Azure REST API reference

Article • 04/01/2024

Welcome to the Azure REST API reference documentation.

Representational State Transfer (REST) APIs are service endpoints that support sets of HTTP operations (methods), which provide create, retrieve, update, or delete access to the service's resources. This article walks you through:

How to call Azure REST APIs with curl

The basic components of a REST API request/response pair.

How to register your client application with Microsoft Entra ID to secure your REST requests.

Overviews of creating and sending a REST request, and handling the response.

The process described in the following blog post shows how to call an Azure REST API using curl . You might consider using curl in unattended scripts. For example, in DevOps automation scenarios.

Calling Azure REST API via curl

A REST API request/response pair can be separated into five components:

1. The request URI, which consists of: {URI-scheme} :// {URI-host} / {resource□path} ? {query-string} . Although the request URI is included in the request message header, we call it out separately here because most languages or frameworks require you to pass it separately from the request message.

□ Tip

Most Azure service REST APIs have client libraries that provide a native interface for using Azure services:

.NET | Java | Node.js | Python | Go | C++

How to call Azure REST APIs with curl

Components of a REST API request/response

URI scheme: Indicates the protocol used to transmit the request. For example, http or https .

URI host: Specifies the domain name or IP address of the server where the REST service endpoint is hosted, such as graph.microsoft.com .

Resource path: Specifies the resource or resource collection, which may include multiple segments used by the service in determining the selection of those resources. For example: beta/applications/00003f25-7e1f-4278-9488-efc7bac53c4a/owners can be used to query the list a specific application's owners within the applications collection.

Query string (optional): Provides additional simple parameters, such as the API version or resource selection criteria.

2. HTTP request message header fields:

A required HTTP method (also known as an operation or verb), which tells the service what type of operation you're requesting. Azure REST APIs support GET, HEAD, PUT, POST, and PATCH methods.

Optional additional header fields, as required by the specified URI and HTTP method. For example, an Authorization header that provides a bearer token containing client authorization information for the request.

3. Optional HTTP request message body fields, to support the URI and HTTP operation. For example, POST operations contain MIME-encoded objects that are passed as complex parameters. For POST or PUT operations, the MIME-encoding type for the body should be specified in the Content-type request header as well. Some services require you to use a specific MIME type, such as application/json .

4. HTTP response message header fields:

An HTTP status code , ranging from 2xx success codes to 4xx or 5xx error codes. Alternatively, a service-defined status code may be returned, as

indicated in the API documentation.

Optional additional header fields, as required to support the request's response, such as a Content-type response header.

5. Optional HTTP response message body fields:

MIME-encoded response objects are returned in the HTTP response body, such as a response from a GET method that is returning data. Typically, these objects are returned in a structured format such as JSON or XML, as indicated by the Content-type response header. For example, when you request an access token from Microsoft Entra ID, it's returned in the response body as the access_token element, one of several name/value paired objects in a data collection. In this example, a response header of Content-Type: application/json is also included.