

[Get Started for Free](#)[Contact Us](#)

[What is Cloud Computing?](#) / [Cloud Computing Concepts Hub](#) / [Application Integration](#)

What is an Api?

[View Application Integration Free Tier](#)

Explore Free Application Integration Offers

View free offers for Application Integration services in the cloud



Check out Application Integration Services

Innovate faster with the most comprehensive set of Application Integration services



Browse Application Integration Trainings

Get started on Application Integration training with content built by AWS experts

[Read Application Integration Blogs](#)

Read about the latest AWS Application Integration product news and best practices



What is an API?

APIs are mechanisms that enable two software components to communicate with each other using a set of definitions and protocols. For example, the weather bureau's software system contains daily weather data. The weather app on your phone "talks" to this system via APIs and shows you daily weather updates on your phone.

What does API stand for?

API stands for Application Programming Interface. In the context of APIs, the word Application refers to any software with a distinct function. Interface can be thought of as a contract of service between two applications. This contract defines how the two communicate with each other using requests and responses. Their API documentation contains information on how developers are to structure those requests and responses.

How do APIs work?

API architecture is usually explained in terms of client and server. The application sending the request is called the client, and the application sending the response is called the server. So in the weather example, the bureau's weather database is the server, and the mobile app is the client.

There are four different ways that APIs can work depending on when and why they were created.

SOAP APIs

These APIs use Simple Object Access Protocol. Client and server exchange messages using XML. This is a less flexible API that was more popular in the past.

RPC APIs

These APIs are called Remote Procedure Calls. The client completes a function (or procedure) on the server, and the server sends the output back to the client.

Websocket APIs

Websocket API is another modern web API development that uses JSON objects to pass data. A WebSocket API supports two-way communication between client apps and the server. The server can send callback messages to connected clients, making it more efficient than REST API.

REST APIs

These are the most popular and flexible APIs found on the web today. The client sends requests to the server as data. The server uses this client input to start internal functions and returns output data back to the client. Let's look at REST APIs in more detail below.

What are REST APIs?

REST stands for Representational State Transfer. REST defines a set of functions like GET, PUT, DELETE, etc. that clients can use to access server data. Clients and servers exchange data using HTTP.

The main feature of **REST API** is statelessness. Statelessness means that servers do not save client data between requests. Client requests to the server are similar to URLs you type in your browser to visit a website. The response from the server is plain data, without the typical graphical rendering of a web page.

What is web API?

A Web API or Web Service API is an application processing interface between a web server and web browser. All web services are APIs but not all APIs are web services. REST API is a special type of Web API that uses the standard architectural style explained above.

The different terms around APIs, like Java API or service APIs, exist because historically, APIs were created before the world wide web. Modern web APIs are REST APIs and the terms can be used interchangeably.

What are API integrations?

API integrations are software components that automatically update data between clients and servers. Some examples of API integrations are when automatic data sync to the cloud from your phone image gallery, or the time and date automatically sync on your laptop when you travel to another time zone. Enterprises can also use them to efficiently automate many system functions.

What are the benefits of REST APIs?

REST APIs offer four main benefits:

1. Integration

APIs are used to integrate new applications with existing software systems. This increases development speed because each functionality doesn't have to be written from scratch. You can use APIs to leverage existing code.

2. Innovation

Entire industries can change with the arrival of a new app. Businesses need to respond quickly and support the rapid deployment of innovative services. They can do this by making changes at the API level without having to re-write the whole code.

3. Expansion

APIs present a unique opportunity for businesses to meet their clients' needs across different platforms. For example, maps API allows map information integration via websites, Android, iOS, etc. Any business can give similar access to their internal databases by using free or paid APIs.

4. Ease of maintenance

The API acts as a gateway between two systems. Each system is obliged to make internal changes so that the API is not impacted. This way, any future code changes by one party do not impact the other party.

What are the different types of APIs?

APIs are classified both according to their architecture and scope of use. We have already explored the main types of API architectures so let's take a look at the scope of use.

Private APIs

These are internal to an enterprise and only used for connecting systems and data within the business.

Public APIs

These are open to the public and may be used by anyone. There may or not be some authorization and cost associated with these types of APIs.

Partner APIs

These are only accessible by authorized external developers to aid business-to-business partnerships.

Composite APIs

These combine two or more different APIs to address complex system requirements or behaviors.

What is an API endpoint and why is it important?

API endpoints are the final touchpoints in the API communication system. These include server URLs, services, and other specific digital locations from where information is sent and received between systems. API endpoints are critical to enterprises for two main reasons:

1. Security

API endpoints make the system vulnerable to attack. API monitoring is crucial for preventing misuse.

2. Performance

API endpoints, especially high traffic ones, can cause bottlenecks and affect system performance.

How to secure a REST API?

All APIs must be secured through proper authentication and monitoring. The two main ways to secure REST APIs include:

1. Authentication tokens

These are used to authorize users to make the API call. Authentication tokens check that the users are who they claim to be and that they have access rights for that particular API call. For example, when you log in to your email server, your email client uses authentication tokens for secure access.

2. API keys

API keys verify the program or application making the API call. They identify the application and ensure it has the access rights required to make the particular API call. API keys are not as secure as tokens but they allow API monitoring in order to gather data on usage. You may have noticed a long string of characters and numbers in your browser URL when you visit different websites. This string is an API key the website uses to make internal API calls.

How to create an API?

Due diligence and effort are required to build an API that other developers will want to work with and trust. These are the five steps required for high-quality API design:

1. Plan the API

API specifications, like OpenAPI, provide the blueprint for your API design. It is better to think about different use cases in advance and ensure the API adheres to current API development standards.

2. Build the API

API designers prototype APIs using boilerplate code. Once the prototype is tested, developers can customize it to internal specifications.

3. Test the API

API testing is the same as software testing and must be done to prevent bugs and defects. API testing tools can be used to strength test the API against cyber attacks.

4. Document the API

While APIs are self-explanatory, API documentation acts as a guide to improve usability. Well-documented APIs that offer a range of functions and use cases tend to be more popular in a service-oriented architecture.

5. Market the API

Just as Amazon is an online marketplace for retail, API marketplaces exist for developers to buy and sell other APIs. Listing your API can allow you to monetize it.

What is API testing?

API testing strategies are similar to other software testing methodologies. The main focus is on validating server responses. API testing includes:

- Making multiple requests to API endpoints for performance testing.
- Writing unit tests for checking business logic and functional correctness.
- Security testing by simulating system attacks.

How to write API documentation?

Writing comprehensive API documentation is part of the [API management process](#). API documentation can be auto-generated using tools or written manually. Some best practices include:

- Writing explanations in simple, easy-to-read English. Documents generated by tools can become wordy and require editing.
- Using code samples to explain functionality.
- Maintaining the documentation so it is accurate and up-to-date.
- Aiming the writing style at beginners
- Covering all the problems the API can solve for the users.

How to use an API?

The steps to implement a new API include:

1. Obtaining an API key. This is done by creating a verified account with the API provider.
2. Set up an HTTP API client. This tool allows you to structure API requests easily using the API keys received.
3. If you don't have an API client, you can try to structure the request yourself in your browser by referring to the API documentation.
4. Once you are comfortable with the new API syntax, you can start using it in your code.

Where can I find new APIs?

New web APIs can be found on API marketplaces and API directories. API marketplaces are open platforms where anyone can list an API for sale. API directories are controlled repositories regulated by the directory owner. Expert API designers may assess and test a new API before adding it to their directory.

Some popular API websites include:

- Rapid API – The largest global API market with over 10,000 public APIs and 1 million active developers on site. RapidAPI allows users to test APIs directly on the platform before committing to purchase.
- Public APIs – The platform groups remote APIs into 40 niche categories, making it easier to browse and find the right one to meet your needs.
- APIForThat and APIList – Both these websites have lists of 500+ web APIs, along with in-depth information on how to use them.

What is an API gateway?

An API Gateway is an API management tool for enterprise clients that use a broad range of back-end services. API gateways typically handle common tasks like user authentication, statistics, and rate management that are applicable across all API calls.

[Amazon API Gateway](#) is a fully managed service that makes it easy for developers to create, publish, maintain, monitor, and secure APIs at any scale. It handles all the tasks involved in accepting and processing thousands of concurrent API calls, including traffic management, CORS support, authorization, and access control, throttling, monitoring, and API version management.

What is GraphQL?

GraphQL is a query language that was developed specifically for APIs. It prioritizes giving clients exactly the data they request and no more. It is designed to make APIs fast, flexible, and developer-friendly. As an alternative to REST, GraphQL gives front-end developers the ability to query multiple databases, microservices, and APIs with a single GraphQL endpoint. Organizations choose to build APIs with GraphQL because it helps them develop applications faster. [Read more about GraphQL here.](#)

AWS AppSync is a fully managed service that makes it easy to develop GraphQL APIs by handling the heavy lifting of securely connecting to data sources like AWS DynamoDB, AWS Lambda, and more AWS AppSync can push real-time data updates over Websockets to millions of clients. For mobile and web applications, AppSync also provides local data access when devices go offline. Once deployed, AWS AppSync automatically scales GraphQL API execution engine up and down to meet API request volumes.

How to get Amazon API services?

Application Programming Interface management is an important part of modern software development. It is worth investing in API infrastructure, including tools, gateway, and microservices architecture for both internal and external users.

[Amazon API Gateway](#) comes with a [full range of features](#) to manage multiple APIs simultaneously and efficiently. You can make up to one million API calls for free by [signing up at the AWS Portal](#).

[AWS AppSync](#) offers fully managed GraphQL API setup, administration, and maintenance, with high-availability serverless infrastructure built-in. You pay only for what you use with no minimum fees or mandatory service usage. To get started, please sign in to the AWS AppSync Console.

Next steps on AWS



Check out additional product-related resources

[Learn more about IDE Services »](#)



Sign up for a free account

Instantly get access to the AWS free tier.

[Sign up »](#)



Start building in the console

Get started building with API gateway in the AWS management console.

[Sign in »](#)

[Sign In to the Console](#)

Learn About AWS

[What Is AWS?](#)

[What Is Cloud Computing?](#)

[AWS Diversity, Equity & Inclusion](#)

[What Is DevOps?](#)

[What Is a Container?](#)

[What Is a Data Lake?](#)

[AWS Cloud Security](#)

[What's New](#)

[Blogs](#)

[Press Releases](#)

Resources for AWS

[Getting Started](#)

[Training and Certification](#)

[AWS Solutions Portfolio](#)

[Architecture Center](#)

[Product and Technical FAQs](#)

[Analyst Reports](#)

[AWS Partners](#)

Developers on AWS

[Developer Center](#)

[SDKs & Tools](#)

[.NET on AWS](#)

[Python on AWS](#)

[Java on AWS](#)

[PHP on AWS](#)

[JavaScript on AWS](#)

Help

[Contact Us](#)

[File a Support Ticket](#)

[Knowledge Center](#)

[AWS re:Post](#)

[AWS Support Overview](#)

[Legal](#)

[AWS Careers](#)

[Create an AWS Account](#)



Amazon is an Equal Opportunity Employer: *Minority / Women / Disability / Veteran / Gender Identity / Sexual Orientation / Age.*

Language

[عربي](#) |
[Bahasa Indonesia](#) |
[Deutsch](#) |
[English](#) |
[Español](#) |
[Français](#) |
[Italiano](#) |
[Português](#) |
[Tiếng Việt](#) |
[Türkçe](#) |
[Русский](#) |
[ไทย](#) |
[日本語](#) |
[한국어](#) |
[中文 \(简体\)](#) |
[中文 \(繁體\)](#)

[Privacy](#)

|

[Site Terms](#)

|

[Cookie Preferences](#)

|

© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.