [az cognitiveservices account list-usage](#) command to get the usage for your Cognitive Service resource.

Azure CLI

```
az cognitiveservices account list-usage --name anomaly-detector-resource --  
resource-group cognitive-services-resource-group --subscription  
subscription-name
```

Clean up resources

If you want to clean up and remove a Cognitive Services resource, you can delete it or the resource group. Deleting the resource group also deletes any other resources contained in the group.

To remove the resource group and its associated resources, use the `az group delete` command.

Azure CLI

```
az group delete --name cognitive-services-resource-group
```

If you need to recover a deleted resource, see [Recover deleted Cognitive Services resources](#).

See also

- See [Authenticate requests to Azure Cognitive Services](#) on how to securely work with Cognitive Services.
- See [What are Azure Cognitive Services?](#) to get a list of different categories within Cognitive Services.
- See [Natural language support](#) to see the list of natural languages that Cognitive Services supports.
- See [Use Cognitive Services as containers](#) to understand how to use Cognitive Services on-prem.
- See [Plan and manage costs for Cognitive Services](#) to estimate cost of using Cognitive Services.

Quickstart: Create a Cognitive Services resource using the Azure Management client library

Article • 09/11/2022 • 29 minutes to read

Choose a programming language

C#

Java

JavaScript

Python

Use this quickstart to create and manage Azure Cognitive Services resources using the Azure Management client library.

Azure Cognitive Services are cloud-based artificial intelligence (AI) services that help developers build cognitive intelligence into applications without having direct AI or data science skills or knowledge. They are available through REST APIs and client library SDKs in popular development languages. Azure Cognitive Services enables developers to easily add cognitive features into their applications with cognitive solutions that can see, hear, speak, and analyze.

Individual AI services are represented by Azure [resources](#) that you create under your Azure subscription. After you create a resource, you can use the keys and endpoint generated to authenticate your applications.

[Reference documentation](#) | [Library source code](#) | [Package \(NuGet\)](#) | [Samples](#)

C# prerequisites

- A valid Azure subscription - [Create one for free](#).
- The current version of [.NET Core](#).
- Your Azure account must have a Cognitive Services Contributor role assigned in order for you to agree to the responsible AI terms and create a resource. To get this role assigned to your account, follow the steps in the [Assign roles](#) documentation, or contact your administrator.
- You must create your first Face, Language service, or Computer Vision resources from the Azure portal to review and acknowledge the terms and conditions. You can do so here: [Face](#), [Language service](#), [Computer Vision](#). After that, you can create subsequent resources using any deployment tool (SDK, CLI, or ARM template, etc) under the same Azure subscription.

Create an Azure Service Principal

To have your application interact with your Azure account, you need an Azure service principal to manage permissions. Follow the instructions in [Create an Azure service principal](#).

When you create a service principal, you'll see it has a secret value, an ID, and an application ID. Save the application ID and secret to a temporary location for later steps.

Create a resource group

Before you create a Cognitive Services resource, your account must have an Azure resource group to contain the resource. If you don't already have a resource group, create one in the [Azure portal](#) before continuing.

Create a new C# application

Create a new .NET Core application. In a console window (such as cmd, PowerShell, or Bash), use the `dotnet new` command to create a new console app with the name `azure-management-quickstart`. This command creates a simple "Hello World" C# project with a single source file: *program.cs*.

```
Console
```

```
dotnet new console -n azure-management-quickstart
```

Change your directory to the newly created app folder. You can build the application with:

```
Console
```

```
dotnet build
```

The build output should contain no warnings or errors.

```
Console
```

```
...
Build succeeded.
  0 Warning(s)
  0 Error(s)
...
```

Install the client library

Within the application directory, install the Azure Management client library for .NET with the following command:

Console

```
dotnet add package Microsoft.Azure.Management.CognitiveServices  
dotnet add package Microsoft.Azure.Management.Fluent  
dotnet add package Microsoft.Azure.Management.ResourceManager.Fluent
```

If you're using the Visual Studio IDE, the client library is available as a downloadable NuGet package.

Import libraries

Open *program.cs* and add the following `using` statements to the top of the file:

C#

```
using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Text;  
using System.Threading.Tasks;  
using Microsoft.Azure.Management.Fluent;  
using Microsoft.Azure.Management.ResourceManager.Fluent;  
using Microsoft.Azure.Management.ResourceManager.Fluent.Authentication;  
using Microsoft.Azure.Management.CognitiveServices;  
using Microsoft.Azure.Management.CognitiveServices.Models;
```

Authenticate the client

Add the following fields to the root of *program.cs* and populate their values, using the service principal you created and your Azure account information.

C#

```
const string service_principal_application_id =  
    "PASTE_YOUR_SERVICE_PRINCIPAL_APPLICATION_ID_HERE";  
const string service_principal_secret =  
    "PASTE_YOUR_SERVICE_PRINCIPAL_SECRET_HERE";  
  
/* The ID of your Azure subscription. You can find this in the Azure  
Dashboard under Home > Subscriptions. */  
const string subscription_id = "PASTE_YOUR_SUBSCRIPTION_ID_HERE";
```

```

/* The Active Directory tenant ID. You can find this in the Azure Dashboard
under Home > Azure Active Directory. */
const string tenant_id = "PASTE_YOUR_TENANT_ID_HERE";

/* The name of the Azure resource group in which you want to create the
resource.
You can find resource groups in the Azure Dashboard under Home > Resource
groups. */
const string resource_group_name = "PASTE_YOUR_RESOURCE_GROUP_NAME_HERE";

/* The name of the custom subdomain to use when you create the resource.
This is optional.
For example, if you create a Bing Search v7 resource with the custom
subdomain name 'my-search-resource',
your resource would have the endpoint https://my-search-
resource.cognitiveservices.azure.com/.
Note not all Cognitive Services allow custom subdomain names. */
const string subdomain_name = "PASTE_YOUR_SUBDOMAIN_NAME_HERE";

```

Then, in your **Main** method, use these values to construct a **CognitiveServicesManagementClient** object. This object is needed for all of your Azure management operations.

C#

```

var service_principal_credentials = new ServicePrincipalLoginInformation ();
service_principal_credentials.ClientId = service_principal_application_id;
service_principal_credentials.ClientSecret = service_principal_secret;

var credentials =
SdkContext.AzureCredentialsFactory.FromServicePrincipal(service_principal_ap-
plication_id, service_principal_secret, tenant_id,
AzureEnvironment.AzureGlobalCloud);
var client = new CognitiveServicesManagementClient(credentials);
client.SubscriptionId = subscription_id;

```

Call management methods

Add the following code to your **Main** method to list available resources, create a sample resource, list your owned resources, and then delete the sample resource. You'll define these methods in the next steps.

C#

```

// Uncomment to list all available resource kinds, SKUs, and locations
for your Azure account:
//list_available_kinds_skus_locations(client);

```

```

    // Create a resource with kind TextTranslation, F0 (free tier), location
    global.
    create_resource(client, "test_resource", "TextTranslation", "F0",
    "Global");

    // List all resources for your Azure account and resource group:
    list_resources(client);

    // Delete the resource.
    delete_resource(client, "test_resource");

    Console.WriteLine("Press any key to exit.");
    Console.ReadKey();

```

Create a Cognitive Services resource (C#)

To create and subscribe to a new Cognitive Services resource, use the `Create` method. This method adds a new billable resource to the resource group you pass in. When creating your new resource, you'll need to know the "kind" of service you want to use, along with its pricing tier (or SKU) and an Azure location. The following method takes all of these as arguments and creates a resource.

C#

```

static void create_resource(CognitiveServicesManagementClient client, string
resource_name, string kind, string account_tier, string location)
{
    Console.WriteLine("Creating resource: " + resource_name + "...");
    /* NOTE If you do not want to use a custom subdomain name, remove the
customSubDomainName
    property from CognitiveServicesAccountProperties. */
    CognitiveServicesAccount parameters =
        new CognitiveServicesAccount(null, null, kind, location,
resource_name, new CognitiveServicesAccountProperties(customSubDomainName :
subdomain_name), new Sku(account_tier));
    var result = client.Accounts.Create(resource_group_name, resource_name,
parameters);
    Console.WriteLine("Resource created.");
    Console.WriteLine("ID: " + result.Id);
    Console.WriteLine("Kind: " + result.Kind);
    Console.WriteLine();
}

```

Choose a service and pricing tier

When you create a new resource, you'll need to know the "kind" of service you want to use, along with the [pricing tier](#) (or SKU) you want. You'll use this and other

information as parameters when creating the resource. You can find a list of available Cognitive Service "kinds" by calling the following method in your script:

```
C#  
  
static void  
list_available_kinds_skus_locations(CognitiveServicesManagementClient  
client)  
{  
  
    Console.WriteLine("Available SKUs:");  
    var result = client.ResourceSkus.List();  
    Console.WriteLine("Kind\tSKU Name\tSKU Tier\tLocations");  
    foreach (var x in result) {  
        var locations = "";  
        foreach (var region in x.Locations)  
        {  
            locations += region;  
        }  
        Console.WriteLine(x.Kind + "\t" + x.Name + "\t" + x.Tier + "\t" +  
locations);  
    };  
}
```

You can access Azure Cognitive Services through two different resources: A multi-service resource, or a single-service one.

- Multi-service resource:
 - Access multiple Azure Cognitive Services with a single key and endpoint.
 - Consolidates billing from the services you use.
- Single-service resource:
 - Access a single Azure Cognitive Service with a unique key and endpoint for each service created.
 - Use the free tier to try out the service.

See the list of SKUs and pricing information below.

Multi-service

Service	Kind
Multiple services. For more information, see the pricing page.	CognitiveServices

Vision

Service	Kind
Computer Vision	ComputerVision
Custom Vision - Prediction	CustomVision.Prediction
Custom Vision - Training	CustomVision.Training
Face	Face
Form Recognizer	FormRecognizer

Speech

Service	Kind
Speech Services	SpeechServices

Language

Service	Kind
LUIS	LUIS
QnA Maker	QnAMaker
Language service	TextAnalytics
Text Translation	TextTranslation

Decision

Service	Kind
Anomaly Detector	AnomalyDetector
Content Moderator	ContentModerator
Personalizer	Personalizer

Pricing tiers and billing

Pricing tiers (and the amount you get billed) are based on the number of transactions you send using your authentication information. Each pricing tier specifies the:

- maximum number of allowed transactions per second (TPS).
- service features enabled within the pricing tier.
- cost for a predefined number of transactions. Going above this number will cause an extra charge as specified in the [pricing details](#) for your service.

 **Note**

Many of the Cognitive Services have a free tier you can use to try the service. To use the free tier, use `F0` as the SKU for your resource.

View your resources

To view all of the resources under your Azure account (across all resource groups), use the following method:

C#

```
static void list_resources(CognitiveServicesManagementClient client)
{
    Console.WriteLine("Resources in resource group: " +
resource_group_name);
    var result = client.Accounts.ListByResourceGroup(resource_group_name);
    foreach (var x in result)
    {
        Console.WriteLine("ID: " + x.Id);
        Console.WriteLine("Name: " + x.Name);
        Console.WriteLine("Type: " + x.Type);
        Console.WriteLine("Kind: " + x.Kind);
        Console.WriteLine();
    }
}
```

Delete a resource

The following method deletes the specified resource from the given resource group.

C#

```
static void delete_resource(CognitiveServicesManagementClient client, string
resource_name)
{
    Console.WriteLine("Deleting resource: " + resource_name + "...");
    client.Accounts.Delete (resource_group_name, resource_name);

    Console.WriteLine("Resource deleted.");
}
```

```
        Console.WriteLine();  
    }  
}
```

If you need to recover a deleted resource, see [Recover deleted Cognitive Services resources](#).

Run the application

Run the application from your application directory with the `dotnet run` command.

```
dotnet
```

```
dotnet run
```

See also

- See [Authenticate requests to Azure Cognitive Services](#) on how to securely work with Cognitive Services.
- See [What are Azure Cognitive Services?](#) to get a list of different categories within Cognitive Services.
- See [Natural language support](#) to see the list of natural languages that Cognitive Services supports.
- See [Use Cognitive Services as containers](#) to understand how to use Cognitive Services on-prem.
- See [Plan and manage costs for Cognitive Services](#) to estimate cost of using Cognitive Services.
- See [Cognitive Services Management SDK reference documentation](#) for more details on the management SDK.

Quickstart: Create a Cognitive Services resource using Bicep

Article • 01/19/2023 • 4 minutes to read

Follow this quickstart to create Cognitive Services resource using Bicep.

Azure Cognitive Services are cloud-based artificial intelligence (AI) services that help developers build cognitive intelligence into applications without having direct AI or data science skills or knowledge. They are available through REST APIs and client library SDKs in popular development languages. Azure Cognitive Services enables developers to easily add cognitive features into their applications with cognitive solutions that can see, hear, speak, and analyze.

[Bicep](#) is a domain-specific language (DSL) that uses declarative syntax to deploy Azure resources. It provides concise syntax, reliable type safety, and support for code reuse. Bicep offers the best authoring experience for your infrastructure-as-code solutions in Azure.

Things to consider

Using Bicep to create a Cognitive Service resource lets you create a multi-service resource. This enables you to:

- Access multiple Azure Cognitive Services with a single key and endpoint.
- Consolidate billing from the services you use.
- You must create your first Face, Language service, or Computer Vision resources from the Azure portal to review and acknowledge the terms and conditions. You can do so here: [Face](#) , [Language service](#) , [Computer Vision](#). After that, you can create subsequent resources using any deployment tool (SDK, CLI, or ARM template, etc) under the same Azure subscription.

Prerequisites

- If you don't have an Azure subscription, [create one for free](#) .

Review the Bicep file

The Bicep file used in this quickstart is from [Azure Quickstart Templates](#) .

⚠ Note

- If you use a different resource `kind` (listed below), you may need to change the `sku` parameter to match the `pricing` tier you wish to use. For example, the `TextAnalytics` kind uses `S` instead of `S0`.
- Many of the Cognitive Services have a free `F0` pricing tier that you can use to try the service.

Be sure to change the `sku` parameter to the `pricing` instance you want. The `sku` depends on the resource `kind` that you are using. For example, `TextAnalytics`

Bicep

```
@description('That name is the name of our application. It has to be unique. Type a name followed by your resource group name. (<name>-<resourceGroupName>)')  
param cognitiveServiceName string =  
'CognitiveService-${uniqueString(resourceGroup().id)}'  
  
@description('Location for all resources.')  
param location string = resourceGroup().location  
  
@allowed([  
    'S0'  
)  
param sku string = 'S0'  
  
resource cognitiveService 'Microsoft.CognitiveServices/accounts@2021-10-01'  
= {  
    name: cognitiveServiceName  
    location: location  
    sku: {  
        name: sku  
    }  
    kind: 'CognitiveServices'  
    properties: {  
        apiProperties: {  
            statisticsEnabled: false  
        }  
    }  
}
```

One Azure resource is defined in the Bicep file: [Microsoft.CognitiveServices/accounts](#) specifies that it is a Cognitive Services resource. The `kind` field in the Bicep file defines the type of resource.

See the list of SKUs and pricing information below.

Multi-service

Service	Kind
Multiple services. For more information, see the pricing page.	CognitiveServices

Vision

Service	Kind
Computer Vision	ComputerVision
Custom Vision - Prediction	CustomVision.Prediction
Custom Vision - Training	CustomVision.Training
Face	Face
Form Recognizer	FormRecognizer

Speech

Service	Kind
Speech Services	SpeechServices

Language

Service	Kind
LUIS	LUIS
QnA Maker	QnAMaker
Language service	TextAnalytics
Text Translation	TextTranslation

Decision

Service	Kind
Anomaly Detector	AnomalyDetector

Service	Kind
Content Moderator	ContentModerator
Personalizer	Personalizer

Pricing tiers and billing

Pricing tiers (and the amount you get billed) are based on the number of transactions you send using your authentication information. Each pricing tier specifies the:

- maximum number of allowed transactions per second (TPS).
- service features enabled within the pricing tier.
- cost for a predefined number of transactions. Going above this number will cause an extra charge as specified in the [pricing details](#) for your service.

Note

Many of the Cognitive Services have a free tier you can use to try the service. To use the free tier, use `F0` as the SKU for your resource.

Deploy the Bicep file

1. Save the Bicep file as `main.bicep` to your local computer.
2. Deploy the Bicep file using either Azure CLI or Azure PowerShell.

CLI

Azure CLI

```
az group create --name exampleRG --location eastus
az deployment group create --resource-group exampleRG --template-
file main.bicep
```

When the deployment finishes, you should see a message indicating the deployment succeeded.

Review deployed resources

Use the Azure portal, Azure CLI, or Azure PowerShell to list the deployed resources in the resource group.

CLI

Azure CLI

```
az resource list --resource-group exampleRG
```

Clean up resources

When no longer needed, use the Azure portal, Azure CLI, or Azure PowerShell to delete the resource group and its resources.

CLI

Azure CLI

```
az group delete --name exampleRG
```

If you need to recover a deleted resource, see [Recover deleted Cognitive Services resources](#).

See also

- See [Authenticate requests to Azure Cognitive Services](#) on how to securely work with Cognitive Services.
- See [What are Azure Cognitive Services?](#) to get a list of different categories within Cognitive Services.
- See [Natural language support](#) to see the list of natural languages that Cognitive Services supports.
- See [Use Cognitive Services as containers](#) to understand how to use Cognitive Services on-prem.
- See [Plan and manage costs for Cognitive Services](#) to estimate cost of using Cognitive Services.

Additional resources

Documentation

[Create an Azure Cognitive Services resource using ARM templates - Azure Cognitive Services](#)

Create a Azure Cognitive Service resource with ARM template.

[Microsoft.CognitiveServices/accounts - Bicep, ARM template & Terraform AzAPI reference](#)

Azure Microsoft.CognitiveServices/accounts syntax and properties to use in Azure Resource Manager templates for deploying the resource. API version latest

[az cognitiveservices account](#)

[az cognitiveservices](#)

[Recover deleted Cognitive Services resource - Azure Cognitive Services](#)

This article provides instructions on how to recover an already-deleted Cognitive Services resource.

[Quickstart: Deploy using templates - Azure Cognitive Search](#)

You can quickly deploy an Azure Cognitive Search service instance using the Azure Resource Manager template.

[Create a Cognitive Services resource using the Azure CLI - Azure Cognitive Services](#)

Get started with Azure Cognitive Services by using Azure CLI commands to create and subscribe to a resource.

[Quickstart: Deploy using Bicep - Azure Cognitive Search](#)

You can quickly deploy an Azure Cognitive Search service instance using Bicep.

[Show 5 more](#)

Training

Learning paths and modules

[Provision and manage Azure Cognitive Services - Training](#)

Provision and manage Azure Cognitive Services

Learning certificate

[Microsoft Certified: Azure AI Engineer Associate - Certifications](#)

Azure AI engineers build, manage, and deploy AI solutions that leverage Azure Cognitive Services and Azure Applied AI services.

Quickstart: Create a Cognitive Services resource using an ARM template

Article • 09/11/2022 • 5 minutes to read

This quickstart describes how to use an Azure Resource Manager template (ARM template) to create Cognitive Services.

Azure Cognitive Services are cloud-based artificial intelligence (AI) services that help developers build cognitive intelligence into applications without having direct AI or data science skills or knowledge. They are available through REST APIs and client library SDKs in popular development languages. Azure Cognitive Services enables developers to easily add cognitive features into their applications with cognitive solutions that can see, hear, speak, and analyze.

Create a resource using an Azure Resource Manager template (ARM template). This multi-service resource lets you:

- Access multiple Azure Cognitive Services with a single key and endpoint.
- Consolidate billing from the services you use.
- You must create your first Face, Language service, or Computer Vision resources from the Azure portal to review and acknowledge the terms and conditions. You can do so here: [Face](#) , [Language service](#) , [Computer Vision](#). After that, you can create subsequent resources using any deployment tool (SDK, CLI, or ARM template, etc) under the same Azure subscription.

An [ARM template](#) is a JavaScript Object Notation (JSON) file that defines the infrastructure and configuration for your project. The template uses declarative syntax. In declarative syntax, you describe your intended deployment without writing the sequence of programming commands to create the deployment.

Prerequisites

- If you don't have an Azure subscription, [create one for free](#).

Review the template

The template used in this quickstart is from [Azure Quickstart Templates](#).

JSON

```
{
    "$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentTemplate.json#",
    "contentVersion": "1.0.0.0",
    "metadata": {
        "_generator": {
            "name": "bicep",
            "version": "0.8.9.13224",
            "templateHash": "13077289615008073304"
        }
    },
    "parameters": {
        "cognitiveServiceName": {
            "type": "string",
            "defaultValue": "[format('CognitiveService-{0}', uniqueString(resourceGroup().id))]",
            "metadata": {
                "description": "That name is the name of our application. It has to be unique. Type a name followed by your resource group name. (<name>-<resourceGroupName>)"
            }
        },
        "location": {
            "type": "string",
            "defaultValue": "[resourceGroup().location]",
            "metadata": {
                "description": "Location for all resources."
            }
        },
        "sku": {
            "type": "string",
            "defaultValue": "S0",
            "allowedValues": [
                "S0"
            ]
        }
    },
    "resources": [
        {
            "type": "Microsoft.CognitiveServices/accounts",
            "apiVersion": "2021-10-01",
            "name": "[parameters('cognitiveServiceName')]",
            "location": "[parameters('location')]",
            "sku": {
                "name": "[parameters('sku')]"
            },
            "kind": "CognitiveServices",
            "properties": {
                "apiProperties": {
                    "statisticsEnabled": false
                }
            }
        }
    ]
}
```

```
]  
}
```

One Azure resource is defined in the Bicep file: [Microsoft.CognitiveServices/accounts](#) specifies that it is a Cognitive Services resource. The `kind` field in the Bicep file defines the type of resource.

See the list of SKUs and pricing information below.

Multi-service

Service	Kind
Multiple services. For more information, see the pricing page.	<code>CognitiveServices</code>

Vision

Service	Kind
Computer Vision	<code>ComputerVision</code>
Custom Vision - Prediction	<code>CustomVision.Prediction</code>
Custom Vision - Training	<code>CustomVision.Training</code>
Face	<code>Face</code>
Form Recognizer	<code>FormRecognizer</code>

Speech

Service	Kind
Speech Services	<code>SpeechServices</code>

Language

Service	Kind
Luis	<code>Luis</code>
QnA Maker	<code>QnAMaker</code>

Service	Kind
Language service	TextAnalytics
Text Translation	TextTranslation

Decision

Service	Kind
Anomaly Detector	AnomalyDetector
Content Moderator	ContentModerator
Personalizer	Personalizer

Pricing tiers and billing

Pricing tiers (and the amount you get billed) are based on the number of transactions you send using your authentication information. Each pricing tier specifies the:

- maximum number of allowed transactions per second (TPS).
- service features enabled within the pricing tier.
- cost for a predefined number of transactions. Going above this number will cause an extra charge as specified in the [pricing details](#) for your service.

 **Note**

Many of the Cognitive Services have a free tier you can use to try the service. To use the free tier, use `F0` as the SKU for your resource.

Deploy the template

Azure portal	
1. Click the Deploy to Azure button.	

2. Enter the following values.

Value	Description
Subscription	Select an Azure subscription.
Resource group	Select Create new , enter a unique name for the resource group, and then click OK .
Region	Select a region. For example, East US
Cognitive Service Name	Replace with a unique name for your resource. You will need the name in the next section when you validate the deployment.
Location	Replace with the region used above.
Sku	The pricing tier for your resource.

The screenshot shows the Microsoft Azure (Preview) portal interface for creating a Cognitive Service Universal Key. The title bar says "Cognitive Service Universal Key". The main area displays a "Template" titled "101-cognitive-services-universalkey" which creates "1 resource". Below this, there's a "Deployment scope" section where users can select their subscription and resource group. The "Subscription" dropdown is set to "My subscription" and the "Resource group" dropdown is set to "(New) quickstartResourceGroup" with a "Create new" option. Further down, there are fields for "Region" (set to "East US"), "Cognitive Service Name" (set to "quickstart"), "Location" (set to "East US"), and "Sku" (set to "S0"). At the bottom, there are "Review + create" and "Next : Review + create >" buttons.

3. Select **Review + Create**, then **Create**. After the resource has successfully finished deploying, the **Go to resource** button will be highlighted.

Tip

If your subscription doesn't allow you to create a Cognitive Service resource, you may need to enable the privilege of that **Azure resource provider** using the **Azure portal**, **PowerShell command** or an **Azure CLI command**. If you are not the subscription owner, ask the *Subscription Owner* or someone with a role of *admin* to complete the registration for you or ask for the **/register/action** privileges to be granted to your account.

Review deployed resources

Portal

When your deployment finishes, you will be able to click the **Go to resource** button to see your new resource. You can also find the resource group by:

1. Selecting **Resource groups** from the left navigation menu.
2. Selecting the resource group name.

Clean up resources

If you want to clean up and remove a Cognitive Services subscription, you can delete the resource or resource group. Deleting the resource group also deletes any other resources contained in the group.

Azure portal

1. In the Azure portal, expand the menu on the left side to open the menu of services, and choose **Resource Groups** to display the list of your resource groups.
2. Locate the resource group containing the resource to be deleted
3. Right-click on the resource group listing. Select **Delete resource group**, and confirm.

If you need to recover a deleted resource, see [Recover deleted Cognitive Services resources](#).

See also

- See [Authenticate requests to Azure Cognitive Services](#) on how to securely work with Cognitive Services.
- See [What are Azure Cognitive Services?](#) to get a list of different categories within Cognitive Services.
- See [Natural language support](#) to see the list of natural languages that Cognitive Services supports.
- See [Use Cognitive Services as containers](#) to understand how to use Cognitive Services on-prem.
- See [Plan and manage costs for Cognitive Services](#) to estimate cost of using Cognitive Services.

Plan and manage costs for Azure Cognitive Services

Article • 09/20/2022 • 7 minutes to read

This article describes how you plan for and manage costs for Azure Cognitive Services. First, you use the Azure pricing calculator to help plan for Cognitive Services costs before you add any resources for the service to estimate costs. Next, as you add Azure resources, review the estimated costs. After you've started using Cognitive Services resources (for example Speech, Computer Vision, LUIS, Language service, Translator, etc.), use Cost Management features to set budgets and monitor costs. You can also review forecasted costs and identify spending trends to identify areas where you might want to act. Costs for Cognitive Services are only a portion of the monthly costs in your Azure bill. Although this article explains how to plan for and manage costs for Cognitive Services, you're billed for all Azure services and resources used in your Azure subscription, including the third-party services.

Prerequisites

Cost analysis in Cost Management supports most Azure account types, but not all of them. To view the full list of supported account types, see [Understand Cost Management data](#). To view cost data, you need at least read access for an Azure account. For information about assigning access to Azure Cost Management data, see [Assign access to data](#).

Estimate costs before using Cognitive Services

Use the [Azure pricing calculator](#) to estimate costs before you add Cognitive Services.

Cognitive Services

API: Computer Vision REGION: West US INSTANCE: S1

ⓘ Up to 10 transactions per second.

Features

Tag, Face, GetThumbnail Color, Image Type

20000	Transactions	= \$20.00
-------	--------------	-----------

OCR, Adult, Celebrity, and Landmark

10000	Transactions	= \$15.00
-------	--------------	-----------

Describe and Recognize Text

100000	Transactions	× \$2.50 Per 1,000 transactions	= \$250.00
--------	--------------	---------------------------------	------------

Upfront cost \$0.00
Monthly cost \$285.00

Support

SUPPORT: Developer = \$29.00

Programs and Offers

LICENSING PROGRAM: Microsoft Online Services Agreement ⓘ

SHOW DEV/TEST PRICING ⓘ

Estimated upfront cost \$0.00
Estimated monthly cost \$314.00

As you add new resources to your workspace, return to this calculator and add the same resource here to update your cost estimates.

For more information, see [Azure Cognitive Services pricing ↗](#).

Understand the full billing model for Cognitive Services

Cognitive Services runs on Azure infrastructure that [accrues costs ↗](#) when you deploy the new resource. It's important to understand that more infrastructure might accrue costs. You need to manage that cost when you make changes to deployed resources.

When you create or use Cognitive Services resources, you might get charged based on the services that you use. There are two billing models available for Cognitive Services: pay-as-you-go, and commitment tier.

Pay-as-you-go

With Pay-As-You-Go pricing, you are billed according to the Cognitive Services offering you use, based on its billing information.

Service	Instance(s)	Billing information
Vision		
Computer Vision ↗	Free, Standard (S1)	Billed by the number of transactions. Price per transaction varies per feature (Read, OCR, Spatial Analysis). For full details, see Pricing ↗ .
Custom Vision ↗	Free, Standard	<ul style="list-style-type: none"> Predictions are billed by the number of transactions. Training is billed by compute hour(s). Image storage is billed by number of images (up to 6 MB per image).
Face ↗	Free, Standard	Billed by the number of transactions.
Speech		
Speech service ↗	Free, Standard	Billing varies by feature (speech-to-text, text-to-speech, speech translation, speaker recognition). Primarily, billing is by transaction count or character count. For full details, see Pricing ↗ .
Language		
Language Understanding (LUIS) ↗	Free Authoring, Free Prediction, Standard	Billed by number of transactions. Price per transaction varies by feature (speech requests, text requests). For full details, see Pricing ↗ .
QnA Maker ↗	Free, Standard	Subscription fee billed monthly. For full details, see Pricing ↗ .
Language service ↗	Free, Standard	Billed by number of text records.
Translator ↗	Free, Pay-as-you-go (S1), Volume discount (S2, S3, S4, C2, C3, C4, D3)	<p>Pricing varies by meter and feature. For full details, see Pricing ↗.</p> <ul style="list-style-type: none"> Text translation is billed by number of characters translated. Document translation is billed by characters translated. Custom translation is billed by characters of source and target training data.
Decision		
Anomaly Detector ↗	Free, Standard	Billed by the number of transactions.

Service	Instance(s)	Billing information
Content Moderator ↗	Free, Standard	Billed by the number of transactions.
Personalizer ↗	Free, Standard (S0)	Billed by transactions per month. There are storage and transaction quotas. For full details, see Pricing ↗ .

Commitment tier

In addition to the pay-as-you-go model, Cognitive Services has commitment tiers, which let you commit to using several service features for a fixed fee, enabling you to have a predictable total cost based on the needs of your workload.

With commitment tier pricing, you are billed according to the plan you choose. See [Quickstart: purchase commitment tier pricing](#) for information on available services, how to sign up, and considerations when purchasing a plan.

Note

If you use the resource above the quota provided by the commitment plan, you will be charged for the additional usage as per the overage amount mentioned in the Azure portal when you purchase a commitment plan. For more information, see [Azure Cognitive Services pricing ↗](#).

Costs that typically accrue with Cognitive Services

Typically, after you deploy an Azure resource, costs are determined by your pricing tier and the API calls you make to your endpoint. If the service you're using has a commitment tier, going over the allotted calls in your tier may incur an overage charge.

Extra costs may accrue when using these services:

QnA Maker

When you create resources for QnA Maker, resources for other Azure services may also be created. They include:

- [Azure App Service \(for the runtime\) ↗](#)
- [Azure Cognitive Search \(for the data\) ↗](#)

Costs that might accrue after resource deletion

QnA Maker

After you delete QnA Maker resources, the following resources might continue to exist. They continue to accrue costs until you delete them.

- [Azure App Service \(for the runtime\) ↗](#)
- [Azure Cognitive Search \(for the data\) ↗](#)

Using Azure Prepayment credit with Cognitive Services

You can pay for Cognitive Services charges with your Azure Prepayment (previously called monetary commitment) credit. However, you can't use Azure Prepayment credit to pay for charges for third-party products and services including those from the Azure Marketplace.

Monitor costs

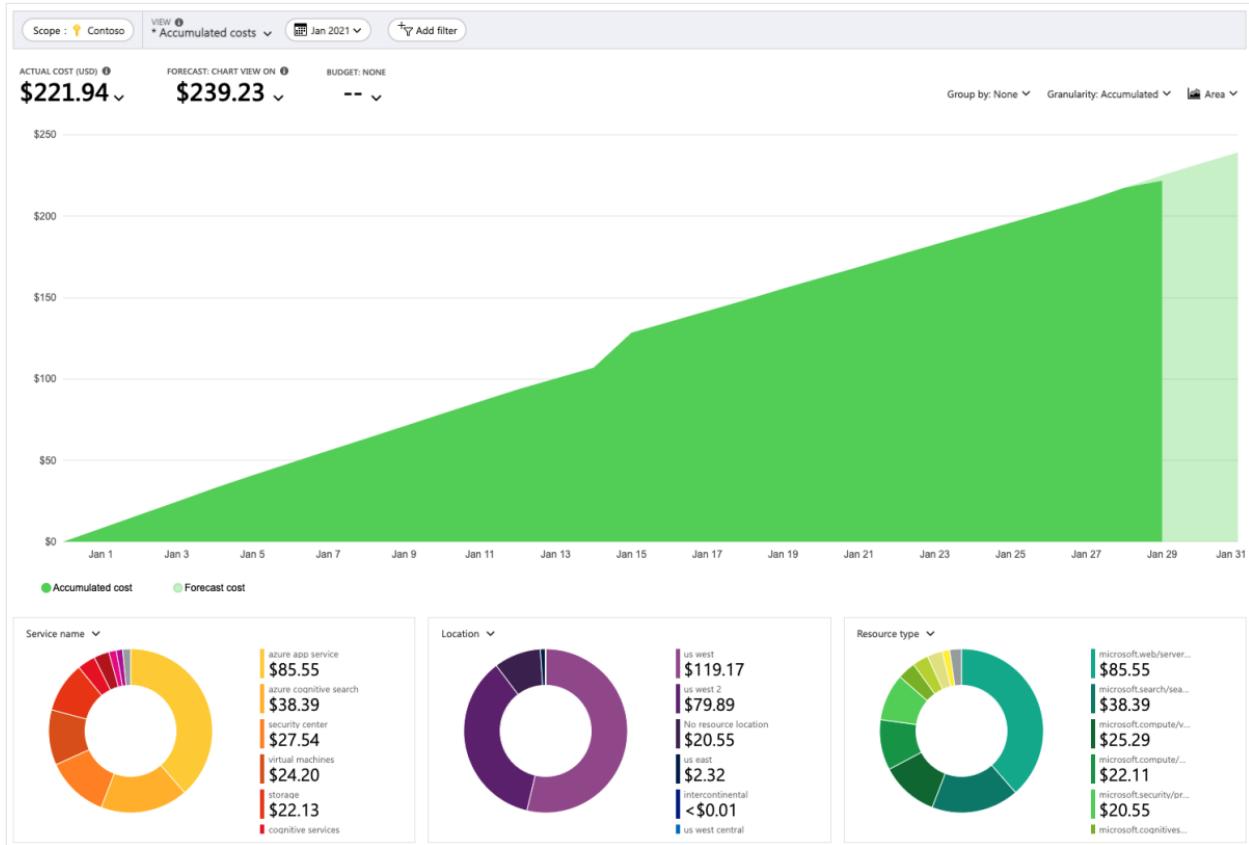
As you use Azure resources with Cognitive Services, you incur costs. Azure resource usage unit costs vary by time intervals (seconds, minutes, hours, and days) or by unit usage (bytes, megabytes, and so on). As soon as use of a Cognitive Service (or Cognitive Services) starts, costs are incurred and you can see the costs in [cost analysis](#).

When you use cost analysis, you view Cognitive Services costs in graphs and tables for different time intervals. Some examples are by day, current and prior month, and year. You also view costs against budgets and forecasted costs. Switching to longer views over time can help you identify spending trends. And you see where overspending might have occurred. If you've created budgets, you can also easily see where they're exceeded.

To view Cognitive Services costs in cost analysis:

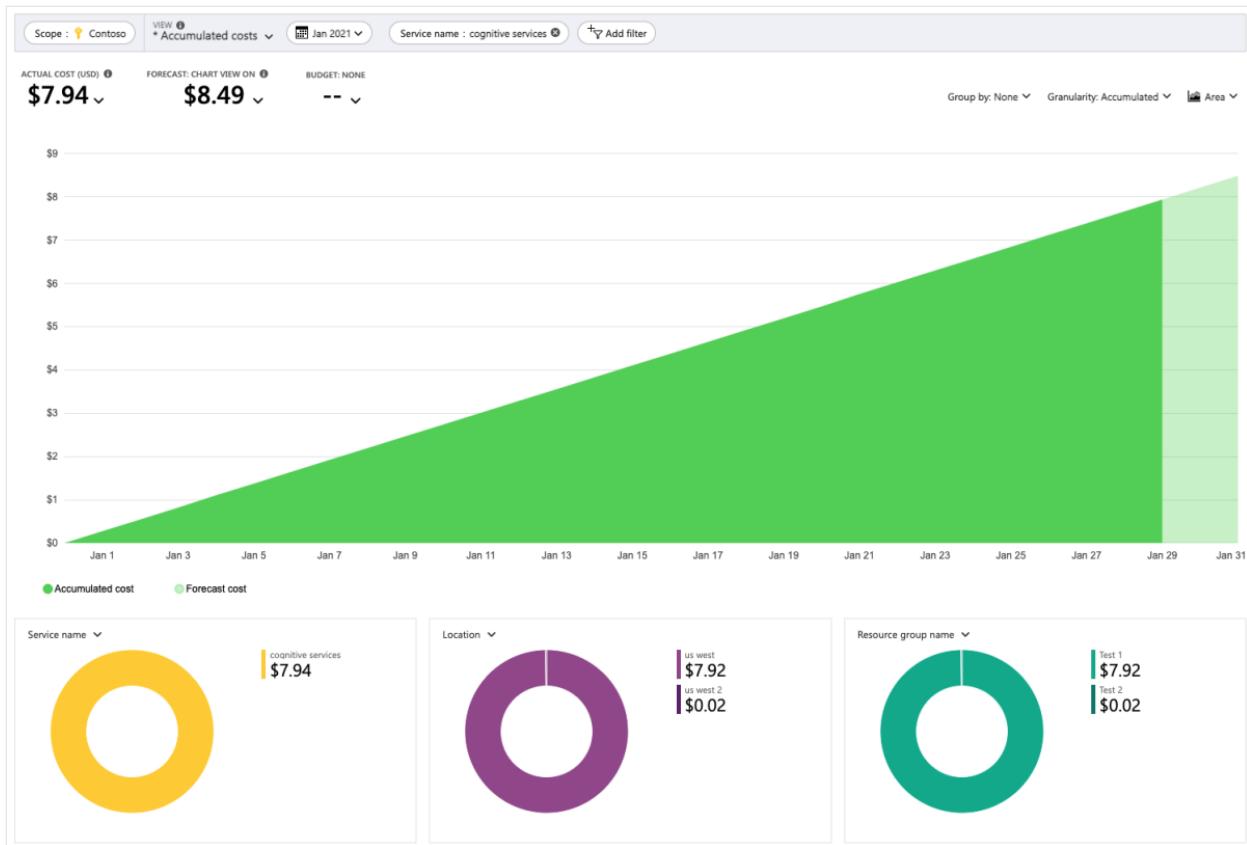
1. Sign in to the Azure portal.
2. Open the scope in the Azure portal and select **Cost analysis** in the menu. For example, go to **Subscriptions**, select a subscription from the list, and then select **Cost analysis** in the menu. Select **Scope** to switch to a different scope in cost analysis.
3. By default, cost for services are shown in the first donut chart. Select the area in the chart labeled Cognitive Services.

Actual monthly costs are shown when you initially open cost analysis. Here's an example showing all monthly usage costs.



- To narrow costs for a single service, like Cognitive Services, select **Add filter** and then select **Service name**. Then, select **Cognitive Services**.

Here's an example showing costs for just Cognitive Services.



In the preceding example, you see the current cost for the service. Costs by Azure regions (locations) and Cognitive Services costs by resource group are also shown. From here, you can explore costs on your own.

Create budgets

You can create [budgets](#) to manage costs and create [alerts](#) that automatically notify stakeholders of spending anomalies and overspending risks. Alerts are based on spending compared to budget and cost thresholds. Budgets and alerts are created for Azure subscriptions and resource groups, so they're useful as part of an overall cost monitoring strategy.

Budgets can be created with filters for specific resources or services in Azure if you want more granularity present in your monitoring. Filters help ensure that you don't accidentally create new resources that cost you more money. For more about the filter options when you create a budget, see [Group and filter options](#).

Export cost data

You can also [export your cost data](#) to a storage account. This is helpful when you or others need to do more data analysis for costs. For example, finance teams can analyze the data using Excel or Power BI. You can export your costs on a daily, weekly, or

monthly schedule and set a custom date range. Exporting cost data is the recommended way to retrieve cost datasets.

Next steps

- Learn [how to optimize your cloud investment with Azure Cost Management](#).
- Learn more about managing costs with [cost analysis](#).
- Learn about how to [prevent unexpected costs](#).
- Take the [Cost Management](#) guided learning course.

Cognitive Services autoscale feature

Article • 11/15/2022 • 3 minutes to read

This article provides guidance for how customers can access higher rate limits on their Cognitive Service resources.

Overview

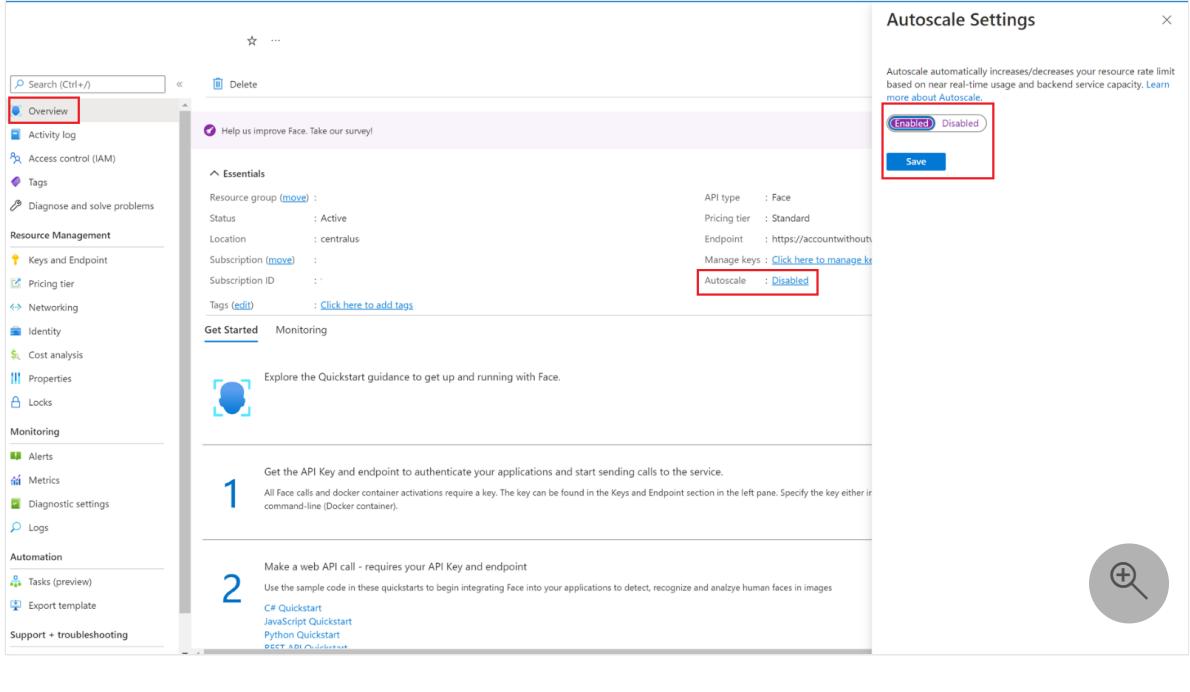
Each Cognitive Services resource has a pre-configured static call rate (transactions per second) which limits the number of concurrent calls that customers can make to the backend service in a given time frame. The autoscale feature will automatically increase/decrease a customer's resource's rate limits based on near-real-time resource usage metrics and backend service capacity metrics.

Get started with the autoscale feature

This feature is disabled by default for every new resource. Follow these instructions to enable it.

Azure portal

Go to your resource's page in the Azure portal, and select the **Overview** tab on the left pane. Under the **Essentials** section, find the **Autoscale** line and select the link to view the **Autoscale Settings** pane and enable the feature.



The screenshot shows the Azure portal interface for a Face resource. On the left, the navigation menu includes 'Overview' (which is selected and highlighted with a red box), 'Activity log', 'Access control (IAM)', 'Tags', 'Diagnose and solve problems', 'Resource Management', 'Keys and Endpoint', 'Pricing tier', 'Networking', 'Identity', 'Cost analysis', 'Properties', 'Locks', 'Monitoring', 'Alerts', 'Metrics', 'Diagnostic settings', 'Logs', 'Automation', 'Tasks (preview)', 'Export template', and 'Support + troubleshooting'. The main content area shows the 'Overview' tab selected. In the 'Essentials' section, there are details about the resource: Resource group (move), Status: Active, Location: centralus, Subscription (move), Subscription ID, Tags (edit), and API type: Face, Pricing tier: Standard, Endpoint: https://accountwithouthttps, Manage keys: Click here to manage key, and Autoscale: Disabled (which is also highlighted with a red box). Below this, there are sections for 'Get Started' (with a Quickstart guide for Face) and 'Automation' (with steps 1 and 2 for integrating Face into applications). A 'Save' button is located at the bottom right of the 'Autoscale Settings' pane.

Frequently asked questions

Does enabling the autoscale feature mean my resource will never be throttled again?

No, you may still get 429 errors for rate limit excess. If your application triggers a spike, and your resource reports a 429 response, autoscale will check the available capacity projection section to see whether the current capacity can accommodate a rate limit increase and respond within five minutes.

If the available capacity is enough for an increase, autoscale will gradually increase the rate limit cap of your resource. If you continue to call your resource at a high rate that results in more 429 throttling, your TPS rate will continue to increase over time. If this continues for one hour or more, you should reach the maximum rate (up to 1000 TPS) currently available at that time for that resource.

If the available capacity is not enough for an increase, the autoscale feature will wait five minutes and check again.

What if I need a higher default rate limit?

By default, Cognitive Service resources have a default rate limit of 10 TPS. If you need a higher default TPS, submit a ticket by following the [New Support Request](#) link on your resource's page in the Azure portal. Remember to include a business justification in the request.

Will this feature increase my Azure spend?

Cognitive Services pricing hasn't changed and can be accessed [here](#). We'll only bill for successful calls made to Cognitive Services APIs. However, increased call rate limits mean more transactions will be completed, and you may receive a higher bill.

Be aware of potential errors and their consequences. If a bug in your client application causes it to call the service hundreds of times per second, that would likely lead to a much higher bill, whereas the cost would be much more limited under a fixed rate limit. Errors of this kind are your responsibility, so we highly recommend that you perform development and client update tests against a resource with a fixed rate limit prior to using the autoscale feature.

Can I disable this feature if I'd rather limit the rate than have unpredictable spending?

Yes, you can disable the autoscale feature through Azure portal or CLI and return to your default call rate limit setting. If your resource was previously approved for a higher default TPS, it will go back to that rate. It can take up to five minutes for the changes to go into effect.

Which services support the autoscale feature?

Autoscale feature is available for the following services:

- [Cognitive Services multi-key](#)
- [Computer Vision](#)
- [Language](#) (only available for sentiment analysis, key phrase extraction, named entity recognition, and text analytics for health)
- [Anomaly Detector](#)
- [Content Moderator](#)
- [Custom Vision \(Prediction\)](#)
- [Immersive Reader](#)
- [LUIS](#)
- [Metrics Advisor](#)
- [Personalizer](#)
- [QnAMaker](#)
- [Form Recognizer](#)

Can I test this feature using a free subscription?

No, the autoscale feature is not available to free tier subscriptions.

Next steps

- [Plan and Manage costs for Azure Cognitive Services.](#)
- [Optimize your cloud investment with Azure Cost Management.](#)
- Learn about how to [prevent unexpected costs](#).
- Take the [Cost Management](#) guided learning course.

Purchase commitment tier pricing

Article • 12/01/2022 • 3 minutes to read

Cognitive Services offers commitment tier pricing, each offering a discounted rate compared to the pay-as-you-go pricing model. With commitment tier pricing, you can commit to using the following Cognitive Services features for a fixed fee, enabling you to have a predictable total cost based on the needs of your workload:

- Speech to Text (Standard)
- Text to Speech (Neural)
- Text Translation (Standard)
- Language Understanding standard (Text Requests)
- Azure Cognitive Service for Language
 - Sentiment Analysis
 - Key Phrase Extraction
 - Language Detection
- Computer Vision - OCR

Commitment tier pricing is also available for the following Applied AI service:

- Form Recognizer – Custom/Invoice

For more information, see [Azure Cognitive Services pricing ↗](#).

Create a new resource

Note

To purchase and use a commitment plan, your resource must have the Standard pricing tier. You cannot purchase a commitment plan (or see the option) for a resource that is on the free tier.

1. Sign into the [Azure portal ↗](#) and select **Create a new resource** for one of the applicable Cognitive Services or Applied AI services listed above.
2. Enter the applicable information to create your resource. Be sure to select the standard pricing tier.

The screenshot shows the Microsoft Azure portal's 'Create Form Recognizer' wizard. On the left, a sidebar lists various services under 'FAVORITES'. The main form is titled 'Create Form Recognizer' and includes tabs for 'Basics', 'Network', 'Identity', 'Tags', and 'Review + create'. In the 'Basics' tab, fields are filled with 'Subscription' (my-subscription), 'Resource group' (my-test-group), 'Region' (West Europe), 'Name' (my-resource), and 'Pricing tier' (Standard S0). A note indicates that the free tier (F0) is already in use. Navigation buttons at the bottom include 'Review + create', '< Previous', 'Next : Network >', and a search icon.

- Once your resource is created, you will be able to change your pricing from pay-as-you-go, to a commitment plan.

Purchase a commitment plan by updating your Azure resource

- Sign in to the [Azure portal](#) with your Azure subscription.
- In your Azure resource for one of the applicable features listed above, select **Commitment tier pricing**.

Note

You will only see the option to purchase a commitment plan if the resource is using the standard pricing tier.

- Select **Change** to view the available commitments for hosted API and container usage. Choose a commitment plan for one or more of the following offerings:
 - Web:** web-based APIs, where you send data to Azure for processing.

- **Connected container:** Docker containers that enable you to [deploy Cognitive services on premises](#), and maintain an internet connection for billing and metering.

my-resource | Commitment tier pricing

Search (Ctrl+/)

- Overview
- Activity log
- Access control (IAM)
- Tags
- Diagnose and solve problems

Resource Management

- Keys and Endpoint
- Commitment tier pricing**
- Pricing tier
- Networking
- Identity
- Cost analysis
- Properties
- Locks

Monitoring

- Alerts
- Metrics

Choose a monthly commitment plan based on the needs of your workload. You can customize the 'Standard' pricing tier by selecting one of several commitment tier options. Each commitment tier offers a discounted rate compared to the pay-as-you-go pricing model. Commitment plans are charged monthly, except the first month upon purchase which is pro-rated(cost and quota) based on the number of days remaining in that month. For the subsequent months, the charge is incurred on the first day of the month. Any overage will be charged as and when incurred after you have consumed the allowed quota for your chosen tier. [Learn more about commitment tier pricing](#)

Commitment plans for: Web

Custom, Invoice ([Change](#))

Tier : No commitment (Pay as you go)

Commitment plans for: Connected container

Custom, Invoice ([Change](#))

Tier : No commitment (Pay as you go)

4. In the window that appears, select both a **Tier** and **Auto-renewal** option.

- **Commitment tier** - The commitment tier for the feature. The commitment tier will be enabled immediately when you click **Purchase** and you will be charged the commitment amount on a pro-rated basis.
- **Auto-renewal** - Choose how you want to renew, change, or cancel the current commitment plan starting with the next billing cycle. If you decide to auto-renew, the **Auto-renewal date** is the date (in your local timezone) when you will be charged for the next billing cycle. This date coincides with the start of the calendar month.

✖ Caution

Once you click **Purchase** you will be charged for the tier you select. Once purchased, the commitment plan is non-refundable.

Commitment plans are charged monthly, except the first month upon purchase which is pro-rated (cost and quota) based on the number of days remaining in that month. For the subsequent months, the charge is incurred on the first day of the month.



Purchase commitment plan

X

Purchase a commitment plan here. Commitment plans are charged monthly, except the first month upon purchase which is pro-rated (cost and quota) based on the number of days remaining in that month. For the subsequent months, the charge is incurred on the first day of the month. You can also configure or cancel your commitment settings for the next month. If you want to cancel your commitment plan after the current month, choose "Do not auto-renew" in the 'Manage your plan' dropdown below. [Learn more about commitment plan management.](#)

Hosting model ⓘ

Web

Capabilities ⓘ

Custom, Invoice

Commitment tier * ⓘ

Manage your plan ⓘ

Auto-renewal date ⓘ

2021-11-30

Purchase

Cancel



Overage pricing

If you use the resource above the quota provided, you will be charged for the additional usage as per the overage amount mentioned in the commitment tier.

Purchase a different commitment plan

The commitment plans have a calendar month commitment period. You can purchase a commitment plan at any time from the default pay-as-you-go pricing model. When you purchase a plan, you will be charged a pro-rated price for the remaining month. During the commitment period, you cannot change the commitment plan for the current month. However, you can choose a different commitment plan for the next calendar month. The billing for the next month would happen on the first day of the next month.

If you need a larger commitment plan than any of the ones offered, contact

csgate@microsoft.com.

End a commitment plan

If you decide that you don't want to continue purchasing a commitment plan, you can set your resource's auto-renewal to **Do not auto-renew**. Your commitment plan will expire on the displayed commitment end date. After this date, you won't be charged for the commitment plan. You will be able to continue using the Azure resource to make API calls, charged at pay-as-you-go pricing. You have until midnight (UTC) on the last day of each month to end a commitment plan, and not be charged for the following month.

See also

- [Azure Cognitive Services pricing ↗](#).

Enable diagnostic logging for Azure Cognitive Services

Article • 02/11/2022 • 4 minutes to read

This guide provides step-by-step instructions to enable diagnostic logging for an Azure Cognitive Service. These logs provide rich, frequent data about the operation of a resource that are used for issue identification and debugging. Before you continue, you must have an Azure account with a subscription to at least one Cognitive Service, such as [Speech Services](#), or [LUIS](#).

Prerequisites

To enable diagnostic logging, you'll need somewhere to store your log data. This tutorial uses Azure Storage and Log Analytics.

- [Azure storage](#) - Retains diagnostic logs for policy audit, static analysis, or backup. The storage account does not have to be in the same subscription as the resource emitting logs as long as the user who configures the setting has appropriate Azure RBAC access to both subscriptions.
- [Log Analytics](#) - A flexible log search and analytics tool that allows for analysis of raw logs generated by an Azure resource.

ⓘ Note

- Additional configuration options are available. To learn more, see [Collect and consume log data from your Azure resources](#).
- "Trace" in diagnostic logging is only available for [Custom question answering](#).

Enable diagnostic log collection

Let's start by enabling diagnostic logging using the Azure portal.

ⓘ Note

To enable this feature using PowerShell or the Azure CLI, use the instructions provided in [Collect and consume log data from your Azure resources](#).

1. Navigate to the Azure portal. Then locate and select a Cognitive Services resource. For example, your subscription to Speech Services.
2. Next, from the left-hand navigation menu, locate **Monitoring** and select **Diagnostic settings**. This screen contains all previously created diagnostic settings for this resource.
3. If there is a previously created resource that you'd like to use, you can select it now. Otherwise, select **+ Add diagnostic setting**.
4. Enter a name for the setting. Then select **Archive to a storage account** and **Send to log Analytics**.
5. When prompted to configure, select the storage account and OMS workspace that you'd like to use to store your diagnostic logs. **Note:** If you don't have a storage account or OMS workspace, follow the prompts to create one.
6. Select **Audit**, **RequestResponse**, and **AllMetrics**. Then set the retention period for your diagnostic log data. If a retention policy is set to zero, events for that log category are stored indefinitely.
7. Click **Save**.

It can take up to two hours before logging data is available to query and analyze. So don't worry if you don't see anything right away.

View and export diagnostic data from Azure Storage

Azure Storage is a robust object storage solution that is optimized for storing large amounts of unstructured data. In this section, you'll learn to query your storage account for total transactions over a 30-day timeframe and export the data to excel.

1. From the Azure portal, locate the Azure Storage resource that you created in the last section.
2. From the left-hand navigation menu, locate **Monitoring** and select **Metrics**.
3. Use the available drop-downs to configure your query. For this example, let's set the time range to **Last 30 days** and the metric to **Transaction**.
4. When the query is complete, you'll see a visualization of transaction over the last 30 days. To export this data, use the **Export to Excel** button located at the top of the page.

Learn more about what you can do with diagnostic data in [Azure Storage](#).

View logs in Log Analytics

Follow these instructions to explore log analytics data for your resource.

1. From the Azure portal, locate and select **Log Analytics** from the left-hand navigation menu.
2. Locate and select the resource you created when enabling diagnostics.
3. Under **General**, locate and select **Logs**. From this page, you can run queries against your logs.

Sample queries

Here are a few basic Kusto queries you can use to explore your log data.

Run this query for all diagnostic logs from Azure Cognitive Services for a specified time period:

```
Kusto  
  
AzureDiagnostics  
| where ResourceProvider == "MICROSOFT.COGNITIVESERVICES"
```

Run this query to see the 10 most recent logs:

```
Kusto  
  
AzureDiagnostics  
| where ResourceProvider == "MICROSOFT.COGNITIVESERVICES"  
| take 10
```

Run this query to group operations by **Resource**:

```
Kusto  
  
AzureDiagnostics  
| where ResourceProvider == "MICROSOFT.COGNITIVESERVICES" |  
summarize count() by Resource
```

Run this query to find the average time it takes to perform an operation:

```
Kusto  
  
AzureDiagnostics  
| where ResourceProvider == "MICROSOFT.COGNITIVESERVICES"  
| summarize avg(DurationMs)  
by OperationName
```

Run this query to view the volume of operations over time split by OperationName with counts binned for every 10s.

Kusto

```
AzureDiagnostics
| where ResourceProvider == "MICROSOFT.COGNITIVESERVICES"
| summarize count()
by bin(TimeGenerated, 10s), OperationName
| render areachart kind=unstacked
```

Next steps

- To understand how to enable logging, and also the metrics and log categories that are supported by the various Azure services, read both the [Overview of metrics](#) in Microsoft Azure and [Overview of Azure Diagnostic Logs](#) articles.
- Read these articles to learn about event hubs:
 - [What is Azure Event Hubs?](#)
 - [Get started with Event Hubs](#)
- Read [Understand log searches in Azure Monitor logs](#).

Recover deleted Cognitive Services resources

Article • 10/19/2022 • 3 minutes to read

This article provides instructions on how to recover a Cognitive Services resource that is already deleted. The article also provides instructions on how to purge a deleted resource.

ⓘ Note

The instructions in this article are applicable to both a multi-service resource and a single-service resource. A multi-service resource enables access to multiple cognitive services using a single key and endpoint. On the other hand, a single-service resource enables access to just that specific cognitive service for which the resource was created.

Prerequisites

- The resource to be recovered must have been deleted within the past 48 hours.
- The resource to be recovered must not have been purged already. A purged resource cannot be recovered.
- Before you attempt to recover a deleted resource, make sure that the resource group for that account exists. If the resource group was deleted, you must recreate it. Recovering a resource group is not possible. For more information, see [Manage resource groups](#).
- If the deleted resource used customer-managed keys with Azure Key Vault and the key vault has also been deleted, then you must restore the key vault before you restore the Cognitive Services resource. For more information, see [Azure Key Vault recovery management](#).
- If the deleted resource used a customer-managed storage and storage account has also been deleted, you must restore the storage account before you restore the Cognitive Services resource. For instructions, see [Recover a deleted storage account](#).

Your subscription must have

`Microsoft.CognitiveServices/locations/resourceGroups/deletedAccounts/delete`

permissions to purge resources, such as [Cognitive Services Contributor](#) or [Contributor](#).

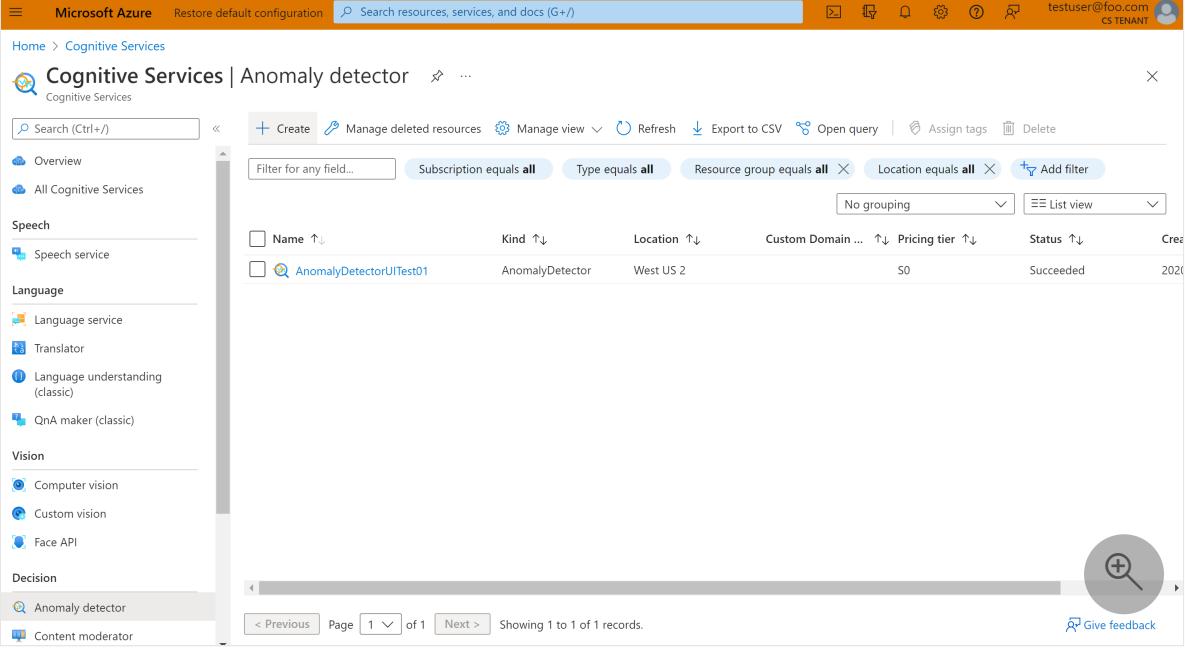
Recover a deleted resource

To recover a deleted cognitive service resource, use the following commands. Where applicable, replace:

- `{subscriptionID}` with your Azure subscription ID
- `{resourceGroup}` with your resource group
- `{resourceName}` with your resource name
- `{location}` with the location of your resource

Azure portal

If you need to recover a deleted resource, navigate to the hub of the cognitive services API type and select "Manage deleted resources" from the menu. For example, if you would like to recover an "Anomaly detector" resource, search for "Anomaly detector" in the search bar and select the service to get to the "Anomaly detector" hub which lists deleted resources.



The screenshot shows the Microsoft Azure portal interface. The top navigation bar includes 'Microsoft Azure', 'Restore default configuration', a search bar ('Search resources, services, and docs (G+)'), and user information ('testuser@foo.com CS TENANT'). Below the navigation is a breadcrumb trail: 'Home > Cognitive Services'. The main title is 'Cognitive Services | Anomaly detector'. On the left, there's a sidebar with categories: Speech (Speech service), Language (Language service, Translator, Language understanding (classic), QnA maker (classic)), Vision (Computer vision, Custom vision, Face API), and Decision (Anomaly detector, Content moderator). The 'Anomaly detector' item is highlighted. The main content area has a header with buttons for '+ Create', 'Manage deleted resources', 'Manage view', 'Refresh', 'Export to CSV', 'Open query', 'Assign tags', and 'Delete'. Below this are filter options: 'Subscription equals all', 'Type equals all', 'Resource group equals all', 'Location equals all', and a 'No grouping' dropdown. A table lists a single deleted resource: 'AnomalyDetectorUITest01' (Kind: AnomalyDetector, Location: West US 2, Status: Succeeded). At the bottom, there are buttons for '< Previous', 'Page 1 of 1', 'Next >', and 'Showing 1 to 1 of 1 records.' A 'Give feedback' button is also present.

Select the subscription in the dropdown list to locate the deleted resource you would like to recover. Select one or more of the deleted resources and click Recover.

The screenshot shows the Microsoft Azure Cognitive Services blade for the 'Anomaly detector' service. On the left, there's a sidebar with navigation links for Speech, Language, Vision, and Decision services. The main area displays a table of deleted resources. The table has columns for Name, Kind, Location, Deletion date, and Scheduled purge date. A search bar and filter buttons are at the top of the table area.

Name	Kind	Location	Deletion date	Scheduled purge date
XYaidjREAnomal	AnomalyDetector	West	8/20/2022, 10:41:53 AM	8/22/2022, 10:41:53 AM
IZhmlrvKAnomal			8/20/2022, 11:01:22 AM	8/22/2022, 11:01:22 AM
IslxtgTmAnomal			8/20/2022, 2:48:12 PM	8/22/2022, 2:48:12 PM
ZLRdDNxAnomal			8/20/2022, 10:50:36 PM	8/22/2022, 10:50:36 PM
dsfaBLnqAnomal			8/21/2022, 5:05:36 AM	8/23/2022, 5:05:36 AM
QalbmPsQAnomal			8/21/2022, 5:19:40 AM	8/23/2022, 5:19:40 AM
EEBstpjOAnomal			8/21/2022, 10:44:50 AM	8/23/2022, 10:44:50 AM
IpZrbIpMAnomal			8/21/2022, 11:03:35 AM	8/23/2022, 11:03:35 AM
ekVRLZQhAnomal			8/21/2022, 2:46:33 PM	8/23/2022, 2:46:33 PM
McfsUpTKAnomal			8/21/2022, 10:50:09 PM	8/23/2022, 10:50:09 PM
vYrazUNZAnomal			8/22/2022, 4:51:57 AM	8/24/2022, 4:51:57 AM
yJlhGfsAnomal			8/22/2022, 5:00:13 AM	8/24/2022, 5:00:13 AM

⚠ Note

It can take a couple of minutes for your deleted resource(s) to recover and show up in the list of the resources. Click on the **Refresh** button in the menu to update the list of resources.

Purge a deleted resource

Once you delete a resource, you won't be able to create another one with the same name for 48 hours. To create a resource with the same name, you will need to purge the deleted resource.

To purge a deleted cognitive service resource, use the following commands. Where applicable, replace:

- `{subscriptionID}` with your Azure subscription ID
- `{resourceGroup}` with your resource group
- `{resourceName}` with your resource name
- `{location}` with the location of your resource

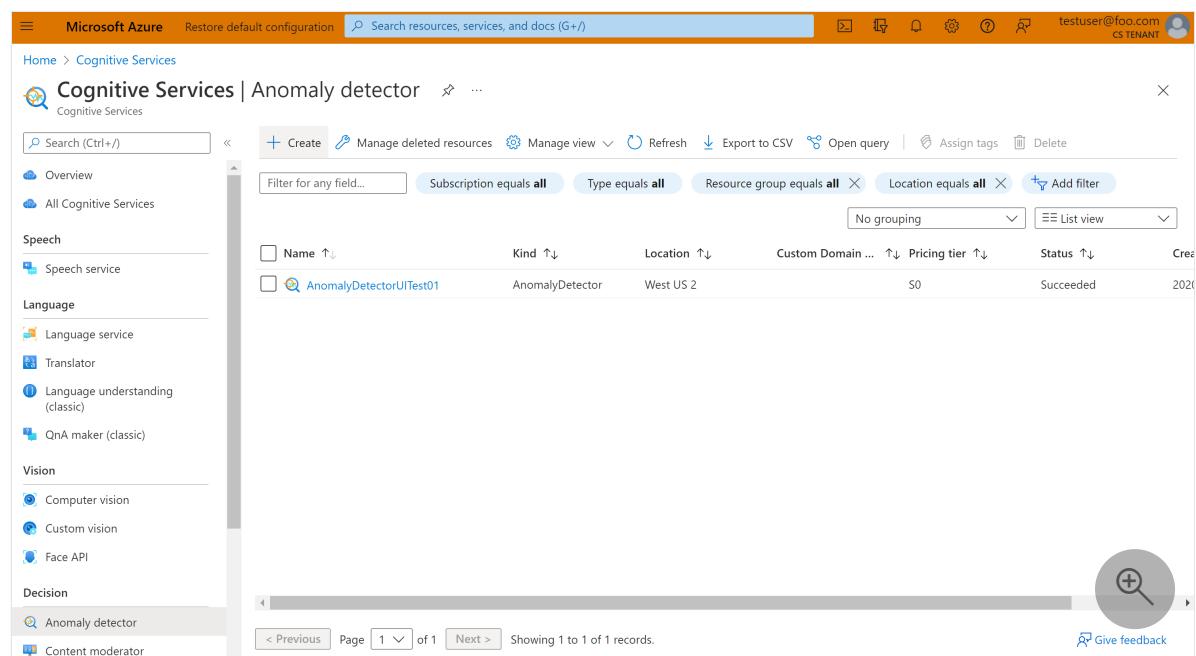
⚠ Note

Once a resource is purged, it is permanently deleted and cannot be restored. You will lose all data and keys associated with the resource.

If you need to purge a deleted resource, the steps are similar to recovering a deleted resource.

Navigate to the hub of the cognitive services API type of your deleted resource. For example, if you would like to purge an "Anomaly detector" resource, search for "Anomaly detector" in the search bar. Select the service to get to the "Anomaly detector" hub which lists deleted resources.

Select **Manage deleted resources** from the menu.



The screenshot shows the Microsoft Azure portal interface. The top navigation bar includes the Microsoft Azure logo, a search bar, and user information (testuser@foo.com, CS TENANT). Below the navigation bar, the breadcrumb path shows 'Home > Cognitive Services'. The main title is 'Cognitive Services | Anomaly detector'. On the left, there is a sidebar with categories: Speech (Speech service), Language (Language service, Translator, Language understanding (classic), QnA maker (classic)), Vision (Computer vision, Custom vision, Face API), and Decision (Anomaly detector, Content moderator). The 'Anomaly detector' item is highlighted. The main content area displays a table of deleted resources. The table has columns: Name, Kind, Location, Custom Domain, Pricing tier, Status, and Created. One row is visible, showing 'AnomalyDetectorUITest01' as the Name, 'AnomalyDetector' as the Kind, 'West US 2' as the Location, '\$0' as the Pricing tier, 'Succeeded' as the Status, and '2020' as the Created date. There are filters at the top of the table: 'Subscription equals all', 'Type equals all', 'Resource group equals all', 'Location equals all', and an 'Add filter' button. Below the table, there are buttons for '< Previous', 'Page 1 of 1', and 'Next >', and a message 'Showing 1 to 1 of 1 records.'

Select the subscription in the dropdown list to locate the deleted resource you would like to purge. Select one or more deleted resources and click **Purge**. Purging will permanently delete a Cognitive Services resource.

The screenshot shows the Microsoft Azure portal interface. On the left, the 'Cognitive Services' blade is open, displaying a list of services under categories like Speech, Language, Vision, and Decision. In the center, the 'Anomaly detector' resource is selected. On the right, a modal window titled 'Manage deleted resources' is open, showing a list of deleted resources from the 'CS-Platform-Test-01' subscription. The table includes columns for Name, Kind, Location, Deletion date, and Scheduled purge date. A search bar and buttons for Recover, Purge, Cancel, and Refresh are at the bottom of the modal.

Name	Kind	Location	Deletion date	Scheduled purge date
XYaidjREAnomal	AnomalyDetector	West	8/20/2022, 10:41:53 AM	8/22/2022, 10:41:53 AM
IZhmlrvKAnomal			8/20/2022, 11:01:22 AM	8/22/2022, 11:01:22 AM
IslxtgTmAnomal			8/20/2022, 2:48:12 PM	8/22/2022, 2:48:12 PM
ZLRdDNcAnomal			8/20/2022, 10:50:36 PM	8/22/2022, 10:50:36 PM
dsfaBLnqAnomal			8/21/2022, 5:05:36 AM	8/23/2022, 5:05:36 AM
QalbmPsQAnomal			8/21/2022, 5:19:40 AM	8/23/2022, 5:19:40 AM
EEBstpjOAnomal			8/21/2022, 10:44:50 AM	8/23/2022, 10:44:50 AM
IpZrbIpMAnomal			8/21/2022, 11:03:35 AM	8/23/2022, 11:03:35 AM
ekVRLZQhAnomal			8/21/2022, 2:46:33 PM	8/23/2022, 2:46:33 PM
McSUpTKAnomal			8/21/2022, 10:50:09 PM	8/23/2022, 10:50:09 PM
vYrazUNZAnomal			8/22/2022, 4:51:57 AM	8/24/2022, 4:51:57 AM
yJlhGfsAnomal			8/22/2022, 5:00:13 AM	8/24/2022, 5:00:13 AM

See also

- [Create a new resource using the Azure portal](#)
- [Create a new resource using the Azure CLI](#)
- [Create a new resource using the client library](#)
- [Create a new resource using an ARM template](#)

Cross-region disaster recovery

Article • 01/31/2023 • 2 minutes to read

One of the first decisions every Cognitive Service customer makes is which region to create their resource in. The choice of region provides customers with the benefits of regional compliance by enforcing data residency requirements. Cognitive Services is available in [multiple geographies](#) to ensure customers across the world are supported.

It's rare, but possible, to encounter a network issue that affects an entire region. If your solution needs to always be available, then you should design it to either fail-over into another region or split the workload between two or more regions. Both approaches require at least two resources in different regions and the ability to sync data between them.

Feature overview

The cross-region disaster recovery feature, also known as Single Resource Multiple Region (SRMR), enables this scenario by allowing you to distribute traffic or copy custom models to multiple resources which can exist in any supported geography.

SRMR business scenarios

- To ensure high availability of your application, each Cognitive Service supports a flexible recovery region option that allows you to choose from a list of supported regions.
- Customers with globally distributed end users can deploy resources in multiple regions to optimize the latency of their applications.

Routing profiles

Azure Traffic Manager routes requests among the selected regions. The SRMR currently supports [Priority](#), [Performance](#) and [Weighted](#) profiles and is currently available for the following services:

- [Computer Vision](#)
- [Immersive Reader](#)
- [Univariate Anomaly Detector](#)

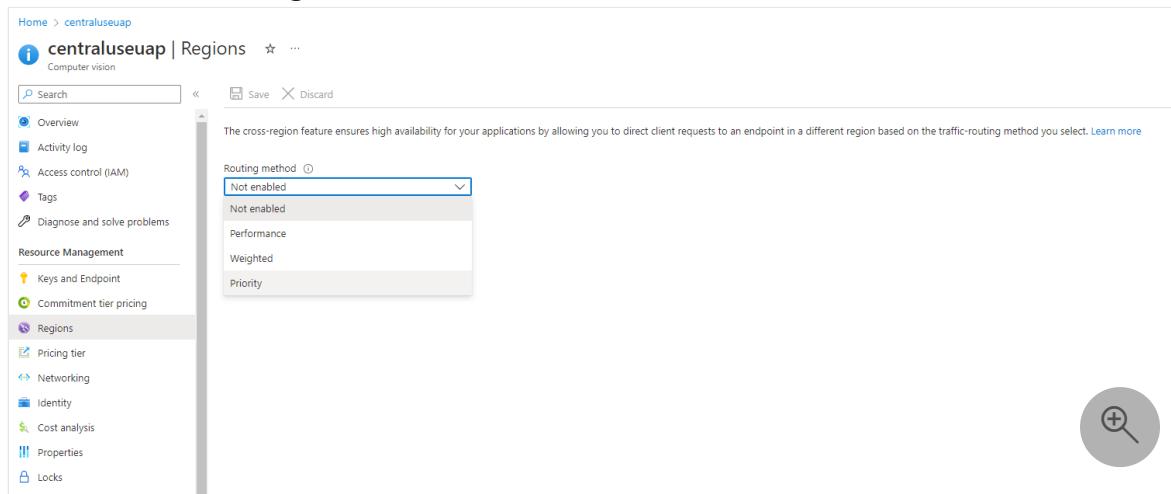
 Note

SRMR is not supported for multi-service resources or free tier resources.

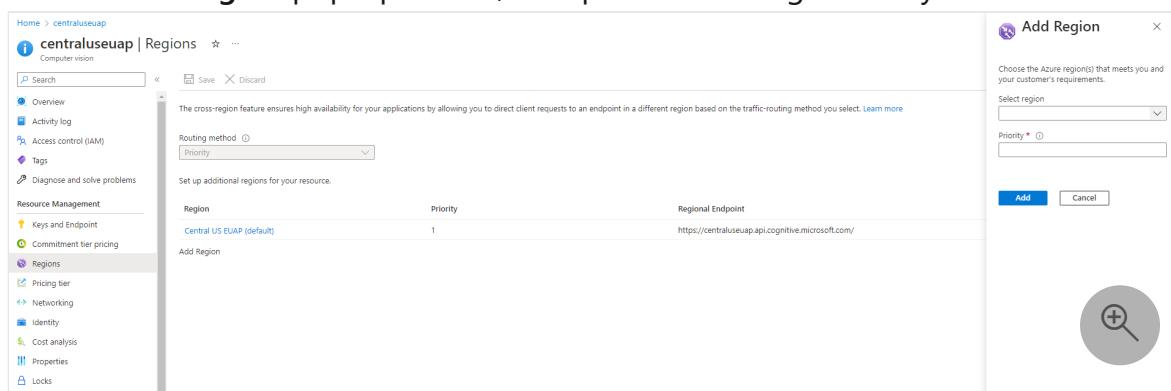
If you use Priority or Weighted traffic manager profiles, your configuration will behave according to the [Traffic Manager documentation](#).

Enable SRMR

1. Sign in to the [Azure portal](#).
2. Navigate to your resource's page.
3. Under the **Resource Management** section on the left pane, select the **Regions** tab and choose a routing method.



4. Select the **Add Region** link.
5. On the **Add Region** pop-up screen, set up additional regions for your resources.



6. Save your changes.

See also

- [Create a new resource using the Azure portal](#)
- [Create a new resource using the Azure CLI](#)
- [Create a new resource using the client library](#)
- [Create a new resource using an ARM template](#)

Additional resources

Azure Cognitive Services containers

Azure Cognitive Services offers several containers to use AI on premises. Containers give you the flexibility to bring Cognitive Services closer to your data for compliance, security or other operational reasons.

About Cognitive Services containers



[What are Cognitive Services containers?](#)

[Frequently Asked Questions](#)

Container recipes



[Container reuse recipe](#)

[Deploy on Azure Container Instances](#)

[Deploy on Kubernetes](#)

[Deploy with Docker compose](#)

Vision containers



[Read OCR](#)

[Spatial analysis](#)

Language containers



[Sentiment Analysis](#)

[Key phrase extraction](#)

[Text language detection](#)

[Text Analytics for health](#)

[LUIS container](#)

[Translator container](#)

Speech containers

 [HOW-TO GUIDE](#)

[Speech to text](#)

[Custom Speech to text](#)

[Neural text to speech](#)

[Speech language identification](#)

Decision container

 [HOW-TO GUIDE](#)

[Anomaly Detector](#)

Disconnected containers

 [HOW-TO GUIDE](#)

[Run containers disconnected from the internet](#)

[Frequently Asked Questions about disconnected containers](#)

What are Azure Cognitive Services containers?

Article • 02/02/2023 • 7 minutes to read

Azure Cognitive Services provides several [Docker containers](#) that let you use the same APIs that are available in Azure, on-premises. Using these containers gives you the flexibility to bring Cognitive Services closer to your data for compliance, security or other operational reasons. Container support is currently available for a subset of Azure Cognitive Services.

<https://www.youtube-nocookie.com/embed/hdfbn4Q8jbo>

Containerization is an approach to software distribution in which an application or service, including its dependencies & configuration, is packaged together as a container image. With little or no modification, a container image can be deployed on a container host. Containers are isolated from each other and the underlying operating system, with a smaller footprint than a virtual machine. Containers can be instantiated from container images for short-term tasks, and removed when no longer needed.

Features and benefits

- **Immutable infrastructure:** Enable DevOps teams to leverage a consistent and reliable set of known system parameters, while being able to adapt to change. Containers provide the flexibility to pivot within a predictable ecosystem and avoid configuration drift.
- **Control over data:** Choose where your data gets processed by Cognitive Services. This can be essential if you can't send data to the cloud but need access to Cognitive Services APIs. Support consistency in hybrid environments – across data, management, identity, and security.
- **Control over model updates:** Flexibility in versioning and updating of models deployed in their solutions.
- **Portable architecture:** Enables the creation of a portable application architecture that can be deployed on Azure, on-premises and the edge. Containers can be deployed directly to [Azure Kubernetes Service](#), [Azure Container Instances](#), or to a [Kubernetes](#) cluster deployed to [Azure Stack](#). For more information, see [Deploy Kubernetes to Azure Stack](#).
- **High throughput / low latency:** Provide customers the ability to scale for high throughput and low latency requirements by enabling Cognitive Services to run physically close to their application logic and data. Containers don't cap

transactions per second (TPS) and can be made to scale both up and out to handle demand if you provide the necessary hardware resources.

- **Scalability:** With the ever growing popularity of containerization and container orchestration software, such as Kubernetes; scalability is at the forefront of technological advancements. Building on a scalable cluster foundation, application development caters to high availability.

Containers in Azure Cognitive Services

Azure Cognitive Services containers provide the following set of Docker containers, each of which contains a subset of functionality from services in Azure Cognitive Services. You can find instructions and image locations in the tables below.

ⓘ Note

See [Install and run Form Recognizer containers for Applied AI Services Form Recognizer](#) container instructions and image locations.

Decision containers

Service	Container	Description	Availability
Anomaly detector	Anomaly Detector (image ↗)	The Anomaly Detector API enables you to monitor and detect abnormalities in your time series data with machine learning.	Generally available

Language containers

Service	Container	Description	Availability
LUIS	LUIS (image ↗)	Loads a trained or published Language Understanding model, also known as a LUIS app, into a docker container and provides access to the query predictions from the container's API endpoints. You can collect query logs from the container and upload these back to the LUIS portal to improve the app's prediction accuracy.	Generally available. This container can also run in disconnected environments .

Service	Container	Description	Availability
Language service	Key Phrase Extraction (image )	Extracts key phrases to identify the main points. For example, for the input text "The food was delicious and there were wonderful staff", the API returns the main talking points: "food" and "wonderful staff".	Generally available. This container can also run in disconnected environments .
Language service	Text Language Detection (image )	For up to 120 languages, detects which language the input text is written in and report a single language code for every document submitted on the request. The language code is paired with a score indicating the strength of the score.	Generally available. This container can also run in disconnected environments .
Language service	Sentiment Analysis (image )	Analyzes raw text for clues about positive or negative sentiment. This version of sentiment analysis returns sentiment labels (for example <i>positive</i> or <i>negative</i>) for each document and sentence within it.	Generally available. This container can also run in disconnected environments .
Language service	Text Analytics for health (image )	Extract and label medical information from unstructured clinical text.	Generally available
Translator	Translator (image )	Translate text in several languages and dialects.	Generally available. Gated - request access . This container can also run in disconnected environments .

Speech containers

 **Note**

To use Speech containers, you will need to complete an [online request form](#).

Service	Container	Description	Availability
Speech Service API	Speech-to-text (image 	Transcribes continuous real-time speech into text.	Generally available. This container can also run in disconnected environments.
Speech Service API	Custom Speech-to-text (image 	Transcribes continuous real-time speech into text using a custom model.	Generally available. This container can also run in disconnected environments.
Speech Service API	Neural Text-to-speech (image 	Converts text to natural-sounding speech using deep neural network technology, allowing for more natural synthesized speech.	Generally available. This container can also run in disconnected environments.
Speech Service API	Speech language detection (image 	Determines the language of spoken audio.	Gated preview

Vision containers

Service	Container	Description	Availability
Computer Vision	Read OCR (image 	The Read OCR container allows you to extract printed and handwritten text from images and documents with support for JPEG, PNG, BMP, PDF, and TIFF file formats. For more information, see the Read API documentation .	Generally Available. Gated - request access  . This container can also run in disconnected environments.
Spatial Analysis	Spatial analysis (image 	Analyzes real-time streaming video to understand spatial relationships between people, their movement, and interactions with objects in physical environments.	Preview

Additionally, some containers are supported in the Cognitive Services [multi-service resource](#) offering. You can create one single Cognitive Services All-In-One resource and use the same billing key across supported services for the following services:

- Computer Vision
- LUIS
- Language service

Prerequisites

You must satisfy the following prerequisites before using Azure Cognitive Services containers:

Docker Engine: You must have Docker Engine installed locally. Docker provides packages that configure the Docker environment on [macOS](#), [Linux](#), and [Windows](#). On Windows, Docker must be configured to support Linux containers. Docker containers can also be deployed directly to [Azure Kubernetes Service](#) or [Azure Container Instances](#).

Docker must be configured to allow the containers to connect with and send billing data to Azure.

Familiarity with Microsoft Container Registry and Docker: You should have a basic understanding of both Microsoft Container Registry and Docker concepts, like registries, repositories, containers, and container images, as well as knowledge of basic `docker` commands.

For a primer on Docker and container basics, see the [Docker overview](#).

Individual containers can have their own requirements, as well, including server and memory allocation requirements.

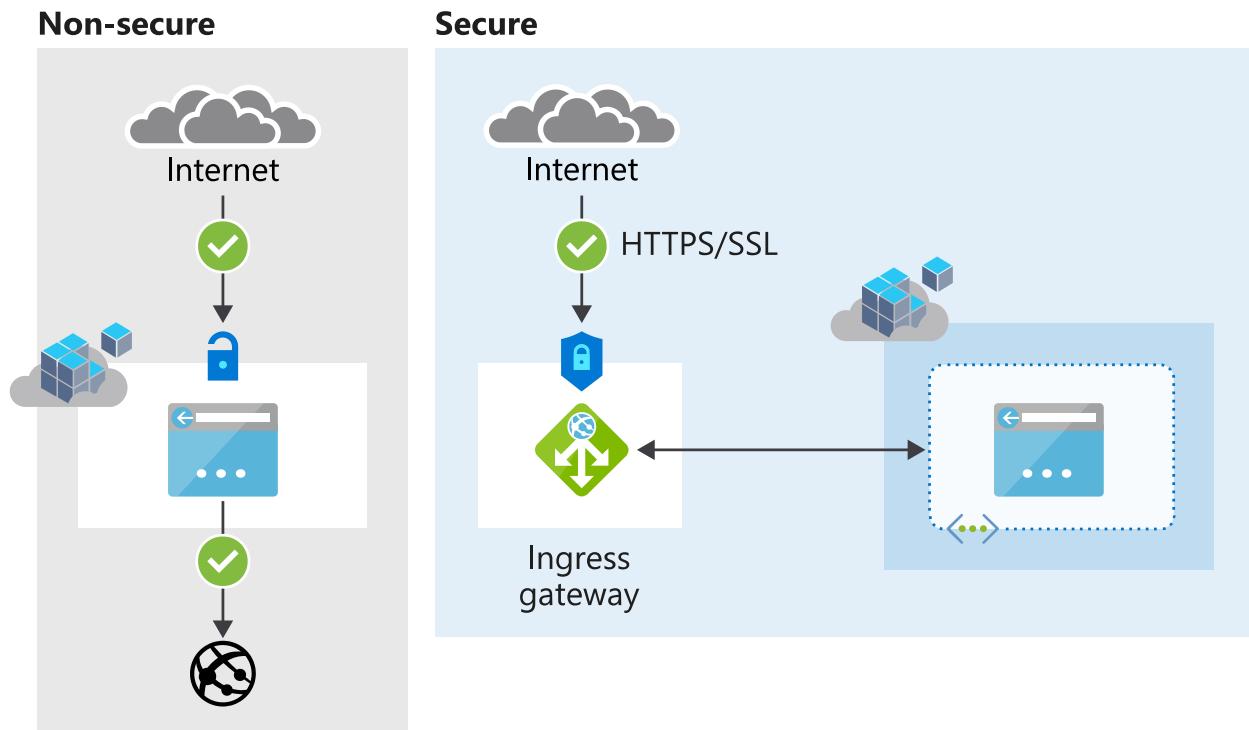
Azure Cognitive Services container security

Security should be a primary focus whenever you're developing applications. The importance of security is a metric for success. When you're architecting a software solution that includes Cognitive Services containers, it's vital to understand the limitations and capabilities available to you. For more information about network security, see [Configure Azure Cognitive Services virtual networks](#).

Important

By default there is *no security* on the Cognitive Services container API. The reason for this is that most often the container will run as part of a pod which is protected from the outside by a network bridge. However, it is possible to enable authentication which works identically to the authentication used when accessing the [cloud-based Cognitive Services](#).

The diagram below illustrates the default and **non-secure** approach:



As an alternative and *secure* approach, consumers of Cognitive Services containers could augment a container with a front-facing component, keeping the container endpoint private. Let's consider a scenario where we use [Istio](#) as an ingress gateway. Istio supports HTTPS/TLS and client-certificate authentication. In this scenario, the Istio frontend exposes the container access, presenting the client certificate that is approved beforehand with Istio.

[Nginx](#) is another popular choice in the same category. Both Istio and Nginx act as a service mesh and offer additional features including things like load-balancing, routing, and rate-control.

Container networking

The Cognitive Services containers are required to submit metering information for billing purposes. Failure to allow list various network channels that the Cognitive Services containers rely on will prevent the container from working.

Allow list Cognitive Services domains and ports

The host should allow list port 443 and the following domains:

- *.cognitive.microsoft.com
- *.cognitiveservices.azure.com

Disable deep packet inspection

[Deep packet inspection ↗](#) (DPI) is a type of data processing that inspects in detail the data being sent over a computer network, and usually takes action by blocking, re-routing, or logging it accordingly.

Disable DPI on the secure channels that the Cognitive Services containers create to Microsoft servers. Failure to do so will prevent the container from functioning correctly.

Developer samples

Developer samples are available at our [GitHub repository ↗](#).

Next steps

Learn about [container recipes](#) you can use with the Cognitive Services.

Install and explore the functionality provided by containers in Azure Cognitive Services:

- Anomaly Detector containers
- Computer Vision containers
- Language Understanding (LUIS) containers
- Speech Service API containers
- Language service containers
- Translator containers

Additional resources

Documentation

[Copy and back up Custom Vision projects - Azure Cognitive Services](#)

Learn how to use the ExportProject and ImportProject APIs to copy and back up your Custom Vision projects.

[Cognitive Services containers FAQ - Azure Cognitive Services](#)

Get answers to frequently asked questions.

[Use Docker containers in disconnected environments - Azure Cognitive Services](#)

Learn how to run Azure Cognitive Services Docker containers disconnected from the internet.

[Plan to manage costs for Azure Cognitive Services - Azure Cognitive Services](#)

Learn how to plan for and manage costs for Azure Cognitive Services by using cost analysis in the Azure portal.

[Examine the Azure Video Indexer output - Azure Video Indexer](#)

This topic examines the Azure Video Indexer output produced by the Get Video Index API.

[Azure Container Instance recipe - Azure Cognitive Services](#)

Learn how to deploy Cognitive Services Containers on Azure Container Instance

[Running Cognitive Services containers in disconnected environments FAQ - Azure Cognitive Services](#)

Get answers to frequently asked questions about running containers disconnected from the internet.

[What is Azure Content Moderator? - Azure Cognitive Services](#)

Learn how to use Content Moderator to track, flag, assess, and filter inappropriate material in user-generated content.

[Show 5 more](#)

Training

Learning paths and modules

[Deploy cognitive services in containers - Training](#)

Deploy cognitive services in containers

FAQ for Azure Cognitive Services containers

FAQ

General questions

What is available?

Azure Cognitive Services containers allow developers to use the same intelligent APIs that are available in Azure, but with the [benefits](#) of containerization. Some containers are available as a gated preview that might require an application to access. Other containers are publicly available as an ungated preview, or are generally available. You can find a full list of containers and their availability in the [Container support in Azure Cognitive Services](#) article. You can also view the containers in [Docker Hub](#).

Is there any difference between the Cognitive Services cloud and the containers?

Cognitive Services containers are an alternative to the Cognitive Services cloud. Containers offer the same capabilities as the corresponding cloud services. Customers can deploy the containers on-premises or in Azure. The core AI technology, pricing tiers, API keys, and API signature are the same between the container and the corresponding cloud services. There are [features and benefits](#) for choosing containers over their cloud service equivalent.

How do I access and use a gated preview container?

Previously, gated preview containers were hosted on the `containerpreview.azurecr.io` repository. As of September 22, 2020, these containers are hosted on the Microsoft Container Registry. Downloading them doesn't require you to use the `docker login` command. You'll be able to run a gated preview container if your Azure resource was created with the approved Azure subscription ID. You won't be able to run the container if your Azure subscription has not been approved after completion of the [request form](#).

Will containers be available for all Cognitive Services offerings, and what's the next set of containers I should expect?

We'd like to make more Cognitive Services offerings available as containers. Contact your local Microsoft account manager to get updates on new container releases and other Cognitive Services announcements.

What will the service-level agreement (SLA) be for Cognitive Services containers?

Important

To learn more about service-level agreements for Cognitive Services, [visit our SLA page](#).

Cognitive Services provides SLAs which can be viewed on our SLA page at the link provided above.

However, Cognitive Services in containers do not provide an SLA. Customers control Cognitive Services container configurations of resources, so Microsoft won't offer an SLA for general availability (GA). Customers are free to deploy containers on-premises and define the host environments.

Are these containers available in Sovereign clouds?

Cognitive Services containers can be used in the Sovereign clouds. The containers can be run and can connect to the billing endpoint in these clouds, but container images will need to be pulled from the Public cloud container registry.

Versioning

How are containers updated to the latest version?

Customers can choose when to update the containers that they've deployed. Containers will be marked with standard [Docker tags](#) such as `latest` to indicate the most recent version. We encourage customers to pull the latest versions of containers as they're released. For details on how to get notified when an image is updated, see [Azure Container Registry webhooks](#).

What versions will be supported?

The current and last major version of the container will be supported. However, we encourage customers to stay current to get the latest technology.

How are updates versioned?

Major version changes indicate that there's a breaking change to the API signature. We anticipate that this will generally coincide with major version changes to the corresponding Cognitive Services cloud offering. Minor version changes indicate bug fixes, model updates, or new features that don't make a breaking change to the API signature.

Technical questions

How can I diagnose potential errors in my deployment environment?

If you're having trouble running a Cognitive Services container, you can try using the Microsoft diagnostics container. Use this container to diagnose common errors in your deployment environment that might prevent Cognitive Services containers from functioning as expected.

To get the container, use the following `docker pull` command:

```
Bash
```

```
docker pull mcr.microsoft.com/azure-cognitive-services/diagnostic
```

Then run the container. Replace `{ENDPOINT_URI}` with your endpoint, and replace `{API_KEY}` with your key to your resource:

```
Bash
```

```
docker run --rm mcr.microsoft.com/azure-cognitive-services/diagnostic \
Eula=accept \
Billing={ENDPOINT_URI} \
ApiKey={API_KEY}
```

The container will test for network connectivity to the billing endpoint.

How should I run the Cognitive Services containers on IoT devices?

Whether you don't have a reliable internet connection, you want to save on bandwidth cost, you have low-latency requirements, or you're dealing with sensitive data that needs to be analyzed on-site, [Azure IoT Edge with Cognitive Services containers](#) gives you consistency with the cloud.

Are these containers compatible with OpenShift?

We don't test containers with OpenShift, but generally, Cognitive Services containers should run on any platform that supports Docker images. If you're using OpenShift, we recommend running the containers as `root-user`.

How do I provide product feedback and feature recommendations?

We encourage customers to [voice their concerns](#) publicly and to up-vote others who have done the same where potential issues overlap. You can use the feedback tool for both product feedback and feature recommendations.

What status messages and errors do Cognitive Services containers return?

Here are the status messages and errors:

Status	Description
Valid	Your API key is valid. No action is needed.

Status	Description
Invalid	Your API key is invalid. You must provide a valid API key to run the container. Find your API key and service region in the Azure portal, in the Keys and Endpoint section for your Cognitive Services resource.
Mismatch	You've provided an API key or endpoint for a different kind of Cognitive Services resource. Find your API key and service region in the Azure portal, in the Keys and Endpoint section for your Cognitive Services resource.
CouldNotConnect	The container couldn't connect to the billing endpoint. Check the <code>Retry-After</code> value and wait for this period to end before you make more requests.
OutOfQuota	The API key has exceeded the quota. You can either upgrade your pricing tier or wait for additional quota to become available. Find your tier in the Azure portal, in the Pricing Tier section of your Cognitive Service resource.
BillingEndpointBusy	The billing endpoint is currently busy. Check the <code>Retry-After</code> value and wait for this period to end before you make more requests.
ContainerUseUnauthorized	The provided API key is not authorized for use with this container. You're likely using a gated container, so make sure your Azure subscription ID is approved by submitting an online request .
Unknown	The server is currently unable to process billing requests.

How do I get support?

Customer support channels are the same as for the Cognitive Services cloud-based APIs. All Cognitive Services containers include logging features that will help us and the community support customers. Here are options for more support:

- Find contact information in your [Azure support plan](#).
- Try these areas of [Stack Overflow](#), a question-and-answer site for professional and enthusiast programmers:
 - [Azure Cognitive Services](#)

- Microsoft Cognitive ↗

How does billing work?

Customers are charged based on consumption, similar to the Cognitive Services cloud. The containers need to be configured to send metering data to Azure, and transactions will be billed accordingly. Resources used across the hosted and on-premises services will add to the single quota with tiered pricing, counting against both usages. For more information, see the following pricing pages:

- Anomaly Detector
- Computer Vision
- Face
- Form Recognizer
- Language Understanding (LUIS)
- Speech Service API
- Language service - sentiment analysis
- Language service - Text Analytics for health
- Language service - language detection
- Language service - key phrase extraction

ⓘ Important

Cognitive Services containers are *not licensed* to run without being connected to Azure for metering. Customers need to enable the containers to communicate billing information with the metering service at all times. Cognitive Services containers do not send customer data to Microsoft.

Here's an example of the information that a container communicates for metering:

JSON

```
{  
    "SubscriptionId": "Customer subscription ID",  
    "apiType": "textanalytics",  
    "id": "billing id",  
    "containerType": "keyphrase",  
    "containerVersion": "1.1.00864.0001",  
    "containerId": "dep--keyphrase-",  
    "meter": {  
        "name": "CognitiveServices.TextAnalytics.Container.BatchScoring",  
        "quantity": 10.0  
    },  
    "RequestTime": "The time at which the billing record was created"  
}
```

What is the current support warranty for containers?

There is no warranty for previews. Microsoft's standard warranty for enterprise software will apply when containers are formally announced as general availability (GA).

What happens to Cognitive Services containers when internet connectivity is lost?

Cognitive Services containers are *not licensed* to run without being connected to Azure for metering. Customers need to enable the containers to communicate with the metering service at all times.

How long can the container operate without being connected to Azure?

Cognitive Services containers are *not licensed* to run without being connected to Azure for metering. Customers need to enable the containers to communicate with the metering service at all times.

What hardware is required to run these containers?

Cognitive Services containers are x64-based containers that can run any compatible Linux node, VM, and edge device that supports x64 Linux Docker containers. They all require CPU processors. You can find the minimum and recommended configurations for each container offering on these pages:

- [Anomaly Detector](#)
- [Computer Vision](#)
- [Face](#)
- [Form Recognizer](#)
- [Language Understanding \(LUIS\)](#)
- [Speech Service API](#)
- [Language service - sentiment analysis](#)
- [Language service - Text Analytics for health](#)
- [Language service - language detection](#)
- [Language service - key phrase extraction](#)

Are these containers currently supported on Windows?

Cognitive Services containers are Linux containers, but there is some support for Linux containers on Windows. For more information about Linux containers on Windows, see the [Docker documentation](#).

How do Cognitive Services containers compare to AWS and Google offerings?

Microsoft is the first cloud provider to move its pre-trained AI models in containers with simple billing per transaction as though customers are using a cloud service. Microsoft believes a hybrid cloud gives customers more choice.

What compliance certifications do containers have?

Cognitive Services containers don't have any compliance certifications.

What regions are Cognitive Services containers available in?

Containers can be run anywhere in any region, but they need a key and need to call back to Azure for metering. All supported regions for the cloud service are supported for the containers' metering call.

- [How to use Cognitive Services containers in disconnected environments](#)

Use Docker containers in disconnected environments

Article • 02/01/2023 • 10 minutes to read

Containers enable you to run Cognitive Services APIs in your own environment, and are great for your specific security and data governance requirements. Disconnected containers enable you to use several of these APIs disconnected from the internet.

Currently, the following containers can be run in this manner:

- [Speech-to-Text](#)
- [Custom Speech-to-Text](#)
- [Neural Text-to-Speech](#)
- [Text Translation \(Standard\)](#)
- [Language Understanding \(LUIS\)](#)
- Azure Cognitive Service for Language
 - [Sentiment Analysis](#)
 - [Key Phrase Extraction](#)
 - [Language Detection](#)
- Computer Vision - Read

Disconnected container usage is also available for the following Applied AI service:

- [Form Recognizer](#)

Before attempting to run a Docker container in an offline environment, make sure you know the steps to successfully download and use the container. For example:

- Host computer requirements and recommendations.
- The Docker `pull` command you'll use to download the container.
- How to validate that a container is running.
- How to send queries to the container's endpoint, once it's running.

Request access to use containers in disconnected environments

Fill out and submit the [request form](#) to request access to the containers disconnected from the internet.

The form requests information about you, your company, and the user scenario for which you'll use the container. After you submit the form, the Azure Cognitive Services

team reviews it and emails you with a decision within 10 business days.

ⓘ Important

- On the form, you must use an email address associated with an Azure subscription ID.
- The Azure resource you use to run the container must have been created with the approved Azure subscription ID.
- Check your email (both inbox and junk folders) for updates on the status of your application from Microsoft.

After you're approved, you'll be able to run the container after you download it from the Microsoft Container Registry (MCR), described later in the article.

You won't be able to run the container if your Azure subscription hasn't been approved.

Access is limited to customers that meet the following requirements:

- Your organization should be identified as strategic customer or partner with Microsoft.
- Disconnected containers are expected to run fully offline, hence your use cases must meet one of below or similar requirements:
 - Environment or device(s) with zero connectivity to internet.
 - Remote location that occasionally has internet access.
 - Organization under strict regulation of not sending any kind of data back to cloud.
- Application completed as instructed - Please pay close attention to guidance provided throughout the application to ensure you provide all the necessary information required for approval.

Purchase a commitment plan to use containers in disconnected environments

Create a new resource

1. Sign into the [Azure portal](#) and select **Create a new resource** for one of the applicable Cognitive Services or Applied AI services listed above.
2. Enter the applicable information to create your resource. Be sure to select **Commitment tier disconnected containers** as your pricing tier.

 **Note**

- You will only see the option to purchase a commitment tier if you have been approved by Microsoft.
- Pricing details are for example only.

Create Form Recognizer ...

Basics Network Identity Tags Review + create

Accelerate your business processes by automating information extraction. Form Recognizer applies advanced machine learning to accurately extract text, key/value pairs, and tables from documents. With just a few samples, Form Recognizer tailors its understanding to your documents, both on-premises and in the cloud. Turn forms into usable data at a fraction of the time and cost, so you can focus more time acting on the information rather than compiling it. [Learn more.](#)

Project Details

Subscription * ⓘ my-subscription

Resource group * ⓘ my-resource-group

[Create new](#)

Instance Details

Region ⓘ West US 2

Name * ⓘ my-resource

Pricing tier * ⓘ Commitment tier disconnected containers DCO

[View full pricing details](#)

Commitment tier pricing (charged annually)

Select a commitment tier pricing for the capabilities you wish you use. With commitment tier pricing, you will be charged a fixed amount every year for a given quota. If you need additional quota, you could either increase the number of units. [Learn more](#)

Custom API/Invoice * ⓘ Tier 2 (100,000 pages/month for 31,200 USD per unit)

Unit count * ⓘ 1 units

Annual cost (in USD): \$31200

Annual quota (pages/month): 100000

 You will be charged for the full year as soon as the resource is created

I agree to getting charged for the full year as per the commitment plan tier price at the time of resource creation



[Review + create](#)

[< Previous](#)

[Next : Network >](#)



3. Select **Review + Create** at the bottom of the page. Review the information, and select **Create**.

Gather required parameters

There are three primary parameters for all Cognitive Services' containers that are required. The end-user license agreement (EULA) must be present with a value of *accept*. Additionally, both an endpoint URL and API key are needed when you first run the container, to configure it for disconnected usage.

You can find the key and endpoint on the **Key and endpoint** page for your resource.

ⓘ Important

You will only use your key and endpoint to configure the container to be run in a disconnected environment. After you configure the container, you won't need them to send API requests. Store them securely, for example, using Azure Key Vault. Only one key is necessary for this process.

Download a Docker container with `docker pull`

After you have a license file, download the Docker container you have approval to run in a disconnected environment. For example:

Docker

```
docker pull mcr.microsoft.com/azure-cognitive-services/form-  
recognizer/invoice:latest
```

Configure the container to be run in a disconnected environment

Now that you've downloaded your container, you'll need to run the container with the `DownloadLicense=True` parameter in your `docker run` command. This parameter will download a license file that will enable your Docker container to run when it isn't connected to the internet. It also contains an expiration date, after which the license file will be invalid to run the container. You can only use a license file with the appropriate container that you've been approved for. For example, you can't use a license file for a speech-to-text container with a form recognizer container. Please do not rename or modify the license file as this will prevent the container from running successfully.

ⓘ Important

- **Translator container only:**

- You must include a parameter to download model files for the [languages](#) you want to translate. For example: `-e Languages=en,es`
- The container will generate a `docker run` template that you can use to run the container, containing parameters you will need for the downloaded models and configuration file. Make sure you save this template.

The following example shows the formatting of the `docker run` command you'll use, with placeholder values. Replace these placeholder values with your own values.

Placeholder	Value	Format or example
{IMAGE}	The container image you want to use.	<code>mcr.microsoft.com/azure-cognitive-services/form-recognizer/invoice</code>
{LICENSE_MOUNT}	The path where the license will be downloaded, and mounted.	<code>/host/license:/path/to/license/directory</code>
{ENDPOINT_URI}	The endpoint for authenticating your service request. You can find it on your resource's Key and endpoint page, on the Azure portal.	<code>https://<your-custom-subdomain>.cognitiveservices.azure.com</code>
{API_KEY}	The key for your Text Analytics resource. You can find it on your resource's Key and endpoint page, on the Azure portal.	<code>xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx</code>
{CONTAINER_LICENSE_DIRECTORY}	Location of the license folder on the container's local filesystem.	<code>/path/to/license/directory</code>

Bash

```
docker run --rm -it -p 5000:5000 \
-v {LICENSE_MOUNT} \
{IMAGE} \
eula=accept \
billing={ENDPOINT_URI} \
apikey={API_KEY} \
DownloadLicense=True \
Mounts:License={CONTAINER_LICENSE_DIRECTORY}
```

After you've configured the container, use the next section to run the container in your environment with the license, and appropriate memory and CPU allocations.

Run the container in a disconnected environment

Important

If you're using the Translator, Neural text-to-speech, or Speech-to-text containers, read the **Additional parameters** section below for information on commands or additional parameters you will need to use.

Once the license file has been downloaded, you can run the container in a disconnected environment. The following example shows the formatting of the `docker run` command you'll use, with placeholder values. Replace these placeholder values with your own values.

Wherever the container is run, the license file must be mounted to the container and the location of the license folder on the container's local filesystem must be specified with `Mounts:License=`. An output mount must also be specified so that billing usage records can be written.

Placeholder	Value	Format or example
{IMAGE}	The container image you want to use.	<code>mcr.microsoft.com/azure-cognitive-services/form-recognizer/invoice</code>
{MEMORY_SIZE}	The appropriate size of memory to allocate for your container.	<code>4g</code>

Placeholder	Value	Format or example
{NUMBER_CPUS}	The appropriate number of CPUs to allocate for your container.	4
{LICENSE_MOUNT}	The path where the license will be located and mounted.	/host/license:/path/to/license/directory
{OUTPUT_PATH}	The output path for logging usage records.	/host/output:/path/to/output/directory
{CONTAINER_LICENSE_DIRECTORY}	Location of the license folder on the container's local filesystem.	/path/to/license/directory
{CONTAINER_OUTPUT_DIRECTORY}	Location of the output folder on the container's local filesystem.	/path/to/output/directory

Bash

```
docker run --rm -it -p 5000:5000 --memory {MEMORY_SIZE} --cpus {NUMBER_CPUS}
 \
-v {LICENSE_MOUNT} \
-v {OUTPUT_PATH} \
{IMAGE} \
eula=accept \
Mounts:License={CONTAINER_LICENSE_DIRECTORY}
Mounts:Output={CONTAINER_OUTPUT_DIRECTORY}
```

Additional parameters and commands

See the following sections for additional parameters and commands you may need to run the container.

Translator container

If you're using the [Translator container](#), you'll need to add parameters for the downloaded translation models and container configuration. These values are generated

and displayed in the container output when you [configure the container](#) as described above. For example:

```
Bash
```

```
-e MODELS= /path/to/model1/, /path/to/model2/
-e TRANSLATORSYSTEMCONFIG=/path/to/model/config/translatorsystemconfig.json
```

Speech containers

Speech-to-text

The [Speech-to-Text](#) container provides a default directory for writing the license file and billing log at runtime. The default directories are `/license` and `/output` respectively.

When you're mounting these directories to the container with the `docker run -v` command, make sure the local machine directory is set ownership to `user:group nonroot:nonroot` before running the container.

Below is a sample command to set file/directory ownership.

```
Bash
```

```
sudo chown -R nonroot:nonroot <YOUR_LOCAL_MACHINE_PATH_1>
<YOUR_LOCAL_MACHINE_PATH_2> ...
```

Additional resources

Documentation

[Cognitive Services containers FAQ - Azure Cognitive Services](#)

Get answers to frequently asked questions.

[Use Azure Cognitive Services Containers on-premises - Azure Cognitive Services](#)

Learn how to use Docker containers to use Cognitive Services on-premises.

[Azure Container Instance recipe - Azure Cognitive Services](#)

Learn how to deploy Cognitive Services Containers on Azure Container Instance

[Running Cognitive Services containers in disconnected environments FAQ - Azure Cognitive Services](#)

Get answers to frequently asked questions about running containers disconnected from the internet.

[Sign up for Azure Video Indexer and upload your first video - Azure - Azure Video Indexer](#)

Learn how to sign up and upload your first video using the Azure Video Indexer website.

[Configure Read OCR containers - Computer Vision - Azure Cognitive Services](#)

This article shows you how to configure both required and optional settings for Read OCR containers in Computer Vision.

[Frequently asked questions - Computer Vision - Azure Cognitive Services](#)

Get answers to frequently asked questions about the Computer Vision Service in Azure Cognitive Services.

[Computer Vision 3.2 GA Read OCR container - Azure Cognitive Services](#)

Use the Read 3.2 OCR containers from Computer Vision to extract text from images and documents, on-premises.

[Show 5 more](#)

FAQ for Azure Cognitive Services disconnected containers

FAQ

General questions

Which services are available as disconnected containers?

Azure Cognitive Services containers allow developers to use the same intelligent APIs that are available in Azure, but with the [benefits](#) of containerization. Disconnected containers are offered through commitment tier pricing, each offering a discounted rate compared to the pay-as-you-go pricing model. With commitment tier pricing, you can commit to using the following Cognitive Services features for a fixed fee, enabling you to have a predictable total cost based on the needs of your workload.

- Speech Services
 - Speech-to-Text
 - Custom Speech-to-Text
 - Neural Text-to-Speech
- Text Translation (Standard)
- Language Understanding (LUIS)
- Azure Cognitive Service for Language
 - Sentiment Analysis
 - Key Phrase Extraction
 - Language Detection
- Computer Vision - Read

Commitment tier pricing is also available for the following Applied AI service:

- Form Recognizer

For more information about commitment tier pricing, reach out to your Microsoft account team or contact at Microsoft. You can also email csgate@microsoft.com.

Will containers be available for other Cognitive Services offerings, and what's the next set of

containers I should expect?

We'd like to make more Cognitive Services offerings available as containers. Contact your local Microsoft account manager to get updates on new container releases and other Cognitive Services announcements.

How do I get access to disconnected container?

Fill out and submit the [request form](#) to request access to the container. Access is limited to customers that meet the following requirements:

- Your organization must have an enterprise agreement or an equivalent agreement and should be identified as strategic customer or partner with Microsoft.
- Valid business scenario/use case - Disconnected containers are expected to run fully offline hence, your use cases must meet one of below or similar requirements:
 - Environment or device(s) with zero connectivity to internet.
 - Remote location that occasionally has internet access.
 - Organization under strict regulation of not sending any kind of data back to cloud.
- Application completed as instructed - Please pay close attention to guidance provided throughout the application to ensure you provide all the necessary information required for approval.

What if my use case can't satisfy the requirements listed above?

If your use case can't satisfy above requirements but you're interested in running containers on premises, you may be able to use [connected containers](#).

What are some reasons my application may be denied?

Possible causes for a denied application are as follows:

- Not being an existing Microsoft partner or enterprise agreement customer
- Invalid Azure subscription ID. your ID must belong to the applicant organization, and you can't apply on behalf of other organizations.
- The application was submitted with a personal email address (For example: @hotmail.com, @gmail.com, @yahoo.com, etc.)
- Invalid justification or business scenario provided

- Other missing or inaccurate data

How long does it take to approve my application to access disconnected containers?

After you submit the form, the Azure Cognitive Services team will review it and email you a decision within 10 business days.

Once your application is approved, the Cognitive Services gating team will communicate details for purchasing the commitment tier pricing, and instructions to download and run the containers.

If you have any questions on the application, gating process or other information needed, email csgate@microsoft.com

How do I download the disconnected containers?

These containers are hosted on the Microsoft Container Registry and available for download on [Microsoft Artifact Registry](#) and [Docker Hub](#). You won't be able to run the container if your Azure subscription has not been approved after completion of the request form. Once you're approved, you can follow the steps in [Use Docker containers in disconnected environments](#)

What is the service-level agreement (SLA) for Cognitive Services containers?

Cognitive Services disconnected containers don't have an SLA, because customers control configuration of the resources used. Customers are free to deploy containers on-premises and define the host environments.

Are these containers available in sovereign clouds?

Not everyone is familiar with the term sovereign cloud, so let's begin with definition: The sovereign cloud consists of the [Azure Government](#), [Azure Germany](#), and [Azure China 21Vianet](#) clouds. Currently, customers are able to create disconnected containers resource from public cloud.

Versioning

How are containers updated to the latest version?

Customers can choose when to update the containers that they've deployed. Containers will be marked with standard Docker tags such as latest to indicate the most recent version. We encourage customers to pull the latest versions of containers as they're released. For details on how to get notified when an image is updated, see Azure Container Registry webhooks.

What versions will be supported?

The current and last major version of the container will be supported. However, we encourage customers to stay current to get the latest technology to be compliant.

How are updates versioned?

Major version changes indicate that there's a breaking change to the API signature. We anticipate that this will generally coincide with major version changes to the corresponding Cognitive Services cloud offering. Minor version changes indicate bug fixes, model updates, or new features that don't make a breaking change to the API signature.

Technical questions

How can I diagnose potential errors in my deployment environment?

If you're having trouble running a Cognitive Services container, you can try using the Microsoft diagnostics container. Use this container to diagnose common errors in your deployment environment that might prevent Cognitive Services containers from functioning as expected.

To get the container, use the following `docker pull` command:

Bash

```
docker pull mcr.microsoft.com/azure-cognitive-services/diagnostic
```

Then run the container. Replace `{ENDPOINT_URI}` with your endpoint, and replace `{API_KEY}` with your key to your resource:

Bash

```
docker run --rm mcr.microsoft.com/azure-cognitive-services/diagnostic \
eula=accept \
Billing={ENDPOINT_URI} \
ApiKey={API_KEY}
```

The container will test for network connectivity to the billing endpoint.

Are these containers compatible with OpenShift?

We don't test containers with OpenShift, but generally, Cognitive Services containers should run on any platform that supports Docker images. If you're using OpenShift, we recommend running the containers as root-user.

How do I provide product feedback and feature recommendations?

We encourage customers to [voice their concerns](#) publicly and to up-vote others who have done the same where potential issues overlap. You can use the feedback tool for both product feedback and feature recommendations.

What status messages and errors do Cognitive Services containers return?

Here are the status messages and errors:

Status	Description
Valid	Your API key is valid. No action is needed.
Invalid	Your API key is invalid. You must provide a valid API key to configure the container for use in disconnected environments. Find your API key and service region in the Azure portal, in the Keys and Endpoint section for your Cognitive Services resource.

Status	Description
Mismatch	You've provided an API key or endpoint for a different kind of Cognitive Services resource. Find your API key and service region in the Azure portal, in the Keys and Endpoint section for your Cognitive Services resource.
CouldNotConnect	The container couldn't connect to the endpoint.
OutOfQuota	The API key has exceeded the quota. You can either upgrade your pricing tier or wait for additional quota to become available. Find your tier in the Azure portal, in the Pricing Tier section of your Cognitive Service resource.
ContainerUseUnauthorized	The provided API key isn't authorized for use with this container. You're likely using a gated container, so make sure your Azure subscription ID is approved by submitting an online request .
Unknown	The server is currently unable to process requests.

How do I get support?

Customer support channels are the same as for the Cognitive Services cloud-based APIs. All Cognitive Services containers include logging features that will help us and the community support customers. Here are options for more support:

- Find contact information in your [Azure support plan](#).
- Try these areas of Stack Overflow, a question-and-answer site for professional and enthusiast programmers:
- [Azure Cognitive Services](#)
- [Microsoft Cognitive](#)

How does billing work?

Customers are charged based on the [commitment tier](#) they purchase. Choose an annual commitment plan based on the needs of your workload. You can customize the "Commitment tier disconnected containers DC0" pricing tier by selecting one of several commitment tier options.

Each commitment tier offers a discounted rate as compared to the pay-as-you-go pricing model. Commitment plans are charged annually starting from the day the resource was created. Additional units can be added at any time in the year and are charged pro-rated based on the number of days remaining in the billing year. For more information, see the following pricing pages:

💡 Tip

To find the pricing information for disconnected containers on these pricing pages, scroll down to **Commitment Tiers** section.

- [Speech Services](#) for:
 - Speech-to-Text
 - Custom Speech-to-Text
 - Neural Text-to-Speech
- [Text Translation \(Standard\)](#)
- [Language Understanding \(LUIS\)](#)
- [Azure Cognitive Service for Language pricing](#) for
- Sentiment Analysis
- Key Phrase Extraction
- Language Detection
- [Computer Vision - Read](#)
- [Form Recognizer - Custom/Invoice](#)

How do I monitor the container's usage?

Docker containers running in a disconnected environment will write usage records to a volume where they're collected over time. These containers also provide a REST endpoint to generate a report about usage. See [Use containers in disconnected environments](#) for more information.

What is the current support warranty for containers?

Microsoft's standard warranty for enterprise software will apply when a container image is formally announced as General Available (GA). There's no warranty for container images in preview.

What hardware is required to run these containers?

Cognitive Services containers are x64-based containers that can run any compatible Linux node, VM, and edge device that supports x64 Linux Docker containers. They all require CPU processors. You can find the minimum and recommended configurations for each container offering on these pages:

- [Speech Services](#) for:
 - [Speech-to-Text](#)
 - [Custom Speech-to-Text](#)
 - [Neural Text-to-Speech](#)
- [Text Translation \(Standard\)](#)
- [Language Understanding \(LUIS\)](#)
- Azure Cognitive Service for Language documentation for
 - [Sentiment Analysis](#)
 - [Key Phrase Extraction](#)
 - [Language Detection](#)
- [Computer Vision - Read](#)
- [Form Recognizer - Custom/Invoice](#)

Are these containers currently supported on Windows?

Cognitive Services containers are Linux containers, but there's some support for Linux containers on Windows. For more information about Linux containers on Windows, see the [Docker documentation](#).

What compliance certifications do containers have?

Cognitive Services containers don't have any compliance certifications.

Create containers for reuse

Article • 02/11/2022 • 4 minutes to read

Use these container recipes to create Cognitive Services Containers that can be reused. Containers can be built with some or all configuration settings so that they are *not* needed when the container is started.

Once you have this new layer of container (with settings), and you have tested it locally, you can store the container in a container registry. When the container starts, it will only need those settings that are not currently stored in the container. The private registry container provides configuration space for you to pass those settings in.

Docker run syntax

Any `docker run` examples in this document assume a Windows console with a `\` line continuation character. Consider the following for your own use:

- Do not change the order of the arguments unless you are very familiar with docker containers.
- If you are using an operating system other than Windows, or a console other than Windows console, use the correct console/terminal, folder syntax for mounts, and line continuation character for your console and system. Because the Cognitive Services container is a Linux operating system, the target mount uses a Linux-style folder syntax.
- `docker run` examples use the directory off the `c:\` drive to avoid any permission conflicts on Windows. If you need to use a specific directory as the input directory, you may need to grant the docker service permission.

Store no configuration settings in image

The example `docker run` commands for each service do not store any configuration settings in the container. When you start the container from a console or registry service, those configuration settings need to pass in. The private registry container provides configuration space for you to pass those settings in.

Reuse recipe: store all configuration settings with container

In order to store all configuration settings, create a `Dockerfile` with those settings.

Issues with this approach:

- The new container has a separate name and tag from the original container.
- In order to change these settings, you will have to change the values of the Dockerfile, rebuild the image, and republish to your registry.
- If someone gets access to your container registry or your local host, they can run the container and use the Cognitive Services endpoints.
- If your Cognitive Service doesn't require input mounts, don't add the `COPY` lines to your Dockerfile.

Create Dockerfile, pulling from the existing Cognitive Services container you want to use, then use docker commands in the Dockerfile to set or pull in information the container needs.

This example:

- Sets the billing endpoint, `{BILLING_ENDPOINT}` from the host's environment key using `ENV`.
- Sets the billing API-key, `{ENDPOINT_KEY}` from the host's environment key using ``ENV``.

Reuse recipe: store billing settings with container

This example shows how to build the Language service's sentiment container from a Dockerfile.

Dockerfile

```
FROM mcr.microsoft.com/azure-cognitive-services/sentiment:latest
ENV billing={BILLING_ENDPOINT}
ENV apikey={ENDPOINT_KEY}
ENV EULA=accept
```

Build and run the container [locally](#) or from your [private registry container](#) as needed.

Reuse recipe: store billing and mount settings with container

This example shows how to use Language Understanding, saving billing and models from the Dockerfile.

- Copies the Language Understanding (LUIS) model file from the host's file system using `COPY`.
- The LUIS container supports more than one model. If all models are stored in the same folder, you all need one `COPY` statement.
- Run the docker file from the relative parent of the model input directory. For the following example, run the `docker build` and `docker run` commands from the relative parent of `/input`. The first `/input` on the `COPY` command is the host computer's directory. The second `/input` is the container's directory.

Dockerfile

```
FROM <container-registry>/<cognitive-service-container-name>:<tag>
ENV billing={BILLING_ENDPOINT}
ENV apikey={ENDPOINT_KEY}
ENV EULA=accept
COPY /input /input
```

Build and run the container [locally](#) or from your [private registry container](#) as needed.

How to use container on your local host

To build the Docker file, replace `<your-image-name>` with the new name of the image, then use:

Console

```
docker build -t <your-image-name> .
```

To run the image, and remove it when the container stops (`--rm`):

Console

```
docker run --rm <your-image-name>
```

How to add container to private registry

Follow these steps to use the Dockerfile and place the new image in your private container registry.

1. Create a `Dockerfile` with the text from reuse recipe. A `Dockerfile` doesn't have an extension.

2. Replace any values in the angle brackets with your own values.
3. Build the file into an image at the command line or terminal, using the following command. Replace the values in the angle brackets, <>, with your own container name and tag.

The tag option, -t, is a way to add information about what you have changed for the container. For example, a container name of modified-LUIS indicates the original container has been layered. A tag name of with-billing-and-model indicates how the Language Understanding (LUIS) container has been modified.

Bash

```
docker build -t <your-new-container-name>:<your-new-tag-name> .
```

4. Sign in to Azure CLI from a console. This command opens a browser and requires authentication. Once authenticated, you can close the browser and continue working in the console.

Azure CLI

```
az login
```

5. Sign in to your private registry with Azure CLI from a console.

Replace the values in the angle brackets, <my-registry>, with your own registry name.

Azure CLI

```
az acr login --name <my-registry>
```

You can also sign in with docker login if you are assigned a service principal.

Bash

```
docker login <my-registry>.azurecr.io
```

6. Tag the container with the private registry location. Replace the values in the angle brackets, <my-registry>, with your own registry name.

Bash

```
docker tag <your-new-container-name>:<your-new-tag-name> <my-registry>.azurecr.io/<your-new-container-name-in-registry>:<your-new-tag-name>
```

If you don't use a tag name, `latest` is implied.

7. Push the new image to your private container registry. When you view your private container registry, the container name used in the following CLI command will be the name of the repository.

Bash

```
docker push <my-registry>.azurecr.io/<your-new-container-name-in-registry>:<your-new-tag-name>
```

Next steps

[Create and use Azure Container Instance](#)

Deploy and run container on Azure Container Instance

Article • 02/11/2022 • 7 minutes to read

With the following steps, scale Azure Cognitive Services applications in the cloud easily with Azure [Container Instances](#). Containerization helps you focus on building your applications instead of managing the infrastructure. For more information on using containers, see [features and benefits](#).

Prerequisites

The recipe works with any Cognitive Services container. The Cognitive Service resource must be created before using the recipe. Each Cognitive Service that supports containers has a "How to install" article for installing and configuring the service for a container. Some services require a file or set of files as input for the container, it is important that you understand and have used the container successfully before using this solution.

- An Azure resource for the Azure Cognitive Service you're using.
- Cognitive Service **endpoint URL** - review your specific service's "How to install" for the container, to find where the endpoint URL is from within the Azure portal, and what a correct example of the URL looks like. The exact format can change from service to service.
- Cognitive Service **key** - the keys are on the **Keys** page for the Azure resource. You only need one of the two keys. The key is a string of 32 alpha-numeric characters.
- A single Cognitive Services Container on your local host (your computer). Make sure you can:
 - Pull down the image with a `docker pull` command.
 - Run the local container successfully with all required configuration settings with a `docker run` command.
 - Call the container's endpoint, getting a response of HTTP 2xx and a JSON response back.

All variables in angle brackets, `<>`, need to be replaced with your own values. This replacement includes the angle brackets.

 **Important**

The LUIS container requires a `.gz` model file that is pulled in at runtime. The container must be able to access this model file via a volume mount from the container instance. To upload a model file, follow these steps:

1. [Create an Azure file share](#). Take note of the Azure Storage account name, key, and file share name as you'll need them later.
2. [export your LUIS model \(packaged app\) from the LUIS portal](#).
3. In the Azure portal, navigate to the [Overview](#) page of your storage account resource, and select **File shares**.
4. Select the file share name that you recently created, then select **Upload**. Then upload your packaged app.

Azure portal

Create an Azure Container Instance resource using the Azure portal

1. Go to the [Create](#) page for Container Instances.
2. On the **Basics** tab, enter the following details:

Setting	Value
Subscription	Select your subscription.
Resource group	Select the available resource group or create a new one such as <code>cognitive-services</code> .
Container name	Enter a name such as <code>cognitive-container-instance</code> . The name must be in lower caps.
Location	Select a region for deployment.
Image type	If your container image is stored in a container registry that doesn't require credentials, choose <code>Public</code> . If accessing your container image requires credentials, choose <code>Private</code> . Refer to container repositories and images for details on whether or not the container image is <code>Public</code> or <code>Private</code> ("Public Preview").

Setting	Value
Image name	<p>Enter the Cognitive Services container location. The location is what's used as an argument to the <code>docker pull</code> command. Refer to the container repositories and images for the available image names and their corresponding repository.</p> <p>The image name must be fully qualified specifying three parts. First, the container registry, then the repository, finally the image name: <code><container-registry>/<repository>/<image-name></code>.</p> <p>Here is an example, <code>mcr.microsoft.com/azure-cognitive-services/keyphrase</code> would represent the Key Phrase Extraction image in the Microsoft Container Registry under the Azure Cognitive Services repository. Another example is, <code>containerpreview.azurecr.io/microsoft/cognitive-services-speech-to-text</code> which would represent the Speech to Text image in the Microsoft repository of the Container Preview container registry.</p>
OS type	<code>Linux</code>
Size	<p>Change size to the suggested recommendations for your specific Cognitive Service container:</p> <ul style="list-style-type: none"> 2 CPU cores 4 GB

3. On the **Networking** tab, enter the following details:

Setting	Value
Ports	Set the TCP port to <code>5000</code> . Exposes the container on port 5000.

4. On the **Advanced** tab, enter the required **Environment Variables** for the container billing settings of the Azure Container Instance resource:

Key	Value
<code>ApiKey</code>	Copied from the Keys and endpoint page of the resource. It is a 32 alphanumeric-character string with no spaces or dashes, <code>xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx</code> .
<code>Billing</code>	Your endpoint URL copied from the Keys and endpoint page of the resource.
<code>Eula</code>	<code>accept</code>

5. Click **Review and Create**

6. After validation passes, click **Create** to finish the creation process

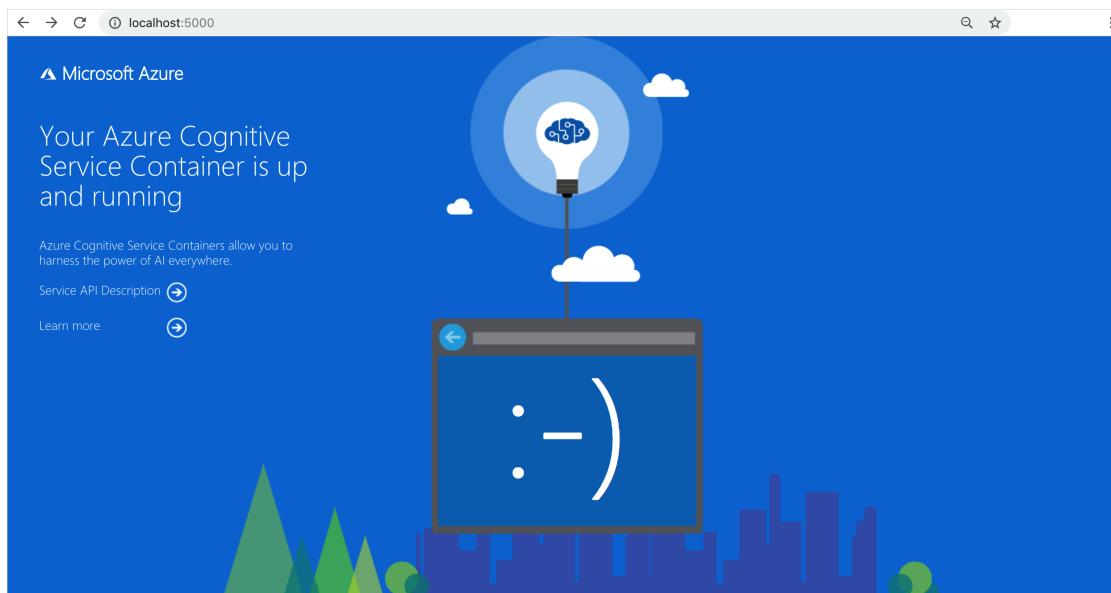
7. When the resource is successfully deployed, it's ready

Use the Container Instance

Azure portal

1. Select the **Overview** and copy the IP address. It will be a numeric IP address such as `55.55.55.55`.

2. Open a new browser tab and use the IP address, for example, `http://<IP-address>:5000` (`http://55.55.55.55:5000`). You will see the container's home page, letting you know the container is running.



3. Select **Service API Description** to view the swagger page for the container.

4. Select any of the **POST** APIs and select **Try it out**. The parameters are displayed including the input. Fill in the parameters.

5. Select **Execute** to send the request to your Container Instance.

You have successfully created and used Cognitive Services containers in Azure Container Instance.

Deploy a language detection container to Azure Kubernetes Service

Article • 02/14/2022 • 10 minutes to read

Learn how to deploy the language detection container. This procedure shows you how to create the local Docker containers, push the containers to your own private container registry, run the container in a Kubernetes cluster, and test it in a web browser.

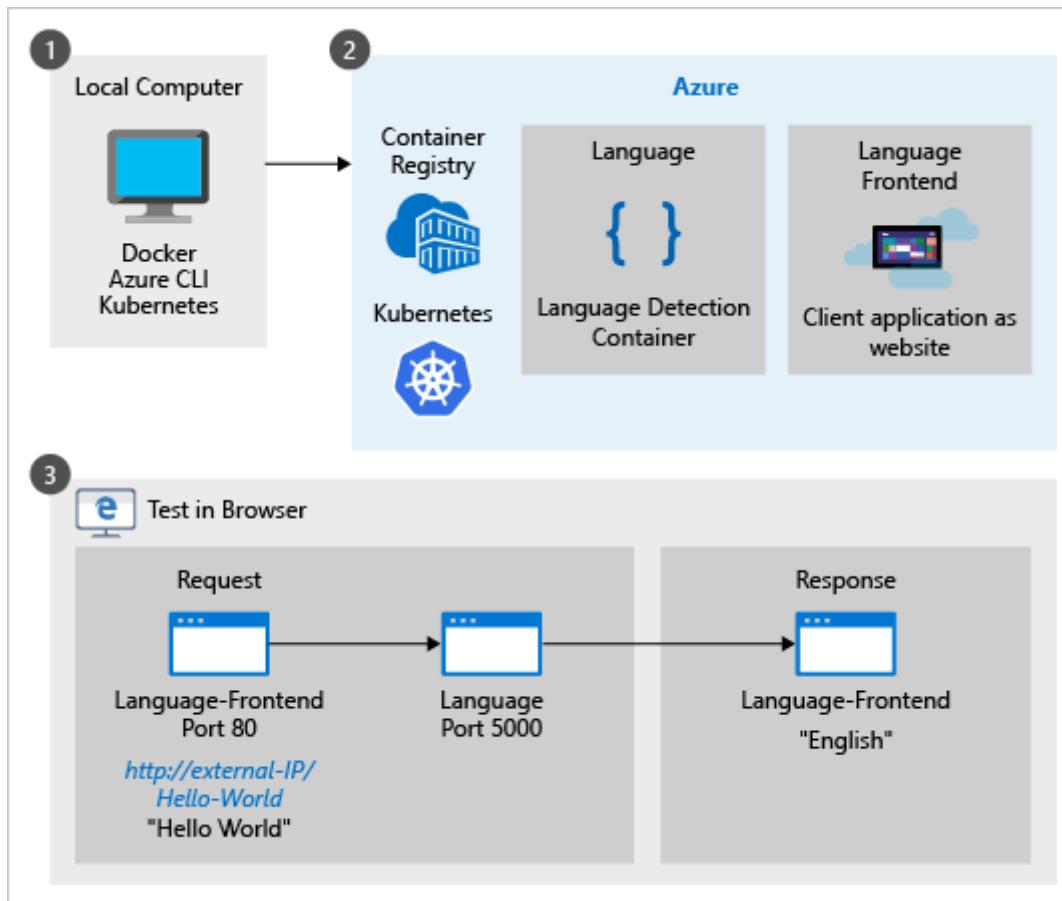
Prerequisites

This procedure requires several tools that must be installed and run locally. Do not use Azure Cloud Shell.

- Use an Azure subscription. If you don't have an Azure subscription, create a [free account](#) before you begin.
- [Git](#) for your operating system so you can clone the [sample](#) used in this procedure.
- [Azure CLI](#).
- [Docker engine](#) and validate that the Docker CLI works in a console window.
- [kubectl](#).
- An Azure resource with the correct pricing tier. Not all pricing tiers work with this container:
 - [Language](#) resource with F0 or Standard pricing tiers only.
 - [Cognitive Services](#) resource with the S0 pricing tier.

Running the sample

This procedure loads and runs the Cognitive Services Container sample for language detection. The sample has two containers, one for the client application and one for the Cognitive Services container. We'll push both of these images to the Azure Container Registry. Once they are on your own registry, create an Azure Kubernetes Service to access these images and run the containers. When the containers are running, use the `kubectl` CLI to watch the containers performance. Access the client application with an HTTP request and see the results.



The sample containers

The sample has two container images, one for the frontend website. The second image is the language detection container returning the detected language (culture) of text. Both containers are accessible from an external IP when you are done.

The language-frontend container

This website is equivalent to your own client-side application that makes requests of the language detection endpoint. When the procedure is finished, you get the detected language of a string of characters by accessing the website container in a browser with `http://<external-IP>/<text-to-analyze>`. An example of this URL is `http://132.12.23.255/helloworld!`. The result in the browser is `English`.

The language container

The language detection container, in this specific procedure, is accessible to any external request. The container hasn't been changed in any way so the standard Cognitive Services container-specific language detection API is available.

For this container, that API is a POST request for language detection. As with all Cognitive Services containers, you can learn more about the container from its hosted Swagger information, <http://<external-IP>:5000/swagger/index.html>.

Port 5000 is the default port used with the Cognitive Services containers.

Create Azure Container Registry service

To deploy the container to the Azure Kubernetes Service, the container images need to be accessible. Create your own Azure Container Registry service to host the images.

1. Sign in to the Azure CLI

```
Azure CLI
```

```
az login
```

2. Create a resource group named `cogserv-container-rg` to hold every resource created in this procedure.

```
Azure CLI
```

```
az group create --name cogserv-container-rg --location westus
```

3. Create your own Azure Container Registry with the format of your name then `registry`, such as `pattyregistry`. Do not use dashes or underline characters in the name.

```
Azure CLI
```

```
az acr create --resource-group cogserv-container-rg --name pattyregistry --sku Basic
```

Save the results to get the `loginServer` property. This will be part of the hosted container's address, used later in the `language.yml` file.

```
Azure CLI
```

```
az acr create --resource-group cogserv-container-rg --name pattyregistry --sku Basic
```

```
Output
```

```
{  
    "adminUserEnabled": false,  
    "creationDate": "2019-01-02T23:49:53.783549+00:00",  
    "id": "/subscriptions/xxxxxxxx-xxxx-xxxx-xxxx-  
xxxxxxxxxx/resourceGroups/cogserv-container-  
rg/providers/Microsoft.ContainerRegistry/registries/pattyregistry",  
    "location": "westus",  
    "loginServer": "pattyregistry.azurecr.io",  
    "name": "pattyregistry",  
    "provisioningState": "Succeeded",  
    "resourceGroup": "cogserv-container-rg",  
    "sku": {  
        "name": "Basic",  
        "tier": "Basic"  
    },  
    "status": null,  
    "storageAccount": null,  
    "tags": {},  
    "type": "Microsoft.ContainerRegistry/registries"  
}
```

4. Sign in to your container registry. You need to login before you can push images to your registry.

Azure CLI

```
az acr login --name pattyregistry
```

Get website Docker image

1. The sample code used in this procedure is in the Cognitive Services containers samples repository. Clone the repository to have a local copy of the sample.

Console

```
git clone https://github.com/Azure-Samples/cognitive-services-  
containers-samples
```

Once the repository is on your local computer, find the website in the [\dotnet\Language\FrontendService](#) directory. This website acts as the client application calling the language detection API hosted in the language detection container.

2. Build the Docker image for this website. Make sure the console is in the [\FrontendService](#) directory where the Dockerfile is located when you run the following command:

Console

```
docker build -t language-frontend -t  
pattyregistry.azurecr.io/language-frontend:v1 .
```

To track the version on your container registry, add the tag with a version format, such as v1.

3. Push the image to your container registry. This may take a few minutes.

Console

```
docker push pattyregistry.azurecr.io/language-frontend:v1
```

If you get an unauthorized: authentication required error, login with the az acr login --name <your-container-registry-name> command.

When the process is done, the results should be similar to:

Output

```
The push refers to repository [pattyregistry.azurecr.io/language-  
frontend]  
82ff52ee6c73: Pushed  
07599c047227: Pushed  
816caf41a9a1: Pushed  
2924be3aed17: Pushed  
45b83a23806f: Pushed  
ef68f6734aa4: Pushed  
v1: digest:  
sha256:31930445deee181605c0cde53dab5a104528dc1ff57e5b3b34324f0d8a0eb286  
size: 1580
```

Get language detection Docker image

1. Pull the latest version of the Docker image to the local machine. This may take a few minutes. If there is a newer version of this container, change the value from 1.1.006770001-amd64-preview to the newer version.

Console

```
docker pull mcr.microsoft.com/azure-cognitive-  
services/language:1.1.006770001-amd64-preview
```

2. Tag image with your container registry. Find the latest version and replace the version `1.1.006770001-amd64-preview` if you have a more recent version.

```
Console
```

```
docker tag mcr.microsoft.com/azure-cognitive-services/language  
pattyregistry.azurecr.io/language:1.1.006770001-amd64-preview
```

3. Push the image to your container registry. This may take a few minutes.

```
Console
```

```
docker push pattyregistry.azurecr.io/language:1.1.006770001-amd64-  
preview
```

Get Container Registry credentials

The following steps are needed to get the required information to connect your container registry with the Azure Kubernetes Service you create later in this procedure.

1. Create service principal.

```
Azure CLI
```

```
az ad sp create-for-rbac
```

Save the results `appId` value for the assignee parameter in step 3, `<appId>`. Save the `password` for the next section's client-secret parameter `<client-secret>`.

```
Output
```

```
{  
  "appId": "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",  
  "displayName": "azure-cli-2018-12-31-18-39-32",  
  "name": "http://azure-cli-2018-12-31-18-39-32",  
  "password": "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",  
  "tenant": "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"  
}
```

2. Get your container registry ID.

```
Azure CLI
```

```
az acr show --resource-group cogserv-container-rg --name pattyregistry  
--query "id" --o table
```

Save the output for the scope parameter value, `<acrId>`, in the next step. It looks like:

Output

```
/subscriptions/xxxxxxxx-xxxx-xxxx-xxxx-  
xxxxxxxxxxxx/resourceGroups/cogserv-container-  
rg/providers/Microsoft.ContainerRegistry/registries/pattyregistry
```

Save the full value for step 3 in this section.

- To grant the correct access for the AKS cluster to use images stored in your container registry, create a role assignment. Replace `<appId>` and `<acrId>` with the values gathered in the previous two steps.

Azure CLI

```
az role assignment create --assignee <appId> --scope <acrId> --role Reader
```

Create Azure Kubernetes Service

1. Create the Kubernetes cluster. All the parameter values are from previous sections except the name parameter. Choose a name that indicates who created it and its purpose, such as `patty-kube`.

Azure CLI

```
az aks create --resource-group cogserv-container-rg --name patty-kube --node-count 2 --service-principal <appId> --client-secret <client-secret> --generate-ssh-keys
```

This step may take a few minutes. The result is:

Output

```
{  
  "aadProfile": null,  
  "addonProfiles": null,  
  "agentPoolProfiles": [  
    {  
      "count": 2,  
      "dnsPrefix": null,  
      "fqdn": null,  
      "maxPods": 110,
```

```

        "name": "nodepool1",
        "osDiskSizeGb": 30,
        "osType": "Linux",
        "ports": null,
        "storageProfile": "ManagedDisks",
        "vmSize": "Standard_DS1_v2",
        "vnetSubnetId": null
    }
],
"dnsPrefix": "patty-kube--65a101",
"enableRbac": true,
"fqdn": "patty-kube--65a101-341f1f54.hcp.westus.azmk8s.io",
"id": "/subscriptions/xxxxxxxxx-xxxx-xxxx-xxxx-
xxxxxxxxxx/resourcegroups/cogserv-container-
rg/providers/Microsoft.ContainerService/managedClusters/patty-kube",
"kubernetesVersion": "1.9.11",
"linuxProfile": {
    "adminUsername": "azureuser",
    "ssh": {
        "publicKeys": [
            {
                "keyData": "ssh-rsa AAAAB3NzaC...ohR2d81mFC
            }
        ]
    }
},
"location": "westus",
"name": "patty-kube",
"networkProfile": {
    "dnsServiceIp": "10.0.0.10",
    "dockerBridgeCidr": "172.17.0.1/16",
    "networkPlugin": "kubenet",
    "networkPolicy": null,
    "podCidr": "10.244.0.0/16",
    "serviceCidr": "10.0.0.0/16"
},
"nodeResourceGroup": "MC_patty_westus",
"provisioningState": "Succeeded",
"resourceGroup": "cogserv-container-rg",
"servicePrincipalProfile": {
    "clientId": "xxxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
    "keyVaultSecretRef": null,
    "secret": null
},
"tags": null,
"type": "Microsoft.ContainerService/ManagedClusters"
}

```

The service is created but it doesn't have the website container or language detection container yet.

2. Get credentials of the Kubernetes cluster.

Azure CLI

```
az aks get-credentials --resource-group cogserv-container-rg --name patty-kube
```

Load the orchestration definition into your Kubernetes service

This section uses the **kubectl** CLI to talk with the Azure Kubernetes Service.

1. Before loading the orchestration definition, check **kubectl** has access to the nodes.

Console

```
kubectl get nodes
```

The response looks like:

Output

NAME	STATUS	ROLES	AGE	VERSION
aks-nodepool1-13756812-0	Ready	agent	6m	v1.9.11
aks-nodepool1-13756812-1	Ready	agent	6m	v1.9.11

2. Copy the following file and name it `language.yml`. The file has a `service` section and a `deployment` section each for the two container types, the `language-frontend` website container and the `language` detection container.

yml

```
# A service which exposes the .net frontend app container through a dependable hostname: http://language-frontend:5000
apiVersion: v1
kind: Service
metadata:
  name: language-frontend
  labels:
    run: language-frontend
spec:
  selector:
    app: language-frontend
  type: LoadBalancer
  ports:
    - name: front
      port: 80
      targetPort: 80
```

```

    protocol: TCP
    ---

# A deployment declaratively indicating how many instances of the .net
frontend app container we want up
apiVersion: apps/v1beta1
kind: Deployment
metadata:
  name: language-frontend
spec:
  replicas: 1
  template:
    metadata:
      labels:
        app: language-frontend
    spec:
      containers:
        - name: language-frontend
          image: # < URI of the Frontend App image >
          ports:
            - name: public-port
              containerPort: 80
          livenessProbe:
            httpGet:
              path: /status
              port: public-port
            initialDelaySeconds: 30
            timeoutSeconds: 1
            periodSeconds: 10
          imagePullSecrets:
            - name: # < Name of the registry secret providing access to the
              frontend image >
          automountServiceAccountToken: false
  ---

# A service which exposes the cognitive-service containers through a
dependable hostname: http://language:5000
apiVersion: v1
kind: Service
metadata:
  name: language
  labels:
    run: language
spec:
  selector:
    app: language
  type: LoadBalancer
  ports:
    - name: language
      port: 5000
      targetPort: 5000
      protocol: TCP
  ---

# A deployment declaratively indicating how many instances of the
cognitive-service container we want up
apiVersion: apps/v1beta1
kind: Deployment

```

```

metadata:
  name: language
spec:
  replicas: 1
  template:
    metadata:
      labels:
        app: language
    spec:
      containers:
        - name: language
          image: # < URI of the Language Image >
          ports:
            - name: public-port
              containerPort: 5000
      livenessProbe:
        httpGet:
          path: /status
          port: public-port
        initialDelaySeconds: 30
        timeoutSeconds: 1
        periodSeconds: 10
      args:
        - "eula=accept"
        - "apikey=" # < API Key for the Language Service >
        - "billing=" # < Language billing endpoint URI >
  imagePullSecrets:
    - name: # < Name of the registry secret providing access to the
      Language image >
  automountServiceAccountToken: false

```

3. Change the language-frontend deployment lines of `language.yml` based on the following table to add your own container registry image names, client secret, and Language service settings.

Language-frontend deployment settings	Purpose
Line 32 <code>image</code> property	Image location for the frontend image in your Container Registry <code><container-registry-name>.azurecr.io/language-frontend:v1</code>
Line 44 <code>name</code> property	Container Registry secret for the image, referred to as <code><client-secret></code> in a previous section.

4. Change the language deployment lines of `language.yml` based on the following table to add your own container registry image names, client secret, and Language service settings.

Language deployment settings	Purpose
Line 78 <code>image</code> property	Image location for the language image in your Container Registry <code><container-registry-name>.azurecr.io/language:1.1.006770001-amd64-preview</code>
Line 95 <code>name</code> property	Container Registry secret for the image, referred to as <code><client-secret></code> in a previous section.
Line 91 <code>apiKey</code> property	Your Language service resource key
Line 92 <code>billing</code> property	The billing endpoint for your Language service resource. <code>https://westus.api.cognitive.microsoft.com/text/analytics/v2.1</code>

Because the **apiKey** and **billing endpoint** are set as part of the Kubernetes orchestration definition, the website container doesn't need to know about these or pass them as part of the request. The website container refers to the language detection container by its orchestrator name `language`.

5. Load the orchestration definition file for this sample from the folder where you created and saved the `language.yml`.

Console

```
kubectl apply -f language.yml
```

The response is:

Output

```
service "language-frontend" created
deployment.apps "language-frontend" created
service "language" created
deployment.apps "language" created
```

Get external IPs of containers

For the two containers, verify the `language-frontend` and `language` services are running and get the external IP address.

Console

```
kubectl get all
```

Output

NAME	READY	STATUS	RESTARTS	AGE
pod/language-586849d8dc-7zvz5	1/1	Running	0	13h
pod/language-frontend-68b9969969-bz9bg	1/1	Running	1	13h

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP
service/kubernetes	ClusterIP	10.0.0.1	<none>
443/TCP	14h		
service/language	LoadBalancer	10.0.39.169	104.42.172.68
5000:30161/TCP	13h		
service/language-frontend	LoadBalancer	10.0.42.136	104.42.37.219
80:30943/TCP	13h		

NAME	DESIRED	CURRENT	UP-TO-DATE
AVAILABLE AGE			
deployment.extensions/language 13h	1	1	1
deployment.extensions/language-frontend 13h	1	1	1

NAME	DESIRED	CURRENT
READY AGE		
replicaset.extensions/language-586849d8dc 13h	1	1
replicaset.extensions/language-frontend-68b9969969 13h	1	1

NAME	DESIRED	CURRENT	UP-TO-DATE
AVAILABLE AGE			
deployment.apps/language 13h	1	1	1
deployment.apps/language-frontend 13h	1	1	1

NAME	DESIRED	CURRENT	READY
AGE			
replicaset.apps/language-586849d8dc 13h	1	1	1
replicaset.apps/language-frontend-68b9969969 13h	1	1	1

If the `EXTERNAL-IP` for the service is shown as pending, rerun the command until the IP address is shown before moving to the next step.

Test the language detection container

Open a browser and navigate to the external IP of the `language` container from the previous section: `http://<external-ip>:5000/swagger/index.html`. You can use the Try it feature of the API to test the language detection endpoint.

The screenshot shows the Swagger UI interface for the Language Detection Cognitive Service API. At the top, there's a green header with the word "swagger" and a dropdown menu "Select a spec" set to "Language Detection Cognitive Service API V1". Below the header, the title "Language Detection Cognitive Service API" is followed by a small "v2" icon. A note below the title says: "The API returns the detected language and a numeric score between 0 and 1. Scores close to 1 indicate 100% certainty that the identified language is true. A total of 120 languages are supported." There are two main sections: "Detect Language" and "Status". The "Detect Language" section has a green "POST" button next to the URL "/text/analytics/v2.0/languages Detect Language". The "Status" section has a blue "GET" button next to the URL "/status". Both sections have a small downward arrow icon at the end of their respective sections.

Test the client application container

Change the URL in the browser to the external IP of the `language-frontend` container using the following format: `http://<external-ip>/helloworld`. The English culture text of `helloworld` is predicted as `English`.

Clean up resources

When you are done with the cluster, delete the Azure resource group.

```
Azure CLI
az group delete --name cogserv-container-rg
```

Related information

- [kubectl for Docker Users ↗](#)

Next steps

[Cognitive Services Containers](#)

Use Docker Compose to deploy multiple containers

Article • 02/11/2022 • 5 minutes to read

This article shows you how to deploy multiple Azure Cognitive Services containers. Specifically, you'll learn how to use Docker Compose to orchestrate multiple Docker container images.

Docker Compose [↗](#) is a tool for defining and running multi-container Docker applications. In Compose, you use a YAML file to configure your application's services. Then, you create and start all the services from your configuration by running a single command.

It can be useful to orchestrate multiple container images on a single host computer. In this article, we'll pull together the Read and Form Recognizer containers.

Prerequisites

This procedure requires several tools that must be installed and run locally:

- An Azure subscription. If you don't have one, create a [free account](#) [↗](#) before you begin.
- [Docker Engine](#) [↗](#). Confirm that the Docker CLI works in a console window.
- An Azure resource with the correct pricing tier. Only the following pricing tiers work with this container:
 - **Computer Vision** resource with F0 or Standard pricing tier only.
 - **Form Recognizer** resource with F0 or Standard pricing tier only.
 - **Cognitive Services** resource with the S0 pricing tier.
- If you're using a gated preview container, You will need to complete the [online request form](#) [↗](#) to use it.

Docker Compose file

The YAML file defines all the services to be deployed. These services rely on either a [DockerFile](#) or an existing container image. In this case, we'll use two preview images.

Copy and paste the following YAML file, and save it as *docker-compose.yaml*. Provide the appropriate **apikey**, **billing**, and **EndpointUri** values in the file.

[YAML](#)

```

version: '3.7'
services:
  forms:
    image: "mcr.microsoft.com/azure-cognitive-services/form-
recognizer/layout"
    environment:
      eula: accept
      billing: # < Your form recognizer billing URL >
      apikey: # < Your form recognizer API key >
      FormRecognizer__ComputerVisionApiKey: # < Your form recognizer API
key >
      FormRecognizer__ComputerVisionEndpointUri: # < Your form recognizer
URI >
    volumes:
      - type: bind
        source: E:\publicpreview\output
        target: /output
      - type: bind
        source: E:\publicpreview\input
        target: /input
    ports:
      - "5010:5000"

  ocr:
    image: "mcr.microsoft.com/azure-cognitive-services/vision/read:3.1-
preview"
    environment:
      eula: accept
      apikey: # < Your computer vision API key >
      billing: # < Your computer vision billing URL >
    ports:
      - "5021:5000"

```

Important

Create the directories on the host machine that are specified under the **volumes** node. This approach is required because the directories must exist before you try to mount an image by using volume bindings.

Start the configured Docker Compose services

A Docker Compose file enables the management of all the stages in a defined service's life cycle: starting, stopping, and rebuilding services; viewing the service status; and log streaming. Open a command-line interface from the project directory (where the docker-compose.yaml file is located).

Note

To avoid errors, make sure that the host machine correctly shares drives with Docker Engine. For example, if `E:\publicpreview` is used as a directory in the `docker-compose.yaml` file, share drive **E** with Docker.

From the command-line interface, execute the following command to start (or restart) all the services defined in the `docker-compose.yaml` file:

```
Console
```

```
docker-compose up
```

The first time Docker executes the `docker-compose up` command by using this configuration, it pulls the images configured under the `services` node and then downloads and mounts them:

```
Console
```

```
Pulling forms (mcr.microsoft.com/azure-cognitive-services/form-
recognizer/layout:)...  
latest: Pulling from azure-cognitive-services/form-recognizer/layout  
743f2d6c1f65: Pull complete  
72befba99561: Pull complete  
2a40b9192d02: Pull complete  
c7715c9d5c33: Pull complete  
f0b33959f1c4: Pull complete  
b8ab86c6ab26: Pull complete  
41940c21ed3c: Pull complete  
e3d37dd258d4: Pull complete  
cdb5eb761109: Pull complete  
fd93b5f95865: Pull complete  
ef41dcbe5857: Pull complete  
4d05c86a4178: Pull complete  
34e811d37201: Pull complete  
Pulling ocr (mcr.microsoft.com/azure-cognitive-services/vision/read:3.1-
preview:)...  
latest: Pulling from /azure-cognitive-services/vision/read:3.1-preview  
f476d66f5408: Already exists  
8882c27f669e: Already exists  
d9af21273955: Already exists  
f5029279ec12: Already exists  
1a578849dcd1: Pull complete  
45064b1ab0bf: Download complete  
4bb846705268: Downloading [=====>]  
] 187.1MB/222.8MB  
c56511552241: Waiting  
e91d2aa0f1ad: Downloading [=====>]  
] 162.2MB/176.1MB
```

After the images are downloaded, the image services are started:

Console

```
Starting docker_ocr_1 ... done
Starting docker_forms_1 ... doneAttaching to docker_ocr_1,
docker_forms_1forms_1 | forms_1 | forms_1 | Notice: This Preview is made
available to you on the condition that you agree to the Supplemental Terms
of Use for Microsoft Azure Previews [https://go.microsoft.com/fwlink/?linkid=2018815], which supplement your agreement
[https://go.microsoft.com/fwlink/?linkid=2018657] governing your use of
Azure. If you do not have an existing agreement governing your use of Azure,
you agree that your agreement governing use of Azure is the Microsoft Online
Subscription Agreement [https://go.microsoft.com/fwlink/?linkid=2018755]
(which incorporates the Online Services Terms
[https://go.microsoft.com/fwlink/?linkid=2018760]). By using the Preview you
agree to these terms.

forms_1 |
forms_1 |
forms_1 | Using '/input' for reading models and other read-only data.
forms_1 | Using '/output/forms/812d811d1bcc' for writing logs and other
output data.
forms_1 | Logging to console.
forms_1 | Submitting metering to
'https://westus2.api.cognitive.microsoft.com/'.
forms_1 | WARNING: No access control enabled!
forms_1 | warn: Microsoft.AspNetCore.Server.Kestrel[0]
forms_1 |       Overriding address(es) 'http://+:80'. Binding to endpoints
defined in UseKestrel() instead.
forms_1 | Hosting environment: Production
forms_1 | Content root path: /app/forms
forms_1 | Now listening on: http://0.0.0.0:5000
forms_1 | Application started. Press Ctrl+C to shut down.

ocr_1 |
ocr_1 |
ocr_1 | Notice: This Preview is made available to you on the condition
that you agree to the Supplemental Terms of Use for Microsoft Azure Previews
[https://go.microsoft.com/fwlink/?linkid=2018815], which supplement your
agreement [https://go.microsoft.com/fwlink/?linkid=2018657] governing your
use of Azure. If you do not have an existing agreement governing your use of
Azure, you agree that your agreement governing use of Azure is the Microsoft
Online Subscription Agreement [https://go.microsoft.com/fwlink/?linkid=2018755]
(which incorporates the Online Services Terms
[https://go.microsoft.com/fwlink/?linkid=2018760]). By using the Preview you
agree to these terms.

ocr_1 |
ocr_1 |
ocr_1 | Logging to console.
ocr_1 | Submitting metering to
'https://westcentralus.api.cognitive.microsoft.com/'.
ocr_1 | WARNING: No access control enabled!
ocr_1 | Hosting environment: Production
```

```
ocr_1 | Content root path: /
ocr_1 | Now listening on: http://0.0.0.0:5000
ocr_1 | Application started. Press Ctrl+C to shut down.
```

Verify the service availability

Tip

You can use the `docker images` command to list your downloaded container images. For example, the following command lists the ID, repository, and tag of each downloaded container image, formatted as a table:

```
docker images --format "table {{.ID}}\t{{.Repository}}\t{{.Tag}}"

IMAGE ID          REPOSITORY          TAG
<image-id>      <repository-path/name>  <tag-name>
```

Here's some example output:

```
IMAGE ID          REPOSITORY          TAG
TAG
2ce533f88e80      mcr.microsoft.com/azure-cognitive-services/form-
recognizer/layout    latest
4be104c126c5      mcr.microsoft.com/azure-cognitive-
services/vision/read:3.1-preview    latest
```

Test containers

Open a browser on the host machine and go to `localhost` by using the specified port from the `docker-compose.yaml` file, such as `http://localhost:5021/swagger/index.html`. For example, you could use the **Try It** feature in the API to test the Form Recognizer endpoint. Both containers swagger pages should be available and testable.

The screenshot shows the Swagger UI interface for the Cognitive Service Form Recognizer API V1 (Preview). At the top, there's a header bar with the URL "localhost:5010/swagger/index.html", the "swagger" logo, and a dropdown menu "Select a spec" set to "Cognitive Service Form Recognizer API V1 (Preview)". Below the header, the main title is "Cognitive Service Form Recognizer API V1 (Preview) ⓘ". A link to "/swagger/v1/swagger.json" is provided. A descriptive text states: "Extracts information from forms and images into structured data based on a model created by a set of representative training forms." The interface is divided into sections:

- Form**:
 - POST** /formrecognizer/v1.0-preview/custom/train Train a model using a set of documents.
 - GET** /formrecognizer/v1.0-preview/custom/models/{id}/keys Get keys of a model.
 - GET** /formrecognizer/v1.0-preview/custom/models/{id} Get information about a model.
 - DELETE** /formrecognizer/v1.0-preview/custom/models/{id} Delete a model and all associated pre-processing data.
 - GET** /formrecognizer/v1.0-preview/custom/models Get all models.
 - POST** /formrecognizer/v1.0-preview/custom/models/{id}/analyze Analyze a document to extract key-value pairs and table information.
- Status**:
 - GET** /status

Next steps

[Cognitive Services containers](#)

Tutorial: Create a container image for deployment to Azure Container Instances

Article • 01/10/2023 • 3 minutes to read

Azure Container Instances enables deployment of Docker containers onto Azure infrastructure without provisioning any virtual machines or adopting a higher-level service. In this tutorial, you package a small Node.js web application into a container image that can be run using Azure Container Instances.

In this article, part one of the series, you:

- ✓ Clone application source code from GitHub
- ✓ Create a container image from application source
- ✓ Test the image in a local Docker environment

In tutorial parts two and three, you upload your image to Azure Container Registry, and then deploy it to Azure Container Instances.

Before you begin

You must satisfy the following requirements to complete this tutorial:

Azure CLI: You must have Azure CLI version 2.0.29 or later installed on your local computer. Run `az --version` to find the version. If you need to install or upgrade, see [Install the Azure CLI](#).

Docker: This tutorial assumes a basic understanding of core Docker concepts like containers, container images, and basic `docker` commands. For a primer on Docker and container basics, see the [Docker overview](#).

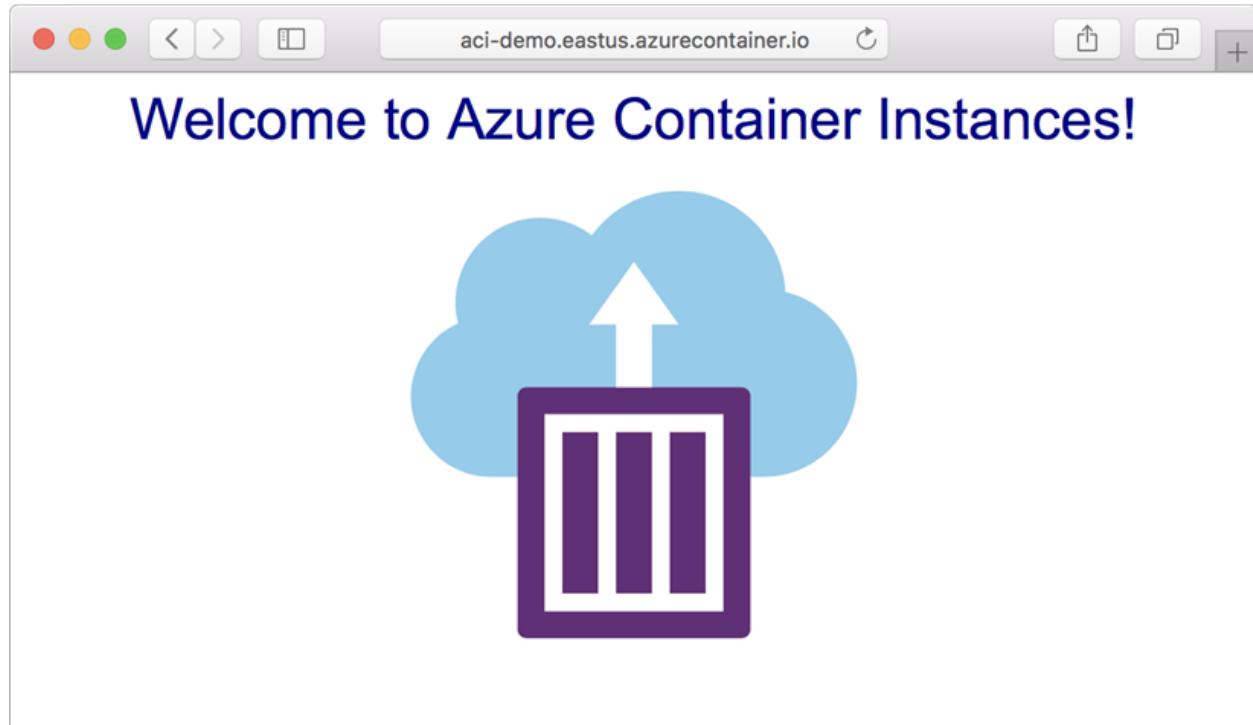
Docker: To complete this tutorial, you need Docker installed locally. Docker provides packages that configure the Docker environment on [macOS](#), [Windows](#), and [Linux](#).

ⓘ Important

Because the Azure Cloud shell does not include the Docker daemon, you *must* install both the Azure CLI and Docker Engine on your *local computer* to complete this tutorial. You cannot use the Azure Cloud Shell for this tutorial.

Get application code

The sample application in this tutorial is a simple web app built in [Node.js](#). The application serves a static HTML page, and looks similar to the following screenshot:



Use Git to clone the sample application's repository:

```
Bash
```

```
git clone https://github.com/Azure-Samples/aci-helloworld.git
```

You can also [download the ZIP archive](#) from GitHub directly.

Build the container image

The Dockerfile in the sample application shows how the container is built. It starts from an [official Node.js image](#) based on [Alpine Linux](#), a small distribution that is well suited for use with containers. It then copies the application files into the container, installs dependencies using the Node Package Manager, and finally, starts the application.

```
Dockerfile
```

```
FROM node:8.9.3-alpine
RUN mkdir -p /usr/src/app
COPY ./app/* /usr/src/app/
WORKDIR /usr/src/app
```

```
RUN npm install  
CMD node /usr/src/app/index.js
```

Use the [docker build](#) command to create the container image and tag it as *aci-tutorial-app*:

Bash

```
docker build ./aci-helloworld -t aci-tutorial-app
```

Output from the [docker build](#) command is similar to the following (truncated for readability):

Console

```
$ docker build ./aci-helloworld -t aci-tutorial-app  
Sending build context to Docker daemon 119.3kB  
Step 1/6 : FROM node:8.9.3-alpine  
8.9.3-alpine: Pulling from library/node  
88286f41530e: Pull complete  
84f3a4bf8410: Pull complete  
d0d9b2214720: Pull complete  
Digest:  
sha256:c73277ccc763752b42bb2400d1aaecb4e3d32e3a9dbedd0e49885c71bea07354  
Status: Downloaded newer image for node:8.9.3-alpine  
--> 90f5ee24bee2  
...  
Step 6/6 : CMD node /usr/src/app/index.js  
--> Running in f4a1ea099eec  
--> 6edad76d09e9  
Removing intermediate container f4a1ea099eec  
Successfully built 6edad76d09e9  
Successfully tagged aci-tutorial-app:latest
```

Use the [docker images](#) command to see the built image:

Bash

```
docker images
```

Your newly built image should appear in the list:

Console

```
$ docker images  
REPOSITORY          TAG      IMAGE ID      CREATED        SIZE  
aci-tutorial-app   latest   5c745774dfa9   39 seconds ago  68.1 MB
```

Run the container locally

Before you deploy the container to Azure Container Instances, use [docker run](#) to run it locally and confirm that it works. The `-d` switch lets the container run in the background, while `-p` allows you to map an arbitrary port on your computer to port 80 in the container.

Bash

```
docker run -d -p 8080:80 aci-tutorial-app
```

Output from the `docker run` command displays the running container's ID if the command was successful:

Console

```
$ docker run -d -p 8080:80 aci-tutorial-app  
a2e3e4435db58ab0c664ce521854c2e1a1bda88c9cf2fcff46aedef48df86cccf
```

Now, navigate to `http://localhost:8080` in your browser to confirm that the container is running. You should see a web page similar to the following:



Next steps

In this tutorial, you created a container image that can be deployed in Azure Container Instances, and verified that it runs locally. So far, you've done the following:

- ✓ Cloned the application source from GitHub

- ✓ Created a container image from the application source
- ✓ Tested the container locally

Advance to the next tutorial in the series to learn about storing your container image in Azure Container Registry:

[Push image to Azure Container Registry](#)

Additional resources

Documentation

[Tutorial - Deploy container application to container instance - Azure Container Instances](#)

Azure Container Instances tutorial part 3 of 3 - Deploy container application to Azure Container Instances

[Tutorial - Prepare container registry to deploy image - Azure Container Instances](#)

Azure Container Instances tutorial part 2 of 3 - Prepare an Azure container registry and push an image

[Quickstart - Deploy Docker container to container instance - Docker CLI - Azure Container Instances](#)

In this quickstart, you use the Docker CLI to quickly deploy a containerized web app that runs in an isolated Azure container instance

[Quickstart - Deploy Docker container to container instance - Azure CLI - Azure Container Instances](#)

In this quickstart, you use the Azure CLI to quickly deploy a containerized web app that runs in an isolated Azure container instance

[Quickstart - Deploy Docker container to container instance - Portal - Azure Container Instances](#)

In this quickstart, you use the Azure portal to quickly deploy a containerized web app that runs in an isolated Azure container instance

[Tutorial - Deploy multi-container group - template - Azure Container Instances](#)

In this tutorial, you learn how to deploy a container group with multiple containers in Azure Container Instances by using an Azure Resource Manager template with the Azure CLI.

[Tutorial - Deploy multi-container group - YAML - Azure Container Instances](#)

In this tutorial, you learn how to deploy a container group with multiple containers in Azure Container Instances by using a YAML file with the Azure CLI.

[Update container group - Azure Container Instances](#)

Learn how to update running containers in your Azure Container Instances container groups.

[Show 5 more](#)

Quickstart: Create a private container registry using the Azure CLI

Article • 10/12/2022 • 4 minutes to read

Azure Container Registry is a private registry service for building, storing, and managing container images and related artifacts. In this quickstart, you create an Azure container registry instance with the Azure CLI. Then, use Docker commands to push a container image into the registry, and finally pull and run the image from your registry.

This quickstart requires that you are running the Azure CLI (version 2.0.55 or later recommended). Run `az --version` to find the version. If you need to install or upgrade, see [Install Azure CLI](#).

You must also have Docker installed locally. Docker provides packages that easily configure Docker on any [macOS](#), [Windows](#), or [Linux](#) system.

Because the Azure Cloud Shell doesn't include all required Docker components (the `dockerd` daemon), you can't use the Cloud Shell for this quickstart.

Create a resource group

Create a resource group with the `az group create` command. An Azure resource group is a logical container into which Azure resources are deployed and managed.

The following example creates a resource group named *myResourceGroup* in the *eastus* location.

Azure CLI

```
az group create --name myResourceGroup --location eastus
```

Create a container registry

In this quickstart you create a *Basic* registry, which is a cost-optimized option for developers learning about Azure Container Registry. For details on available service tiers, see [Container registry service tiers](#).

Create an ACR instance using the `az acr create` command. The registry name must be unique within Azure, and contain 5-50 lowercase alphanumeric characters. In the following example, *mycontainerregistry* is used. Update this to a unique value.

Azure CLI

```
az acr create --resource-group myResourceGroup \
--name mycontainerregistry --sku Basic
```

When the registry is created, the output is similar to the following:

JSON

```
{
  "adminUserEnabled": false,
  "creationDate": "2019-01-08T22:32:13.175925+00:00",
  "id": "/subscriptions/00000000-0000-0000-0000-
0000000000/resourceGroups/myResourceGroup/providers/Microsoft.ContainerReg
istry/registries/mycontainerregistry",
  "location": "eastus",
  "loginServer": "mycontainerregistry.azurecr.io",
  "name": "mycontainerregistry",
  "provisioningState": "Succeeded",
  "resourceGroup": "myResourceGroup",
  "sku": {
    "name": "Basic",
    "tier": "Basic"
  },
  "status": null,
  "storageAccount": null,
  "tags": {},
  "type": "Microsoft.ContainerRegistry/registries"
}
```

Take note of `loginServer` in the output, which is the fully qualified registry name (all lowercase). Throughout the rest of this quickstart `<registry-name>` is a placeholder for the container registry name, and `<login-server>` is a placeholder for the registry's login server name.

💡 Tip

In this quickstart, you create a *Basic* registry, which is a cost-optimized option for developers learning about Azure Container Registry. Choose other tiers for increased storage and image throughput, and capabilities such as connection using a **private endpoint**. For details on available service tiers (SKUs), see [Container registry service tiers](#).

Log in to registry

Before pushing and pulling container images, you must log in to the registry. To do so, use the `az acr login` command. Specify only the registry resource name when logging in with the Azure CLI. Don't use the fully qualified login server name.

```
Azure CLI
```

```
az acr login --name <registry-name>
```

Example:

```
Azure CLI
```

```
az acr login --name mycontainerregistry
```

The command returns a `Login Succeeded` message once completed.

Push image to registry

To push an image to an Azure Container registry, you must first have an image. If you don't yet have any local container images, run the following `docker pull` command to pull an existing public image. For this example, pull the `hello-world` image from Microsoft Container Registry.

```
docker pull mcr.microsoft.com/hello-world
```

Before you can push an image to your registry, you must tag it with the fully qualified name of your registry login server. The login server name is in the format `<registry-name>.azurecr.io` (must be all lowercase), for example, `mycontainerregistry.azurecr.io`.

Tag the image using the `docker tag` command. Replace `<login-server>` with the login server name of your ACR instance.

```
docker tag mcr.microsoft.com/hello-world <login-server>/hello-world:v1
```

Example:

```
docker tag mcr.microsoft.com/hello-world  
mycontainerregistry.azurecr.io/hello-world:v1
```

Finally, use [docker push](#) to push the image to the registry instance. Replace <login-server> with the login server name of your registry instance. This example creates the **hello-world** repository, containing the `hello-world:v1` image.

```
docker push <login-server>/hello-world:v1
```

After pushing the image to your container registry, remove the `hello-world:v1` image from your local Docker environment. (Note that this [docker rmi](#) command does not remove the image from the **hello-world** repository in your Azure container registry.)

```
docker rmi <login-server>/hello-world:v1
```

List container images

The following example lists the repositories in your registry:

Azure CLI

```
az acr repository list --name <registry-name> --output table
```

Output:

```
Result  
-----  
hello-world
```

The following example lists the tags on the **hello-world** repository.

Azure CLI

```
az acr repository show-tags --name <registry-name> --repository hello-world  
--output table
```

Output:

```
Result
```

```
-----
```

```
v1
```

Run image from registry

Now, you can pull and run the `hello-world:v1` container image from your container registry by using [docker run ↗](#):

```
docker run <login-server>/hello-world:v1
```

Example output:

```
Unable to find image 'mycontainerregistry.azurecr.io/hello-world:v1' locally
v1: Pulling from hello-world
Digest:
sha256:662dd8e65ef7ccf13f417962c2f77567d3b132f12c95909de6c85ac3c326a345
Status: Downloaded newer image for mycontainerregistry.azurecr.io/hello-
world:v1
```

```
Hello from Docker!
```

```
This message shows that your installation appears to be working correctly.
```

```
[...]
```

Clean up resources

When no longer needed, you can use the [az group delete](#) command to remove the resource group, the container registry, and the container images stored there.

```
Azure CLI
```

```
az group delete --name myResourceGroup
```

Next steps

In this quickstart, you created an Azure Container Registry with the Azure CLI, pushed a container image to the registry, and pulled and ran the image from the registry. Continue to the Azure Container Registry tutorials for a deeper look at ACR.

[Azure Container Registry tutorials](#)

[Azure Container Registry Tasks tutorials](#)

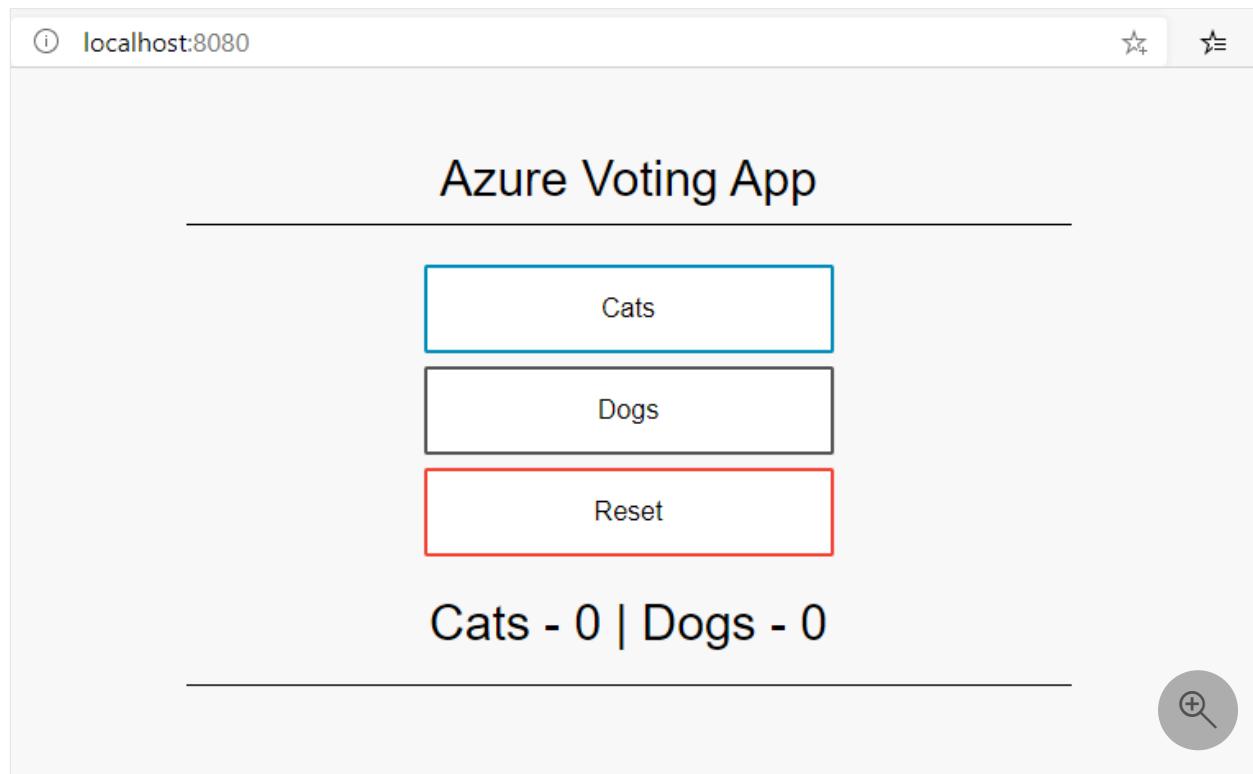
Tutorial: Prepare an application for Azure Kubernetes Service (AKS)

Article • 12/06/2022 • 3 minutes to read

In this tutorial, part one of seven, you prepare a multi-container application to use in Kubernetes. You use existing development tools like Docker Compose to locally build and test the application. You learn how to:

- ✓ Clone a sample application source from GitHub
- ✓ Create a container image from the sample application source
- ✓ Test the multi-container application in a local Docker environment

Once completed, the following application runs in your local development environment:



In later tutorials, you upload the container image to an Azure Container Registry (ACR), and then deploy it into an AKS cluster.

Before you begin

This tutorial assumes a basic understanding of core Docker concepts such as containers, container images, and `docker` commands. For a primer on container basics, see [Get started with Docker ↗](#).

To complete this tutorial, you need a local Docker development environment running Linux containers. Docker provides packages that configure Docker on a [Mac](#), [Windows](#), or [Linux](#) system.

ⓘ Note

Azure Cloud Shell does not include the Docker components required to complete every step in these tutorials. Therefore, we recommend using a full Docker development environment.

Get application code

The [sample application](#) used in this tutorial is a basic voting app consisting of a front-end web component and a back-end Redis instance. The web component is packaged into a custom container image. The Redis instance uses an unmodified image from Docker Hub.

Use [git](#) to clone the sample application to your development environment.

Console

```
git clone https://github.com/Azure-Samples/azure-voting-app-redis.git
```

Change into the cloned directory.

Console

```
cd azure-voting-app-redis
```

The directory contains the application source code, a pre-created Docker compose file, and a Kubernetes manifest file. These files are used throughout the tutorial set. The contents and structure of the directory are as follows:

Output

```
azure-voting-app-redis
|   azure-vote-all-in-one-redis.yaml
|   docker-compose.yaml
|   LICENSE
|   README.md

|___azure-vote
    |   app_init.supervisord.conf
    |   Dockerfile
```

```
Dockerfile-for-app-service
sshd_config

azure-vote
    config_file.cfg
    main.py

static
    default.css

templates
    index.html

jenkins-tutorial
    config-jenkins.sh
    deploy-jenkins-vm.sh
```

Create container images

[Docker Compose](#) can be used to automate building container images and the deployment of multi-container applications.

The following command uses the sample `docker-compose.yaml` file to create the container image, download the Redis image, and start the application.

```
Console
docker-compose up -d
```

When completed, use the [docker images](#) command to see the created images. Three images are downloaded or created. The *azure-vote-front* image contains the front-end application and uses the *nginx-flask* image as a base. The *redis* image is used to start a Redis instance.

```
$ docker images

REPOSITORY          TAG      IMAGE ID
CREATED            SIZE
mcr.microsoft.com/azuredocs/azure-vote-front   v1
84b41c268ad9      9 seconds ago    944MB
mcr.microsoft.com/oss/bitnami/redis           6.0.8
3a54a920bb6c      2 days ago     103MB
tiangolo/uwsgi-nginx-flask                   python3.6
a16ce562e863      6 weeks ago    944MB
```

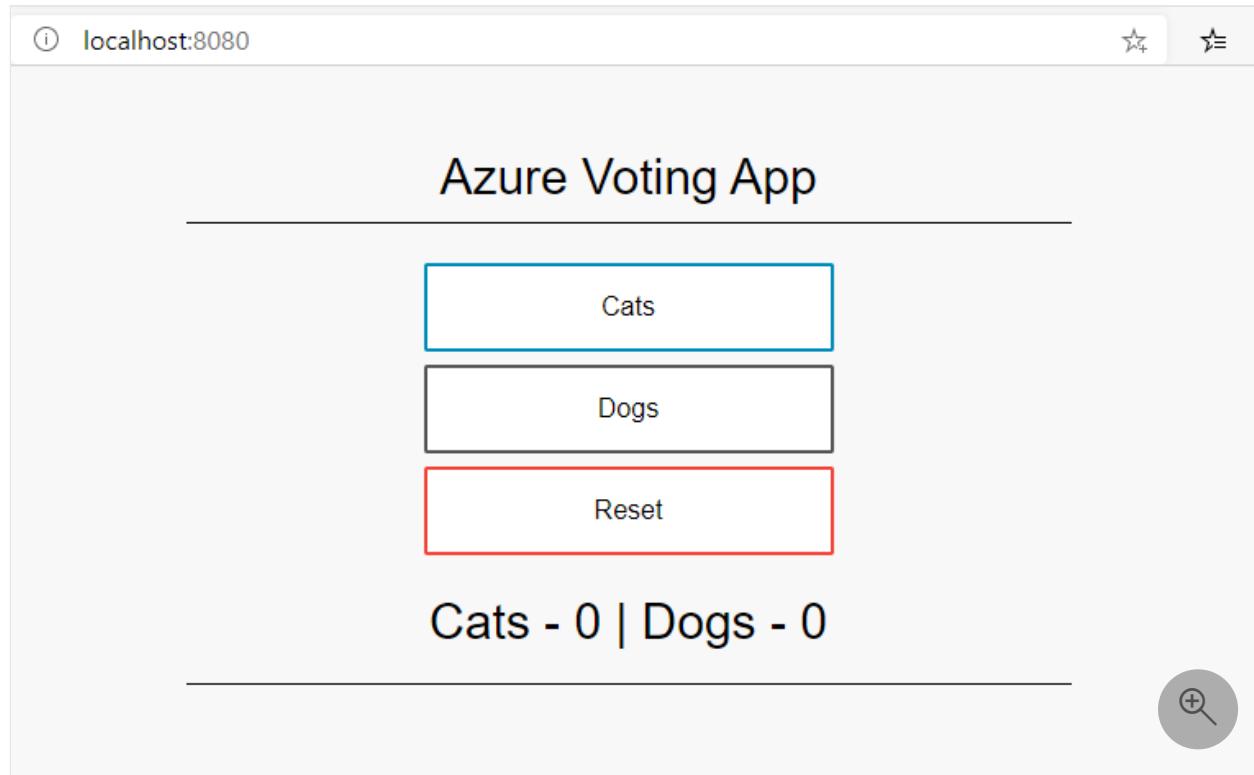
Run the [docker ps](#) command to see the running containers.

```
$ docker ps
```

CONTAINER ID	IMAGE	CREATED	STATUS	PORTS
COMMAND	NAMES			
d10e5244f237	mcr.microsoft.com/azuredocs/azure-vote-front:v1	"/entrypoint.sh /sta..."	3 minutes ago	Up 3 minutes
0.0.0.0:8080->80/tcp	azure-vote-front			443/tcp,
21574cb38c1f	mcr.microsoft.com/oss/bitnami/redis:6.0.8	"./opt/bitnami/script..."	3 minutes ago	Up 3 minutes
0.0.0.0:6379->6379/tcp	azure-vote-back			

Test application locally

To see your running application, navigate to <http://localhost:8080> in a local web browser. The sample application loads, as shown in the following example:



Clean up resources

Now that the application's functionality has been validated, the running containers can be stopped and removed. ***Do not delete the container images*** - in the next tutorial, you'll upload the *azure-vote-front* image to an ACR instance.

To stop and remove the container instances and resources, use the [docker-compose down](#) command.

Console

```
docker-compose down
```

When the local application has been removed, you have a Docker image that contains the Azure Vote application, *azure-vote-front*, to use in the next tutorial.

Next steps

In this tutorial, you created a sample application, created container images for the application, and then tested the application. You learned how to:

- Clone a sample application source from GitHub
- Create a container image from the sample application source
- Test the multi-container application in a local Docker environment

In the next tutorial, you'll learn how to store container images in an ACR.

[Push images to Azure Container Registry](#)

Azure Cognitive Services security

Article • 01/11/2023 • 5 minutes to read

Security should be considered a top priority in the development of all applications, and with the growth of artificial intelligence enabled applications, security is even more important. This article outlines various security features available for Azure Cognitive Services. Each feature addresses a specific liability, so multiple features can be used in the same workflow.

For a comprehensive list of Azure service security recommendations see the [Cognitive Services security baseline](#) article.

Security features

Feature	Description
Transport Layer Security (TLS)	All of the Cognitive Services endpoints exposed over HTTP enforce the TLS 1.2 protocol. With an enforced security protocol, consumers attempting to call a Cognitive Services endpoint should follow these guidelines: <ul style="list-style-type: none">- The client operating system (OS) needs to support TLS 1.2.- The language (and platform) used to make the HTTP call need to specify TLS 1.2 as part of the request. Depending on the language and platform, specifying TLS is done either implicitly or explicitly.- For .NET users, consider the Transport Layer Security best practices.
Authentication options	Authentication is the act of verifying a user's identity. Authorization, by contrast, is the specification of access rights and privileges to resources for a given identity. An identity is a collection of information about a principal , and a principal can be either an individual user or a service. <p>By default, you authenticate your own calls to Cognitive Services using the subscription keys provided; this is the simplest method but not the most secure. The most secure authentication method is to use managed roles in Azure Active Directory. To learn about this and other authentication options, see Authenticate requests to Cognitive Services.</p>
Key rotation	Each Cognitive Services resource has two API keys to enable secret rotation. This is a security precaution that lets you regularly change the keys that can access your service, protecting the privacy of your service in the event that a key gets leaked. To learn about this and other authentication options, see Rotate keys .

Feature	Description
Environment variables	<p>Environment variables are name-value pairs that are stored within a specific development environment. You can store your credentials in this way as a more secure alternative to using hardcoded values in your code. However, if your environment is compromised, the environment variables are compromised as well, so this is not the most secure approach.</p> <p>For instructions on how to use environment variables in your code, see the Environment variables guide.</p>
Customer-managed keys (CMK)	<p>This feature is for services that store customer data at rest (longer than 48 hours). While this data is already double-encrypted on Azure servers, users can get extra security by adding another layer of encryption, with keys they manage themselves. You can link your service to Azure Key Vault and manage your data encryption keys there.</p> <p>You need special approval to get the E0 SKU for your service, which enables CMK. Within 3-5 business days after you submit the request form, you'll get an update on the status of your request. Depending on demand, you may be placed in a queue and approved as space becomes available. Once you're approved for using the E0 SKU, you'll need to create a new resource from the Azure portal and select E0 as the Pricing Tier. You won't be able to upgrade from F0 to the new E0 SKU.</p> <p>Only some services can use CMK; look for your service on the Customer-managed keys page.</p>
Virtual networks	<p>Virtual networks allow you to specify which endpoints can make API calls to your resource. The Azure service will reject API calls from devices outside of your network. You can set a formula-based definition of the allowed network, or you can define an exhaustive list of endpoints to allow. This is another layer of security that can be used in combination with others.</p>
Data loss prevention	<p>The data loss prevention feature lets an administrator decide what types of URIs their Azure resource can take as inputs (for those API calls that take URLs as input). This can be done to prevent the possible exfiltration of sensitive company data: If a company stores sensitive information (such as a customer's private data) in URL parameters, a bad actor inside that company could submit the sensitive URLs to an Azure service, which surfaces that data outside the company. Data loss prevention lets you configure the service to reject certain URL forms on arrival.</p>

Feature	Description
Customer Lockbox	<p>The Customer Lockbox feature provides an interface for customers to review and approve or reject data access requests. It's used in cases where a Microsoft engineer needs to access customer data during a support request. For information on how Customer Lockbox requests are initiated, tracked, and stored for later reviews and audits, see the Customer Lockbox guide.</p> <p>Customer Lockbox is available for the following services:</p> <ul style="list-style-type: none"> - Azure OpenAI - Translator - Conversational language understanding - Custom text classification - Custom named entity recognition - Orchestration workflow <p>Similarly, the following services offer the E0 tier (accessed by this request form), and Microsoft engineers will not access any customer data from that tier:</p> <ul style="list-style-type: none"> - Language Understanding - Face - Content Moderator - Personalizer
Bring your own storage (BYOS)	<p>The Speech service doesn't currently support Customer Lockbox. However, you can arrange for your service-specific data to be stored in your own storage resource using bring-your-own-storage (BYOS). BYOS allows you to achieve similar data controls to Customer Lockbox. Keep in mind that Speech service data stays and is processed in the Azure region where the Speech resource was created. This applies to any data at rest and data in transit. For customization features like Custom Speech and Custom Voice, all customer data is transferred, stored, and processed in the same region where the Speech service resource and BYOS resource (if used) reside.</p> <p>To use BYOS with Speech, follow the Speech encryption of data at rest guide.</p> <p>Microsoft does not use customer data to improve its Speech models. Additionally, if endpoint logging is disabled and no customizations are used, then no customer data is stored by Speech.</p>

Next steps

- Explore [Cognitive Services](#) and choose a service to get started.

Additional resources

Documentation

[Frequently asked questions - Computer Vision - Azure Cognitive Services](#)

Get answers to frequently asked questions about the Computer Vision Service in Azure Cognitive Services.

[Use the autoscale feature - Azure Cognitive Services](#)

Learn how to use the autoscale feature for Cognitive Services to dynamically adjust the rate limit of your service.

[Plan to manage costs for Azure Cognitive Services - Azure Cognitive Services](#)

Learn how to plan for and manage costs for Azure Cognitive Services by using cost analysis in the Azure portal.

[Quickstart - Custom named entity recognition \(NER\) - Azure Cognitive Services](#)

Use this article to quickly get started using Custom Named Entity Recognition (NER) with Language Studio

[Data limits for Language service features - Azure Cognitive Services](#)

Data and service limitations for Azure Cognitive Service for Language features.

[Custom text classification limits - Azure Cognitive Services](#)

Learn about the data and rate limits when using custom text classification.

[Post-call transcription and analytics quickstart - Speech service - Azure Cognitive Services](#)

In this quickstart, you perform sentiment analysis and conversation summarization of call center transcriptions.

[Using Azure resources in custom NER - Azure Cognitive Services](#)

Learn about the steps for using Azure resources with custom NER.

[Show 5 more](#)

Authenticate requests to Azure Cognitive Services

Article • 09/02/2022 • 8 minutes to read

Each request to an Azure Cognitive Service must include an authentication header. This header passes along a subscription key or authentication token, which is used to validate your subscription for a service or group of services. In this article, you'll learn about three ways to authenticate a request and the requirements for each.

- Authenticate with a [single-service](#) or [multi-service](#) subscription key
- Authenticate with a [token](#)
- Authenticate with [Azure Active Directory \(AAD\)](#)

Prerequisites

Before you make a request, you need an Azure account and an Azure Cognitive Services subscription. If you already have an account, go ahead and skip to the next section. If you don't have an account, we have a guide to get you set up in minutes: [Create a Cognitive Services account for Azure](#).

You can get your subscription key from the [Azure portal](#) after [creating your account](#) ↗.

Authentication headers

Let's quickly review the authentication headers available for use with Azure Cognitive Services.

Header	Description
Ocp-Apim-Subscription-Key	Use this header to authenticate with a subscription key for a specific service or a multi-service subscription key.
Ocp-Apim-Subscription-Region	This header is only required when using a multi-service subscription key with the Translator service . Use this header to specify the subscription region.
Authorization	Use this header if you are using an access token. The steps to perform a token exchange are detailed in the following sections. The value provided follows this format: <code>Bearer <TOKEN></code> .

Authenticate with a single-service subscription key

The first option is to authenticate a request with a subscription key for a specific service, like Translator. The keys are available in the Azure portal for each resource that you've created. To use a subscription key to authenticate a request, it must be passed along as the `Ocp-Apim-Subscription-Key` header.

These sample requests demonstrates how to use the `Ocp-Apim-Subscription-Key` header. Keep in mind, when using this sample you'll need to include a valid subscription key.

This is a sample call to the Bing Web Search API:

cURL

```
curl -X GET 'https://api.cognitive.microsoft.com/bing/v7.0/search?  
q=Welsch%20Pembroke%20Corgis' \  
-H 'Ocp-Apim-Subscription-Key: YOUR_SUBSCRIPTION_KEY' | json_pp
```

This is a sample call to the Translator service:

cURL

```
curl -X POST 'https://api.cognitive.microsofttranslator.com/translate?api-  
version=3.0&from=en&to=de' \  
-H 'Ocp-Apim-Subscription-Key: YOUR_SUBSCRIPTION_KEY' \  
-H 'Content-Type: application/json' \  
--data-binary '[{"text": "How much for the cup of coffee?" }]' | json_pp
```

The following video demonstrates using a Cognitive Services key.

Authenticate with a multi-service subscription key

⚠ Warning

At this time, the multi-service key doesn't support: QnA Maker, Immersive Reader, Personalizer, and Anomaly Detector.

This option also uses a subscription key to authenticate requests. The main difference is that a subscription key is not tied to a specific service, rather, a single key can be used to

authenticate requests for multiple Cognitive Services. See [Cognitive Services pricing](#) for information about regional availability, supported features, and pricing.

The subscription key is provided in each request as the `Ocp-Apim-Subscription-Key` header.



[↗](#)

Supported regions

When using the multi-service subscription key to make a request to `api.cognitive.microsoft.com`, you must include the region in the URL. For example: `westus.api.cognitive.microsoft.com`.

When using multi-service subscription key with the Translator service, you must specify the subscription region with the `Ocp-Apim-Subscription-Region` header.

Multi-service authentication is supported in these regions:

- `australiaeast`
- `brazilsouth`
- `canadacentral`
- `centralindia`
- `eastasia`
- `eastus`
- `japaneast`
- `northeurope`
- `southcentralus`

- southeastasia
- uksouth
- westcentralus
- westeurope
- westus
- westus2
- francecentral
- koreacentral
- northcentralus
- southafricanorth
- uaenorth
- switzerlandnorth

Sample requests

This is a sample call to the Bing Web Search API:

cURL

```
curl -X GET 'https://YOUR-
REGION.api.cognitive.microsoft.com/bing/v7.0/search?
q=Welsch%20Pembroke%20Corgis' \
-H 'Ocp-Apim-Subscription-Key: YOUR_SUBSCRIPTION_KEY' | json_pp
```

This is a sample call to the Translator service:

cURL

```
curl -X POST 'https://api.cognitive.microsofttranslator.com/translate?api-
version=3.0&from=en&to=de' \
-H 'Ocp-Apim-Subscription-Key: YOUR_SUBSCRIPTION_KEY' \
-H 'Ocp-Apim-Subscription-Region: YOUR_SUBSCRIPTION_REGION' \
-H 'Content-Type: application/json' \
--data-raw '[{"text": "How much for the cup of coffee?" }]' | json_pp
```

Authenticate with an access token

Some Azure Cognitive Services accept, and in some cases require, an access token. Currently, these services support access tokens:

- Text Translation API
- Speech Services: Speech-to-text API

- Speech Services: Text-to-speech API

ⓘ Note

QnA Maker also uses the Authorization header, but requires an endpoint key. For more information, see [QnA Maker: Get answer from knowledge base](#).

⚠ Warning

The services that support access tokens may change over time, please check the API reference for a service before using this authentication method.

Both single service and multi-service subscription keys can be exchanged for authentication tokens. Authentication tokens are valid for 10 minutes. They're stored in JSON Web Token (JWT) format and can be queried programmatically using the [JWT libraries](#).

Access tokens are included in a request as the `Authorization` header. The token value provided must be preceded by `Bearer`, for example: `Bearer YOUR_AUTH_TOKEN`.

Sample requests

Use this URL to exchange a subscription key for an access token: `https://YOUR-REGION.api.cognitive.microsoft.com/sts/v1.0/issueToken`.

cURL

```
curl -v -X POST \
"https://YOUR-REGION.api.cognitive.microsoft.com/sts/v1.0/issueToken" \
-H "Content-type: application/x-www-form-urlencoded" \
-H "Content-length: 0" \
-H "Ocp-Apim-Subscription-Key: YOUR_SUBSCRIPTION_KEY"
```

These multi-service regions support token exchange:

- `australiaeast`
- `brazilsouth`
- `canadacentral`
- `centralindia`
- `eastasia`
- `eastus`

- `japaneast`
- `northeurope`
- `southcentralus`
- `southeastasia`
- `uksouth`
- `westcentralus`
- `westeurope`
- `westus`
- `westus2`

After you get an access token, you'll need to pass it in each request as the `Authorization` header. This is a sample call to the Translator service:

CURL

```
curl -X POST 'https://api.cognitive.microsofttranslator.com/translate?api-version=3.0&from=en&to=de' \
-H 'Authorization: Bearer YOUR_AUTH_TOKEN' \
-H 'Content-Type: application/json' \
--data-raw '[{"text": "How much for the cup of coffee?" }]' | json_pp
```

Authenticate with Azure Active Directory

ⓘ Important

Azure AD authentication always needs to be used together with custom subdomain name of your Azure resource. **Regional endpoints** do not support Azure AD authentication.

In the previous sections, we showed you how to authenticate against Azure Cognitive Services using a single-service or multi-service subscription key. While these keys provide a quick and easy path to start development, they fall short in more complex scenarios that require Azure [role-based access control \(Azure RBAC\)](#). Let's take a look at what's required to authenticate using Azure Active Directory (Azure AD).

In the following sections, you'll use either the Azure Cloud Shell environment or the Azure CLI to create a subdomain, assign roles, and obtain a bearer token to call the Azure Cognitive Services. If you get stuck, links are provided in each section with all available options for each command in Azure Cloud Shell/Azure CLI.

Create a resource with a custom subdomain

The first step is to create a custom subdomain. If you want to use an existing Cognitive Services resource which does not have custom subdomain name, follow the instructions in [Cognitive Services Custom Subdomains](#) to enable custom subdomain for your resource.

1. Start by opening the Azure Cloud Shell. Then [select a subscription](#):

```
PowerShell
```

```
Set-AzContext -SubscriptionName <SubscriptionName>
```

2. Next, [create a Cognitive Services resource](#) with a custom subdomain. The subdomain name needs to be globally unique and cannot include special characters, such as: ".", "!", ",", ".".

```
PowerShell
```

```
$account = New-AzCognitiveServicesAccount -ResourceGroupName  
<RESOURCE_GROUP_NAME> -name <ACCOUNT_NAME> -Type <ACCOUNT_TYPE> -  
SkuName <SUBSCRIPTION_TYPE> -Location <REGION> -CustomSubdomainName  
<UNIQUE_SUBDOMAIN>
```

3. If successful, the **Endpoint** should show the subdomain name unique to your resource.

Assign a role to a service principal

Now that you have a custom subdomain associated with your resource, you're going to need to assign a role to a service principal.

 **Note**

Keep in mind that Azure role assignments may take up to five minutes to propagate.

1. First, let's register an [Azure AD application](#).

```
PowerShell
```

```
$SecureStringPassword = ConvertTo-SecureString -String <YOUR_PASSWORD>  
-AsPlainText -Force
```

```
$app = New-AzADApplication -DisplayName <APP_DISPLAY_NAME> -  
IdentifierUris <APP_URIS> -Password $SecureStringPassword
```

You're going to need the **ApplicationId** in the next step.

2. Next, you need to [create a service principal](#) for the Azure AD application.

PowerShell

```
New-AzADServicePrincipal -ApplicationId <APPLICATION_ID>
```

 **Note**

If you register an application in the Azure portal, this step is completed for you.

3. The last step is to [assign the "Cognitive Services User" role](#) to the service principal (scoped to the resource). By assigning a role, you're granting service principal access to this resource. You can grant the same service principal access to multiple resources in your subscription.

 **Note**

The ObjectId of the service principal is used, not the ObjectId for the application. The ACCOUNT_ID will be the Azure resource Id of the Cognitive Services account you created. You can find Azure resource Id from "properties" of the resource in Azure portal.

Azure CLI

```
New-AzRoleAssignment -ObjectId <SERVICE_PRINCIPAL_OBJECTID> -Scope  
<ACCOUNT_ID> -RoleDefinitionName "Cognitive Services User"
```

Sample request

In this sample, a password is used to authenticate the service principal. The token provided is then used to call the Computer Vision API.

1. Get your **TenantId**:

PowerShell

```
$context=Get-AzContext  
$context.Tenant.Id
```

2. Get a token:

! Note

If you're using Azure Cloud Shell, the `SecureClientSecret` class isn't available.

PowerShell

PowerShell

```
$authContext = New-Object  
"Microsoft.IdentityModel.Clients.ActiveDirectory.AuthenticationCont  
ext" -ArgumentList "https://login.windows.net/<TENANT_ID>"  
$secureSecretObject = New-Object  
"Microsoft.IdentityModel.Clients.ActiveDirectory.SecureClientSecret  
" -ArgumentList $SecureStringPassword  
$clientCredential = New-Object  
"Microsoft.IdentityModel.Clients.ActiveDirectory.ClientCredential"  
-ArgumentList $app.ApplicationId, $secureSecretObject  
$token=$authContext.AcquireTokenAsync("https://cognitiveservices.azure.com/", $clientCredential).Result  
$token
```

3. Call the Computer Vision API:

PowerShell

```
$url = $account.Endpoint+"vision/v1.0/models"  
$result = Invoke-RestMethod -Uri $url -Method Get -Headers  
@{ "Authorization"=$token.CreateAuthorizationHeader() } -Verbose  
$result | ConvertTo-Json
```

Alternatively, the service principal can be authenticated with a certificate. Besides service principal, user principal is also supported by having permissions delegated through another Azure AD application. In this case, instead of passwords or certificates, users would be prompted for two-factor authentication when acquiring token.

Authorize access to managed identities

Cognitive Services support Azure Active Directory (Azure AD) authentication with [managed identities for Azure resources](#). Managed identities for Azure resources can authorize access to Cognitive Services resources using Azure AD credentials from applications running in Azure virtual machines (VMs), function apps, virtual machine scale sets, and other services. By using managed identities for Azure resources together with Azure AD authentication, you can avoid storing credentials with your applications that run in the cloud.

Enable managed identities on a VM

Before you can use managed identities for Azure resources to authorize access to Cognitive Services resources from your VM, you must enable managed identities for Azure resources on the VM. To learn how to enable managed identities for Azure Resources, see:

- [Azure portal](#)
- [Azure PowerShell](#)
- [Azure CLI](#)
- [Azure Resource Manager template](#)
- [Azure Resource Manager client libraries](#)

For more information about managed identities, see [Managed identities for Azure resources](#).

Use Azure key vault to securely access credentials

You can [use Azure Key Vault](#) to securely develop Cognitive Services applications. Key Vault enables you to store your authentication credentials in the cloud, and reduces the chances that secrets may be accidentally leaked, because you won't store security information in your application.

Authentication is done via Azure Active Directory. Authorization may be done via Azure role-based access control (Azure RBAC) or Key Vault access policy. Azure RBAC can be used for both management of the vaults and access data stored in a vault, while key vault access policy can only be used when attempting to access data stored in a vault.

See also

- [What is Cognitive Services?](#)

- Cognitive Services pricing ↗
- Custom subdomains

Rotate subscription keys in Cognitive Services

Article • 11/15/2022 • 2 minutes to read

Each Cognitive Services resource has two API keys to enable secret rotation. This is a security precaution that lets you regularly change the keys that can access your service, protecting the privacy of your resource if a key gets leaked.

How to rotate keys

Keys can be rotated using the following procedure:

1. If you're using both keys in production, change your code so that only one key is in use. In this guide, assume it's key 1.

This is a necessary step because once a key is regenerated, the older version of that key will stop working immediately. This would cause clients using the older key to get 401 access denied errors.

2. Once you have only key 1 in use, you can regenerate the key 2. Go to your resource's page on the Azure portal, select the **Keys and Endpoint** tab, and select the **Regenerate Key 2** button at the top of the page.
3. Next, update your code to use the newly generated key 2.

It will help to have logs or availability to check that users of the key have successfully swapped from using key 1 to key 2 before you proceed.

4. Now you can regenerate the key 1 using the same process.
5. Finally, update your code to use the new key 1.

See also

- [What is Cognitive Services?](#)
- [Cognitive Services security features](#)

Use environment variables with Cognitive Services

Article • 10/05/2022 • 2 minutes to read

This guide shows you how to set and retrieve environment variables to handle your Cognitive Services subscription credentials in a more secure way when you test out applications.

Set an environment variable

To set environment variables, use one the following commands, where the `ENVIRONMENT_VARIABLE_KEY` is the named key and `value` is the value stored in the environment variable.

Command Line

Use the following command to create and assign a persisted environment variable, given the input value.

CMD

```
:: Assigns the env var to the value  
setx ENVIRONMENT_VARIABLE_KEY "value"
```

In a new instance of the Command Prompt, use the following command to read the environment variable.

CMD

```
:: Prints the env var value  
echo %ENVIRONMENT_VARIABLE_KEY%
```

Tip

After you set an environment variable, restart your integrated development environment (IDE) to ensure that the newly added environment variables are available.

Retrieve an environment variable

To use an environment variable in your code, it must be read into memory. Use one of the following code snippets, depending on which language you're using. These code snippets demonstrate how to get an environment variable given the `ENVIRONMENT_VARIABLE_KEY` and assign the value to a program variable named `value`.

C#

For more information, see [Environment.GetEnvironmentVariable](#) .

C#

```
using static System.Environment;

class Program
{
    static void Main()
    {
        // Get the named env var, and assign it to the value variable
        var value =
            GetEnvironmentVariable(
                "ENVIRONMENT_VARIABLE_KEY");
    }
}
```

Next steps

- Explore [Cognitive Services](#) and choose a service to get started.

Configure customer-managed keys with Azure Key Vault for Cognitive Services

Article • 12/06/2022 • 2 minutes to read

The process to enable Customer-Managed Keys with Azure Key Vault for Cognitive Services varies by product. Use these links for service-specific instructions:

Vision

- [Custom Vision encryption of data at rest](#)
- [Face Services encryption of data at rest](#)
- [Form Recognizer encryption of data at rest](#)

Language

- [Language Understanding service encryption of data at rest](#)
- [QnA Maker encryption of data at rest](#)
- [Translator encryption of data at rest](#)
- [Language service encryption of data at rest](#)

Speech

- [Speech encryption of data at rest](#)

Decision

- [Content Moderator encryption of data at rest](#)
- [Personalizer encryption of data at rest](#)

Azure OpenAI

- [Azure OpenAI encryption of data at rest](#)

Next steps

- [What is Azure Key Vault?](#)
- [Cognitive Services Customer-Managed Key Request Form ↗](#)

Configure Azure Cognitive Services virtual networks

Article • 08/03/2022 • 16 minutes to read

Azure Cognitive Services provides a layered security model. This model enables you to secure your Cognitive Services accounts to a specific subset of networks. When network rules are configured, only applications requesting data over the specified set of networks can access the account. You can limit access to your resources with request filtering. Allowing only requests originating from specified IP addresses, IP ranges or from a list of subnets in [Azure Virtual Networks](#).

An application that accesses a Cognitive Services resource when network rules are in effect requires authorization. Authorization is supported with [Azure Active Directory](#) (Azure AD) credentials or with a valid API key.

Important

Turning on firewall rules for your Cognitive Services account blocks incoming requests for data by default. In order to allow requests through, one of the following conditions needs to be met:

- The request should originate from a service operating within an Azure Virtual Network (VNet) on the allowed subnet list of the target Cognitive Services account. The endpoint in requests originated from VNet needs to be set as the [custom subdomain](#) of your Cognitive Services account.
- Or the request should originate from an allowed list of IP addresses.

Requests that are blocked include those from other Azure services, from the Azure portal, from logging and metrics services, and so on.

Note

To interact with Azure, the Azure Az PowerShell module is recommended. See [Install Azure PowerShell](#) to get started. To learn how to migrate to the Az PowerShell module, see [Migrate Azure PowerShell from AzureRM to Az.](#)

Scenarios

To secure your Cognitive Services resource, you should first configure a rule to deny access to traffic from all networks (including internet traffic) by default. Then, you should configure rules that grant access to traffic from specific VNets. This configuration enables you to build a secure network boundary for your applications. You can also configure rules to grant access to traffic from select public internet IP address ranges, enabling connections from specific internet or on-premises clients.

Network rules are enforced on all network protocols to Azure Cognitive Services, including REST and WebSocket. To access data using tools such as the Azure test consoles, explicit network rules must be configured. You can apply network rules to existing Cognitive Services resources, or when you create new Cognitive Services resources. Once network rules are applied, they're enforced for all requests.

Supported regions and service offerings

Virtual networks (VNets) are supported in [regions where Cognitive Services are available](#). Cognitive Services supports service tags for network rules configuration. The services listed below are included in the **CognitiveServicesManagement** service tag.

- ✓ Anomaly Detector
- ✓ Computer Vision
- ✓ Content Moderator
- ✓ Custom Vision
- ✓ Face
- ✓ Language Understanding (LUIS)
- ✓ Personalizer
- ✓ Speech service
- ✓ Language service
- ✓ QnA Maker
- ✓ Translator Text

Note

If you're using LUIS, Speech Services, or Language services, the **CognitiveServicesManagement** tag only enables you use the service using the SDK or REST API. To access and use LUIS portal , Speech Studio or Language Studio from a virtual network, you will need to use the following tags:

- **AzureActiveDirectory**
- **AzureFrontDoor.Frontend**
- **AzureResourceManager**

- CognitiveServicesManagement

Change the default network access rule

By default, Cognitive Services resources accept connections from clients on any network. To limit access to selected networks, you must first change the default action.

⚠ Warning

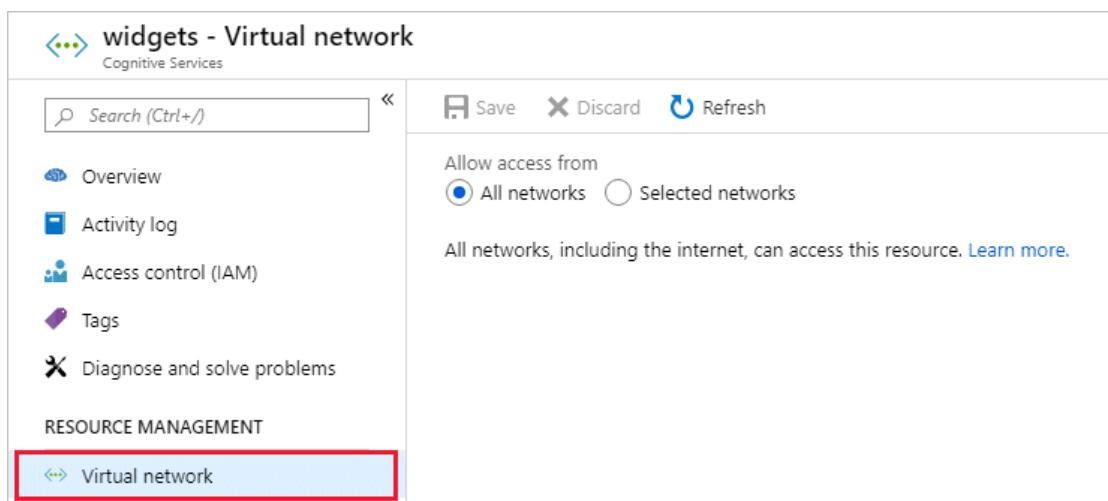
Making changes to network rules can impact your applications' ability to connect to Azure Cognitive Services. Setting the default network rule to **deny** blocks all access to the data unless specific network rules that **grant** access are also applied. Be sure to grant access to any allowed networks using network rules before you change the default rule to deny access. If you are allowing listing IP addresses for your on-premises network, be sure to add all possible outgoing public IP addresses from your on-premises network.

Managing default network access rules

You can manage default network access rules for Cognitive Services resources through the Azure portal, PowerShell, or the Azure CLI.

Azure portal

1. Go to the Cognitive Services resource you want to secure.
2. Select the **RESOURCE MANAGEMENT** menu called **Virtual network**.



3. To deny access by default, choose to allow access from **Selected networks**.

With the **Selected networks** setting alone, unaccompanied by configured **Virtual networks** or **Address ranges** - all access is effectively denied. When all access is denied, requests attempting to consume the Cognitive Services resource aren't permitted. The Azure portal, Azure PowerShell or, Azure CLI can still be used to configure the Cognitive Services resource.

4. To allow traffic from all networks, choose to allow access from **All networks**.

The screenshot shows the Azure portal interface for managing a Cognitive Service. The left sidebar has a 'Virtual network' section selected. The main content area shows a note about firewall settings. Below it, under 'Allow access from', there are two options: 'All networks' and 'Selected networks'. The 'Selected networks' option is selected and highlighted with a red box. At the bottom, there's a 'ADDRESS RANGE' input field with the placeholder 'IP address or CIDR'.

5. Select **Save** to apply your changes.

Grant access from a virtual network

You can configure Cognitive Services resources to allow access only from specific subnets. The allowed subnets may belong to a VNet in the same subscription, or in a different subscription, including subscriptions belonging to a different Azure Active Directory tenant.

Enable a [service endpoint](#) for Azure Cognitive Services within the VNet. The service endpoint routes traffic from the VNet through an optimal path to the Azure Cognitive Services service. The identities of the subnet and the virtual network are also transmitted with each request. Administrators can then configure network rules for the Cognitive Services resource that allow requests to be received from specific subnets in a VNet. Clients granted access via these network rules must continue to meet the authorization requirements of the Cognitive Services resource to access the data.

Each Cognitive Services resource supports up to 100 virtual network rules, which may be combined with [IP network rules](#).

Required permissions

To apply a virtual network rule to a Cognitive Services resource, the user must have the appropriate permissions for the subnets being added. The required permission is the default *Contributor* role, or the *Cognitive Services Contributor* role. Required permissions can also be added to custom role definitions.

Cognitive Services resource and the virtual networks granted access may be in different subscriptions, including subscriptions that are a part of a different Azure AD tenant.

Note

Configuration of rules that grant access to subnets in virtual networks that are a part of a different Azure Active Directory tenant are currently only supported through PowerShell, CLI and REST APIs. Such rules cannot be configured through the Azure portal, though they may be viewed in the portal.

Managing virtual network rules

You can manage virtual network rules for Cognitive Services resources through the Azure portal, PowerShell, or the Azure CLI.

Azure portal

1. Go to the Cognitive Services resource you want to secure.
2. Select the **RESOURCE MANAGEMENT** menu called **Virtual network**.
3. Check that you've selected to allow access from **Selected networks**.
4. To grant access to a virtual network with an existing network rule, under **Virtual networks**, select **Add existing virtual network**.

widgets - Virtual network
Cognitive Services

Save Discard Refresh

Firewall settings allowing access to cognitive service will remain in effect for up to three minutes after saving updated settings restricting access.

Allow access from
 All networks Selected networks

Configure network security for your cognitive service. [Learn more.](#)

Virtual networks
Secure your cognitive service with virtual networks. [+ Add existing virtual network](#) [+ Add new virtual network](#)

VIRTUAL NETWORK	SUBNET	ADDRESS RANGE
No network selected.		

Firewall
Add IP ranges to allow access from the internet or your on-premises networks. [Learn more.](#)

Add your client IP address? [?](#)

ADDRESS RANGE

5. Select the **Virtual networks** and **Subnets** options, and then select **Enable**.

Add networks X

* Subscription

* Virtual networks

* Subnets

Information
The following networks don't have service endpoints enabled for 'Microsoft.CognitiveServices'. Enabling access will take up to 15 minutes to complete. After starting this operation, it is safe to leave and return later if you do not wish to wait.

VIRTUAL NETWORK	SERVICE ENDPOINT STATUS
▼ widgets-vnet...	...
default	Not enabled

Enable

6. To create a new virtual network and grant it access, select **Add new virtual network**.

The screenshot shows a Microsoft Azure Cognitive Services interface for managing virtual networks. At the top, there's a header bar with a back arrow, the title 'widgets - Virtual network', and 'Cognitive Services'. Below the header are buttons for 'Save', 'Discard', and 'Refresh'. A status message says: 'Firewall settings allowing access to cognitive service will remain in effect for up to three minutes after saving updated settings restricting access.' Under 'Allow access from', the 'Selected networks' option is selected. A note says: 'Configure network security for your cognitive service. [Learn more.](#)' Below that, under 'Virtual networks', there's a table with columns 'VIRTUAL NETWORK', 'SUBNET', and 'ADDRESS RANGE'. A sub-note says: 'Secure your cognitive service with virtual networks.' Below the table, there's a section for 'Firewall' with a note: 'Add IP ranges to allow access from the internet or your on-premises networks. [Learn more.](#)' There's also a checkbox for 'Add your client IP address?' followed by a help icon. At the bottom, there's a 'ADDRESS RANGE' input field with a placeholder 'IP address or CIDR'.

7. Provide the information necessary to create the new virtual network, and then select **Create**.

Create virtual network

★ Name
widgets-vnet ✓

★ Address space ⓘ
10.1.0.0/16
10.1.0.0 - 10.1.255.255 (65536 addresses)

★ Subscription
widgets-subscription ▾

★ Resource group
widgets-resource-group ▾
[Create new](#)

★ Location
(US) West US 2 ▾

Subnet

★ Name
default

★ Address range ⓘ
10.1.0.0/24 ✓
10.1.0.0 - 10.1.0.255 (256 addresses)

DDoS protection ⓘ
 Basic Standard

Service endpoint ⓘ
Microsoft.CognitiveServices

Firewall ⓘ

(!) Note

If a service endpoint for Azure Cognitive Services wasn't previously configured for the selected virtual network and subnets, you can configure it as part of this operation.

Presently, only virtual networks belonging to the same Azure Active Directory tenant are shown for selection during rule creation. To grant access to a subnet in a virtual network belonging to another tenant, please use PowerShell, CLI or REST APIs.

8. To remove a virtual network or subnet rule, select ... to open the context menu for the virtual network or subnet, and select Remove.

The screenshot shows the Azure portal interface for managing a virtual network. At the top, it says "widgets - Virtual network" under "Cognitive Services". Below that are "Save", "Discard", and "Refresh" buttons. A message box states: "Firewall settings allowing access to cognitive service will remain in effect for up to three minutes after saving updated settings restricting access." Under "Allow access from", the "Selected networks" radio button is selected. A note says: "Configure network security for your cognitive service. [Learn more](#)." Below this is a section for "Virtual networks" with "Secure your cognitive service with virtual networks." buttons for "Add existing virtual network" and "Add new virtual network". A table lists subnets:

VIRTUAL NETWORK	SUBNET	ADDRESS RANGE	ENDPOINT STATUS	RESOURCE GROUP	SUBSCRIPTION
widgets-vnet	1	default 10.1.0.0/24	✓ Enabled	widgets-	widgets-resource-gr... widgets-subscription ...

A context menu is open over the "Remove" button for the first subnet rule. The menu items are "Remove" (highlighted with a red box) and "...". Below the table is a "Firewall" section with a note to "Add IP ranges to allow access from the internet or your on-premises networks." and a checkbox for "Add your client IP address". There is also an "ADDRESS RANGE" input field with "IP address or CIDR" placeholder text.

9. Select Save to apply your changes.

ⓘ Important

Be sure to **set the default rule to deny**, or network rules have no effect.

Grant access from an internet IP range

You can configure Cognitive Services resources to allow access from specific public internet IP address ranges. This configuration grants access to specific services and on-premises networks, effectively blocking general internet traffic.

Provide allowed internet address ranges using [CIDR notation](#) in the form `16.17.18.0/24` or as individual IP addresses like `16.17.18.19`.

💡 Tip

Small address ranges using "/31" or "/32" prefix sizes are not supported. These ranges should be configured using individual IP address rules.

IP network rules are only allowed for **public internet** IP addresses. IP address ranges reserved for private networks (as defined in [RFC 1918](#)) aren't allowed in IP rules. Private networks include addresses that start with `10.*`, `172.16.* - 172.31.*`, and `192.168.*`.

Only IPV4 addresses are supported at this time. Each Cognitive Services resource supports up to 100 IP network rules, which may be combined with [Virtual network rules](#).

Configuring access from on-premises networks

To grant access from your on-premises networks to your Cognitive Services resource with an IP network rule, you must identify the internet facing IP addresses used by your network. Contact your network administrator for help.

If you're using [ExpressRoute](#) on-premises for public peering or Microsoft peering, you'll need to identify the NAT IP addresses. For public peering, each ExpressRoute circuit by default uses two NAT IP addresses. Each is applied to Azure service traffic when the traffic enters the Microsoft Azure network backbone. For Microsoft peering, the NAT IP addresses that are used are either customer provided or are provided by the service provider. To allow access to your service resources, you must allow these public IP addresses in the resource IP firewall setting. To find your public peering ExpressRoute circuit IP addresses, open a support ticket with [ExpressRoute](#) via the Azure portal. Learn more about [NAT for ExpressRoute public and Microsoft peering](#).

Managing IP network rules

You can manage IP network rules for Cognitive Services resources through the Azure portal, PowerShell, or the Azure CLI.

Azure portal

1. Go to the Cognitive Services resource you want to secure.
2. Select the **RESOURCE MANAGEMENT** menu called **Virtual network**.
3. Check that you've selected to allow access from **Selected networks**.
4. To grant access to an internet IP range, enter the IP address or address range (in [CIDR format](#)) under **Firewall > Address Range**. Only valid public IP (non-reserved) addresses are accepted.

widgets - Virtual network
Cognitive Services

» Save Discard Refresh

Firewall settings allowing access to cognitive service will remain in effect for up to three minutes after saving updated settings restricting access.

Allow access from
 All networks Selected networks

Configure network security for your cognitive service. [Learn more](#).

Virtual networks
Secure your cognitive service with virtual networks. [+ Add existing virtual network](#) [+ Add new virtual network](#)

VIRTUAL NETWORK	SUBNET	ADDRESS RANGE
No network selected.		

Firewall
Add IP ranges to allow access from the internet or your on-premises networks. [Learn more](#).

Add your client IP address?

ADDRESS RANGE

IP address or CIDR
173.0.0.0/16

5. To remove an IP network rule, select the trash can icon next to the address range.

widgets - Virtual network
Cognitive Services

» Save Discard Refresh

Firewall settings allowing access to cognitive service will remain in effect for up to three minutes after saving updated settings restricting access.

Allow access from
 All networks Selected networks

Configure network security for your cognitive service. [Learn more](#).

Virtual networks
Secure your cognitive service with virtual networks. [+ Add existing virtual network](#) [+ Add new virtual network](#)

VIRTUAL NETWORK	SUBNET	ADDRESS RANGE
No network selected.		

Firewall
Add IP ranges to allow access from the internet or your on-premises networks. [Learn more](#).

Add your client IP address?

ADDRESS RANGE

IP address or CIDR
173.0.0.0/16

6. Select **Save** to apply your changes.

Important

Be sure to **set the default rule to deny**, or network rules have no effect.

Use private endpoints

You can use [private endpoints](#) for your Cognitive Services resources to allow clients on a virtual network (VNet) to securely access data over a [Private Link](#). The private endpoint uses an IP address from the VNet address space for your Cognitive Services resource. Network traffic between the clients on the VNet and the resource traverses the VNet and a private link on the Microsoft backbone network, eliminating exposure from the public internet.

Private endpoints for Cognitive Services resources let you:

- Secure your Cognitive Services resource by configuring the firewall to block all connections on the public endpoint for the Cognitive Services service.
- Increase security for the VNet, by enabling you to block exfiltration of data from the VNet.
- Securely connect to Cognitive Services resources from on-premises networks that connect to the VNet using [VPN](#) or [ExpressRoutes](#) with private-peering.

Conceptual overview

A private endpoint is a special network interface for an Azure resource in your [VNet](#). Creating a private endpoint for your Cognitive Services resource provides secure connectivity between clients in your VNet and your resource. The private endpoint is assigned an IP address from the IP address range of your VNet. The connection between the private endpoint and the Cognitive Services service uses a secure private link.

Applications in the VNet can connect to the service over the private endpoint seamlessly, using the same connection strings and authorization mechanisms that they would use otherwise. The exception is the Speech Services, which require a separate endpoint. See the section on [Private endpoints with the Speech Services](#). Private endpoints can be used with all protocols supported by the Cognitive Services resource, including REST.

Private endpoints can be created in subnets that use [Service Endpoints](#). Clients in a subnet can connect to one Cognitive Services resource using private endpoint, while using service endpoints to access others.

When you create a private endpoint for a Cognitive Services resource in your VNet, a consent request is sent for approval to the Cognitive Services resource owner. If the user requesting the creation of the private endpoint is also an owner of the resource, this consent request is automatically approved.

Cognitive Services resource owners can manage consent requests and the private endpoints, through the '*Private endpoints*' tab for the Cognitive Services resource in the [Azure portal](#).

Private endpoints

When creating the private endpoint, you must specify the Cognitive Services resource it connects to. For more information on creating a private endpoint, see:

- [Create a private endpoint using the Private Link Center in the Azure portal](#)
- [Create a private endpoint using Azure CLI](#)
- [Create a private endpoint using Azure PowerShell](#)

Connecting to private endpoints

Clients on a VNet using the private endpoint should use the same connection string for the Cognitive Services resource as clients connecting to the public endpoint. The exception is the Speech Services, which require a separate endpoint. See the section on [Private endpoints with the Speech Services](#). We rely upon DNS resolution to automatically route the connections from the VNet to the Cognitive Services resource over a private link.

We create a [private DNS zone](#) attached to the VNet with the necessary updates for the private endpoints, by default. However, if you're using your own DNS server, you may need to make additional changes to your DNS configuration. The section on [DNS changes](#) below describes the updates required for private endpoints.

Private endpoints with the Speech Services

See [Using Speech Services with private endpoints provided by Azure Private Link](#).

DNS changes for private endpoints

When you create a private endpoint, the DNS CNAME resource record for the Cognitive Services resource is updated to an alias in a subdomain with the prefix '*privatelink*'. By default, we also create a [private DNS zone](#), corresponding to the '*privatelink*' subdomain, with the DNS A resource records for the private endpoints.

When you resolve the endpoint URL from outside the VNet with the private endpoint, it resolves to the public endpoint of the Cognitive Services resource. When resolved from

the VNet hosting the private endpoint, the endpoint URL resolves to the private endpoint's IP address.

This approach enables access to the Cognitive Services resource using the same connection string for clients in the VNet hosting the private endpoints and clients outside the VNet.

If you are using a custom DNS server on your network, clients must be able to resolve the fully qualified domain name (FQDN) for the Cognitive Services resource endpoint to the private endpoint IP address. Configure your DNS server to delegate your private link subdomain to the private DNS zone for the VNet.

💡 Tip

When using a custom or on-premises DNS server, you should configure your DNS server to resolve the Cognitive Services resource name in the 'privatelink' subdomain to the private endpoint IP address. You can do this by delegating the 'privatelink' subdomain to the private DNS zone of the VNet, or configuring the DNS zone on your DNS server and adding the DNS A records.

For more information on configuring your own DNS server to support private endpoints, refer to the following articles:

- [Name resolution for resources in Azure virtual networks](#)
- [DNS configuration for private endpoints](#)

Pricing

For pricing details, see [Azure Private Link pricing](#).

Next steps

- Explore the various [Azure Cognitive Services](#)
- Learn more about [Azure Virtual Network Service Endpoints](#)

Develop Azure Cognitive Services applications with Key Vault

Article • 11/11/2022 • 18 minutes to read

Use this article to learn how to develop Cognitive Services applications securely by using [Azure Key Vault](#).

Key Vault reduces the chances that secrets may be accidentally leaked, because you won't store security information in your application.

Prerequisites

- A valid Azure subscription - [Create one for free](#) ↗
- [Visual Studio IDE](#) ↗
- An [Azure Key Vault](#)
- [A multi-service resource or a resource for a specific service](#)

ⓘ Note

Review the documentation and quickstart articles for the Cognitive Service you're using to get an understanding of:

- The credentials and other information you will need to send API calls.
- The packages and code you will need to run your application.

Get your credentials from your Cognitive Services resource

Before you add your credential information to your Azure key vault, you need to retrieve them from your Cognitive Services resource. For example, if your service needs a key and endpoint you would find them using the following steps:

1. Navigate to your Azure resource in the [Azure portal](#) ↗ .
2. From the collapsible menu on the left, select **Keys and Endpoint**.

The screenshot shows the 'MyDemoResource | Keys and Endpoint' page in the Azure portal. On the left, a sidebar lists various service management options like Overview, Activity log, Access control (IAM), Tags, and Diagnose and solve problems. Under Resource Management, the 'Keys and Endpoint' option is highlighted with a red box. The main content area displays two keys (KEY 1 and KEY 2) and their corresponding endpoints. A callout box with an exclamation mark provides a note about securely storing keys. A 'Show Keys' button is visible above the key fields. The endpoint field for KEY 1 is also highlighted with a red box. The location is set to 'westus'. A magnifying glass icon is located in the bottom right corner.

Some Cognitive Services require different information to authenticate API calls, such as a key and region. Make sure to retrieve this information before continuing on.

Add your credentials to your key vault

For your application to retrieve and use your credentials to authenticate API calls, you will need to add them to your [key vault secrets](#).

Repeat these steps to generate a secret for each required resource credential. For example, a key and endpoint. These secret names will be used later to authenticate your application.

1. Open a new browser tab or window. Navigate to your key vault in the [Azure portal](#).
2. From the collapsible menu on the left, select **Objects > Secrets**.
3. Select **Generate/Import**.

4. On the **Create a secret** screen, enter the following values:

Name	Value
Upload options	Manual
Name	A secret name for your key or endpoint. For example: "CognitiveServicesKey" or "CognitiveServicesEndpoint"
Value	Your Azure Cognitive Services resource key or endpoint.

Later your application will use the secret "Name" to securely access the "Value".

5. Leave the other values as their defaults. Select **Create**.

💡 Tip

Make sure to remember the names that you set for your secrets, as you'll use them later in your application.

You should now have named secrets for your resource information.

Create an environment variable for your key vault's name

We recommend creating an environment variable for your Azure key vault's name. Your application will read this environment variable at runtime to retrieve your key and endpoint information.

To set environment variables, use one the following commands. `KEY_VAULT_NAME` with the name of the environment variable, and replace `Your-Key-Vault-Name` with the name of your key vault, which will be stored in the environment variable.

Azure CLI

Create and assign persisted environment variable, given the value.

CMD

```
setx KEY_VAULT_NAME "Your-Key-Vault-Name"
```

In a new instance of the **Command Prompt**, read the environment variable.

CMD

```
echo %KEY_VAULT_NAME%
```

Authenticate to Azure using Visual Studio

Developers using Visual Studio 2017 or later can authenticate an Azure Active Directory account through Visual Studio. This enables you to access secrets in your key vault by signing into your Azure subscription from within the IDE.

To authenticate in Visual Studio, select **Tools** from the top navigation menu, and select **Options**. Navigate to the **Azure Service Authentication** option to sign in with your user name and password.

Authenticate using the command line

Before you can grant access to your key vault, you must authenticate with your Azure Active Directory user name and password.

Azure CLI

To authenticate with the [Azure CLI](#), run the `az login` command.

Azure CLI

```
az login
```

On systems with a default web browser, the Azure CLI will launch the browser to authenticate. For systems without a default web browser, the `az login` command will use the device code authentication flow. You can also force the Azure CLI to use the device code flow rather than launching a browser by specifying the `--use-device-code` argument.

If you have multiple subscriptions, make sure to [select the Azure subscription](#) that contains your key vault.

Grant access to your key vault

Create an access policy for your key vault that grants secret permissions to your user account.

Azure CLI

To set the access policy, run the `az keyvault set-policy` command. Replace `Your-Key-Vault-Name` with the name of your key vault. Replace `user@domain.com` with your Azure Active Directory user name.

Azure CLI

```
az keyvault set-policy --name Your-Key-Vault-Name --upn user@domain.com  
--secret-permissions delete get list set purge
```

Create a new C# application

Using the Visual Studio IDE, create a new .NET Core console app. This will create a "Hello World" project with a single C# source file: `program.cs`.

Install the following client libraries by right-clicking on the solution in the **Solution Explorer** and selecting **Manage NuGet Packages**. In the package manager that opens select **Browse** and search for the following libraries, and select **Install** for each:

- `Azure.Security.KeyVault.Secrets`
- `Azure.Identity`

Import the example code

Copy the following example code into your `program.cs` file. Replace `Your-Key-Secret-Name` and `Your-Endpoint-Secret-Name` with the secret names that you set in your key vault.

C#

```
using System;
using System.Threading.Tasks;
using Azure;
using Azure.Identity;
using Azure.Security.KeyVault.Secrets;
using System.Net;

namespace key_vault_console_app
{
    class Program
    {
        static async Task Main(string[] args)
        {
            //Name of your key vault
            var keyVaultName =
                Environment.GetEnvironmentVariable("KEY_VAULT_NAME");

            //variables for retrieving the key and endpoint from your key
            //vault.
            //Set these variables to the names you created for your secrets
            const string keySecretName = "Your-Key-Secret-Name";
            const string endpointSecretName = "Your-Endpoint-Secret-Name";

            //Endpoint for accessing your key vault
            var kvUri = $"https://'{keyVaultName}'.vault.azure.net";

            var keyVaultClient = new SecretClient(new Uri(kvUri), new
DefaultAzureCredential());

            Console.WriteLine($"Retrieving your secrets from
{keyVaultName}.");

            //Key and endpoint secrets retrieved from your key vault
            var keySecret = await
keyVaultClient.GetSecretAsync(keySecretName);
            var endpointSecret = await
keyVaultClient.GetSecretAsync(endpointSecretName);
            Console.WriteLine($"Your key secret value is:
{keySecret.Value.Value}");
            Console.WriteLine($"Your endpoint secret value is:
{endpointSecret.Value.Value}");
            Console.WriteLine("Secrets retrieved successfully");

        }
    }
}
```

Run the application

Run the application by selecting the **Debug** button at the top of Visual studio. Your key and endpoint secrets will be retrieved from your key vault.

Send a test Language service call (optional)

If you're using a multi-service resource or Language resource, you can update [your application](#) by following these steps to send an example Named Entity Recognition call by retrieving a key and endpoint from your key vault.

1. Install the `Azure.AI.TextAnalytics` library by right-clicking on the solution in the **Solution Explorer** and selecting **Manage NuGet Packages**. In the package manager that opens select **Browse** and search for the following libraries, and select **Install** for each:
 - Microsoft.Azure.CognitiveServices.Language.TextAnalytics
 - Microsoft.Azure.CognitiveServices.Core
2. Add the following directive to the top of your `program.cs` file.

```
C#  
  
using Azure.AI.TextAnalytics;
```

3. Add the following code sample to your application.

```
C#  
  
// Example method for extracting named entities from text  
private static void EntityRecognitionExample(string keySecret, string endpointSecret)  
{  
    //String to be sent for Named Entity Recognition  
    var exampleString = "I had a wonderful trip to Seattle last week.";  
  
    AzureKeyCredential azureKeyCredential = new  
    AzureKeyCredential(keySecret);  
    Uri endpoint = new Uri(endpointSecret);  
    var languageServiceClient = new TextAnalyticsClient(endpoint,  
    azureKeyCredential);  
  
    Console.WriteLine($"Sending a Named Entity Recognition (NER)  
request");  
    var response =  
    languageServiceClient.RecognizeEntities(exampleString);  
    Console.WriteLine("Named Entities:");  
    foreach (var entity in response.Value)  
    {  
        Console.WriteLine($"{entity.Text}, {entity.Category}:
```

```
{entity.Category},\tSub-Category: {entity.SubCategory}");  
    Console.WriteLine($"\\t\\tScore:  
{entity.ConfidenceScore:F2},\\tLength: {entity.Length},\\tOffset:  
{entity.Offset}\\n");  
}  
}
```

4. Add the following code to call `EntityRecognitionExample()` from your main method, with your key and endpoint values.

C#

```
EntityRecognitionExample(keySecret.Value.Value,  
endpointSecret.Value.Value);
```

5. Run the application.

Next steps

- See [What are Cognitive Services](#) for available features you can develop along with [Azure key vault](#).
- For additional information on secure application development, see:
 - [Best practices for using Azure Key Vault](#)
 - [Cognitive Services security](#)
 - [Azure security baseline for Cognitive Services](#)

Configure data loss prevention for Azure Cognitive Services

Article • 03/18/2022 • 2 minutes to read

Cognitive Services data loss prevention capabilities allow customers to configure the list of outbound URLs their Cognitive Services resources are allowed to access. This creates another level of control for customers to prevent data loss. In this article, we'll cover the steps required to enable the data loss prevention feature for Cognitive Services resources.

Prerequisites

Before you make a request, you need an Azure account and an Azure Cognitive Services subscription. If you already have an account, go ahead and skip to the next section. If you don't have an account, we have a guide to get you set up in minutes: [Create a Cognitive Services account for Azure](#).

You can get your subscription key from the [Azure portal](#) after [creating your account](#).

Enabling data loss prevention

There are two parts to enable data loss prevention. First the property `restrictOutboundNetworkAccess` must be set to true. When this is set to true, you also need to provide the list of approved URLs. The list of URLs is added to the `allowedFqdnList` property. The `allowedFqdnList` property contains an array of comma-separated URLs.

ⓘ Note

- The `allowedFqdnList` property value supports a maximum of 1000 URLs.
- The property supports both IP addresses and fully qualified domain names i.e., `www.microsoft.com`, values.
- It can take up to 15 minutes for the updated list to take effect.

Azure CLI

1. Install the [Azure CLI](#) and [sign in](#), or select [Try it](#).

2. View the details of the Cognitive Services resource.

Azure CLI

```
az cognitiveservices account show \
-g "myresourcegroup" -n "myaccount" \
```

3. View the current properties of the Cognitive Services resource.

Azure CLI

```
az rest -m get \
-u /subscriptions/{subscription ID}/resourceGroups/{resource
group}/providers/Microsoft.CognitiveServices/accounts/{account
name}?api-version=2021-04-30 \
```

4. Configure the restrictOutboundNetworkAccess property and update the allowed FqdnList with the approved URLs

Azure CLI

```
az rest -m patch \
-u /subscriptions/{subscription ID}/resourceGroups/{resource
group}/providers/Microsoft.CognitiveServices/accounts/{account
name}?api-version=2021-04-30 \
-b '{"properties": { "restrictOutboundNetworkAccess": true,
"allowedFqdnList": [ "microsoft.com" ] }}'
```

Supported services

The following services support data loss prevention configuration:

- Computer Vision
- Content Moderator
- Custom Vision
- Face
- Form Recognizer
- Speech Service
- QnA Maker

Next steps

- Configure Virtual Networks

Azure security baseline for Cognitive Services

Article • 01/15/2023 • 12 minutes to read

This security baseline applies guidance from the [Microsoft cloud security benchmark version 1.0](#) to Cognitive Services. The Microsoft cloud security benchmark provides recommendations on how you can secure your cloud solutions on Azure. The content is grouped by the security controls defined by the Microsoft cloud security benchmark and the related guidance applicable to Cognitive Services.

You can monitor this security baseline and its recommendations using Microsoft Defender for Cloud. Azure Policy definitions will be listed in the Regulatory Compliance section of the Microsoft Defender for Cloud dashboard.

When a feature has relevant Azure Policy Definitions, they are listed in this baseline to help you measure compliance to the Microsoft cloud security benchmark controls and recommendations. Some recommendations may require a paid Microsoft Defender plan to enable certain security scenarios.

ⓘ Note

Features not applicable to Cognitive Services have been excluded. To see how Cognitive Services completely maps to the Microsoft cloud security benchmark, see the [full Cognitive Services security baseline mapping file ↗](#).

Security profile

The security profile summarizes high-impact behaviors of Cognitive Services, which may result in increased security considerations.

Service Behavior Attribute	Value
Product Category	AI+ML
Customer can access HOST / OS	No Access
Service can be deployed into customer's virtual network	False
Stores customer content at rest	True

Network security

For more information, see the [Microsoft cloud security benchmark: Network security](#).

NS-1: Establish network segmentation boundaries

Features

Virtual Network Integration

Description: Service supports deployment into customer's private Virtual Network (VNet). [Learn more](#).

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

Configuration Guidance: This feature is not supported to secure this service.

Network Security Group Support

Description: Service network traffic respects Network Security Groups rule assignment on its subnets. [Learn more](#).

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

Feature notes: Though Network Security Groups for this service is not supported, a service-level firewall may be configured. For more information, please visit: [Managing IP network rules](#)

Configuration Guidance: This feature is not supported to secure this service.

NS-2: Secure cloud services with network controls

Features

Azure Private Link

Description: Service native IP filtering capability for filtering network traffic (not to be confused with NSG or Azure Firewall). [Learn more](#).

Supported	Enabled By Default	Configuration Responsibility
True	False	Customer

Configuration Guidance: Deploy private endpoints for all Azure resources that support the Private Link feature, to establish a private access point for the resources.

Reference: [Use private endpoints](#)

Disable Public Network Access

Description: Service supports disabling public network access either through using service-level IP ACL filtering rule (not NSG or Azure Firewall) or using a 'Disable Public Network Access' toggle switch. [Learn more](#).

Supported	Enabled By Default	Configuration Responsibility
True	False	Customer

Configuration Guidance: Disable public network access either using the service-level IP ACL filtering rule or a toggling switch for public network access.

Reference: [Change the default network access rule](#)

Microsoft Defender for Cloud monitoring

Azure Policy built-in definitions - Microsoft.CognitiveServices:

Name (Azure portal)	Description	Effect(s)	Version (GitHub)
Cognitive Services accounts should disable public network access ↗	Disabling public network access improves security by ensuring that Cognitive Services account isn't exposed on the public internet. Creating private endpoints can limit exposure of Cognitive Services account. Learn more at: https://go.microsoft.com/fwlink/?linkid=2129800 .	Audit, Deny, Disabled	2.0.0 ↗

Name	Description	Effect(s)	Version
(Azure portal)		(GitHub)	
Cognitive Services accounts should restrict network access ↗	<p>Network access to Cognitive Services accounts should be restricted. Configure network rules so only applications from allowed networks can access the Cognitive Services account.</p> <p>To allow connections from specific internet or on-premises clients, access can be granted to traffic from specific Azure virtual networks or to public internet IP address ranges.</p>	Audit, Deny, Disabled	2.0.0 ↗

Identity management

For more information, see the [Microsoft cloud security benchmark: Identity management](#).

IM-1: Use centralized identity and authentication system

Features

Azure AD Authentication Required for Data Plane Access

Description: Service supports using Azure AD authentication for data plane access.

[Learn more](#).

Supported	Enabled By Default	Configuration Responsibility
True	False	Customer

Configuration Guidance: Use Azure Active Directory (Azure AD) as the default authentication method to control your data plane access.

Reference: [Authenticate with Azure Active Directory](#)

Local Authentication Methods for Data Plane Access

Description: Local authentications methods supported for data plane access, such as a local username and password. [Learn more](#).

Supported	Enabled By Default	Configuration Responsibility
True	False	Customer

Feature notes: While you can authenticate against Azure Cognitive Services using a single-service or multi-service subscription key, or use those keys to authenticate with access tokens, these authentication methods fall short in more complex scenarios that require Azure role-based access control (Azure RBAC). Avoid the usage of local authentication methods or accounts, these should be disabled wherever possible. Instead use Azure AD to authenticate where possible.

Configuration Guidance: Restrict the use of local authentication methods for data plane access. Instead, use Azure Active Directory (Azure AD) as the default authentication method to control your data plane access.

Reference: [Authenticate with an access token](#)

IM-3: Manage application identities securely and automatically

Features

Managed Identities

Description: Data plane actions support authentication using managed identities. [Learn more.](#)

Supported	Enabled By Default	Configuration Responsibility
True	False	Customer

Configuration Guidance: Use Azure managed identities instead of service principals when possible, which can authenticate to Azure services and resources that support Azure Active Directory (Azure AD) authentication. Managed identity credentials are fully managed, rotated, and protected by the platform, avoiding hard-coded credentials in source code or configuration files.

Reference: [Authorize access to managed identities](#)

Service Principals

Description: Data plane supports authentication using service principals. [Learn more.](#)

Supported	Enabled By Default	Configuration Responsibility
True	False	Customer

Configuration Guidance: There is no current Microsoft guidance for this feature configuration. Please review and determine if your organization wants to configure this security feature.

Reference: [Authenticate requests to Azure Cognitive Services](#)

IM-7: Restrict resource access based on conditions

Features

Conditional Access for Data Plane

Description: Data plane access can be controlled using Azure AD Conditional Access Policies. [Learn more.](#)

Supported	Enabled By Default	Configuration Responsibility
True	False	Customer

Configuration Guidance: Define the applicable conditions and criteria for Azure Active Directory (Azure AD) conditional access in the workload. Consider common use cases such as blocking or granting access from specific locations, blocking risky sign-in behavior, or requiring organization-managed devices for specific applications.

IM-8: Restrict the exposure of credential and secrets

Features

Service Credential and Secrets Support Integration and Storage in Azure Key Vault

Description: Data plane supports native use of Azure Key Vault for credential and secrets store. [Learn more.](#)

Supported	Enabled By Default	Configuration Responsibility
True	False	Customer

Configuration Guidance: Ensure that secrets and credentials are stored in secure locations such as Azure Key Vault, instead of embedding them into code or configuration files.

Privileged access

For more information, see the [Microsoft cloud security benchmark: Privileged access](#).

PA-1: Separate and limit highly privileged/administrative users

Features

Local Admin Accounts

Description: Service has the concept of a local administrative account. [Learn more](#).

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

Configuration Guidance: This feature is not supported to secure this service.

PA-7: Follow just enough administration (least privilege) principle

Features

Azure RBAC for Data Plane

Description: Azure Role-Based Access Control (Azure RBAC) can be used to manage access to service's data plane actions. [Learn more](#).

Supported	Enabled By Default	Configuration Responsibility
True	False	Customer

Configuration Guidance: Use Azure role-based access control (Azure RBAC) to manage Azure resource access through built-in role assignments. Azure RBAC roles can be assigned to users, groups, service principals, and managed identities.

Reference: [Authenticate with Azure Active Directory](#)

PA-8: Determine access process for cloud provider support

Features

Customer Lockbox

Description: Customer Lockbox can be used for Microsoft support access. [Learn more.](#)

Supported	Enabled By Default	Configuration Responsibility
True	False	Customer

Configuration Guidance: In support scenarios where Microsoft needs to access your data, use Customer Lockbox to review, then approve or reject each of Microsoft's data access requests.

Data protection

For more information, see the [Microsoft cloud security benchmark: Data protection](#).

DP-1: Discover, classify, and label sensitive data

Features

Sensitive Data Discovery and Classification

Description: Tools (such as Azure Purview or Azure Information Protection) can be used for data discovery and classification in the service. [Learn more.](#)

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

Configuration Guidance: This feature is not supported to secure this service.

DP-2: Monitor anomalies and threats targeting sensitive data

Features

Data Leakage/Loss Prevention

Description: Service supports DLP solution to monitor sensitive data movement (in customer's content). [Learn more](#).

Supported	Enabled By Default	Configuration Responsibility
True	False	Customer

Configuration Guidance: Cognitive Services data loss prevention capabilities allow customers to configure the list of outbound URLs their Cognitive Services resources are allowed to access. This creates another level of control for customers to prevent data loss.

Reference: [Configure data loss prevention for Azure Cognitive Services](#)

DP-3: Encrypt sensitive data in transit

Features

Data in Transit Encryption

Description: Service supports data in-transit encryption for data plane. [Learn more](#).

Supported	Enabled By Default	Configuration Responsibility
True	True	Microsoft

Configuration Guidance: No additional configurations are required as this is enabled on a default deployment.

Reference: [Azure Cognitive Services security](#)

DP-4: Enable data at rest encryption by default

Features

Data at Rest Encryption Using Platform Keys

Description: Data at-rest encryption using platform keys is supported, any customer content at rest is encrypted with these Microsoft managed keys. [Learn more](#).

Supported	Enabled By Default	Configuration Responsibility
True	True	Microsoft

Configuration Guidance: No additional configurations are required as this is enabled on a default deployment.

Reference: [Configure customer-managed keys with Azure Key Vault for Cognitive Services](#)

DP-5: Use customer-managed key option in data at rest encryption when required

Features

Data at Rest Encryption Using CMK

Description: Data at-rest encryption using customer-managed keys is supported for customer content stored by the service. [Learn more](#).

Supported	Enabled By Default	Configuration Responsibility
True	False	Customer

Configuration Guidance: If required for regulatory compliance, define the use case and service scope where encryption using customer-managed keys are needed. Enable and implement data at rest encryption using customer-managed key for those services.

Reference: [Configure customer-managed keys with Azure Key Vault for Cognitive Services](#)

Microsoft Defender for Cloud monitoring

Azure Policy built-in definitions - Microsoft.CognitiveServices:

Name	Description	Effect(s)	Version
(Azure portal)			(GitHub)

Name (Azure portal)	Description	Effect(s)	Version (GitHub)
Cognitive Services accounts should enable data encryption with a customer-managed key ↗	Customer-managed keys are commonly required to meet regulatory compliance standards. Customer-managed keys enable the data stored in Cognitive Services to be encrypted with an Azure Key Vault key created and owned by you. You have full control and responsibility for the key lifecycle, including rotation and management. Learn more about customer-managed keys at https://go.microsoft.com/fwlink/?linkid=2121321 .	Audit, Deny, Disabled	2.0.0 ↗

DP-6: Use a secure key management process

Features

Key Management in Azure Key Vault

Description: The service supports Azure Key Vault integration for any customer keys, secrets, or certificates. [Learn more](#).

Supported	Enabled By Default	Configuration Responsibility
True	False	Customer

Configuration Guidance: Use Azure Key Vault to create and control the life cycle of your encryption keys, including key generation, distribution, and storage. Rotate and revoke your keys in Azure Key Vault and your service based on a defined schedule or when there is a key retirement or compromise. When there is a need to use customer-managed key (CMK) in the workload, service, or application level, ensure you follow the best practices for key management: Use a key hierarchy to generate a separate data encryption key (DEK) with your key encryption key (KEK) in your key vault. Ensure keys are registered with Azure Key Vault and referenced via key IDs from the service or application. If you need to bring your own key (BYOK) to the service (such as importing HSM-protected keys from your on-premises HSMs into Azure Key Vault), follow recommended guidelines to perform initial key generation and key transfer.

Reference: [Configure customer-managed keys with Azure Key Vault for Cognitive Services](#)

DP-7: Use a secure certificate management process

Features

Certificate Management in Azure Key Vault

Description: The service supports Azure Key Vault integration for any customer certificates. [Learn more.](#)

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

Configuration Guidance: This feature is not supported to secure this service.

Asset management

For more information, see the [Microsoft cloud security benchmark: Asset management](#).

AM-2: Use only approved services

Features

Azure Policy Support

Description: Service configurations can be monitored and enforced via Azure Policy. [Learn more.](#)

Supported	Enabled By Default	Configuration Responsibility
True	False	Customer

Configuration Guidance: Use Microsoft Defender for Cloud to configure Azure Policy to audit and enforce configurations of your Azure resources. Use Azure Monitor to create alerts when there is a configuration deviation detected on the resources. Use Azure Policy [deny] and [deploy if not exists] effects to enforce secure configuration across Azure resources.

Reference: [Azure Policy built-in policy definitions for Azure Cognitive Services](#)

Logging and threat detection

For more information, see the [Microsoft cloud security benchmark: Logging and threat detection](#).

LT-1: Enable threat detection capabilities

Features

Microsoft Defender for Service / Product Offering

Description: Service has an offering-specific Microsoft Defender solution to monitor and alert on security issues. [Learn more](#).

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

Configuration Guidance: This feature is not supported to secure this service.

LT-4: Enable logging for security investigation

Features

Azure Resource Logs

Description: Service produces resource logs that can provide enhanced service-specific metrics and logging. The customer can configure these resource logs and send them to their own data sink like a storage account or log analytics workspace. [Learn more](#).

Supported	Enabled By Default	Configuration Responsibility
True	False	Customer

Configuration Guidance: Enable resource logs for the service. For example, Key Vault supports additional resource logs for actions that get a secret from a key vault or and Azure SQL has resource logs that track requests to a database. The content of resource logs varies by the Azure service and resource type.

Reference: [Enable diagnostic logging for Azure Cognitive Services](#)

Backup and recovery

For more information, see the [Microsoft cloud security benchmark: Backup and recovery](#).

BR-1: Ensure regular automated backups

Features

Azure Backup

Description: The service can be backed up by the Azure Backup service. [Learn more](#).

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

Configuration Guidance: This feature is not supported to secure this service.

Service Native Backup Capability

Description: Service supports its own native backup capability (if not using Azure Backup). [Learn more](#).

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

Configuration Guidance: This feature is not supported to secure this service.

Next steps

- See the [Microsoft cloud security benchmark overview](#)
- Learn more about [Azure security baselines](#)

Azure Policy Regulatory Compliance controls for Azure Cognitive Services

Article • 01/05/2023 • 10 minutes to read

[Regulatory Compliance in Azure Policy](#) provides Microsoft created and managed initiative definitions, known as *built-ins*, for the **compliance domains** and **security controls** related to different compliance standards. This page lists the **compliance domains** and **security controls** for Azure Cognitive Services. You can assign the built-ins for a **security control** individually to help make your Azure resources compliant with the specific standard.

The title of each built-in policy definition links to the policy definition in the Azure portal. Use the link in the **Policy Version** column to view the source on the [Azure Policy GitHub repo](#).

Important

Each control is associated with one or more [Azure Policy](#) definitions. These policies might help you [assess compliance](#) with the control. However, there often isn't a one-to-one or complete match between a control and one or more policies. As such, **Compliant** in Azure Policy refers only to the policies themselves. This doesn't ensure that you're fully compliant with all requirements of a control. In addition, the compliance standard includes controls that aren't addressed by any Azure Policy definitions at this time. Therefore, compliance in Azure Policy is only a partial view of your overall compliance status. The associations between controls and Azure Policy Regulatory Compliance definitions for these compliance standards can change over time.

Azure Security Benchmark

The [Azure Security Benchmark](#) provides recommendations on how you can secure your cloud solutions on Azure. To see how this service completely maps to the Azure Security Benchmark, see the [Azure Security Benchmark mapping files](#).

To review how the available Azure Policy built-ins for all Azure services map to this compliance standard, see [Azure Policy Regulatory Compliance - Azure Security Benchmark](#).

Domain	Control ID	Control title	Policy (Azure portal)	Policy version (GitHub)
Network Security	NS-2	Secure cloud services with network controls	Cognitive Services accounts should disable public network access ↗	3.0.1 ↗
Network Security	NS-2	Secure cloud services with network controls	Cognitive Services accounts should restrict network access ↗	3.0.0 ↗
Data Protection	DP-5	Use customer-managed key option in data at rest encryption when required	Cognitive Services accounts should enable data encryption with a customer-managed key ↗	2.0.0 ↗

CMMC Level 3

To review how the available Azure Policy built-ins for all Azure services map to this compliance standard, see [Azure Policy Regulatory Compliance - CMMC Level 3](#). For more information about this compliance standard, see [Cybersecurity Maturity Model Certification \(CMMC\) ↗](#).

Domain	Control ID	Control title	Policy (Azure portal)	Policy version (GitHub)
Access Control	AC.1.001	Limit information system access to authorized users, processes acting on behalf of authorized users, and devices (including other information systems).	Cognitive Services accounts should disable public network access ↗	3.0.1 ↗
Access Control	AC.1.001	Limit information system access to authorized users, processes acting on behalf of authorized users, and devices (including other information systems).	Cognitive Services accounts should restrict network access ↗	3.0.0 ↗
Access Control	AC.1.002	Limit information system access to the types of transactions and functions that authorized users are permitted to execute.	Cognitive Services accounts should disable public network access ↗	3.0.1 ↗

Domain	Control ID	Control title	Policy (Azure portal)	Policy version (GitHub)
Access Control	AC.1.002	Limit information system access to the types of transactions and functions that authorized users are permitted to execute.	Cognitive Services accounts should restrict network access ↗	3.0.0 ↗
Access Control	AC.2.016	Control the flow of CUI in accordance with approved authorizations.	Cognitive Services accounts should disable public network access ↗	3.0.1 ↗
Access Control	AC.2.016	Control the flow of CUI in accordance with approved authorizations.	Cognitive Services accounts should restrict network access ↗	3.0.0 ↗
Configuration Management	CM.3.068	Restrict, disable, or prevent the use of nonessential programs, functions, ports, protocols, and services.	Cognitive Services accounts should disable public network access ↗	3.0.1 ↗
Configuration Management	CM.3.068	Restrict, disable, or prevent the use of nonessential programs, functions, ports, protocols, and services.	Cognitive Services accounts should restrict network access ↗	3.0.0 ↗
System and Communications Protection	SC.1.175	Monitor, control, and protect communications (i.e., information transmitted or received by organizational systems) at the external boundaries and key internal boundaries of organizational systems.	Cognitive Services accounts should disable public network access ↗	3.0.1 ↗

Domain	Control ID	Control title	Policy (Azure portal)	Policy version (GitHub)
System and Communications Protection	SC.1.175	Monitor, control, and protect communications (i.e., information transmitted or received by organizational systems) at the external boundaries and key internal boundaries of organizational systems.	Cognitive Services accounts should restrict network access ↗	3.0.0 ↗
System and Communications Protection	SC.3.177	Employ FIPS-validated cryptography when used to protect the confidentiality of CUI.	Cognitive Services accounts should enable data encryption with a customer-managed key ↗	2.0.0 ↗
System and Communications Protection	SC.3.183	Deny network communications traffic by default and allow network communications traffic by exception (i.e., deny all, permit by exception).	Cognitive Services accounts should disable public network access ↗	3.0.1 ↗
System and Communications Protection	SC.3.183	Deny network communications traffic by default and allow network communications traffic by exception (i.e., deny all, permit by exception).	Cognitive Services accounts should restrict network access ↗	3.0.0 ↗

FedRAMP High

To review how the available Azure Policy built-ins for all Azure services map to this compliance standard, see [Azure Policy Regulatory Compliance - FedRAMP High](#). For more information about this compliance standard, see [FedRAMP High](#) ↗.

Domain	Control ID	Control title	Policy (Azure portal)	Policy version (GitHub)
Access Control	AC-2	Account Management	Cognitive Services accounts should have local authentication methods disabled ↗	1.0.0 ↗

Domain	Control ID	Control title	Policy (Azure portal)	Policy version (GitHub)
Access Control	AC-2 (1)	Automated System Account Management	Cognitive Services accounts should have local authentication methods disabled ↗	1.0.0 ↗
Access Control	AC-2 (7)	Role-Based Schemes	Cognitive Services accounts should have local authentication methods disabled ↗	1.0.0 ↗
Access Control	AC-3	Access Enforcement	Cognitive Services accounts should have local authentication methods disabled ↗	1.0.0 ↗
Access Control	AC-4	Information Flow Enforcement	Cognitive Services accounts should disable public network access ↗	3.0.1 ↗
Access Control	AC-4	Information Flow Enforcement	Cognitive Services accounts should restrict network access ↗	3.0.0 ↗
Access Control	AC-4	Information Flow Enforcement	Cognitive Services should use private link ↗	3.0.0 ↗
Access Control	AC-17	Remote Access	Cognitive Services should use private link ↗	3.0.0 ↗
Access Control	AC-17 (1)	Automated Monitoring / Control	Cognitive Services should use private link ↗	3.0.0 ↗
Identification And Authentication	IA-2	Identification And Authentication		
(Organizational Users)	Cognitive Services accounts should have local authentication methods disabled ↗	1.0.0 ↗		
Identification And Authentication	IA-4	Identifier Management	Cognitive Services accounts should have local authentication methods disabled ↗	1.0.0 ↗

Domain	Control ID	Control title	Policy (Azure portal)	Policy version (GitHub)
System And Communications Protection	SC-7	Boundary Protection	Cognitive Services accounts should disable public network access ↗	3.0.1 ↗
System And Communications Protection	SC-7	Boundary Protection	Cognitive Services accounts should restrict network access ↗	3.0.0 ↗
System And Communications Protection	SC-7	Boundary Protection	Cognitive Services should use private link ↗	3.0.0 ↗
System And Communications Protection	SC-7 (3)	Access Points	Cognitive Services accounts should disable public network access ↗	3.0.1 ↗
System And Communications Protection	SC-7 (3)	Access Points	Cognitive Services accounts should restrict network access ↗	3.0.0 ↗
System And Communications Protection	SC-7 (3)	Access Points	Cognitive Services should use private link ↗	3.0.0 ↗
System And Communications Protection	SC-12	Cryptographic Key Establishment And Management	Cognitive Services accounts should enable data encryption with a customer-managed key ↗	2.0.0 ↗

FedRAMP Moderate

To review how the available Azure Policy built-ins for all Azure services map to this compliance standard, see [Azure Policy Regulatory Compliance - FedRAMP Moderate](#). For more information about this compliance standard, see [FedRAMP Moderate ↗](#).

Domain	Control ID	Control title	Policy (Azure portal)	Policy version (GitHub)
Access Control	AC-2	Account Management	Cognitive Services accounts should have local authentication methods disabled ↗	1.0.0 ↗

Domain	Control ID	Control title	Policy (Azure portal)	Policy version (GitHub)
Access Control	AC-2 (1)	Automated System Account Management	Cognitive Services accounts should have local authentication methods disabled	1.0.0 ↗
Access Control	AC-2 (7)	Role-Based Schemes	Cognitive Services accounts should have local authentication methods disabled	1.0.0 ↗
Access Control	AC-3	Access Enforcement	Cognitive Services accounts should have local authentication methods disabled	1.0.0 ↗
Access Control	AC-4	Information Flow Enforcement	Cognitive Services accounts should disable public network access	3.0.1 ↗
Access Control	AC-4	Information Flow Enforcement	Cognitive Services accounts should restrict network access	3.0.0 ↗
Access Control	AC-4	Information Flow Enforcement	Cognitive Services should use private link	3.0.0 ↗
Access Control	AC-17 (1)	Remote Access Automated Monitoring / Control	Cognitive Services should use private link	3.0.0 ↗
Identification And Authentication	IA-2	Identification And Authentication (Organizational Users)	Cognitive Services accounts should have local authentication methods disabled	1.0.0 ↗
Identification And Authentication	IA-4	Identifier Management	Cognitive Services accounts should have local authentication methods disabled	1.0.0 ↗
System And Communications Protection	SC-7	Boundary Protection	Cognitive Services accounts should disable public network access	3.0.1 ↗
System And Communications Protection	SC-7	Boundary Protection	Cognitive Services accounts should restrict network access	3.0.0 ↗

Domain	Control ID	Control title	Policy (Azure portal)	Policy version (GitHub)
System And Communications Protection	SC-7	Boundary Protection	Cognitive Services should use private link ↗	3.0.0 ↗
System And Communications Protection	SC-7 (3)	Access Points	Cognitive Services accounts should disable public network access ↗	3.0.1 ↗
System And Communications Protection	SC-7 (3)	Access Points	Cognitive Services accounts should restrict network access ↗	3.0.0 ↗
System And Communications Protection	SC-7 (3)	Access Points	Cognitive Services should use private link ↗	3.0.0 ↗
System And Communications Protection	SC-12	Cryptographic Key Establishment And Management	Cognitive Services accounts should enable data encryption with a customer-managed key ↗	2.0.0 ↗

New Zealand ISM Restricted

To review how the available Azure Policy built-ins for all Azure services map to this compliance standard, see [Azure Policy Regulatory Compliance - New Zealand ISM Restricted](#). For more information about this compliance standard, see [New Zealand ISM Restricted](#) [↗](#).

Domain	Control ID	Control title	Policy (Azure portal)	Policy version (GitHub)
Cryptography	CR-3	17.1.46 Reducing storage and physical transfer requirements	Cognitive Services accounts should enable data encryption with a customer-managed key ↗	2.0.0 ↗
Gateway security	GS-2	19.1.11 Using Gateways	Cognitive Services accounts should disable public network access ↗	3.0.1 ↗
Gateway security	GS-3	19.1.12 Configuration of Gateways	Cognitive Services accounts should restrict network access ↗	3.0.0 ↗

To review how the available Azure Policy built-ins for all Azure services map to this compliance standard, see [Azure Policy Regulatory Compliance - NIST SP 800-53 Rev. 5](#). For more information about this compliance standard, see [NIST SP 800-53 Rev. 5](#).

Domain	Control ID	Control title	Policy (Azure portal)	Policy version (GitHub)
Access Control	AC-2	Account Management	Cognitive Services accounts should have local authentication methods disabled	1.0.0 ↗
Access Control	AC-2 (1)	Automated System Account Management	Cognitive Services accounts should have local authentication methods disabled	1.0.0 ↗
Access Control	AC-2 (7)	Privileged User Accounts	Cognitive Services accounts should have local authentication methods disabled	1.0.0 ↗
Access Control	AC-3	Access Enforcement	Cognitive Services accounts should have local authentication methods disabled	1.0.0 ↗
Access Control	AC-4	Information Flow Enforcement	Cognitive Services accounts should disable public network access	3.0.1 ↗
Access Control	AC-4	Information Flow Enforcement	Cognitive Services accounts should restrict network access	3.0.0 ↗
Access Control	AC-4	Information Flow Enforcement	Cognitive Services should use private link	3.0.0 ↗
Access Control	AC-17	Remote Access	Cognitive Services should use private link	3.0.0 ↗
Access Control	AC-17 (1)	Monitoring and Control	Cognitive Services should use private link	3.0.0 ↗
Identification and Authentication	IA-2	Identification and Authentication (organizational Users)	Cognitive Services accounts should have local authentication methods disabled	1.0.0 ↗
Identification and Authentication	IA-4	Identifier Management	Cognitive Services accounts should have local authentication methods disabled	1.0.0 ↗

Domain	Control ID	Control title	Policy (Azure portal)	Policy version (GitHub)
System and Communications Protection	SC-7	Boundary Protection	Cognitive Services accounts should disable public network access ↗	3.0.1 ↗
System and Communications Protection	SC-7	Boundary Protection	Cognitive Services accounts should restrict network access ↗	3.0.0 ↗
System and Communications Protection	SC-7	Boundary Protection	Cognitive Services should use private link ↗	3.0.0 ↗
System and Communications Protection	SC-7 (3)	Access Points	Cognitive Services accounts should disable public network access ↗	3.0.1 ↗
System and Communications Protection	SC-7 (3)	Access Points	Cognitive Services accounts should restrict network access ↗	3.0.0 ↗
System and Communications Protection	SC-7 (3)	Access Points	Cognitive Services should use private link ↗	3.0.0 ↗
System and Communications Protection	SC-12	Cryptographic Key Establishment and Management	Cognitive Services accounts should enable data encryption with a customer-managed key ↗	2.0.0 ↗

NZ ISM Restricted v3.5

To review how the available Azure Policy built-ins for all Azure services map to this compliance standard, see [Azure Policy Regulatory Compliance - NZ ISM Restricted v3.5](#). For more information about this compliance standard, see [NZ ISM Restricted v3.5 ↗](#).

Domain	Control ID	Control title	Policy (Azure portal)	Policy version (GitHub)
Cryptography	NZISM Security Benchmark CR-3	17.1.53 Reducing storage and physical transfer requirements	Cognitive Services accounts should enable data encryption with a customer-managed key ↗	2.0.0 ↗

Domain	Control ID	Control title	Policy (Azure portal)	Policy version (GitHub)
Gateway security	NZISM Security Benchmark GS-2	19.1.11 Using Gateways	Cognitive Services accounts should disable public network access ↗	3.0.1 ↗
Gateway security	NZISM Security Benchmark GS-3	19.1.12 Configuration of Gateways	Cognitive Services accounts should restrict network access ↗	3.0.0 ↗

Reserve Bank of India IT Framework for Banks v2016

To review how the available Azure Policy built-ins for all Azure services map to this compliance standard, see [Azure Policy Regulatory Compliance - RBI ITF Banks v2016](#). For more information about this compliance standard, see [RBI ITF Banks v2016 \(PDF\) \[↗\]\(#\)](#).

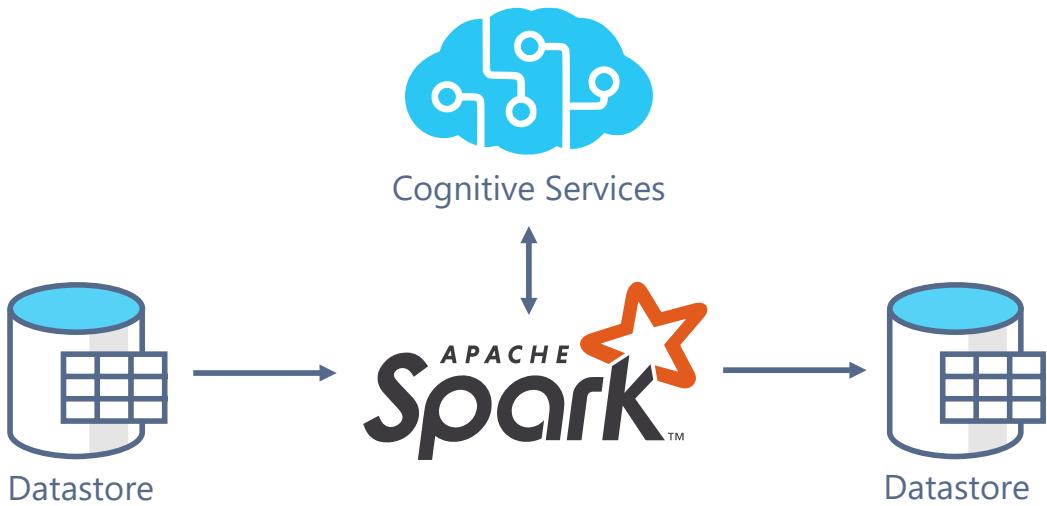
Domain	Control ID	Control title	Policy (Azure portal)	Policy version (GitHub)
Anti-Phishing		Anti-Phishing-14.1	Cognitive Services accounts should disable public network access ↗	3.0.1 ↗
Metrics		Metrics-21.1	Cognitive Services accounts should enable data encryption with a customer-managed key ↗	2.0.0 ↗
Advanced Real-Timethreat Defenceand Management		Advanced Real-Timethreat Defenceand Management-13.4	Cognitive Services accounts should enable data encryption with a customer-managed key ↗	2.0.0 ↗
Metrics		Metrics-21.1	Cognitive Services accounts should enable data encryption with a customer-managed key ↗	2.0.0 ↗
Anti-Phishing		Anti-Phishing-14.1	Cognitive Services accounts should restrict network access ↗	3.0.0 ↗

Next steps

- Learn more about [Azure Policy Regulatory Compliance](#).
- See the built-ins on the [Azure Policy GitHub repo](#)↗.

Azure Cognitive Services for big data

Article • 11/01/2022 • 5 minutes to read



Azure Cognitive Services for big data lets users channel terabytes of data through Cognitive Services using [Apache Spark™](#) and open source libraries for distributed machine learning workloads. With Cognitive Services for big data, it's easy to create large-scale intelligent applications with any datastore.

Using the resources and libraries described in this article, you can embed continuously improving, intelligent models directly into Apache Spark™ and SQL computations. These tools liberate developers from low-level networking details, so that they can focus on creating smart, distributed applications.

Features and benefits

Cognitive Services for big data can use resources from any [supported region](#), as well as [containerized Cognitive Services](#). Containers support low or no connectivity deployments with ultra-low latency responses. Containerized Cognitive Services can be run locally, directly on the worker nodes of your Spark cluster, or on an external orchestrator like Kubernetes.

Supported services

[Cognitive Services](#), accessed through APIs and SDKs, help developers build intelligent applications without having AI or data science skills. With Cognitive Services you can make your applications see, hear, speak, and understand. To use Cognitive Services, your application must send data to the service over the network. Once received, the service

sends an intelligent response in return. The following Cognitive Services resources are available for big data workloads:

Vision

Service Name	Service Description
Computer Vision	The Computer Vision service provides you with access to advanced algorithms for processing images and returning information.
Face	The Face service provides access to advanced face algorithms, enabling face attribute detection and recognition.

Speech

Service Name	Service Description
Speech service	The Speech service provides access to features like speech recognition, speech synthesis, speech translation, and speaker verification and identification.

Decision

Service Name	Service Description
Anomaly Detector	The Anomaly Detector service allows you to monitor and detect abnormalities in your time series data.

Language

Service Name	Service Description
Language service	The Language service provides natural language processing over raw text for sentiment analysis, key-phrase extraction, and language detection.

Search

Service Name	Service Description

Service Name	Service Description
Bing Image Search	The Bing Image Search service returns a display of images determined to be relevant to the user's query.

Supported programming languages for Cognitive Services for big data

Cognitive Services for big data are built on Apache Spark. Apache Spark is a distributed computing library that supports Java, Scala, Python, R, and many other languages. See [SynapseML](#) for documentation, samples, and blog posts.

The following languages are currently supported.

Python

We provide a PySpark API for current and legacy libraries:

- [synapseml.cognitive](#)
- [mmlspark.cognitive](#)

For more information, see the [Python Developer API](#). For usage examples, see the [Python Samples](#).

Scala and Java

We provide a Scala and Java-based Spark API for current and legacy libraries:

- [com.microsoft.synapseml.cognitive](#)
- [com.microsoft.ml.spark.cognitive](#)

For more information, see the [Scala Developer API](#). For usage examples, see the [Scala Samples](#).

Supported platforms and connectors

Big data scenarios require Apache Spark. There are several Apache Spark platforms that support Cognitive Services for big data.

Azure Databricks

[Azure Databricks](#) is an Apache Spark-based analytics platform optimized for the Microsoft Azure cloud services platform. It provides one-click setup, streamlined workflows, and an interactive workspace that supports collaboration between data scientists, data engineers, and business analysts.

Azure Synapse Analytics

[Azure Synapse Analytics](#) is an enterprise data warehouse that uses massive parallel processing. With Synapse Analytics, you can quickly run complex queries across petabytes of data. Azure Synapse Analytics provides managed Spark Pools to run Spark Jobs with an intuitive Jupyter Notebook Interface.

Azure Kubernetes Service

[Azure Kubernetes Service \(AKS\)](#) orchestrates Docker Containers and distributed applications at massive scales. AKS is a managed Kubernetes offering that simplifies using Kubernetes in Azure. Kubernetes can enable fine-grained control of Cognitive Service scale, latency, and networking. However, we recommend using Azure Databricks or Azure Synapse Analytics if you're unfamiliar with Apache Spark.

Data Connectors

Once you have a Spark Cluster, the next step is connecting to your data. Apache Spark has a broad collection of database connectors. These connectors allow applications to work with large datasets no matter where they're stored. For more information about supported databases and connectors, see the [list of supported datasources for Azure Databricks](#).

Concepts

Spark

[Apache Spark™](#) is a unified analytics engine for large-scale data processing. Its parallel processing framework boosts performance of big data and analytic applications. Spark can operate as both a batch and stream processing system, without changing core application code.

The basis of Spark is the DataFrame: a tabular collection of data distributed across the Apache Spark worker nodes. A Spark DataFrame is like a table in a relational database or a data frame in R/Python, but with limitless scale. DataFrames can be constructed from many sources such as: structured data files, tables in Hive, or external databases. Once your data is in a Spark DataFrame, you can:

- Do SQL-style computations such as join and filter tables.
- Apply functions to large datasets using MapReduce style parallelism.
- Apply Distributed Machine Learning using Microsoft Machine Learning for Apache Spark.
- Use Cognitive Services for big data to enrich your data with ready-to-use intelligent services.

Microsoft Machine Learning for Apache Spark (MMLSpark)

[Microsoft Machine Learning for Apache Spark](#) (MMLSpark) is an open-source, distributed machine learning library (ML) built on Apache Spark. Cognitive Services for big data is included in this package. Additionally, MMLSpark contains several other ML tools for Apache Spark, such as LightGBM, Vowpal Wabbit, OpenCV, LIME, and more. With MMLSpark, you can build powerful predictive and analytical models from any Spark datasource.

HTTP on Spark

Cognitive Services for big data is an example of how we can integrate intelligent web services with big data. Web services power many applications across the globe and most services communicate through the Hypertext Transfer Protocol (HTTP). To work with *arbitrary* web services at large scales, we provide HTTP on Spark. With HTTP on Spark, you can pass terabytes of data through any web service. Under the hood, we use this technology to power Cognitive Services for big data.

Developer samples

- [Recipe: Predictive Maintenance](#)
- [Recipe: Intelligent Art Exploration](#)

Blog posts

- [Learn more about how Cognitive Services work on Apache Spark™](#)

- Saving Snow Leopards with Deep Learning and Computer Vision on Spark
- Microsoft Research Podcast: MMLSpark, empowering AI for Good with Mark Hamilton ↗
- Academic Whitepaper: Large Scale Intelligent Microservices ↗

Webinars and videos

- The Azure Cognitive Services on Spark: Clusters with Embedded Intelligent Services ↗
- Spark Summit Keynote: Scalable AI for Good ↗
- Cognitive Services for big data in Azure Cosmos DB ↗
- Lightning Talk on Large Scale Intelligent Microservices ↗

Next steps

- Getting Started with Cognitive Services for big data
- Simple Python Examples
- Simple Scala Examples

Getting started

Article • 10/18/2022 • 4 minutes to read

Setting up your environment is the first step to building a pipeline for your data. After your environment is ready, running a sample is quick and easy.

In this article, you'll perform these steps to get started:

- ✓ [Create a Cognitive Services resource](#)
- ✓ [Create an Apache Spark cluster](#)
- ✓ [Try a sample](#)

Create a Cognitive Services resource

To work with big data in Cognitive Services, first create a Cognitive Services resource for your workflow. There are two main types of Cognitive Services: cloud services hosted in Azure and containerized services managed by users. We recommend starting with the simpler cloud-based Cognitive Services.

Cloud services

Cloud-based Cognitive Services are intelligent algorithms hosted in Azure. These services are ready for use without training, you just need an internet connection. You can [create a Cognitive Service in the Azure portal](#) or with the [Azure CLI](#).

Containerized services (optional)

If your application or workload uses large datasets, requires private networking, or can't contact the cloud, communicating with cloud services might be impossible. In this situation, containerized Cognitive Services have these benefits:

- **Low Connectivity:** You can deploy containerized Cognitive Services in any computing environment, both on-cloud and off. If your application can't contact the cloud, consider deploying containerized Cognitive Services on your application.
- **Low Latency:** Because containerized services don't require the round-trip communication to/from the cloud, responses are returned with much lower latencies.
- **Privacy and Data Security:** You can deploy containerized services into private networks, so that sensitive data doesn't leave the network.

- **High Scalability:** Containerized services don't have "rate limits" and run on user-managed computers. So, you can scale Cognitive Services without end to handle much larger workloads.

Follow [this guide](#) to create a containerized Cognitive Service.

Create an Apache Spark cluster

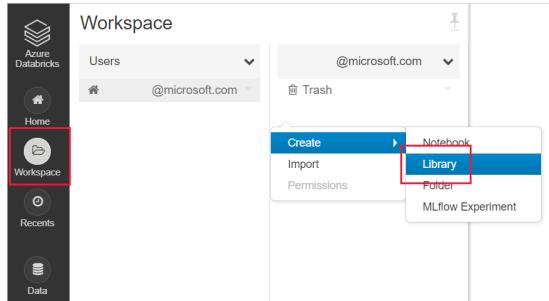
Apache Spark™ [↗](#) is a distributed computing framework designed for big-data data processing. Users can work with Apache Spark in Azure with services like Azure Databricks, Azure Synapse Analytics, HDInsight, and Azure Kubernetes Services. To use the big data Cognitive Services, you must first create a cluster. If you already have a Spark cluster, feel free to try an example.

Azure Databricks

Azure Databricks is an Apache Spark-based analytics platform with a one-click setup, streamlined workflows, and an interactive workspace. It's often used to collaborate between data scientists, engineers, and business analysts. To use the big data Cognitive Services on Azure Databricks, follow these steps:

1. [Create an Azure Databricks workspace](#)
2. [Create a Spark cluster in Databricks](#)
3. Install the SynapseML open-source library (or MMLSpark library if you're supporting a legacy application):

- Create a new library in your databricks workspace



- For SynapseML: input the following maven coordinates Coordinates:
`com.microsoft.azure:synapsem1_2.12:0.10.0` Repository: default
- For MMLSpark (legacy): input the following maven coordinates Coordinates:
`com.microsoft.ml.spark:mmlspark_2.11:1.0.0-rc3` Repository:

<https://mmlspark.azureedge.net/maven>

Create Library

Library Source
Upload DBFS PyPI Maven CRAN

Coordinates
com.microsoft.ml.spark:mmlspark_2.11:1.0.0-rc1

Repository

Exclusions
Dependencies to exclude (log4j:log4j:junit:junit)

- Install the library onto a cluster

Status on running clusters

Install automatically on all clusters This option does not work on clusters with Runtime version >= 7.0 Learn more

Click Uninstall or Install to apply action on the selected item(s)

Status	Cluster Name	Message
<input checked="" type="checkbox"/> Status	mmlspark-test	
<input checked="" type="checkbox"/> Installed		

Azure Synapse Analytics (optional)

Optionally, you can use Synapse Analytics to create a spark cluster. Azure Synapse Analytics brings together enterprise data warehousing and big data analytics. It gives you the freedom to query data on your terms, using either serverless on-demand or provisioned resources at scale. To get started using Azure Synapse Analytics, follow these steps:

1. [Create a Synapse Workspace \(preview\)](#).
2. [Create a new serverless Apache Spark pool \(preview\) using the Azure portal](#).

In Azure Synapse Analytics, big data for Cognitive Services is installed by default.

Azure Kubernetes Service

If you're using containerized Cognitive Services, one popular option for deploying Spark alongside containers is the Azure Kubernetes Service.

To get started on Azure Kubernetes Service, follow these steps:

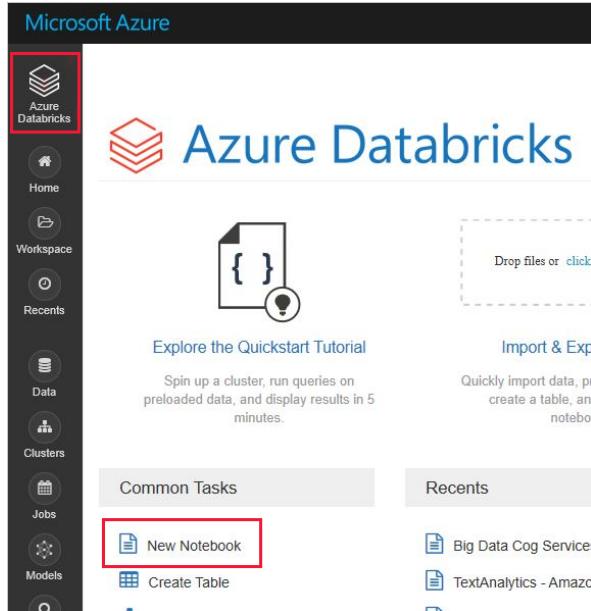
1. [Deploy an Azure Kubernetes Service \(AKS\) cluster using the Azure portal](#)
2. [Install the Apache Spark 2.4.0 helm chart](#) ↗
3. [Install a cognitive service container using Helm](#)

Try a sample

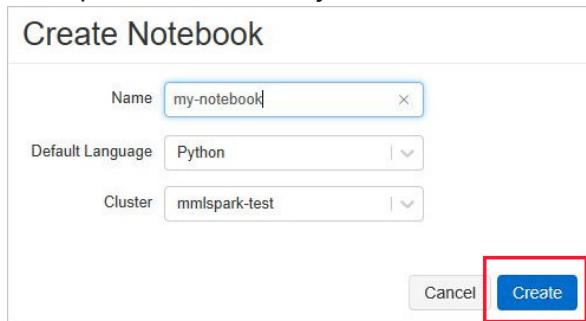
After you set up your Spark cluster and environment, you can run a short sample. This sample assumes Azure Databricks and the `mmlspark.cognitive` package. For an example using `synapseml.cognitive`, see [Add search to AI-enriched data from Apache Spark using SynapseML](#).

First, you can create a notebook in Azure Databricks. For other Spark cluster providers, use their notebooks or Spark Submit.

1. Create a new Databricks notebook, by choosing **New Notebook** from the **Azure Databricks** menu.



2. In the **Create Notebook**, enter a name, select **Python** as the language, and select the Spark cluster that you created earlier.



Select **Create**.

3. Paste this code snippet into your new notebook.

```
Python

from mmlspark.cognitive import *
from pyspark.sql.functions import col

# Add your region and subscription key from the Language service (or a
# general Cognitive Service key)
# If using a multi-region Cognitive Services resource, delete the
```

```

placeholder text: service_region = ""
service_key = "ADD-SUBSCRIPTION-KEY-HERE"
service_region = "ADD-SERVICE-REGION-HERE"

df = spark.createDataFrame([
    ("I am so happy today, its sunny!", "en-US"),
    ("I am frustrated by this rush hour traffic", "en-US"),
    ("The cognitive services on spark aint bad", "en-US"),
], ["text", "language"])

sentiment = (TextSentiment()
    .setTextCol("text")
    .setLocation(service_region)
    .setSubscriptionKey(service_key)
    .setOutputCol("sentiment")
    .setErrorCol("error")
    .setLanguageCol("language"))

results = sentiment.transform(df)

# Show the results in a table
display(results.select("text", col("sentiment")
[0].getItem("score").alias("sentiment")))

```

4. Get your region and subscription key from the **Keys and Endpoint** menu from your Language resource in the Azure portal.
5. Replace the region and subscription key placeholders in your Databricks notebook code with values that are valid for your resource.
6. Select the play, or triangle, symbol in the upper right of your notebook cell to run the sample. Optionally, select **Run All** at the top of your notebook to run all cells. The answers will display below the cell in a table.

Expected results

text	sentiment
I am so happy today, its sunny!	0.978959
I am frustrated by this rush hour traffic	0.0237956
The cognitive services on spark aint bad	0.888896

Next steps

- [Short Python Examples](#)

- Short Scala Examples
- Recipe: Predictive Maintenance
- Recipe: Intelligent Art Exploration

Python Samples for Cognitive Services for big data

Article • 11/01/2022 • 5 minutes to read

The following snippets are ready to run and will help get you started with using Cognitive Services on Spark with Python.

The samples in this article use these Cognitive Services:

- Language service - get the sentiment (or mood) of a set of sentences.
- Computer Vision - get the tags (one-word descriptions) associated with a set of images.
- Speech-to-text - transcribe audio files to extract text-based transcripts.
- Anomaly Detector - detect anomalies within a time series data.

Prerequisites

1. Follow the steps in [Getting started](#) to set up your Azure Databricks and Cognitive Services environment. This tutorial shows you how to install MMLSpark and how to create your Spark cluster in Databricks.
2. After you create a new notebook in Azure Databricks, copy the **Shared code** below and paste into a new cell in your notebook.
3. Choose a service sample, below, and copy paste it into a second new cell in your notebook.
4. Replace any of the service subscription key placeholders with your own key.
5. Choose the run button (triangle icon) in the upper right corner of the cell, then select **Run Cell**.
6. View results in a table below the cell.

Shared code

To get started, we'll need to add this code to the project:

Python

```
from mmlspark.cognitive import *

# A general Cognitive Services key for the Language service and Computer Vision (or use separate
# keys that belong to each service)
service_key = "ADD_YOUR_SUBSCRIPTION_KEY"
# An Anomaly Dector subscription key
anomaly_key = "ADD_YOUR_SUBSCRIPTION_KEY"

# Validate the key
assert service_key != "ADD_YOUR_SUBSCRIPTION_KEY"
```

Language service sample

The [Language service](#) provides several algorithms for extracting intelligent insights from text. For example, we can find the sentiment of given input text. The service will return a score between 0.0 and 1.0 where low scores indicate negative sentiment and high score indicates positive sentiment. This sample uses three simple sentences and returns the sentiment for each.

Python

```
from pyspark.sql.functions import col

# Create a dataframe that's tied to its column names
```

```

df = spark.createDataFrame([
    ("I am so happy today, its sunny!", "en-US"),
    ("I am frustrated by this rush hour traffic", "en-US"),
    ("The cognitive services on spark aint bad", "en-US"),
], ["text", "language"])

# Run the Language service with options
sentiment = (TextSentiment()
    .setTextCol("text")
    .setLocation("eastus")
    .setSubscriptionKey(service_key)
    .setOutputCol("sentiment")
    .setErrorCol("error")
    .setLanguageCol("language"))

# Show the results of your text query in a table format
display(sentiment.transform(df).select("text", col("sentiment")
[0].getItem("sentiment").alias("sentiment")))

```

Expected result

text	sentiment
I am so happy today, its sunny!	positive
I am frustrated by this rush hour traffic	negative
The cognitive services on spark aint bad	positive

Computer Vision sample

Computer Vision analyzes images to identify structure such as faces, objects, and natural-language descriptions. In this sample, we tag a list of images. Tags are one-word descriptions of things in the image like recognizable objects, people, scenery, and actions.

Python

```

# Create a dataframe with the image URLs
df = spark.createDataFrame([
    ("https://raw.githubusercontent.com/Azure-Samples/cognitive-services-sample-data-
files/master/ComputerVision/Images/objects.jpg", ),
    ("https://raw.githubusercontent.com/Azure-Samples/cognitive-services-sample-data-
files/master/ComputerVision/Images/dog.jpg", ),
    ("https://raw.githubusercontent.com/Azure-Samples/cognitive-services-sample-data-
files/master/ComputerVision/Images/house.jpg", )
], ["image", ])

# Run the Computer Vision service. Analyze Image extracts information from/about the images.
analysis = (AnalyzeImage()
    .setLocation("eastus")
    .setSubscriptionKey(service_key)
    .setVisualFeatures(["Categories", "Color", "Description", "Faces", "Objects", "Tags"])
    .setOutputCol("analysis_results")
    .setImageUrlCol("image")
    .setErrorCol("error"))

# Show the results of what you wanted to pull out of the images.
display(analysis.transform(df).select("image", "analysis_results.description.tags"))

```

Expected result

image	tags
https://raw.githubusercontent.com/Azure-Samples/cognitive-services-sample-data-files/master/ComputerVision/Images/objects.jpg ↗	['skating' 'person' 'man' 'outdoor' 'riding' 'sport' 'skateboard' 'young' 'board' 'shirt' 'air' 'black' 'park' 'boy' 'side' 'jumping' 'trick' 'ramp' 'doing' 'flying']
https://raw.githubusercontent.com/Azure-Samples/cognitive-services-sample-data-files/master/ComputerVision/Images/dog.jpg ↗	['dog' 'outdoor' 'fence' 'wooden' 'small' 'brown' 'building' 'sitting' 'front' 'bench' 'standing' 'table' 'walking' 'board' 'beach' 'white' 'holding' 'bridge' 'track']
https://raw.githubusercontent.com/Azure-Samples/cognitive-services-sample-data-files/master/ComputerVision/Images/house.jpg ↗	['outdoor' 'grass' 'house' 'building' 'old' 'home' 'front' 'small' 'church' 'stone' 'large' 'grazing' 'yard' 'green' 'sitting' 'leading' 'sheep' 'brick' 'bench' 'street' 'white' 'country' 'clock' 'sign' 'parked' 'field' 'standing' 'garden' 'water' 'red' 'horse' 'man' 'tall' 'fire' 'group']

Speech-to-Text sample

The [Speech-to-text](#) service converts streams or files of spoken audio to text. In this sample, we transcribe two audio files. The first file is easy to understand, and the second is more challenging.

Python

```
# Create a dataframe with our audio URLs, tied to the column called "url"
df = spark.createDataFrame([( "https://mmlspark.blob.core.windows.net/datasets/Speech/audio2.wav" ,),
                           ("https://mmlspark.blob.core.windows.net/datasets/Speech/audio3.mp3" ,)
                          ], [ "url" ])

# Run the Speech-to-text service to translate the audio into text
speech_to_text = (SpeechToTextSDK()
                  .setSubscriptionKey(service_key)
                  .setLocation("eastus")
                  .setOutputCol("text")
                  .setAudioDataCol("url")
                  .setLanguage("en-US")
                  .setProfanity("Masked"))

# Show the results of the translation
display(speech_to_text.transform(df).select("url", "text.DisplayText"))
```

Expected result

url	DisplayText
https://mmlspark.blob.core.windows.net/datasets/Speech/audio2.wav ↗	Custom speech provides tools that allow you to visually inspect the recognition quality of a model by comparing audio data with the corresponding recognition result from the custom speech portal. You can playback uploaded audio and determine if the provided recognition result is correct. This tool allows you to quickly inspect quality of Microsoft's baseline speech to text model or a trained custom model without having to transcribe any audio data.
https://mmlspark.blob.core.windows.net/datasets/Speech/audio3.mp3 ↗	Add a gentleman Sir thinking visual check.
https://mmlspark.blob.core.windows.net/datasets/Speech/audio3.mp3 ↗	I hear me.

url	DisplayText
https://mmlspark.blob.core.windows.net/datasets/Speech/audio3.mp3	I like the reassurance for radio that I can hear it as well.

Anomaly Detector sample

[Anomaly Detector](#) is great for detecting irregularities in your time series data. In this sample, we use the service to find anomalies in the entire time series.

Python

```
from pyspark.sql.functions import lit

# Create a dataframe with the point data that Anomaly Detector requires
df = spark.createDataFrame([
    ("1972-01-01T00:00:00Z", 826.0),
    ("1972-02-01T00:00:00Z", 799.0),
    ("1972-03-01T00:00:00Z", 890.0),
    ("1972-04-01T00:00:00Z", 900.0),
    ("1972-05-01T00:00:00Z", 766.0),
    ("1972-06-01T00:00:00Z", 805.0),
    ("1972-07-01T00:00:00Z", 821.0),
    ("1972-08-01T00:00:00Z", 20000.0),
    ("1972-09-01T00:00:00Z", 883.0),
    ("1972-10-01T00:00:00Z", 898.0),
    ("1972-11-01T00:00:00Z", 957.0),
    ("1972-12-01T00:00:00Z", 924.0),
    ("1973-01-01T00:00:00Z", 881.0),
    ("1973-02-01T00:00:00Z", 837.0),
    ("1973-03-01T00:00:00Z", 9000.0)
], ["timestamp", "value"]).withColumn("group", lit("series1"))

# Run the Anomaly Detector service to look for irregular data
anomaly_detector = (SimpleDetectAnomalies()
    .setSubscriptionKey(anomaly_key)
    .setLocation("eastus")
    .setTimestampCol("timestamp")
    .setValueCol("value")
    .setOutputCol("anomalies")
    .setGroupbyCol("group")
    .setGranularity("monthly"))

# Show the full results of the analysis with the anomalies marked as "True"
display(anomaly_detector.transform(df).select("timestamp", "value", "anomalies.isAnomaly"))
```

Expected result

timestamp	value	isAnomaly
1972-01-01T00:00:00Z	826	False
1972-02-01T00:00:00Z	799	False
1972-03-01T00:00:00Z	890	False
1972-04-01T00:00:00Z	900	False
1972-05-01T00:00:00Z	766	False
1972-06-01T00:00:00Z	805	False

timestamp	value	isAnomaly
1972-07-01T00:00:00Z	821	False
1972-08-01T00:00:00Z	20000	True
1972-09-01T00:00:00Z	883	False
1972-10-01T00:00:00Z	898	False
1972-11-01T00:00:00Z	957	False
1972-12-01T00:00:00Z	924	False
1973-01-01T00:00:00Z	881	False
1973-02-01T00:00:00Z	837	False
1973-03-01T00:00:00Z	9000	True

Arbitrary web APIs

With HTTP on Spark, any web service can be used in your big data pipeline. In this example, we use the [World Bank API](#) to get information about various countries around the world.

Python

```
from requests import Request
from mmlspark.io.http import HTTPTransformer, http_udf
from pyspark.sql.functions import udf, col

# Use any requests from the Python requests library
def world_bank_request(country):
    return Request("GET", "http://api.worldbank.org/v2/country/{}?format=json".format(country))

# Create a dataframe with specifics which countries we want data on
df = (spark.createDataFrame([('br',), ('usa',)], ["country"])
      .withColumn("request", http_udf(world_bank_request)(col("country"))))

# Much faster for big data because of the concurrency :
client = (HTTPTransformer()
           .setConcurrency(3)
           .setInputCol("request")
           .setOutputCol("response"))

# Get the body of the response
def get_response_body(resp):
    return resp.entity.content.decode()

# Show the details of the country data returned
display(client.transform(df).select("country", udf(get_response_body)
                                      (col("response")).alias("response")))
```

Expected result

country	response
---------	----------

country response

br	[{"page":1,"pages":1,"per_page":"50","total":1},[{"id":"BRA","iso2Code":"BR","name":"Brazil","region":[{"id":"LCN","iso2code":"ZJ","value":"Latin America & Caribbean "}, {"adminregion":[{"id":"LAC","iso2code":"XJ","value":"Latin America & Caribbean (excluding high income)"}], "incomeLevel":[{"id":"UMC","iso2code":"XT","value":"Upper middle income"}], "lendingType":[{"id":"IBD","iso2code":"XF","value":"IBRD"}], "capitalCity":"Brasilia","longitude": -47.9292,"latitude": -15.7801}]]
usa	[{"page":1,"pages":1,"per_page":"50","total":1},[{"id":"USA","iso2Code":"US","name":"United States","region":[{"id":"NAC","iso2code":"XU","value":"North America"}, {"adminregion":[{"id":"","iso2code":"","value":""}], "incomeLevel":[{"id":"HIC","iso2code":"XD","value":"High income"}], "lendingType":[{"id":"LNX","iso2code":"XX","value":"Not classified"}], "capitalCity":"Washington D.C.","longitude": -77.032,"latitude": 38.8895}]]

See also

- [Recipe: Anomaly Detection](#)
- [Recipe: Art Explorer](#)

Quick Examples

Article • 11/01/2022 • 4 minutes to read

The following snippets are ready to run and will help get you started with using Cognitive Services on Spark. The samples below are in Scala.

The samples use these Cognitive Services:

- Language service - get the sentiment (or mood) of a set of sentences.
- Computer Vision - get the tags (one-word descriptions) associated with a set of images.
- Speech-to-text - transcribe audio files to extract text-based transcripts.
- Anomaly Detector - detect anomalies within a time series data.

Prerequisites

1. Follow the steps in [Getting started](#) to set up your Azure Databricks and Cognitive Services environment. This tutorial will include how to install MMLSpark and how to create your Spark cluster in Databricks.
2. After you create a new notebook in Azure Databricks, copy the **Shared code** below and paste into a new cell in your notebook.
3. Choose a service sample, below, and copy paste it into a second new cell in your notebook.
4. Replace any of the service subscription key placeholders with your own key.
5. Choose the run button (triangle icon) in the upper right corner of the cell, then select **Run Cell**.
6. View results in a table below the cell.

Shared code

To get started, add this code to your project:

```
import com.microsoft.ml.spark.cognitive._  
import spark.implicits._  
  
val serviceKey = "ADD-YOUR-SUBSCRIPTION-KEY"  
val location = "eastus"
```

Language service

The [Language service](#) provides several algorithms for extracting intelligent insights from text. For example, we can find the sentiment of given input text. The service will return a score between `0.0` and `1.0` where low scores indicate negative sentiment and high score indicates positive sentiment. The sample below uses three simple sentences and returns the sentiment score for each.

```
import org.apache.spark.sql.functions.col

val df = Seq(
  ("I am so happy today, its sunny!", "en-US"),
  ("I am frustrated by this rush hour traffic", "en-US"),
  ("The cognitive services on spark aint bad", "en-US")
).toDF("text", "language")

val sentiment = new TextSentiment()
  .setTextCol("text")
  .setLocation(location)
  .setSubscriptionKey(serviceKey)
  .setOutputCol("sentiment")
  .setErrorCol("error")
  .setLanguageCol("language")

display(sentiment.transform(df).select(col("text"), col("sentiment")
(0).getItem("score").alias("sentiment"))))
```

Expected result

text	sentiment
I am so happy today, its sunny!	0.9789592027664185
I am frustrated by this rush hour traffic	0.023795604705810547
The cognitive services on spark aint bad	0.8888956308364868

Computer Vision

[Computer Vision](#) analyzes images to identify structure such as faces, objects, and natural-language descriptions. In this sample, we tag a list of images. Tags are one-word descriptions of things in the image like recognizable objects, people, scenery, and actions.

```

// Create a dataframe with the image URLs
val df = Seq(
    ("https://raw.githubusercontent.com/Azure-Samples/cognitive-services-
sample-data-files/master/ComputerVision/Images/objects.jpg"),
    ("https://raw.githubusercontent.com/Azure-Samples/cognitive-services-
sample-data-files/master/ComputerVision/Images/dog.jpg"),
    ("https://raw.githubusercontent.com/Azure-Samples/cognitive-services-
sample-data-files/master/ComputerVision/Images/house.jpg")
).toDF("image")

// Run the Computer Vision service. Analyze Image extracts information
from/about the images.
val analysis = new AnalyzeImage()
    .setLocation(location)
    .setSubscriptionKey(serviceKey)

    .setVisualFeatures(Seq("Categories","Color","Description","Faces","Objects",
"Tags"))
    .setOutputCol("results")
    .setImageUrlCol("image")
    .setErrorCol("error"))

// Show the results of what you wanted to pull out of the images.
display(analysis.transform(df).select(col("image"),
col("results").getItem("tags").getItem("name")).alias("results")))

// Uncomment for full results with all visual feature requests
//display(analysis.transform(df).select(col("image"), col("results")))

```

Expected result

image	tags
https://raw.githubusercontent.com/Azure-Samples/cognitive-services-sample-data-files/master/ComputerVision/Images/objects.jpg ↗	['skating' 'person' 'man' 'outdoor' 'riding' 'sport' 'skateboard' 'young' 'board' 'shirt' 'air' 'black' 'park' 'boy' 'side' 'jumping' 'trick' 'ramp' 'doing' 'flying']
https://raw.githubusercontent.com/Azure-Samples/cognitive-services-sample-data-files/master/ComputerVision/Images/dog.jpg ↗	['dog' 'outdoor' 'fence' 'wooden' 'small' 'brown' 'building' 'sitting' 'front' 'bench' 'standing' 'table' 'walking' 'board' 'beach' 'white' 'holding' 'bridge' 'track']

image	tags
https://raw.githubusercontent.com/Azure-Samples/cognitive-services-sample-data-files/master/ComputerVision/Images/house.jpg	['outdoor' 'grass' 'house' 'building' 'old' 'home' 'front' 'small' 'church' 'stone' 'large' 'grazing' 'yard' 'green' 'sitting' 'leading' 'sheep' 'brick' 'bench' 'street' 'white' 'country' 'clock' 'sign' 'parked' 'field' 'standing' 'garden' 'water' 'red' 'horse' 'man' 'tall' 'fire' 'group']

Speech-to-Text

The [Speech-to-text](#) service converts streams or files of spoken audio to text. In this sample, we transcribe two audio files. The first file is easy to understand, and the second is more challenging.

```
import org.apache.spark.sql.functions.col

// Create a dataframe with audio URLs, tied to the column called "url"
val df =
Seq(("https://mmlspark.blob.core.windows.net/datasets/Speech/audio2.wav"),
("https://mmlspark.blob.core.windows.net/datasets/Speech/audio3.mp3")).toDF(
"url")

// Run the Speech-to-text service to translate the audio into text
val speechToText = new SpeechToTextSDK()
.setSubscriptionKey(serviceKey)
.setLocation("eastus")
.setOutputCol("text")
.setAudioDataCol("url")
.setLanguage("en-US")
.setProfanity("Masked")

// Show the results of the translation
display(speechToText.transform(df).select(col("url"),
col("text").getItem("DisplayText")))
```

Expected result

url	DisplayText

url	DisplayText
https://mmlspark.blob.core.windows.net/datasets/Speech/audio2.wav	Custom speech provides tools that allow you to visually inspect the recognition quality of a model by comparing audio data with the corresponding recognition result from the custom speech portal. You can playback uploaded audio and determine if the provided recognition result is correct. This tool allows you to quickly inspect quality of Microsoft's baseline speech to text model or a trained custom model without having to transcribe any audio data.
https://mmlspark.blob.core.windows.net/datasets/Speech/audio3.mp3	Add a gentleman Sir thinking visual check.
https://mmlspark.blob.core.windows.net/datasets/Speech/audio3.mp3	I hear me.
https://mmlspark.blob.core.windows.net/datasets/Speech/audio3.mp3	I like the reassurance for radio that I can hear it as well.

Anomaly Detector

[Anomaly Detector](#) is great for detecting irregularities in your time series data. In this sample, we use the service to find anomalies in the entire time series.

```
import org.apache.spark.sql.functions.{col, lit}

val anomalyKey = "84a2c303cc7e49f6a44d692c27fb9967"

val df = Seq(
```

```

        ("1972-01-01T00:00:00Z", 826.0),
        ("1972-02-01T00:00:00Z", 799.0),
        ("1972-03-01T00:00:00Z", 890.0),
        ("1972-04-01T00:00:00Z", 900.0),
        ("1972-05-01T00:00:00Z", 766.0),
        ("1972-06-01T00:00:00Z", 805.0),
        ("1972-07-01T00:00:00Z", 821.0),
        ("1972-08-01T00:00:00Z", 20000.0),
        ("1972-09-01T00:00:00Z", 883.0),
        ("1972-10-01T00:00:00Z", 898.0),
        ("1972-11-01T00:00:00Z", 957.0),
        ("1972-12-01T00:00:00Z", 924.0),
        ("1973-01-01T00:00:00Z", 881.0),
        ("1973-02-01T00:00:00Z", 837.0),
        ("1973-03-01T00:00:00Z", 9000.0)
    ).toDF("timestamp", "value").withColumn("group", lit("series1"))

// Run the Anomaly Detector service to look for irregular data
val anamolyDetector = new SimpleDetectAnomalies()
    .setSubscriptionKey(anomalyKey)
    .setLocation("eastus")
    .setTimestampCol("timestamp")
    .setValueCol("value")
    .setOutputCol("anomalies")
    .setGroupbyCol("group")
    .setGranularity("monthly")

// Show the full results of the analysis with the anomalies marked as "True"
display(anamolyDetector.transform(df).select("timestamp", "value",
    "anomalies.isAnomaly"))

```

Expected result

timestamp	value	isAnomaly
1972-01-01T00:00:00Z	826	False
1972-02-01T00:00:00Z	799	False
1972-03-01T00:00:00Z	890	False
1972-04-01T00:00:00Z	900	False
1972-05-01T00:00:00Z	766	False
1972-06-01T00:00:00Z	805	False
1972-07-01T00:00:00Z	821	False
1972-08-01T00:00:00Z	20000	True
1972-09-01T00:00:00Z	883	False

timestamp	value	isAnomaly
1972-10-01T00:00:00Z	898	False
1972-11-01T00:00:00Z	957	False
1972-12-01T00:00:00Z	924	False
1973-01-01T00:00:00Z	881	False
1973-02-01T00:00:00Z	837	False
1973-03-01T00:00:00Z	9000	True

Recipe: Predictive maintenance with the Cognitive Services for big data

Article • 10/12/2022 • 3 minutes to read

This recipe shows how you can use Azure Synapse Analytics and Cognitive Services on Apache Spark for predictive maintenance of IoT devices. We'll follow along with the [Azure Cosmos DB and Synapse Link](#) sample. To keep things simple, in this recipe we'll read the data straight from a CSV file rather than getting streamed data through Azure Cosmos DB and Synapse Link. We strongly encourage you to look over the Synapse Link sample.

Hypothetical scenario

The hypothetical scenario is a Power Plant, where IoT devices are monitoring [steam turbines](#). The IoTSignals collection has Revolutions per minute (RPM) and Megawatts (MW) data for each turbine. Signals from steam turbines are being analyzed and anomalous signals are detected.

There could be outliers in the data in random frequency. In those situations, RPM values will go up and MW output will go down, for circuit protection. The idea is to see the data varying at the same time, but with different signals.

Prerequisites

- An Azure subscription - [Create one for free](#)
- [Azure Synapse workspace](#) configured with a [serverless Apache Spark pool](#)

Setup

Create an Anomaly Detector resource

Azure Cognitive Services are represented by Azure resources that you subscribe to. Create a resource for Translator using the [Azure portal](#) or [Azure CLI](#). You can also:

- View an existing resource in the [Azure portal](#).

Make note of the endpoint and the key for this resource, you'll need it in this guide.

Enter your service keys

Let's start by adding your key and location.

Python

```
service_key = None # Paste your anomaly detector key here
location = None # Paste your anomaly detector location here

assert (service_key is not None)
assert (location is not None)
```

Read data into a DataFrame

Next, let's read the IoTSignals file into a DataFrame. Open a new notebook in your Synapse workspace and create a DataFrame from the file.

Python

```
df_signals =
spark.read.csv("wasbs://publicwasb@mmlspark.blob.core.windows.net/iot/IoTSig
nals.csv", header=True, inferSchema=True)
```

Run anomaly detection using Cognitive Services on Spark

The goal is to find instances where the signals from the IoT devices were outputting anomalous values so that we can see when something is going wrong and do predictive maintenance. To do that, let's use Anomaly Detector on Spark:

Python

```
from pyspark.sql.functions import col, struct
from mmlspark.cognitive import SimpleDetectAnomalies
from mmlspark.core.spark import FluentAPI

detector = (SimpleDetectAnomalies()
    .setSubscriptionKey(service_key)
    . setLocation(location)
    .setOutputCol("anomalies")
    .setGroupbyCol("grouping")
    .setSensitivity(95)
    .setGranularity("secondly"))

df_anomaly = (df_signals
    .where(col("unitSymbol") == 'RPM')
    .withColumn("timestamp", col("dateTime").cast("string")))
```

```

.withColumn("value", col("measureValue").cast("double"))
.withColumn("grouping", struct("deviceId"))
.mlTransform(detector).cache()

df_anomaly.createOrReplaceTempView('df_anomaly')

```

Let's take a look at the data:

Python

```
df_anomaly.select("timestamp", "value", "deviceId", "anomalies.isAnomaly").show(3)
```

This cell should yield a result that looks like:

timestamp	value	deviceId	isAnomaly
2020-05-01 18:33:51	3174	dev-7	False
2020-05-01 18:33:52	2976	dev-7	False
2020-05-01 18:33:53	2714	dev-7	False

Visualize anomalies for one of the devices

IoTSignals.csv has signals from multiple IoT devices. We'll focus on a specific device and visualize anomalous outputs from the device.

Python

```

df_anomaly_single_device = spark.sql("""
select
    timestamp,
    measureValue,
    anomalies.expectedValue,
    anomalies.expectedValue + anomalies.upperMargin as expectedUpperValue,
    anomalies.expectedValue - anomalies.lowerMargin as expectedLowerValue,
    case when anomalies.isAnomaly=true then 1 else 0 end as isAnomaly
from
    df_anomaly
where deviceid = 'dev-1' and timestamp < '2020-04-29'
order by timestamp
limit 200""")

```

Now that we have created a dataframe that represents the anomalies for a particular device, we can visualize these anomalies:

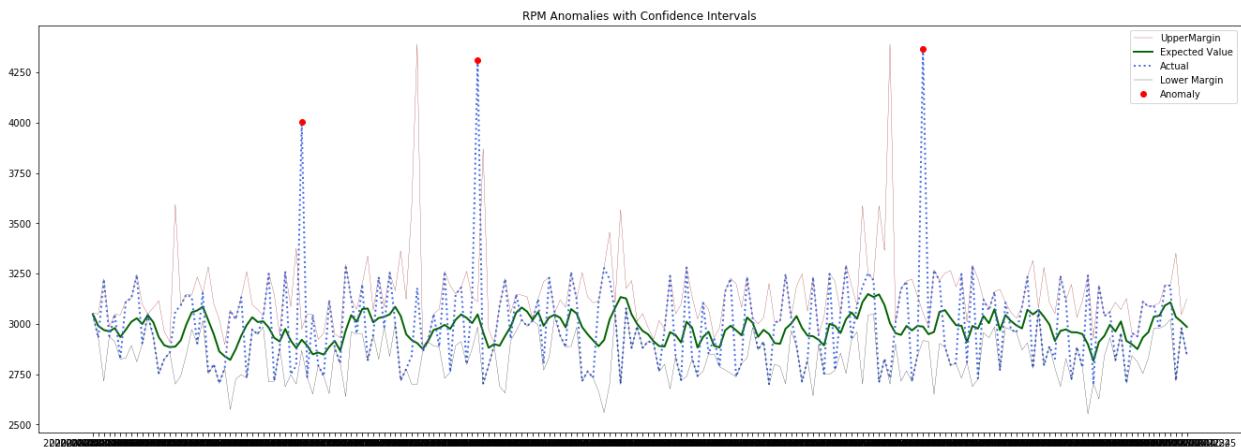
Python

```
import matplotlib.pyplot as plt
from pyspark.sql.functions import col

adf = df_anomaly_single_device.toPandas()
adf_subset = df_anomaly_single_device.where(col("isAnomaly") == 1).toPandas()

plt.figure(figsize=(23,8))
plt.plot(adf['timestamp'],adf['expectedUpperValue'], color='darkred',
linestyle='solid', linewidth=0.25, label='UpperMargin')
plt.plot(adf['timestamp'],adf['expectedValue'], color='darkgreen',
linestyle='solid', linewidth=2, label='Expected Value')
plt.plot(adf['timestamp'],adf['measureValue'], 'b', color='royalblue',
linestyle='dotted', linewidth=2, label='Actual')
plt.plot(adf['timestamp'],adf['expectedLowerValue'], color='black',
linestyle='solid', linewidth=0.25, label='Lower Margin')
plt.plot(adf_subset['timestamp'],adf_subset['measureValue'], 'ro', label =
'Anomaly')
plt.legend()
plt.title('RPM Anomalies with Confidence Intervals')
plt.show()
```

If successful, your output will look like this:



Next steps

Learn how to do predictive maintenance at scale with Azure Cognitive Services, Azure Synapse Analytics, and Azure Cosmos DB. For more information, see the full sample on [GitHub](#).

Recipe: Intelligent Art Exploration with the Cognitive Services for big data

Article • 08/17/2022 • 2 minutes to read

In this example, we'll use the Cognitive Services for big data to add intelligent annotations to the Open Access collection from the Metropolitan Museum of Art (MET). This will enable us to create an intelligent search engine using Azure Search even without manual annotations.

Prerequisites

- You must have a subscription key for Computer Vision and Cognitive Search. Follow the instructions in [Create a Cognitive Services account](#) to subscribe to Computer Vision and get your key.

ⓘ Note

For pricing information, see [Azure Cognitive Search](#).

Import Libraries

Run the following command to import libraries for this recipe.

Python

```
import os, sys, time, json, requests
from pyspark.ml import Transformer, Estimator, Pipeline
from pyspark.ml.feature import SQLTransformer
from pyspark.sql.functions import lit, udf, col, split
```

Set up Subscription Keys

Run the following command to set up variables for service keys. Insert your subscription keys for Computer Vision and Azure Cognitive Search.

Python

```
VISION_API_KEY = 'INSERT_COMPUTER_VISION_SUBSCRIPTION_KEY'
AZURE_SEARCH_KEY = 'INSERT_AZURE_COGNITIVE_SEARCH_SUBSCRIPTION_KEY'
```

```
search_service = "mmlspark-azure-search"
search_index = "test"
```

Read the Data

Run the following command to load data from the MET's Open Access collection.

Python

```
data = spark.read\
    .format("csv")\
    .option("header", True)\

    .load("wasbs://publicwasb@mmlspark.blob.core.windows.net/metartworks_sample.csv")\
    .withColumn("searchAction", lit("upload"))\
    .withColumn("Neighbors", split(col("Neighbors"),
",,").cast("array<string>"))\
    .withColumn("Tags", split(col("Tags"), ",").cast("array<string>"))\
    .limit(25)
```

Analyze the Images

Run the following command to use Computer Vision on the MET's Open Access artworks collection. As a result, you'll get visual features from the artworks.

Python

```
from mmlspark.cognitive import AnalyzeImage
from mmlspark.stages import SelectColumns

#define pipeline
describeImage = (AnalyzeImage()
    .setSubscriptionKey(VISION_API_KEY)
    .setLocation("eastus")
    .setImageUrlCol("PrimaryImageUrl")
    .setOutputCol("RawImageDescription")
    .setErrorCol("Errors")
    .setVisualFeatures(["Categories", "Tags", "Description", "Faces",
"ImageType", "Color", "Adult"])
    .setConcurrency(5))

df2 = describeImage.transform(data)\
    .select("*", "RawImageDescription.*").drop("Errors",
"RawImageDescription")
```

Create the Search Index

Run the following command to write the results to Azure Search to create a search engine of the artworks with enriched metadata from Computer Vision.

Python

```
from mmlspark.cognitive import *
df2.writeToAzureSearch(
    subscriptionKey=AZURE_SEARCH_KEY,
    actionCol="searchAction",
    serviceName=search_service,
    indexName=search_index,
    keyCol="ObjectID"
)
```

Query the Search Index

Run the following command to query the Azure Search index.

Python

```
url = 'https://{}.search.windows.net/indexes/{}/docs/search?api-
version=2019-05-06'.format(search_service, search_index)
requests.post(url, json={"search": "Glass"}, headers = {"api-key": 
AZURE_SEARCH_KEY}).json()
```

Next steps

Learn how to use [Cognitive Services for big data for Anomaly Detection](#).

Cognitive Services development options

Article • 08/17/2022 • 7 minutes to read

This document provides a high-level overview of development and deployment options to help you get started with Azure Cognitive Services.

Azure Cognitive Services are cloud-based AI services that allow developers to build intelligence into their applications and products without deep knowledge of machine learning. With Cognitive Services, you have access to AI capabilities or models that are built, trained, and updated by Microsoft - ready to be used in your applications. In many cases, you also have the option to customize the models for your business needs.

Cognitive Services are organized into four categories: Decision, Language, Speech, and Vision. Typically you would access these services through REST APIs, client libraries, and custom tools (like command-line interfaces) provided by Microsoft. However, this is only one path to success. Through Azure, you also have access to several development options, such as:

- Automation and integration tools like Logic Apps and Power Automate.
- Deployment options such as Azure Functions and the App Service.
- Cognitive Services Docker containers for secure access.
- Tools like Apache Spark, Azure Databricks, Azure Synapse Analytics, and Azure Kubernetes Service for big data scenarios.

Before we jump in, it's important to know that the Cognitive Services are primarily used for two distinct tasks. Based on the task you want to perform, you have different development and deployment options to choose from.

- [Development options for prediction and analysis](#)
- [Tools to customize and configure models](#)

Development options for prediction and analysis

The tools that you will use to customize and configure models are different from those that you'll use to call the Cognitive Services. Out of the box, most Cognitive Services allow you to send data and receive insights without any customization. For example:

- You can send an image to the Computer Vision service to detect words and phrases or count the number of people in the frame

- You can send an audio file to the Speech service and get transcriptions and translate the speech to text at the same time

Azure offers a wide range of tools that are designed for different types of users, many of which can be used with Cognitive Services. Designer-driven tools are the easiest to use, and are quick to set up and automate, but may have limitations when it comes to customization. Our REST APIs and client libraries provide users with more control and flexibility, but require more effort, time, and expertise to build a solution. If you use REST APIs and client libraries, there is an expectation that you're comfortable working with modern programming languages like C#, Java, Python, JavaScript, or another popular programming language.

Let's take a look at the different ways that you can work with the Cognitive Services.

Client libraries and REST APIs

Cognitive Services client libraries and REST APIs provide you direct access to your service. These tools provide programmatic access to the Cognitive Services, their baseline models, and in many cases allow you to programmatically customize your models and solutions.

- **Target user(s):** Developers and data scientists
- **Benefits:** Provides the greatest flexibility to call the services from any language and environment.
- **UI:** N/A - Code only
- **Subscription(s):** Azure account + Cognitive Services resources

If you want to learn more about available client libraries and REST APIs, use our [Cognitive Services overview](#) to pick a service and get started with one of our quickstarts for vision, decision, language, and speech.

Cognitive Services for big data

With Cognitive Services for big data you can embed continuously improving, intelligent models directly into Apache Spark™ and SQL computations. These tools liberate developers from low-level networking details, so that they can focus on creating smart, distributed applications. Cognitive Services for big data support the following platforms and connectors: Azure Databricks, Azure Synapse, Azure Kubernetes Service, and Data Connectors.

- **Target user(s):** Data scientists and data engineers

- **Benefits:** The Azure Cognitive Services for big data let users channel terabytes of data through Cognitive Services using Apache Spark™. It's easy to create large-scale intelligent applications with any datastore.
- **UI:** N/A - Code only
- **Subscription(s):** Azure account + Cognitive Services resources

If you want to learn more about big data for Cognitive Services, a good place to start is with the [overview](#). If you're ready to start building, try our [Python](#) or [Scala](#) samples.

Azure Functions and Azure Service Web Jobs

[Azure Functions](#) and [Azure App Service Web Jobs](#) both provide code-first integration services designed for developers and are built on [Azure App Services](#). These products provide serverless infrastructure for writing code. Within that code you can make calls to our services using our client libraries and REST APIs.

- **Target user(s):** Developers and data scientists
- **Benefits:** Serverless compute service that lets you run event-triggered code.
- **UI:** Yes
- **Subscription(s):** Azure account + Cognitive Services resource + Azure Functions subscription

Azure Logic Apps

[Azure Logic Apps](#) share the same workflow designer and connectors as Power Automate but provide more advanced control, including integrations with Visual Studio and DevOps. Power Automate makes it easy to integrate with your Cognitive Services resources through service-specific connectors that provide a proxy or wrapper around the APIs. These are the same connectors as those available in Power Automate.

- **Target user(s):** Developers, integrators, IT pros, DevOps
- **Benefits:** Designer-first (declarative) development model providing advanced options and integration in a low-code solution
- **UI:** Yes
- **Subscription(s):** Azure account + Cognitive Services resource + Logic Apps deployment

Power Automate

Power Automate is a service in the [Power Platform](#) that helps you create automated workflows between apps and services without writing code. We offer several connectors

to make it easy to interact with your Cognitive Services resource in a Power Automate solution. Power Automate is built on top of Logic Apps.

- **Target user(s):** Business users (analysts) and SharePoint administrators
- **Benefits:** Automate repetitive manual tasks simply by recording mouse clicks, keystrokes and copy paste steps from your desktop!
- **UI tools:** Yes - UI only
- **Subscription(s):** Azure account + Cognitive Services resource + Power Automate Subscription + Office 365 Subscription

AI Builder

[AI Builder](#) is a Microsoft Power Platform capability you can use to improve business performance by automating processes and predicting outcomes. AI Builder brings the power of AI to your solutions through a point-and-click experience. Many cognitive services such as the Language service, and Computer Vision have been directly integrated here and you don't need to create your own Cognitive Services.

- **Target user(s):** Business users (analysts) and SharePoint administrators
- **Benefits:** A turnkey solution that brings the power of AI through a point-and-click experience. No coding or data science skills required.
- **UI tools:** Yes - UI only
- **Subscription(s):** AI Builder

Continuous integration and deployment

You can use Azure DevOps and GitHub Actions to manage your deployments. In the [section below](#), we have two examples of CI/CD integrations to train and deploy custom models for Speech and the Language Understanding (LUIS) service.

- **Target user(s):** Developers, data scientists, and data engineers
- **Benefits:** Allows you to continuously adjust, update, and deploy applications and models programmatically. There is significant benefit when regularly using your data to improve and update models for Speech, Vision, Language, and Decision.
- **UI tools:** N/A - Code only
- **Subscription(s):** Azure account + Cognitive Services resource + GitHub account

Tools to customize and configure models

As you progress on your journey building an application or workflow with the Cognitive Services, you may find that you need to customize the model to achieve the desired

performance. Many of our services allow you to build on top of the pre-built models to meet your specific business needs. For all our customizable services, we provide both a UI-driven experience for walking through the process as well as APIs for code-driven training. For example:

- You want to train a Custom Speech model to correctly recognize medical terms with a word error rate (WER) below 3 percent
- You want to build an image classifier with Custom Vision that can tell the difference between coniferous and deciduous trees
- You want to build a custom neural voice with your personal voice data for an improved automated customer experience

The tools that you will use to train and configure models are different from those that you'll use to call the Cognitive Services. In many cases, Cognitive Services that support customization provide portals and UI tools designed to help you train, evaluate, and deploy models. Let's quickly take a look at a few options:

Pillar	Service	Customization UI	Quickstart
Vision	Custom Vision	https://www.customvision.ai/	Quickstart
Decision	Personalizer	UI is available in the Azure portal under your Personalizer resource.	Quickstart
Language	Language Understanding (LUIS)	https://www.luis.ai/	
Language	QnA Maker	https://www.qnamaker.ai/	Quickstart
Language	Translator/Custom Translator	https://portal.customtranslator.azure.ai/	Quickstart
Speech	Custom Commands	https://speech.microsoft.com/	Quickstart
Speech	Custom Speech	https://speech.microsoft.com/	Quickstart
Speech	Custom Voice	https://speech.microsoft.com/	Quickstart

Continuous integration and delivery with DevOps and GitHub Actions

Language Understanding and the Speech service offer continuous integration and continuous deployment solutions that are powered by Azure DevOps and GitHub

Actions. These tools are used for automated training, testing, and release management of custom models.

- [CI/CD for Custom Speech](#)
- [CI/CD for LUIS](#)

On-premises containers

Many of the Cognitive Services can be deployed in containers for on-premises access and use. Using these containers gives you the flexibility to bring Cognitive Services closer to your data for compliance, security, or other operational reasons. For a complete list of Cognitive Services containers, see [On-premises containers for Cognitive Services](#).

Next steps

- [Create a Cognitive Services resource and start building](#)

Cognitive Services and machine learning

Article • 05/25/2022 • 5 minutes to read

Cognitive Services provides machine learning capabilities to solve general problems such as analyzing text for emotional sentiment or analyzing images to recognize objects or faces. You don't need special machine learning or data science knowledge to use these services.

[Cognitive Services](#) is a group of services, each supporting different, generalized prediction capabilities. The services are divided into different categories to help you find the right service.

Service category	Purpose
Decision ↗	Build apps that surface recommendations for informed and efficient decision-making.
Language ↗	Allow your apps to process natural language with pre-built scripts, evaluate sentiment and learn how to recognize what users want.
Search ↗	Add Bing Search APIs to your apps and harness the ability to comb billions of webpages, images, videos, and news with a single API call.
Speech ↗	Convert speech into text and text into natural-sounding speech. Translate from one language to another and enable speaker verification and recognition.
Vision ↗	Recognize, identify, caption, index, and moderate your pictures, videos, and digital ink content.

Use Cognitive Services when you:

- Can use a generalized solution.
- Access solution from a programming REST API or SDK.

Use other machine-learning solutions when you:

- Need to choose the algorithm and need to train on very specific data.

What is machine learning?

Machine learning is a concept where you bring together data and an algorithm to solve a specific need. Once the data and algorithm are trained, the output is a model that you can use again with different data. The trained model provides insights based on the new data.

The process of building a machine learning system requires some knowledge of machine learning or data science.

Machine learning is provided using [Azure Machine Learning \(AML\) products and services](#).

What is a Cognitive Service?

A Cognitive Service provides part or all of the components in a machine learning solution: data, algorithm, and trained model. These services are meant to require general knowledge about your data without needing experience with machine learning or data science. These services provide both REST API(s) and language-based SDKs. As a result, you need to have programming language knowledge to use the services.

How are Cognitive Services and Azure Machine Learning (AML) similar?

Both have the end-goal of applying artificial intelligence (AI) to enhance business operations, though how each provides this in the respective offerings is different.

Generally, the audiences are different:

- Cognitive Services are for developers without machine-learning experience.
- Azure Machine Learning is tailored for data scientists.

How is a Cognitive Service different from machine learning?

A Cognitive Service provides a trained model for you. This brings data and an algorithm together, available from a REST API(s) or SDK. You can implement this service within minutes, depending on your scenario. A Cognitive Service provides answers to general problems such as key phrases in text or item identification in images.

Machine learning is a process that generally requires a longer period of time to implement successfully. This time is spent on data collection, cleaning, transformation, algorithm selection, model training, and deployment to get to the same level of functionality provided by a Cognitive Service. With machine learning, it is possible to provide answers to highly specialized and/or specific problems. Machine learning problems require familiarity with the specific subject matter and data of the problem under consideration, as well as expertise in data science.

What kind of data do you have?

Cognitive Services, as a group of services, can require none, some, or all custom data for the trained model.

No additional training data required

Services that provide a fully-trained model can be treated as a *opaque box*. You don't need to know how they work or what data was used to train them. You bring your data to a fully trained model to get a prediction.

Some or all training data required

Some services allow you to bring your own data, then train a model. This allows you to extend the model using the Service's data and algorithm with your own data. The output matches your needs. When you bring your own data, you may need to tag the data in a way specific to the service. For example, if you are training a model to identify flowers, you can provide a catalog of flower images along with the location of the flower in each image to train the model.

A service may *allow* you to provide data to enhance its own data. A service may *require* you to provide data.

Real-time or near real-time data required

A service may need real-time or near-real time data to build an effective model. These services process significant amounts of model data.

Service requirements for the data model

The following data categorizes each service by which kind of data it allows or requires.

Cognitive Service	No training data required	You provide some or all training data	Real-time or near real-time data collection
Anomaly Detector	x	x	x
Bing Search	x		
Computer Vision	x		
Content Moderator	x		x

Cognitive Service	No training data required	You provide some or all training data	Real-time or near real-time data collection
Custom Vision		x	
Face	x	x	
Ink Recognizer	x	x	
Language Understanding (LUIS)		x	
Personalizer	x*	x*	x
QnA Maker		x	
Speaker Recognizer		x	
Speech Text-to-speech (TTS)	x	x	
Speech Speech-to-text (STT)	x	x	
Speech Translation	x		
Language service	x		
Translator	x		
Translator - custom translator		x	

*Personalizer only needs training data collected by the service (as it operates in real-time) to evaluate your policy and data. Personalizer does not need large historical datasets for up-front or batch training.

Where can you use Cognitive Services?

The services are used in any application that can make REST API(s) or SDK calls. Examples of applications include web sites, bots, virtual or mixed reality, desktop and mobile applications.

How is Azure Cognitive Search related to Cognitive Services?

Azure Cognitive Search is a separate cloud search service that optionally uses Cognitive Services to add image and natural language processing to indexing workloads. Cognitive Services is exposed in Azure Cognitive Search through [built-in skills](#) that wrap individual APIs. You can use a free resource for walkthroughs, but plan on creating and attaching a [billable resource](#) for larger volumes.

How can you use Cognitive Services?

Each service provides information about your data. You can combine services together to chain solutions such as converting speech (audio) to text, translating the text into many languages, then using the translated languages to get answers from a knowledge base. While Cognitive Services can be used to create intelligent solutions on their own, they can also be combined with traditional machine learning projects to supplement models or accelerate the development process.

Cognitive Services that provide exported models for other machine learning tools:

Cognitive Service	Model information
Custom Vision	Export for Tensorflow for Android, CoreML for iOS11, ONNX for Windows ML

Learn more

- [Architecture Guide - What are the machine learning products at Microsoft?](#)
- [Machine learning - Introduction to deep learning vs. machine learning](#)

Next steps

- Create your Cognitive Service account in the [Azure portal](#) or with [Azure CLI](#).
- Learn how to [authenticate](#) to a Cognitive Service.
- Use [diagnostic logging](#) for issue identification and debugging.
- Deploy a Cognitive Service in a Docker [container](#).
- Keep up to date with [service updates](#) ↗.

Custom subdomain names for Cognitive Services

Article • 10/24/2021 • 2 minutes to read

Azure Cognitive Services use custom subdomain names for each resource created through the [Azure portal](#), [Azure Cloud Shell](#), or [Azure CLI](#). Unlike regional endpoints, which were common for all customers in a specific Azure region, custom subdomain names are unique to the resource. Custom subdomain names are required to enable features like Azure Active Directory (Azure AD) for authentication.

How does this impact existing resources?

Cognitive Services resources created before July 1, 2019 will use the regional endpoints for the associated service. These endpoints will work with existing and new resources.

If you'd like to migrate an existing resource to leverage custom subdomain names, so that you can enable features like Azure AD, follow these instructions:

1. Sign in to the Azure portal and locate the Cognitive Services resource that you'd like to add a custom subdomain name to.
2. In the **Overview** blade, locate and select **Generate Custom Domain Name**.
3. This opens a panel with instructions to create a unique custom subdomain for your resource.

Warning

After you've created a custom subdomain name it **cannot** be changed.

Do I need to update my existing resources?

No. The regional endpoint will continue to work for new and existing Cognitive Services and the custom subdomain name is optional. Even if a custom subdomain name is added the regional endpoint will continue to work with the resource.

What if an SDK asks me for the region for a resource?

Warning

Speech Services use custom subdomains with **private endpoints only**. In all other cases use **regional endpoints** with Speech Services and associated SDKs.

Regional endpoints and custom subdomain names are both supported and can be used interchangeably. However, the full endpoint is required.

Region information is available in the **Overview** blade for your resource in the [Azure portal](#). For the full list of regional endpoints, see [Is there a list of regional endpoints?](#)

Are custom subdomain names regional?

Yes. Using a custom subdomain name doesn't change any of the regional aspects of your Cognitive Services resource.

What are the requirements for a custom subdomain name?

A custom subdomain name is unique to your resource. The name can only include alphanumeric characters and the `-` character; it must be between 2 and 64 characters in length and cannot end with a `-`.

Can I change a custom domain name?

No. After a custom subdomain name is created and associated with a resource it cannot be changed.

Can I reuse a custom domain name?

Each custom subdomain name is unique, so in order to reuse a custom subdomain name that you've assigned to a Cognitive Services resource, you'll need to delete the existing resource. After the resource has been deleted, you can reuse the custom subdomain name.

Is there a list of regional endpoints?

Yes. This is a list of regional endpoints that you can use with Azure Cognitive Services resources.

① Note

The Translator service and Bing Search APIs use global endpoints.

Endpoint type	Region	Endpoint
Public	Global (Translator & Bing)	https://api.cognitive.microsoft.com
	Australia East	https://australiaeast.api.cognitive.microsoft.com
	Brazil South	https://brazilsouth.api.cognitive.microsoft.com
	Canada Central	https://canadacentral.api.cognitive.microsoft.com
	Central US	https://centralus.api.cognitive.microsoft.com
	East Asia	https://eastasia.api.cognitive.microsoft.com
	East US	https://eastus.api.cognitive.microsoft.com
	East US 2	https://eastus2.api.cognitive.microsoft.com
	France Central	https://francecentral.api.cognitive.microsoft.com
	India Central	https://centralindia.api.cognitive.microsoft.com
	Japan East	https://japaneast.api.cognitive.microsoft.com
	Korea Central	https://koreacentral.api.cognitive.microsoft.com
	North Central US	https://northcentralus.api.cognitive.microsoft.com
	North Europe	https://northeurope.api.cognitive.microsoft.com
	South Africa North	https://southafricanorth.api.cognitive.microsoft.com
	South Central US	https://southcentralus.api.cognitive.microsoft.com
	Southeast Asia	https://southeastasia.api.cognitive.microsoft.com
	UK South	https://uksouth.api.cognitive.microsoft.com
	West Central US	https://westcentralus.api.cognitive.microsoft.com
	West Europe	https://westeurope.api.cognitive.microsoft.com

Endpoint type	Region	Endpoint
	West US	https://westus.api.cognitive.microsoft.com
	West US 2	https://westus2.api.cognitive.microsoft.com
US Gov	US Gov Virginia	https://virginia.api.cognitive.microsoft.us
China	China East 2	https://chinaeast2.api.cognitive.azure.cn
	China North	https://chinanorth.api.cognitive.azure.cn

See also

- [What are the Cognitive Services?](#)
- [Authentication](#)

Accounts

Reference

Service: Cognitive Services

API Version: 2021-10-01

In this article

[Operations](#)

Operations

Create	Create Cognitive Services Account. Accounts is a resource group wide resource type. It holds the keys for developer to access intelligent APIs. It's also the re...
Delete	Deletes a Cognitive Services account from the resource group.
Get	Returns a Cognitive Services account specified by the parameters.
List	Returns all the resources of a particular type belonging to a subscription.
List By Resource Group	Returns all the resources of a particular type belonging to a resource group
List Keys	Lists the account keys for the specified Cognitive Services account.
List Skus	List available SKUs for the requested Cognitive Services account
List Usages	Get usages for the requested Cognitive Services account
Regenerate Key	Regenerates the specified account key for the specified Cognitive Services account.
Update	Updates a Cognitive Services account

Microsoft Azure Cognitive Services management client library for .NET

Article • 01/31/2023 • 2 minutes to read

Microsoft Azure Cognitive Services brings AI within reach of every developer and data scientist. With leading models, a variety of use cases can be unlocked. All it takes is an API call to embed the ability to see, hear, speak, search, understand, and accelerate advanced decision-making into your apps. Enable developers and data scientists of all skill levels to easily add AI capabilities to their apps.

This library supports managing Microsoft Azure Cognitive Services resources.

This library follows the [new Azure SDK guidelines](#), and provides many core capabilities:

- Support MSAL.NET, Azure.Identity is out of box for supporting MSAL.NET.
- Support [OpenTelemetry](<https://opentelemetry.io/>) for distributed tracing.
- HTTP pipeline with custom policies.
- Better error-handling.
- Support uniform telemetry across all languages.

Getting started

Install the package

Install the Microsoft Azure Cognitive Services management library for .NET with [NuGet](#):

.NET CLI

```
dotnet add package Azure.ResourceManager.CognitiveServices
```

Prerequisites

- You must have an [Microsoft Azure subscription](#).

Authenticate the Client

To create an authenticated client and start interacting with Microsoft Azure resources, see the [quickstart guide here](#).

Key concepts

Key concepts of the Microsoft Azure SDK for .NET can be found [here](#).

Documentation

Documentation is available to help you learn how to use this package:

- [Quickstart](#).
- [API References](#).
- [Authentication](#).

Examples

Code samples for using the management library for .NET can be found in the following locations

- [.NET Management Library Code Samples](#)

Troubleshooting

- File an issue via [GitHub Issues](#).
- Check [previous questions](#) or ask new ones on Stack Overflow using Azure and .NET tags.

Next steps

For more information about Microsoft Azure SDK, see [this website](#).

Contributing

For details on contributing to this repository, see the [contributing guide](#).

This project welcomes contributions and suggestions. Most contributions require you to agree to a Contributor License Agreement (CLA) declaring that you have the right to, and actually do, grant us the rights to use your contribution. For details, visit <https://cla.microsoft.com>.

When you submit a pull request, a CLA-bot will automatically determine whether you need to provide a CLA and decorate the PR appropriately (for example, label, comment). Follow the instructions provided by the bot. You'll only need to do this action once across all repositories using our CLA.

This project has adopted the [Microsoft Open Source Code of Conduct](#). For more information, see the [Code of Conduct FAQ](#) or contact opencode@microsoft.com with any other questions or comments.

@azure/arm-cognitiveservices package

Reference

In this article

- [Classes](#)
- [Interfaces](#)
- [Type Aliases](#)
- [Enums](#)
- [Functions](#)
- [Function Details](#)

Classes

[CognitiveServicesManagementClient](#)

Interfaces

Account	Cognitive Services account is an Azure resource representing the provisioned account, it's type, location and SKU.
AccountListResult	The list of cognitive services accounts operation response.
AccountModel	Cognitive Services account Model.
AccountModelListResult	The list of cognitive services accounts operation response.
AccountProperties	Properties of Cognitive Services account.
AccountSku	Cognitive Services resource type and SKU.
AccountSkuListResult	The list of cognitive services accounts operation response.
Accounts	Interface representing a Accounts.
AccountsCreateOptionalParams	Optional parameters.
AccountsDeleteOptionalParams	Optional parameters.
AccountsGetOptionalParams	Optional parameters.
AccountsListByResourceGroupNextOptionalParams	Optional parameters.

AccountsListByResourceGroup	Optional parameters.
OptionalParams	
AccountsListKeysOptional	Optional parameters.
Params	
AccountsListModelsNext	Optional parameters.
OptionalParams	
AccountsListModelsOptional	Optional parameters.
Params	
AccountsListNextOptional	Optional parameters.
Params	
AccountsListOptionalParams	Optional parameters.
AccountsListSkusOptional	Optional parameters.
Params	
AccountsListUsagesOptional	Optional parameters.
Params	
AccountsRegenerateKey	Optional parameters.
OptionalParams	
AccountsUpdateOptional	Optional parameters.
Params	
ApiKeys	The access keys for the cognitive services account.
ApiProperties	The api properties for special APIs.
AzureEntityResource	The resource model definition for an Azure Resource Manager resource with an etag.
CallRateLimit	The call rate limit Cognitive Services account.
CheckDomainAvailability	Optional parameters.
OptionalParams	
CheckDomainAvailability	Check Domain availability parameter.
Parameter	
CheckSkuAvailabilityOptional	Optional parameters.
Params	
CheckSkuAvailability	Check SKU availability parameter.
Parameter	
CognitiveServices	Optional parameters.
ManagementClientOptional	
Params	

CommitmentCost	Cognitive Services account commitment cost.
CommitmentPeriod	Cognitive Services account commitment period.
CommitmentPlan	Cognitive Services account commitment plan.
CommitmentPlanListResult	The list of cognitive services accounts operation response.
CommitmentPlanProperties	Properties of Cognitive Services account commitment plan.
CommitmentPlans	Interface representing a CommitmentPlans.
CommitmentPlansCreate OrUpdateOptionalParams	Optional parameters.
CommitmentPlansDelete OptionalParams	Optional parameters.
CommitmentPlansGet OptionalParams	Optional parameters.
CommitmentPlansListNext OptionalParams	Optional parameters.
CommitmentPlansListOptional Params	Optional parameters.
CommitmentQuota	Cognitive Services account commitment quota.
CommitmentTier	Cognitive Services account commitment tier.
CommitmentTierListResult	The list of cognitive services accounts operation response.
CommitmentTiers	Interface representing a CommitmentTiers.
CommitmentTiersListNext OptionalParams	Optional parameters.
CommitmentTiersListOptional Params	Optional parameters.
DeletedAccounts	Interface representing a DeletedAccounts.
DeletedAccountsGetOptional Params	Optional parameters.
DeletedAccountsListNext OptionalParams	Optional parameters.
DeletedAccountsListOptional Params	Optional parameters.
DeletedAccountsPurge	Optional parameters.

OptionalParams	
Deployment	Cognitive Services account deployment.
DeploymentListResult	The list of cognitive services accounts operation response.
DeploymentModel	Properties of Cognitive Services account deployment model.
DeploymentProperties	Properties of Cognitive Services account deployment.
DeploymentScaleSettings	Properties of Cognitive Services account deployment model.
Deployments	Interface representing a Deployments.
DeploymentsCreateOrUpdate OptionalParams	Optional parameters.
DeploymentsDeleteOptional Params	Optional parameters.
DeploymentsGetOptional Params	Optional parameters.
DeploymentsListNextOptional Params	Optional parameters.
DeploymentsListOptional Params	Optional parameters.
DomainAvailability	Domain availability.
Encryption	Properties to configure Encryption
ErrorAdditionalInfo	The resource management error additional info.
ErrorDetail	The error detail.
ErrorResponse	Common error response for all Azure Resource Manager APIs to return error details for failed operations. (This also follows the OData error response format.).
Identity	Identity for the resource.
IpRule	A rule governing the accessibility from a specific ip address or ip range.
KeyVaultProperties	Properties to configure keyVault Properties
MetricName	A metric name.
ModelDeprecationInfo	Cognitive Services account ModelDeprecationInfo.
NetworkRuleSet	A set of rules governing the network accessibility.

Operation	Details of a REST API operation, returned from the Resource Provider Operations API
OperationDisplay	Localized display information for this particular operation.
OperationListResult	A list of REST API operations supported by an Azure Resource Provider. It contains an URL link to get the next set of results.
Operations	Interface representing a Operations.
OperationsListNextOptionalParams	Optional parameters.
OperationsListOptionalParams	Optional parameters.
PrivateEndpoint	The Private Endpoint resource.
PrivateEndpointConnection	The Private Endpoint Connection resource.
PrivateEndpointConnectionListResult	A list of private endpoint connections
PrivateEndpointConnectionProperties	Properties of the PrivateEndpointConnectProperties.
PrivateEndpointConnections	Interface representing a PrivateEndpointConnections.
PrivateEndpointConnectionsCreateOrUpdateOptionalParams	Optional parameters.
PrivateEndpointConnectionsDeleteOptionalParams	Optional parameters.
PrivateEndpointConnectionsGetOptionalParams	Optional parameters.
PrivateEndpointConnectionsListOptionalParams	Optional parameters.
PrivateLinkResource	A private link resource
PrivateLinkResourceListResult	A list of private link resources
PrivateLinkResourceProperties	Properties of a private link resource.
PrivateLinkResources	Interface representing a PrivateLinkResources.
PrivateLinkResourcesListOptionalParams	Optional parameters.
PrivateLinkService ConnectionState	A collection of information about the state of the connection between service consumer and provider.

ProxyResource	The resource model definition for a Azure Resource Manager proxy resource. It will not have tags and a location
QuotaLimit	
RegenerateKeyParameters	Regenerate key parameters.
RequestMatchPattern	
Resource	Common fields that are returned in the response for all Azure Resource Manager resources
ResourceSku	Describes an available Cognitive Services SKU.
ResourceSkuListResult	The Get Skus operation response.
ResourceSkuRestrictionInfo	
ResourceSkuRestrictions	Describes restrictions of a SKU.
ResourceSkus	Interface representing a ResourceSkus.
ResourceSkusListNextOptionalParams	Optional parameters.
ResourceSkusListOptionalParams	Optional parameters.
Sku	The resource model definition representing SKU
SkuAvailability	SKU availability.
SkuAvailabilityListResult	Check SKU availability result list.
SkuCapability	SkuCapability indicates the capability of a certain feature.
SkuChangeInfo	Sku change info of account.
SystemData	Metadata pertaining to creation and last modification of the resource.
ThrottlingRule	
Usage	The usage data for a usage request.
UsageListResult	The response to a list usage request.
UserAssignedIdentity	User-assigned managed identity.
UserOwnedStorage	The user owned storage for Cognitive Services account.
VirtualNetworkRule	A rule governing the accessibility from a specific virtual network.

Type Aliases

AccountsCreateResponse	Contains response data for the create operation.
AccountsGetResponse	Contains response data for the get operation.
AccountsListByResourceGroupNextResponse	Contains response data for the listByResourceGroupNext operation.
AccountsListByResourceGroupResponse	Contains response data for the listByResourceGroup operation.
AccountsListKeysResponse	Contains response data for the listKeys operation.
AccountsListModelsNextResponse	Contains response data for the listModelsNext operation.
AccountsListModelsResponse	Contains response data for the listModels operation.
AccountsListNextResponse	Contains response data for the listNext operation.
AccountsListResponse	Contains response data for the list operation.
AccountsListSkusResponse	Contains response data for the listSkus operation.
AccountsListUsagesResponse	Contains response data for the listUsages operation.
AccountsRegenerateKeyResponse	Contains response data for the regenerateKey operation.
AccountsUpdateResponse	Contains response data for the update operation.
ActionType	Defines values for ActionType. KnownActionType can be used interchangeably with ActionType, this enum contains the known values that the service supports.

Known values supported by the service

	Internal
CheckDomainAvailabilityResponse	Contains response data for the checkDomainAvailability operation.
CheckSkuAvailabilityResponse	Contains response data for the checkSkuAvailability operation.
CommitmentPlansCreateOrUpdateResponse	Contains response data for the createOrUpdate operation.
CommitmentPlansGet	Contains response data for the get operation.

Response	
CommitmentPlansListNext Response	Contains response data for the listNext operation.
CommitmentPlansList Response	Contains response data for the list operation.
CommitmentTiersListNext Response	Contains response data for the listNext operation.
CommitmentTiersList Response	Contains response data for the list operation.
CreatedByType	<p>Defines values for CreatedByType. KnownCreatedByType can be used interchangeably with CreatedByType, this enum contains the known values that the service supports.</p>
Known values supported by the service	
	User Application ManagedIdentity Key
DeletedAccountsGetResponse	Contains response data for the get operation.
DeletedAccountsListNext Response	Contains response data for the listNext operation.
DeletedAccountsListResponse	Contains response data for the list operation.
DeploymentProvisioningState	<p>Defines values for DeploymentProvisioningState. KnownDeploymentProvisioningState can be used interchangeably with DeploymentProvisioningState, this enum contains the known values that the service supports.</p>
Known values supported by the service	
	Accepted Creating Deleting Moving Failed Succeeded

DeploymentScaleType	Defines values for DeploymentScaleType. KnownDeploymentScaleType can be used interchangeably with DeploymentScaleType, this enum contains the known values that the service supports.
-------------------------------------	--

Known values supported by the service

Standard
Manual

DeploymentsCreateOrUpdateResponse	Contains response data for the createOrUpdate operation.
DeploymentsGetResponse	Contains response data for the get operation.
DeploymentsListNextResponse	Contains response data for the listNext operation.
DeploymentsListResponse	Contains response data for the list operation.
HostingModel	Defines values for HostingModel. KnownHostingModel can be used interchangeably with HostingModel, this enum contains the known values that the service supports.

Known values supported by the service

Web
ConnectedContainer
DisconnectedContainer

KeyName	Defines values for KeyName.
KeySource	Defines values for KeySource. KnownKeySource can be used interchangeably with KeySource, this enum contains the known values that the service supports.

Known values supported by the service

Microsoft.CognitiveServices
Microsoft.KeyVault

NetworkRuleAction	Defines values for NetworkRuleAction. KnownNetworkRuleAction can be used interchangeably with
-----------------------------------	--

NetworkRuleAction, this enum contains the known values that the service supports.

Known values supported by the service

Allow

Deny

[OperationsListNextResponse](#) Contains response data for the listNext operation.

[OperationsListResponse](#) Contains response data for the list operation.

[Origin](#) Defines values for Origin.

[KnownOrigin](#) can be used interchangeably with Origin, this enum contains the known values that the service supports.

Known values supported by the service

user

system

user,system

[PrivateEndpointConnection](#) Defines values for PrivateEndpointConnectionProvisioningState.
[ProvisioningState](#)

[KnownPrivateEndpointConnectionProvisioningState](#) can be used interchangeably with
PrivateEndpointConnectionProvisioningState, this enum contains the known values that the service supports.

Known values supported by the service

Succeeded

Creating

Deleting

Failed

[PrivateEndpointConnections](#)
[CreateOrUpdateResponse](#) Contains response data for the createOrUpdate operation.

[PrivateEndpointConnections](#)
[GetResponse](#) Contains response data for the get operation.

[PrivateEndpointConnections](#)
[ListResponse](#) Contains response data for the list operation.

PrivateEndpointServiceConnectionStatus	Defines values for PrivateEndpointServiceConnectionStatus. KnownPrivateEndpointServiceConnectionStatus can be used interchangeably with PrivateEndpointServiceConnectionStatus, this enum contains the known values that the service supports.
--	---

Known values supported by the service

Pending
Approved
Rejected

PrivateLinkResourcesList Response	Contains response data for the list operation.
-----------------------------------	--

ProvisioningState	Defines values for ProvisioningState. KnownProvisioningState can be used interchangeably with ProvisioningState, this enum contains the known values that the service supports.
-------------------	--

Known values supported by the service

Accepted
Creating
Deleting
Moving
Failed
Succeeded
ResolvingDNS

PublicNetworkAccess	Defines values for PublicNetworkAccess. KnownPublicNetworkAccess can be used interchangeably with PublicNetworkAccess, this enum contains the known values that the service supports.
---------------------	--

Known values supported by the service

Enabled
Disabled

QuotaUsageStatus	Defines values for QuotaUsageStatus. KnownQuotaUsageStatus can be used interchangeably with QuotaUsageStatus, this enum contains the known values that the service supports.
------------------	---

Known values supported by the service

Included
Blocked
InOverage
Unknown

ResourceIdentityType	Defines values for ResourceIdentityType.
ResourceSkuRestrictionsReasonCode	Defines values for ResourceSkuRestrictionsReasonCode. KnownResourceSkuRestrictionsReasonCode can be used interchangeably with ResourceSkuRestrictionsReasonCode, this enum contains the known values that the service supports.

Known values supported by the service

QuotaId
NotAvailableForSubscription

ResourceSkuRestrictionsType	Defines values for ResourceSkuRestrictionsType.
ResourceSkusListNextResponse	Contains response data for the listNext operation.
ResourceSkusListResponse	Contains response data for the list operation.
SkuTier	Defines values for SkuTier. KnownSkuTier can be used interchangeably with SkuTier, this enum contains the known values that the service supports.

Known values supported by the service

Free
Basic
Standard
Premium
Enterprise

UnitType	Defines values for UnitType. KnownUnitType can be used interchangeably with UnitType, this enum contains the known values that the service supports.
--------------------------	---

Known values supported by the service

Count
Bytes
Seconds
Percent
CountPerSecond
BytesPerSecond
Milliseconds

Enums

KnownActionType	Known values of ActionType that the service accepts.
KnownCreatedByType	Known values of CreatedByType that the service accepts.
KnownDeploymentProvisioningState	Known values of DeploymentProvisioningState that the service accepts.
KnownDeploymentScaleType	Known values of DeploymentScaleType that the service accepts.
KnownHostingModel	Known values of HostingModel that the service accepts.
KnownKeySource	Known values of KeySource that the service accepts.
KnownNetworkRuleAction	Known values of NetworkRuleAction that the service accepts.
KnownOrigin	Known values of Origin that the service accepts.
KnownPrivateEndpointConnectionProvisioningState	Known values of PrivateEndpointConnectionProvisioningState that the service accepts.
KnownPrivateEndpointServiceConnectionStatus	Known values of PrivateEndpointServiceConnectionStatus that the service accepts.
KnownProvisioningState	Known values of ProvisioningState that the service accepts.
KnownPublicNetworkAccess	Known values of PublicNetworkAccess that the service accepts.
KnownQuotaUsageStatus	Known values of QuotaUsageStatus that the service accepts.
KnownResourceSkuRestrictionsReasonCode	Known values of ResourceSkuRestrictionsReasonCode that the service accepts.
KnownSkuTier	Known values of SkuTier that the service accepts.
KnownUnitType	Known values of UnitType that the service accepts.

Functions

[getContinuationToken\(unknown\)](#)

Given a result page from a pageable operation, returns a continuation token that can be used to begin paging from that point later.

Function Details

getContinuationToken(unknown)

Given a result page from a pageable operation, returns a continuation token that can be used to begin paging from that point later.

TypeScript

```
function getContinuationToken(page: unknown): string | undefined
```

Parameters

page `unknown`

A result object from calling `.byPage()` on a paged operation.

Returns

`string | undefined`

The continuation token that can be passed into `byPage()`.

azure-mgmt-cognitiveservices Package

Reference

In this article

[Packages](#)

Packages

[cognitiveservices](#)

com.microsoft.azure.management.cognitiveservices

Reference

Package: com.microsoft.azure.management.cognitiveservices

Maven Artifact: [com.microsoft.azure:azure-mgmt-cognitiveservices:1.10.0-beta](#) ↗

This package contains the classes for CognitiveServicesManagementClient. Cognitive Services Management Client.

In this article

[Classes](#)

[Enums](#)

Classes

CheckSkuAvailability	Check SKU availability parameter.
Parameter	
CheckSkuAvailabilityResult	Check SKU availability result.
CognitiveServicesAccountUpdateParameters	The parameters to provide for the account.
CognitiveServicesResourceAndSku	Cognitive Services resource type and SKU.
Error	Cognitive Services error object.
ErrorBody	Cognitive Services error body.
ErrorException	Exception thrown for an invalid response with Error information.
Kind	Defines values for Kind.
MetricName	A metric name.
OperationDisplayInfo	The operation supported by Cognitive Services.
ProvisioningState	Defines values for ProvisioningState.
QuotaUsageStatus	Defines values for QuotaUsageStatus.
RegenerateKeyParameters	Regenerate key parameters.
SkuName	Defines values for SkuName.

UnitType	Defines values for UnitType.
Usage	The usage data for a usage request.

Enums

KeyName	Defines values for KeyName.
SkuTier	Defines values for SkuTier.

az cognitiveservices

Reference

Manage Azure Cognitive Services accounts.

This article lists the Azure CLI commands for Azure Cognitive Services account and subscription management only. Refer to the documentation at <https://docs.microsoft.com/azure/cognitive-services/> for individual services to learn how to use the APIs and supported SDKs.

In this article

[Commands](#)

[az cognitiveservices list](#)

Commands

az cognitiveservices account	Manage Azure Cognitive Services accounts.
az cognitiveservices account commitment-plan	Manage commitment plans for Azure Cognitive Services accounts.
az cognitiveservices account commitment-plan create	Create a commitment plan for Azure Cognitive Services account.
az cognitiveservices account commitment-plan delete	Delete a commitment plan from Azure Cognitive Services account.
az cognitiveservices account commitment-plan list	Show all commitment plans from Azure Cognitive Services account.
az cognitiveservices account commitment-plan show	Show a commitment plan from Azure Cognitive Services account.
az cognitiveservices account create	Manage Azure Cognitive Services accounts.
az cognitiveservices account delete	Manage Azure Cognitive Services accounts.
az cognitiveservices account deployment	Manage deployments for Azure Cognitive Services accounts.
az cognitiveservices account deployment create	Create a deployment for Azure Cognitive Services account.
az cognitiveservices account	Delete a deployment from Azure Cognitive Services account.

deployment delete	
az cognitiveservices account deployment list	Show all deployments for Azure Cognitive Services account.
az cognitiveservices account deployment show	Show a deployment for Azure Cognitive Services account.
az cognitiveservices account identity	Manage identity of Cognitive Services accounts.
az cognitiveservices account identity assign	Assign an identity of a Cognitive Services account.
az cognitiveservices account identity remove	Remove the identity from a Cognitive Services account.
az cognitiveservices account identity show	Show the identity of a Cognitive Services account.
az cognitiveservices account keys	Manage Azure Cognitive Services accounts.
az cognitiveservices account keys list	Manage Azure Cognitive Services accounts.
az cognitiveservices account keys regenerate	Manage Azure Cognitive Services accounts.
az cognitiveservices account list	Manage Azure Cognitive Services accounts.
az cognitiveservices account list-deleted	List soft-deleted Azure Cognitive Services accounts.
az cognitiveservices account list-kinds	List all valid kinds for Azure Cognitive Services account.
az cognitiveservices account list-models	Manage Azure Cognitive Services accounts.
az cognitiveservices account list-skus	Manage Azure Cognitive Services accounts.
az cognitiveservices account list-usage	List usages for Azure Cognitive Services account.
az cognitiveservices account network-rule	Manage network rules.
az cognitiveservices account network-rule add	Add a network rule.

az cognitiveservices account network-rule list	List network rules.
az cognitiveservices account network-rule remove	Remove a network rule.
az cognitiveservices account purge	Purge a soft-deleted Azure Cognitive Services account.
az cognitiveservices account recover	Recover a soft-deleted Azure Cognitive Services account.
az cognitiveservices account show	Manage Azure Cognitive Services accounts.
az cognitiveservices account show-deleted	Show a soft-deleted Azure Cognitive Services account.
az cognitiveservices account update	Manage Azure Cognitive Services accounts.
az cognitiveservices commitment-tier	Manage commitment tiers for Azure Cognitive Services.
az cognitiveservices commitment-tier list	Show all commitment tiers for Azure Cognitive Services.
az cognitiveservices list	Manage Azure Cognitive Services accounts.

az cognitiveservices list

 Edit

Manage Azure Cognitive Services accounts.

This article lists the Azure CLI commands for Azure Cognitive Services account and subscription management only. Refer to the documentation at <https://docs.microsoft.com/azure/cognitive-services/> for individual services to learn how to use the APIs and supported SDKs.

Azure CLI

```
az cognitiveservices list [--resource-group]
```

Examples

List all the Cognitive Services accounts in a resource group.

```
az cognitiveservices list -g MyResourceGroup
```

Optional Parameters

--resource-group -g

Name of resource group. You can configure the default group using `az configure --defaults group=<name>`.

▽ Global Parameters

--debug

Increase logging verbosity to show all debug logs.

--help -h

Show this help message and exit.

--only-show-errors

Only show errors, suppressing warnings.

--output -o

Output format.

--query

JMESPath query string. See <http://jmespath.org/> for more information and examples.

--subscription

Name or ID of subscription. You can configure the default subscription using `az account set -s NAME_OR_ID`.

--verbose

Increase logging verbosity. Use --debug for full debug logs.

AzureRM.CognitiveServices

Reference

In this article

[Cognitive Services](#)

This topic displays help topics for the Azure Cognitive Services cmdlets.

Cognitive Services

[Get-AzureRmCognitiveServicesAccount](#)

Gets an account.

✖ Caution

Because Az PowerShell modules now have all the capabilities of AzureRM PowerShell modules and more, we'll retire AzureRM PowerShell modules on 29 February 2024.

To avoid service interruptions, [update your scripts](#) that use AzureRM PowerShell modules to use Az PowerShell modules by 29 February 2024. To automatically update your scripts, follow the [quickstart guide](#).

[Get-AzureRmCognitiveServicesAccountKey](#)

Gets the API keys for an account.

✖ Caution

Because Az PowerShell modules now have all the capabilities of AzureRM PowerShell modules and more, we'll retire AzureRM PowerShell modules on 29 February 2024.

To avoid service interruptions, [update your scripts](#) that use AzureRM PowerShell modules to use Az PowerShell modules by 29 February 2024. To automatically update your scripts, follow the [quickstart guide](#).

[Get-](#)

Gets the available SKUs for an account.

AzureRmCognitiveServicesAccountSkus

⊗ Caution

Because Az PowerShell modules now have all the capabilities of AzureRM PowerShell modules and more, we'll retire AzureRM PowerShell modules on 29 February 2024.

To avoid service interruptions, [update your scripts](#) that use AzureRM PowerShell modules to use Az PowerShell modules by 29 February 2024. To automatically update your scripts, follow the [quickstart guide](#).

Get-AzureRmCognitiveServicesAccountType

Gets the available Cognitive Services Account Types.

⊗ Caution

Because Az PowerShell modules now have all the capabilities of AzureRM PowerShell modules and more, we'll retire AzureRM PowerShell modules on 29 February 2024.

To avoid service interruptions, [update your scripts](#) that use AzureRM PowerShell modules to use Az PowerShell modules by 29 February 2024. To automatically update your scripts, follow the [quickstart guide](#).

Get-AzureRmCognitiveServicesAccountUsage

Get current usages for a Cognitive Services account.

⊗ Caution

Because Az PowerShell modules now have all the capabilities of AzureRM PowerShell modules and more, we'll retire AzureRM PowerShell modules on 29 February 2024.

To avoid service interruptions, [update your scripts](#) that use AzureRM PowerShell modules to use Az PowerShell modules by 29 February 2024. To automatically update your scripts, follow the [quickstart guide](#).

New-AzureRmCognitiveServicesAccount	Creates a Cognitive Services account.
New-AzureRmCognitiveServicesAccountKey	Regenerates an account key.
Remove-AzureRmCognitiveServicesAccount	Deletes a Cognitive Services account.

`Set-AzureRmCognitiveServicesAccount`

Modifies an account.

 **Caution**

Because Az PowerShell modules now have all the capabilities of AzureRM PowerShell modules and more, we'll retire AzureRM PowerShell modules on 29 February 2024.

To avoid service interruptions, [update your scripts](#) that use AzureRM PowerShell modules to use Az PowerShell modules by 29 February 2024. To automatically update your scripts, follow the [quickstart guide](#).

Azure Policy built-in policy definitions for Azure Cognitive Services

Article • 01/05/2023 • 3 minutes to read

This page is an index of [Azure Policy](#) built-in policy definitions for Azure Cognitive Services. For additional Azure Policy built-ins for other services, see [Azure Policy built-in definitions](#).

The name of each built-in policy definition links to the policy definition in the Azure portal. Use the link in the **Version** column to view the source on the [Azure Policy GitHub repo](#).

Azure Cognitive Services

Name (Azure portal)	Description	Effect(s)	Version (GitHub)
Cognitive Services accounts should disable public network access	To improve the security of Cognitive Services accounts, ensure that it isn't exposed to the public internet and can only be accessed from a private endpoint. Disable the public network access property as described in https://go.microsoft.com/fwlink/?linkid=2129800 . This option disables access from any public address space outside the Azure IP range, and denies all logins that match IP or virtual network-based firewall rules. This reduces data leakage risks.	Audit, Deny, Disabled	3.0.1
Cognitive Services accounts should enable data encryption with a customer-managed key	Customer-managed keys are commonly required to meet regulatory compliance standards. Customer-managed keys enable the data stored in Cognitive Services to be encrypted with an Azure Key Vault key created and owned by you. You have full control and responsibility for the key lifecycle, including rotation and management. Learn more about customer-managed keys at https://go.microsoft.com/fwlink/?linkid=2121321 .	Audit, Deny, Disabled	2.0.0

Name	Description	Effect(s)	Version
(Azure portal)			(GitHub)
Cognitive Services accounts should have local authentication methods disabled ↗	Disabling local authentication methods improves security by ensuring that Cognitive Services accounts require Azure Active Directory identities exclusively for authentication. Learn more at: https://aka.ms/cs/auth ↗.	Audit, Deny, Disabled	1.0.0 ↗
Cognitive Services accounts should restrict network access ↗	Network access to Cognitive Services accounts should be restricted. Configure network rules so only applications from allowed networks can access the Cognitive Services account. To allow connections from specific internet or on-premises clients, access can be granted to traffic from specific Azure virtual networks or to public internet IP address ranges.	Audit, Deny, Disabled	3.0.0 ↗
Cognitive Services accounts should use a managed identity ↗	Assigning a managed identity to your Cognitive Service account helps ensure secure authentication. This identity is used by this Cognitive service account to communicate with other Azure services, like Azure Key Vault, in a secure way without you having to manage any credentials.	Audit, Deny, Disabled	1.0.0 ↗
Cognitive Services accounts should use customer owned storage ↗	Use customer owned storage to control the data stored at rest in Cognitive Services. To learn more about customer owned storage, visit https://aka.ms/cogsvc-cmk ↗.	Audit, Deny, Disabled	2.0.0 ↗
Cognitive Services should use private link ↗	Azure Private Link lets you connect your virtual networks to Azure services without a public IP address at the source or destination. The Private Link platform handles the connectivity between the consumer and services over the Azure backbone network. By mapping private endpoints to Cognitive Services, you'll reduce the potential for data leakage. Learn more about private links at: https://go.microsoft.com/fwlink/?linkid=2129800 ↗.	Audit, Disabled	3.0.0 ↗

Name	Description	Effect(s)	Version
(Azure portal)			(GitHub)
Configure Cognitive Services accounts to disable local authentication methods	Disable local authentication methods so that your Cognitive Services accounts require Azure Active Directory identities exclusively for authentication. Learn more at: https://aka.ms/cs/auth .	Modify, Disabled	1.0.0 ↗
Configure Cognitive Services accounts to disable public network access	Disable public network access for your Cognitive Services resource so that it's not accessible over the public internet. This can reduce data leakage risks. Learn more at: https://go.microsoft.com/fwlink/?linkid=2129800 .	Disabled, Modify	3.0.0 ↗
Configure Cognitive Services accounts with private endpoints	Private endpoints connect your virtual networks to Azure services without a public IP address at the source or destination. By mapping private endpoints to Cognitive Services, you'll reduce the potential for data leakage. Learn more about private links at: https://go.microsoft.com/fwlink/?linkid=2129800 .	DeployIfNotExists, Disabled	3.0.0 ↗

Next steps

- See the built-ins on the [Azure Policy GitHub repo](#).
- Review the [Azure Policy definition structure](#).
- Review [Understanding policy effects](#).

Azure Cognitive Services support and help options

Article • 07/22/2022 • 2 minutes to read

Are you just starting to explore the functionality of Azure Cognitive Services? Perhaps you are implementing a new feature in your application. Or after using the service, do you have suggestions on how to improve it? Here are options for where you can get support, stay up-to-date, give feedback, and report bugs for Cognitive Services.

Create an Azure support request

A

Explore the range of [Azure support options and choose the plan](#) that best fits, whether you're a developer just starting your cloud journey or a large organization deploying business-critical, strategic applications. Azure customers can create and manage support requests in the Azure portal.

- [Azure portal](#)
- [Azure portal for the United States government](#)

Post a question on Microsoft Q&A

For quick and reliable answers on your technical product questions from Microsoft Engineers, Azure Most Valuable Professionals (MVPs), or our expert community, engage with us on [Microsoft Q&A](#), Azure's preferred destination for community support.

If you can't find an answer to your problem using search, submit a new question to Microsoft Q&A. Use one of the following tags when you ask your question:

- [Cognitive Services](#)

Vision

- [Computer Vision](#)
- [Custom Vision](#)
- [Face](#)
- [Form Recognizer](#)
- [Video Indexer](#)

Language

- Immersive Reader
- Language Understanding (LUIS)
- QnA Maker
- Language service
- Translator

Speech

- Speech service

Decision

- Anomaly Detector
- Content Moderator
- Metrics Advisor
- Personalizer

Azure OpenAI

- Azure OpenAI

Post a question to Stack Overflow



For answers on your developer questions from the largest community developer ecosystem, ask your question on Stack Overflow.

If you do submit a new question to Stack Overflow, please use one or more of the following tags when you create the question:

- Cognitive Services ↗

Vision

- Computer Vision ↗
- Custom Vision ↗
- Face ↗
- Form Recognizer ↗
- Video Indexer ↗

Language

- Immersive Reader ↗
- Language Understanding (LUIS) ↗

- [QnA Maker](#)
- [Language service](#)
- [Translator](#)

Speech

- [Speech service](#)

Decision

- [Anomaly Detector](#)
- [Content Moderator](#)
- [Metrics Advisor](#)
- [Personalizer](#)

Azure OpenAI

- [Azure OpenAI](#)

Submit feedback

To request new features, post them on <https://feedback.azure.com>. Share your ideas for making Cognitive Services and its APIs work better for the applications you develop.

- [Cognitive Services](#)

Vision

- [Computer Vision](#)
- [Custom Vision](#)
- [Face](#)
- [Form Recognizer](#)
- [Video Indexer](#)

Language

- [Immersive Reader](#)
- [Language Understanding \(LUIS\)](#)
- [QnA Maker](#)
- [Language service](#)
- [Translator](#)

Speech

- [Speech service](#)

Decision

- [Anomaly Detector ↗](#)
- [Content Moderator ↗](#)
- [Metrics Advisor ↗](#)
- [Personalizer ↗](#)

Stay informed

Staying informed about features in a new release or news on the Azure blog can help you find the difference between a programming error, a service bug, or a feature not yet available in Cognitive Services.

- Learn more about product updates, roadmap, and announcements in [Azure Updates ↗](#).
- News about Cognitive Services is shared in the [Azure blog ↗](#).
- [Join the conversation on Reddit ↗](#) about Cognitive Services.

Next steps

[What are Azure Cognitive Services?](#)

Responsible use of AI with Cognitive Services

Article • 01/11/2023 • 2 minutes to read

Azure Cognitive Services provides information and guidelines on how to responsibly use artificial intelligence in applications. Below are the links to articles that provide this guidance for the different services within the Cognitive Services suite.

Anomaly Detector

- [Transparency note and use cases](#)
- [Data, privacy, and security](#)

Computer Vision - OCR

- [Transparency note and use cases](#)
- [Characteristics and limitations](#)
- [Integration and responsible use](#)
- [Data, privacy, and security](#)

Computer Vision - Image Analysis

- [Transparency note](#)
- [Characteristics and limitations](#)
- [Integration and responsible use](#)
- [Data, privacy, and security](#)
- [Limited Access features](#)

Computer Vision - Face

- [Transparency note and use cases](#)
- [Characteristics and limitations](#)
- [Integration and responsible use](#)
- [Data privacy and security](#)
- [Limited Access features](#)

Computer Vision - Spatial Analysis

- Transparency note and use cases
- Characteristics and limitations
- Responsible use in AI deployment
- Disclosure design guidelines
- Research insights
- Data, privacy, and security

Custom Vision

- Transparency note and use cases
- Characteristics and limitations
- Integration and responsible use
- Data, privacy, and security

Language service

- Transparency note
- Integration and responsible use
- Data, privacy, and security

Language - Custom text classification

- Transparency note
- Integration and responsible use
- Characteristics and limitations
- Data, privacy, and security

Language - Named entity recognition

- Transparency note
- Integration and responsible use
- Data, privacy, and security

Language - Custom named entity recognition

- Transparency note
- Integration and responsible use
- Characteristics and limitations
- Data, privacy, and security

Language - Entity linking

- Transparency note
- Integration and responsible use
- Data, privacy, and security

Language - Language detection

- Transparency note
- Integration and responsible use
- Data, privacy, and security

Language - Key phrase extraction

- Transparency note
- Integration and responsible use
- Data, privacy, and security

Language - Personally identifiable information detection

- Transparency note
- Integration and responsible use
- Data, privacy, and security

Language - Question Answering

- Transparency note
- Integration and responsible use
- Data, privacy, and security

Language - Sentiment Analysis and opinion mining

- Transparency note
- Integration and responsible use
- Data, privacy, and security

Language - Text Analytics for health

- Transparency note
- Integration and responsible use
- Data, privacy, and security

Language - Summarization

- Transparency note
- Integration and responsible use
- Characteristics and limitations
- Data, privacy, and security

Language Understanding

- Transparency note and use cases
- Characteristics and limitations
- Integration and responsible use
- Data, privacy, and security

OpenAI

- Transparency note
- Limited access
- Code of conduct
- Data, privacy, and security

Personalizer

- Transparency note and use cases
- Characteristics and limitations
- Integration and responsible use
- Data and privacy

QnA Maker

- Transparency note and use cases
- Characteristics and limitations
- Integration and responsible use

- Data, privacy, and security

Speech - Pronunciation Assessment

- Transparency note and use cases
- Characteristics and limitations

Speech - Speaker Recognition

- Transparency note and use cases
- Characteristics and limitations
- Limited access
- General guidelines
- Data, privacy, and security

Speech - Custom Neural Voice

- Transparency note and use cases
- Characteristics and limitations
- Limited access
- Responsible deployment of synthetic speech
- Disclosure of voice talent
- Disclosure of design guidelines
- Disclosure of design patterns
- Code of conduct
- Data, privacy, and security

Speech - Speech to Text

- Transparency note and use cases
- Characteristics and limitations
- Integration and responsible use
- Data, privacy, and security

Additional resources

[What are Azure Applied AI Services? - Azure Applied AI Services](#)

Applied AI Services description.

[AI enrichment with image and text processing - Azure Architecture Center](#)

Transform unstructured image and text data into full text searchable content with Azure Cognitive Search pre-built and custom skills.

[What is Spatial Analysis? - Azure Cognitive Services](#)

This document explains the basic concepts and features of the Azure Spatial Analysis container.

[Brand detection - Computer Vision - Azure Cognitive Services](#)

Learn about brand and logo detection, a specialized mode of object detection, using the Computer Vision API.

[Characteristics and limitations for Computer Vision spatial analysis - Azure Cognitive Services](#)

This document explains the characteristics and limitations of a Computer Vision spatial analysis container.

[Improving your model - Custom Vision Service - Azure Cognitive Services](#)

In this article you'll learn how the amount, quality and variety of data can improve the quality of your model in the Custom Vision service.

[Image categorization - Computer Vision - Azure Cognitive Services](#)

Learn concepts related to the image categorization feature of the Image Analysis API.

[Domain-specific content - Computer Vision - Azure Cognitive Services](#)

Learn how to specify an image categorization domain to return more detailed information about an image.

[Show 5 more](#)

Limited Access features for Cognitive Services

Article • 12/29/2022 • 4 minutes to read

Our vision is to empower developers and organizations to use AI to transform society in positive ways. We encourage responsible AI practices to protect the rights and safety of individuals. To achieve this, Microsoft has implemented a Limited Access policy grounded in our [AI Principles](#) to support responsible deployment of Azure services.

What is Limited Access?

Limited Access services require registration, and only customers managed by Microsoft, meaning those who are working directly with Microsoft account teams, are eligible for access. The use of these services is limited to the use case selected at the time of registration. Customers must acknowledge that they've reviewed and agree to the terms of service. Microsoft may require customers to reverify this information.

Limited Access services are made available to customers under the terms governing their subscription to Microsoft Azure Services (including the [Service Specific Terms](#)). Review these terms carefully as they contain important conditions and obligations governing your use of Limited Access services.

List of Limited Access services

The following services are Limited Access:

- [Custom Neural Voice](#): Pro features
- [Speaker Recognition](#): All features
- [Face API](#): Identify and Verify features, face ID property
- [Computer Vision](#): Celebrity Recognition feature
- [Azure Video Indexer](#): Celebrity Recognition and Face Identify features
- [Azure OpenAI](#): Azure OpenAI Service, modified abuse monitoring, and modified content filters

Features of these services that aren't listed above are available without registration.

FAQ about Limited Access

How do I register for access?

Submit a registration form for each Limited Access service you would like to use:

- [Custom Neural Voice](#) : Pro features
- [Speaker Recognition](#) : All features
- [Face API](#) : Identify and Verify features
- [Computer Vision](#) : Celebrity Recognition feature
- [Azure Video Indexer](#) : Celebrity Recognition and Face Identify features
- [Azure OpenAI](#): Azure OpenAI Service, modified abuse monitoring, and modified content filters

How long will the registration process take?

Review may take 5-10 business days. You will receive an email as soon as your application is reviewed.

Who is eligible to use Limited Access services?

Limited Access services are available only to customers managed by Microsoft. Additionally, Limited Access services are only available for certain use cases, and customers must select their intended use case in their registration form.

Please use an email address affiliated with your organization in your registration form. Registration forms submitted with personal email addresses will be denied.

If you're not a managed customer, we invite you to submit an application using the same forms and we will reach out to you about any opportunities to join an eligibility program.

What is a managed customer? What if I don't know whether I'm a managed customer?

Managed customers work with Microsoft account teams. We invite you to submit a registration form for the features you'd like to use, and we'll verify your eligibility for access. We are not able to accept requests to become a managed customer at this time.

What happens if I'm an existing customer and I don't register?

Existing customers have until June 30, 2023 to submit a registration form and be approved to continue using Limited Access services after June 30, 2023. We recommend allowing 10 business days for review. Without an approved application, you will be denied access after June 30, 2023.

The registration forms can be found here:

- [Custom Neural Voice](#) : Pro features
- [Speaker Recognition](#) : All features
- [Face API](#) : Identify and Verify features
- [Computer Vision](#) : Celebrity Recognition feature
- [Azure Video Indexer](#) : Celebrity Recognition and Face Identify features
- [Azure OpenAI: [Azure OpenAI service](#), modified abuse monitoring, and modified content filters]

I'm an existing customer who applied for access to Custom Neural Voice or Speaker Recognition, do I have to register to keep using these services?

We're always looking for opportunities to improve our Responsible AI program, and Limited Access is an update to our service gating processes. If you've previously applied for and been granted access to Custom Neural Voice or Speaker Recognition, we request that you submit a new registration form to continue using these services beyond June 30, 2023.

If you're an existing customer using Custom Neural Voice or Speaker Recognition on June 21, 2022, you have until June 30, 2023 to submit a registration form with your selected use case and receive approval to continue using these services after June 30, 2023. We recommend allowing 10 days for application processing. Existing customers can continue using the service until June 30, 2023, after which they must be approved for access. The registration forms can be found here:

- [Custom Neural Voice](#) : Pro features
- [Speaker Recognition](#) : All features

What if my use case isn't on the registration form?

Limited Access features are only available for the use cases listed on the registration forms. If your desired use case isn't listed, let us know in this [feedback form](#) so we can improve our service offerings.

Where can I use Limited Access services?

Search [here](#) for a Limited Access service to view its regional availability. In the Brazil South and UAE North datacenter regions, we are prioritizing access for commercial customers managed by Microsoft.

Detailed information about supported regions for Custom Neural Voice and Speaker Recognition operations can be found [here](#).

What happens to my data if my application is denied?

If you're an existing customer and your application for access is denied, you will no longer be able to use Limited Access features after June 30, 2023. Your data is subject to Microsoft's data retention [policies](#).

How long will the registration process take?

You'll receive communication from us about your application within 10 business days. In some cases, reviews can take longer. You'll receive an email as soon as your application is reviewed.

Help and support

Report abuse of Limited Access services [here](#).

Compare Azure Government and global Azure

Article • 02/10/2023 • 15 minutes to read

Microsoft Azure Government uses same underlying technologies as global Azure, which includes the core components of [Infrastructure-as-a-Service \(IaaS\)](#), [Platform-as-a-Service \(PaaS\)](#), and [Software-as-a-Service \(SaaS\)](#). Both Azure and Azure Government have the same comprehensive security controls in place and the same Microsoft commitment on the safeguarding of customer data. Whereas both cloud environments are assessed and authorized at the FedRAMP High impact level, Azure Government provides an extra layer of protection to customers through contractual commitments regarding storage of customer data in the United States and limiting potential access to systems processing customer data to [screened US persons](#). These commitments may be of interest to customers using the cloud to store or process data subject to US export control regulations.

Export control implications

You're responsible for designing and deploying your applications to meet [US export control requirements](#) such as the requirements prescribed in the EAR, ITAR, and DoE 10 CFR Part 810. In doing so, you shouldn't include sensitive or restricted information in Azure resource names, as explained in [Considerations for naming Azure resources](#).

Guidance for developers

Azure Government services operate the same way as the corresponding services in global Azure, which is why most of the existing online Azure documentation applies equally well to Azure Government. However, there are some key differences that developers working on applications hosted in Azure Government must be aware of. For more information, see [Guidance for developers](#). As a developer, you must know how to connect to Azure Government and once you connect you'll mostly have the same experience as in global Azure.

! Note

This article has been updated to use the new Azure PowerShell Az module. You can still use the AzureRM module, which will continue to receive bug fixes until at least December 2020. To learn more about the new Az module and AzureRM compatibility, see [Introducing the new Azure PowerShell Az module](#). For Az module installation instructions, see [Install the Azure Az PowerShell module](#).

You can use AzureCLI or PowerShell to obtain Azure Government endpoints for services you provisioned:

- Use **Azure CLI** to run the `az cloud show` command and provide `AzureUSGovernment` as the name of the target cloud environment. For example,

```
Azure CLI  
az cloud show --name AzureUSGovernment
```

should get you different endpoints for Azure Government.

- Use a **PowerShell** cmdlet such as `Get-AzEnvironment` to get endpoints and metadata for an instance of Azure service. For example,

```
PowerShell  
Get-AzEnvironment -Name AzureUSGovernment
```

should get you properties for Azure Government. This cmdlet gets environments from your subscription data file.

Table below lists API endpoints in Azure vs. Azure Government for accessing and managing some of the more common services. If you provisioned a service that isn't listed in the table below, see the Azure CLI and PowerShell examples above for suggestions on how to obtain the corresponding Azure Government endpoint.

Service category	Service name	Azure Public	Azure Government	Notes
AI + machine learning	Azure Bot Service	botframework.com	botframework.azure.us	
	Azure Form Recognizer	cognitiveservices.azure.com	cognitiveservices.azure.us	
	Computer Vision	cognitiveservices.azure.com	cognitiveservices.azure.us	
	Custom Vision	cognitiveservices.azure.com	cognitiveservices.azure.us Portal ↗	
	Content Moderator	cognitiveservices.azure.com	cognitiveservices.azure.us	
	Face API	cognitiveservices.azure.com	cognitiveservices.azure.us	
	Language Understanding	cognitiveservices.azure.com	cognitiveservices.azure.us Portal ↗	Part of Cognitive Services for Language
	Personalizer	cognitiveservices.azure.com	cognitiveservices.azure.us	
	QnA Maker	cognitiveservices.azure.com	cognitiveservices.azure.us	Part of Cognitive Services for Language
Speech	Speech service	See STT API docs	Speech Studio ↗ See Speech service endpoints	
			 Speech translation endpoints Virginia: https://usgovvirginia.s2s.speech.azure.us Arizona: https://usgovarizona.s2s.speech.azure.us	
Analytics	Text Analytics	cognitiveservices.azure.com	cognitiveservices.azure.us	Part of Cognitive Services for Language
	Translator	See Translator API docs	cognitiveservices.azure.us	
	Azure HDInsight	azurehdinsight.net	azurehdinsight.us	
Event Hubs	Event Hubs	servicebus.windows.net	servicebus.usgovcloudapi.net	
	Power BI	app.powerbi.com	app.powerbigov.us	Power BI US Gov ↗

Service category	Service name	Azure Public	Azure Government	Notes
Compute	Batch	batch.azure.com	batch.usgovcloudapi.net	
	Cloud Services	cloudapp.net	usgovcloudapp.net	
Containers	Azure Service Fabric	cloudapp.azure.com	cloudapp.usgovcloudapi.net	
	Container Registry	azurecr.io	azurecr.us	
Databases	Azure Cache for Redis	redis.cache.windows.net	redis.cache.usgovcloudapi.net	See How to connect to other clouds
	Azure Cosmos DB	documents.azure.com	documents.azure.us	
	Azure Database for MariaDB	mariadb.database.azure.com	mariadb.database.usgovcloudapi.net	
	Azure Database for MySQL	mysql.database.azure.com	mysql.database.usgovcloudapi.net	
	Azure Database for PostgreSQL	postgres.database.azure.com	postgres.database.usgovcloudapi.net	
	Azure SQL Database	database.windows.net	database.usgovcloudapi.net	
Identity	Azure AD	login.microsoftonline.com	login.microsoftonline.us	
		certauth.login.microsoftonline.com	certauth.login.microsoftonline.us	
		passwordreset.microsoftonline.com	passwordreset.microsoftonline.us	
Integration	Service Bus	servicebus.windows.net	servicebus.usgovcloudapi.net	
Internet of Things	Azure IoT Hub	azure-devices.net	azure-devices.us	
	Azure Maps	atlas.microsoft.com	atlas.azure.us	
Management and governance	Notification Hubs	servicebus.windows.net	servicebus.usgovcloudapi.net	
	Azure Automation	azure-automation.net	azure-automation.us	
Management and governance	Azure Monitor	mms.microsoft.com	oms.microsoft.us	Log Analytics workspace portal
		ods.opinsights.azure.com	ods.opinsights.azure.us	Data collector API
		oms.opinsights.azure.com	oms.opinsights.azure.us	

Service category	Service name	Azure Public	Azure Government	Notes
		portal.loganalytics.io	portal.loganalytics.us	
		api.loganalytics.io	api.loganalytics.us	
		docs.loganalytics.io	docs.loganalytics.us	
		adx.monitor.azure.com	adx.monitor.azure.us	Data Explorer queries
Azure Resource Manager		management.azure.com	management.usgovcloudapi.net	
Cost Management		consumption.azure.com	consumption.azure.us	
Gallery URL		gallery.azure.com	gallery.azure.us	
Microsoft Azure portal		portal.azure.com	portal.azure.us	
Microsoft Intune		enterpriseregistration.windows.net	enterpriseregistration.microsoftonline.us	Enterprise registration
		manage.microsoft.com	manage.microsoft.us	Enterprise enrollment
Migration	Azure Site Recovery	hypervrecoverymanager.windowsazure.com	hypervrecoverymanager.windowsazure.us	Site Recovery service
		backup.windowsazure.com	backup.windowsazure.us	Protection service
		blob.core.windows.net	blob.core.usgovcloudapi.net	Storing VM snapshots
Networking	Traffic Manager	trafficmanager.net	usgovtrafficmanager.net	
Security	Key Vault	vault.azure.net	vault.usgovcloudapi.net	
Storage	Azure Backup	backup.windowsazure.com	backup.windowsazure.us	
	Blob	blob.core.windows.net	blob.core.usgovcloudapi.net	
	Queue	queue.core.windows.net	queue.core.usgovcloudapi.net	
	Table	table.core.windows.net	table.core.usgovcloudapi.net	
	File	file.core.windows.net	file.core.usgovcloudapi.net	
Virtual desktop infrastructure	Azure Virtual Desktop	See AVD docs	See AVD docs	
Web	API Management	management.azure.com	management.usgovcloudapi.net	
	API Management Gateway	azure-api.net	azure-api.us	

Service category	Service name	Azure Public	Azure Government	Notes
	API Management	management.azure-api.net	management.azure-api.us	
	management			
	API Portal	portal.azure-api.net	portal.azure-api.us	
	Management Portal			
App Configuration	azconfig.io		azconfig.azure.us	
App Service	azurewebsites.net		azurewebsites.us	
Azure Cognitive Search	search.windows.net		search.windows.us	
Azure Functions	azurewebsites.net		azurewebsites.us	

Service availability

Microsoft's goal for Azure Government is to match service availability in Azure. For service availability in Azure Government, see [Products available by region](#). Services available in Azure Government are listed by category and whether they're Generally Available or available through Preview. If a service is available in Azure Government, that fact isn't reiterated in the rest of this article. Instead, you're encouraged to review [Products available by region](#) for the latest, up-to-date information on service availability.

In general, service availability in Azure Government implies that all corresponding service features are available to you. Variations to this approach and other applicable limitations are tracked and explained in this article based on the main service categories outlined in the [online directory of Azure services](#). Other considerations for service deployment and usage in Azure Government are also provided.

AI + machine learning

This section outlines variations and considerations when using **Azure Bot Service**, **Azure Machine Learning**, and **Cognitive Services** in the Azure Government environment. For service availability, see [Products available by region](#).

Azure Bot Service

The following Azure Bot Service **features aren't currently available** in Azure Government:

- Bot Framework Composer integration
- Channels (due to availability of dependent services)
 - Direct Line Speech Channel
 - Telephony Channel (Preview)
 - Microsoft Search Channel (Preview)
 - Kik Channel (deprecated)

For more information, see [How do I create a bot that uses US Government data center](#).

Azure Machine Learning

For feature variations and limitations, see [Azure Machine Learning feature availability across cloud regions](#).

Cognitive Services: Content Moderator

The following Content Moderator **features aren't currently available** in Azure Government:

- Review UI and Review APIs.

Cognitive Services: Language Understanding (LUIS)

The following Language Understanding **features aren't currently available** in Azure Government:

- Speech Requests
- Prebuilt Domains

Cognitive Services Language Understanding (LUIS) is part of [Cognitive Services for Language](#).

Cognitive Services: Speech

For feature variations and limitations, including API endpoints, see [Speech service in sovereign clouds](#).

Cognitive Services: Translator

For feature variations and limitations, including API endpoints, see [Translator in sovereign clouds](#).

Analytics

This section outlines variations and considerations when using Analytics services in the Azure Government environment. For service availability, see [Products available by region](#).

Azure HDInsight

For secured virtual networks, you'll want to allow network security groups (NSGs) access to certain IP addresses and ports. For Azure Government, you should allow the following IP addresses (all with an Allowed port of 443):

Region	Allowed IP addresses	Allowed port
US DoD Central	52.180.249.174 52.180.250.239	443
US DoD East	52.181.164.168 52.181.164.151	443
US Gov Texas	52.238.116.212 52.238.112.86	443
US Gov Virginia	13.72.49.126 13.72.55.55 13.72.184.124 13.72.190.110	443
US Gov Arizona	52.127.3.176 52.127.3.178	443

For a demo on how to build data-centric solutions on Azure Government using HDInsight, see Cognitive Services, HDInsight, and Power BI on Azure Government.

Power BI

For usage guidance, feature variations, and limitations, see [Power BI for US government customers](#). For a demo on how to build data-centric solutions on Azure Government using Power BI, see Cognitive Services, HDInsight, and Power BI on Azure Government.

Power BI Embedded

To learn how to embed analytical content within your business process application, see [Tutorial: Embed a Power BI content into your application for national clouds](#).

Databases

This section outlines variations and considerations when using Databases services in the Azure Government environment. For service availability, see [Products available by region](#).

Azure Database for MySQL

The following Azure Database for MySQL **features aren't currently available** in Azure Government:

- Advanced Threat Protection

Azure Database for PostgreSQL

The following Azure Database for PostgreSQL **features aren't currently available** in Azure Government:

- Hyperscale (Citus) deployment option
- The following features of the Single server deployment option
 - Advanced Threat Protection
 - Backup with long-term retention

Azure SQL Managed Instance

The following Azure SQL Managed Instance **features aren't currently available** in Azure Government:

- Long-term retention

Developer tools

This section outlines variations and considerations when using Developer tools in the Azure Government environment. For service availability, see [Products available by region](#).

Enterprise Dev/Test subscription offer

- Enterprise Dev/Test subscription offer in existing or separate tenant is currently available only in Azure public as documented in [Azure EA portal administration](#).

Identity

This section outlines variations and considerations when using Identity services in the Azure Government environment. For service availability, see [Products available by region](#).

Azure Active Directory Premium P1 and P2

For feature variations and limitations, see [Cloud feature availability](#).

The following features have known limitations in Azure Government:

- Limitations with B2B Collaboration in supported Azure US Government tenants:
 - For more information about B2B collaboration limitations in Azure Government and to find out if B2B collaboration is available in your Azure Government tenant, see [Azure AD B2B in government and national clouds](#).
 - B2B collaboration via Power BI isn't supported. When you invite a guest user from within Power BI, the B2B flow isn't used and the guest user won't appear in the tenant's user list. If a guest user is invited through other means, they'll appear in the Power BI user list, but any sharing request to the user will fail and display a 403 Forbidden error.
- Limitations with multi-factor authentication:
 - Trusted IPs isn't supported in Azure Government. Instead, use Conditional Access policies with named locations to establish when multi-factor authentication should and shouldn't be required based off the user's current IP address.

Azure Active Directory B2C

Azure Active Directory B2C is **not available** in Azure Government.

Microsoft Authentication Library (MSAL)

The Microsoft Authentication Library (MSAL) enables developers to acquire security tokens from the Microsoft identity platform to authenticate users and access secured web APIs. For feature variations and limitations, see [National clouds and MSAL](#).

Management and governance

This section outlines variations and considerations when using Management and Governance services in the Azure Government environment. For service availability, see [Products available by region](#).

Automation

The following Automation **features aren't currently available** in Azure Government:

- Automation analytics solution

Azure Advisor

For feature variations and limitations, see [Azure Advisor in sovereign clouds](#).

Azure Lighthouse

The following Azure Lighthouse **features aren't currently available** in Azure Government:

- Managed Service offers published to Azure Marketplace
- Delegation of subscriptions across a national cloud and the Azure public cloud, or across two separate national clouds, isn't supported

- Privileged Identity Management (PIM) feature isn't enabled, for example, just-in-time (JIT) / eligible authorization capability

Azure Monitor

Azure Monitor enables the same features in both Azure and Azure Government.

- System Center Operations Manager 2019 is supported equally well in both Azure and Azure Government.

The following options are available for previous versions of System Center Operations Manager:

- Integrating System Center Operations Manager 2016 with Azure Government requires an updated Advisor management pack that is included with Update Rollup 2 or later.
- System Center Operations Manager 2012 R2 requires an updated Advisor management pack included with Update Rollup 3 or later.

For more information, see [Connect Operations Manager to Azure Monitor](#).

Frequently asked questions

- Can I migrate data from Azure Monitor logs in Azure to Azure Government?
 - No. It isn't possible to move data or your workspace from Azure to Azure Government.
- Can I switch between Azure and Azure Government workspaces from the Operations Management Suite portal?
 - No. The portals for Azure and Azure Government are separate and don't share information.

Application Insights

Application Insights (part of Azure Monitor) enables the same features in both Azure and Azure Government. This section describes the supplemental configuration that is required to use Application Insights in Azure Government.

Visual Studio - In Azure Government, you can enable monitoring on your ASP.NET, ASP.NET Core, Java, and Node.js based applications running on Azure App Service. For more information, see [Application monitoring for Azure App Service overview](#). In Visual Studio, go to Tools|Options|Accounts|Registered Azure Clouds|Add New Azure Cloud and select Azure US Government as the Discovery endpoint. After that, adding an account in File|Account Settings will prompt you for which cloud you want to add from.

SDK endpoint modifications - In order to send data from Application Insights to an Azure Government region, you'll need to modify the default endpoint addresses that are used by the Application Insights SDKs. Each SDK requires slightly different modifications, as described in [Application Insights overriding default endpoints](#).

Firewall exceptions - Application Insights uses several IP addresses. You might need to know these addresses if the app that you're monitoring is hosted behind a firewall. For more information, see [IP addresses used by Azure Monitor](#) from where you can download Azure Government IP addresses.

Note

Although these addresses are static, it's possible that we'll need to change them from time to time. All Application Insights traffic represents outbound traffic except for availability monitoring and webhooks, which require inbound firewall rules.

You need to open some **outgoing ports** in your server's firewall to allow the Application Insights SDK and/or Status Monitor to send data to the portal:

Purpose	URL	IP address	Ports
---------	-----	------------	-------

Purpose	URL	IP address	Ports
Telemetry	dc.applicationinsights.us	23.97.4.113	443

Cost Management and Billing

The following Azure Cost Management + Billing **features aren't currently available** in Azure Government:

- Cost Management + Billing for cloud solution providers (CSPs)

Media

This section outlines variations and considerations when using Media services in the Azure Government environment. For service availability, see [Products available by region](#).

Media Services

For Azure Media Services v3 feature variations in Azure Government, see [Azure Media Services v3 clouds and regions availability](#).

Migration

This section outlines variations and considerations when using Migration services in the Azure Government environment. For service availability, see [Products available by region](#).

Azure Migrate

The following Azure Migrate **features aren't currently available** in Azure Government:

- Containerizing Java Web Apps on Apache Tomcat (on Linux servers) and deploying them on Linux containers on App Service.
- Containerizing Java Web Apps on Apache Tomcat (on Linux servers) and deploying them on Linux containers on Azure Kubernetes Service (AKS).
- Containerizing ASP.NET apps and deploying them on Windows containers on AKS.
- Containerizing ASP.NET apps and deploying them on Windows containers on App Service.
- You can only create assessments for Azure Government as target regions and using Azure Government offers.

For more information, see [Azure Migrate support matrix](#). For a list of Azure Government URLs needed by the Azure Migrate appliance when connecting to the internet, see [Azure Migrate appliance URL access](#).

Networking

This section outlines variations and considerations when using Networking services in the Azure Government environment. For service availability, see [Products available by region](#).

Azure ExpressRoute

For an overview of ExpressRoute, see [What is Azure ExpressRoute?](#). For an overview of how **BGP communities** are used with ExpressRoute in Azure Government, see [BGP community support in National Clouds](#).

Private Link

- For Private Link services availability, see [Azure Private Link availability](#).
- For Private DNS zone names, see [Azure Private Endpoint DNS configuration](#).

Traffic Manager

Traffic Manager health checks can originate from certain IP addresses for Azure Government. Review the [IP addresses in the JSON file](#) to ensure that incoming connections from these IP addresses are allowed at the endpoints to check its health status.

Security

This section outlines variations and considerations when using Security services in the Azure Government environment. For service availability, see [Products available by region](#).

Microsoft Defender for Endpoint

For feature variations and limitations, see [Microsoft Defender for Endpoint for US Government customers](#).

Microsoft Defender for IoT

For feature variations and limitations, see [Cloud feature availability for US Government customers](#).

Azure Information Protection

Azure Information Protection Premium is part of the [Enterprise Mobility + Security](#) suite. For details on this service and how to use it, see [Azure Information Protection Premium Government Service Description](#).

Microsoft Defender for Cloud

For feature variations and limitations, see [Cloud feature availability for US Government customers](#).

Microsoft Sentinel

For feature variations and limitations, see [Cloud feature availability for US Government customers](#).

Storage

This section outlines variations and considerations when using Storage services in the Azure Government environment. For service availability, see [Products available by region](#).

Azure NetApp Files

For Azure NetApp Files feature availability in Azure Government and how to access the Azure NetApp Files service within Azure Government, see [Azure NetApp Files for Azure Government](#).

Azure Import/Export

With Import/Export jobs for US Gov Arizona or US Gov Texas, the mailing address is for US Gov Virginia. The data is loaded into selected storage accounts from the US Gov Virginia region. For all jobs, we recommend that you rotate your storage account keys after the job is complete to remove any access granted during the process. For more information, see [Manage storage account access keys](#).

Web

This section outlines variations and considerations when using Web services in the Azure Government environment. For service availability, see [Products available by region](#).

API Management

The following API Management **features aren't currently available** in Azure Government:

- Azure AD B2C integration

App Service

The following App Service **resources aren't currently available** in Azure Government:

- App Service Certificate
- App Service Managed Certificate
- App Service Domain

The following App Service **features aren't currently available** in Azure Government:

- Deployment
 - Deployment options: only Local Git Repository and External Repository are available

Azure Functions

When connecting your Functions app to Application Insights in Azure Government, make sure you use [APPLICATIONINSIGHTS_CONNECTION_STRING](#), which lets you customize the Application Insights endpoint.

Next steps

Learn more about Azure Government:

- [Acquiring and accessing Azure Government](#)
- [Azure Government overview](#)
- [Azure support for export controls](#)
- [Azure Government compliance](#)
- [Azure Government security](#)
- [Azure guidance for secure isolation](#)

Start using Azure Government:

- [Guidance for developers](#)
- [Connect with the Azure Government portal](#)

Additional resources