# **Tìm hiểu về API Gateway trong AWS**

1. **API là gì?**

* API là viết tắt của Application Programming Interface(Giao diện chương trình ứng dụng) - phương thức trung gian kết nối các ứng dụng và thư viện khác nhau.
* Nó cung cấp khả năng truy xuất đến một tập các hàm hay dùng, từ đó có thể trao đổi dữ liệu giữa các ứng dụng. Nó không phải là một ngôn ngữ lập trình mà chỉ là các hàm hay thủ tục thông thường. Các hàm này được viết trên nhiều ngôn ngữ khác nhau.

1. **API hoạt động như thế nào?**

* Kiến trúc API thường được giải thích dưới dạng máy chủ và máy khách. Ứng dụng gửi yêu cầu được gọi là máy khách, còn ứng dụng gửi phản hồi được gọi là máy chủ. Như vậy, trong ví dụ về thời tiết, cơ sở dữ liệu của cơ quan thời tiết là máy chủ còn ứng dụng di động là máy khách.
* API hoạt động theo 4 cách khác nhau, tùy vào thời điểm và lý do chúng được tạo ra.
  + **API SOAP**: Các API này sử dụng Giao thức truy cập đối tượng đơn giản. Máy chủ và máy khách trao đổi thông đệp bằng XML. Đây là loại API kém linh hoạt được dùng phổ biến trước đây.
  + **API RPC**: Những API này được gọi là Lệnh gọi thủ tục từ xa. Máy khách hoàn thành một hàm (hoặc thủ tục) trên máy chủ còn máy chủ gửi kết quả về cho máy khách.
  + **API Websocket**: API Websocket là một bản phát triển API web hiện đại khác sử dụng các đối tượng JSON để chuyển dữ liệu. API WebSocket hỗ trợ hoạt động giao tiếp hai chiều giữa ứng dụng máy khách và máy chủ. Máy chủ có thể gửi thông điệp gọi lại cho các máy khách được kết nối, điều này khiến loại API này hiệu quả hơn API REST.
  + **API REST**: Đây là loại API phổ biến và linh hoạt nhất trên web hiện nay. Máy khách gửi yêu cầu đến máy chủ dưới dạng dữ liệu. Máy chủ dùng dữ liệu đầu vào từ máy khách này để bắt đầu các hàm nội bộ và trả lại dữ liệu đầu ra cho máy khách. Hãy cùng xem xét API REST chi tiết hơn ở bên dưới.

1. **Các loại API trên AWS**

Có 2 loại API trên AWS là API RESTful và API WEBSOCKET

API được sử dụng trong đồ án đó là API

* 1. **API RESTful**
  + Là một tiêu chuẩn dùng trong việc thiết kế API cho các ứng dụng web (thiết kế Web services) để tiện cho việc quản lý các resource. Nó chú trọng vào tài nguyên hệ thống (tệp văn bản, ảnh, âm thanh, video, hoặc dữ liệu động…), bao gồm các trạng thái tài nguyên được định dạng và được truyền tải qua HTTP.
  + Xây dựng các API RESTful được tối ưu hóa cho khối lượng công việc serverless và backend HTTP bằng API HTTP. API HTTP là lựa chọn tốt nhất để xây dựng những API chỉ yêu cầu chức năng proxy API. Nếu API của bạn yêu cầu cả chức năng proxy API lẫn tính năng quản lý API trong cùng một giải pháp thì API Gateway cũng cung cấp cả API REST.
  + REST hoạt động chủ yếu dựa vào giao thức HTTP. Các hoạt động cơ bản nêu trên sẽ sử dụng những phương thức HTTP riêng.
  + GET (SELECT): Trả về một Resource hoặc một danh sách Resource.
  + POST (CREATE): Tạo mới một Resource.
  + PUT (UPDATE): Cập nhật thông tin cho Resource.
  + DELETE (DELETE): Xoá một Resource.
  1. **API WEBSOCKET**
  + Xây dựng các ứng dụng giao tiếp hai chiều theo thời gian thực, chẳng hạn như ứng dụng trò chuyện và bảng điều khiển truyền phát bằng API WebSocket. API Gateway duy trì kết nối lâu dài để xử lý quá trình truyền tin nhắn giữa dịch vụ backend và máy khách của bạn.

1. **API Gateway**
   1. **API Gateway là gì?**

API Gateway là phương pháp tiếp cận tối ưu cho kiến trúc microservices. Một cổng kết nối API là một máy chủ truy xuất duy nhất vào hệ thống. Nó cũng tương tự như mẫu thiết kế Facade dựa trên thiết kế hướng đối tượng. Cổng kết nối API che giấu đi thông tin kiến trúc hệ thống nội bộ và nó cung cấp các API tùy chỉnh cho mỗi Client. Cổng kết nối API còn có trách nhiệm xác thực, giám sát, cân bằng tải, caching, định hình yêu cầu và quản lí thông tin, xử lí phản hồi tĩnh.

Cổng kết nối API làm nhiệm vụ định tuyến các yêu cầu, kết hợp và chuyển đổi các giao thức. Tất cả yêu cầu từ Client đều đi qua cổng kết nối API. Sau đó cổng kết nối API định tuyến các yêu cầu này tới microservices phù hợp. Cổng kết nối API Gateway sẽ xử lý một yêu cầu người dùng bằng cách gọi đến một loạt microservices rồi tổng hợp các kết quả. Nó có thể chuyển đổi giữa các giao thức web như HTTP, WebSocket và các giao thức nội bộ không thân thiện với web.

* 1. **Tính năng của API Gateway**
  + **Tối ưu đầu cuối**: Khi một request của Client tới dịch vụ yêu cầu thông tin của nhiều thành phần trong hệ thống, ví dụ: Customer, Order, Invoices... API Gateway sẽ thực hiện request đồng thời tới các thành phần, lấy về các thông tin từ từng thành phần, ghép lại và phản hồi lại kết quả mong muốn cho Client.
  + **Ngắt kết nối API**: Khi mỗi thành phần trong hệ thống dừng hoạt động, hoặc trả về số lượng lớn lỗi đạt đến ngưỡng đã được quy định, API Gateway có thể tự động thực hiện dừng gửi request tới thành phần gặp lỗi. Việc này sẽ cho đội vận hành có thời gian để phân tích log, xử lý lỗi và update lại thành phần.
  + **Thuận tiện nâng cấp các thành phần**: Khi các thành phần trong dịch vụ được nâng cấp, phát triển thêm tính năng API Gateway có các hình thức để có thể điều phối tỉ lệ các request giữa phiên bản hiện tại và phiên bản mới.
  + **Kết nối tập trung**: Client sử dụng API sẽ kết nối tập trung qua API Gateway, API Gateway sẽ thực hiện các việc xác thực request của client, log các request, giám sát các Microservices, báo cáo/phân tích log...
  + **Cân bằng tải**: API Gateway có thể phối hợp cùng Service-discovery tool để thực hiện cân bằng tải, phân phối request tới các cụm API.

1. **Những ưu điểm nổi trội của API**

* **Phát triển API hiệu quả**: Chạy đồng thời nhiều phiên bản của cùng một API với API Gateway, từ đó cho phép bạn lặp lại, kiểm tra và phát hành các phiên bản mới một cách nhanh chóng. Bạn trả tiền cho các lệnh gọi được thực hiện với các API của mình và chuyển dữ liệu ra và không có lệ phí tối thiểu hay cam kết trả trước.
* **Hoạt động ở mọi quy mô**: Cung cấp cho người dùng cuối độ trễ thấp nhất có thể cho các yêu cầu và phản hồi API bằng cách tận dụng mạng lưới các vị trí biên bằng cách sử dụng Amazon CloudFront. Điều tiết lưu lượng truy cập và cấp phép các cuộc gọi API nhằm đảm bảo rằng các hoạt động backend chịu được các lưu lượng truy cập tăng đột biến và các hệ thống backend không được gọi một cách không cần thiết.
* **Tiết kiệm chi phí ở quy mô lớn**: API Gateway cung cấp mô hình giá theo bậc cho các yêu cầu API. Với mức giá API thấp 0.90 USD trên một triệu yêu cầu ở bậc cao nhất, bạn có thể giảm chi phí khi mức độ sử dụng API tăng lên trên mỗi khu vực trong toàn bộ các tài khoản AWS.
* **Theo dõi dễ dàng**: Giám sát các số liệu về hiệu năng và thông tin về các lệnh gọi API, độ trễ dữ liệu và tỷ lệ lỗi từ bảng điều khiển API Gateway, nhờ đó bạn có thể giám sát trực quan các lệnh gọi đến dịch vụ của bạn thông qua Amazon CloudWatch.
* **Kiểm soát bảo mật linh hoạt**: Cho phép truy cập các API của bạn bằng AWS Identity and Access Management (IAM) và Amazon Cognito. Nếu bạn sử dụng token OAuth, API Gateway có hỗ trợ OIDC gốc và OAuth2. Để hỗ trợ các yêu cầu cấp phép tùy chỉnh, bạn có thể thực thi trình cấp quyền Lambda từ AWS Lambda.