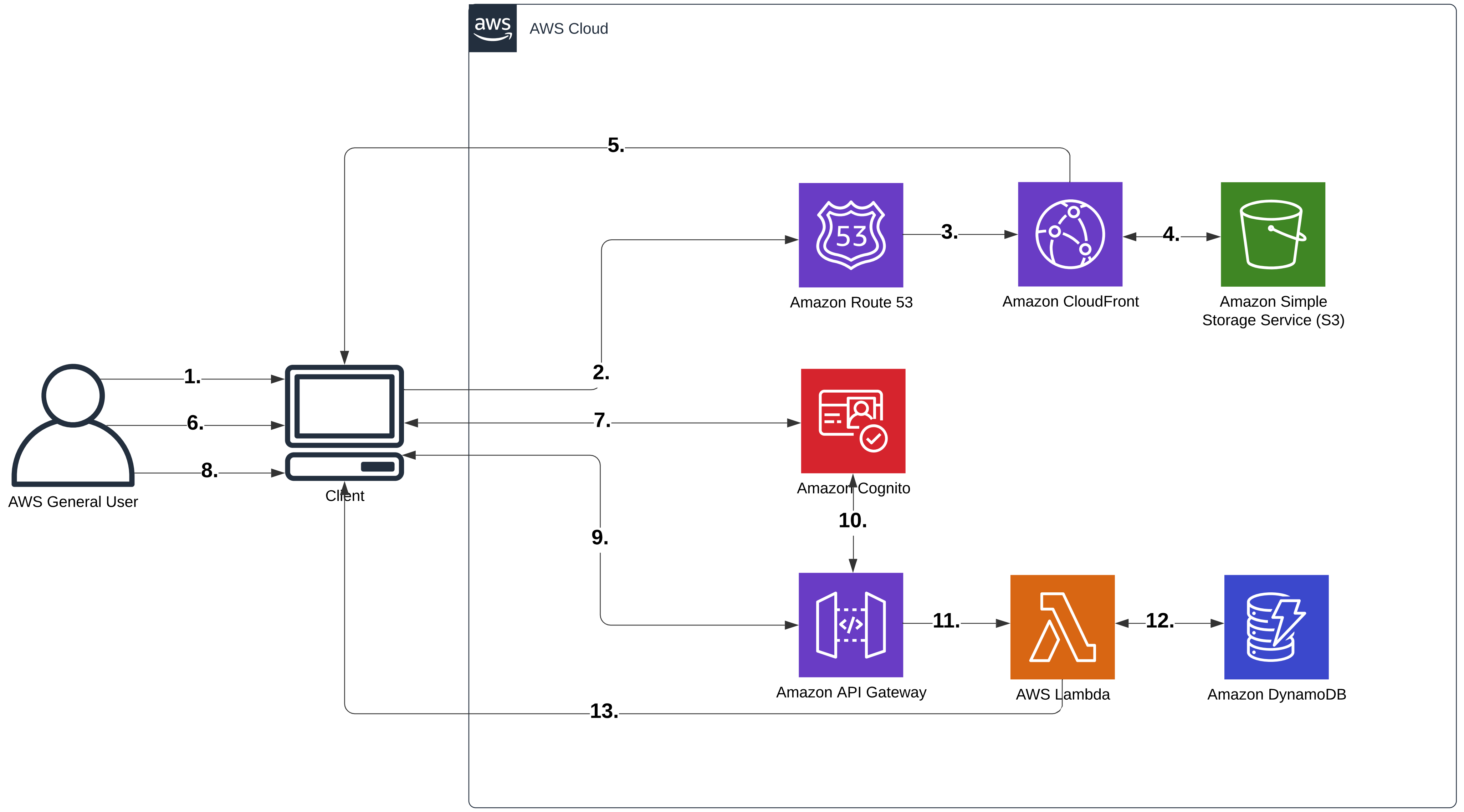
Building a Serverless API with Cognito, Lambda, and DynamoDB

By Hou Chia | June 10, 2020



System Elements

1. The ***User*** visits the website using a public website domain.

2. The ***Client*** sends a HTTP GET request to ***Amazon Route 53,*** where the website domain is registered.

3. *Amazon Route 53,* which contains an alias record pointing to ***Amazon CloudFront,*** routes the DNS to that ***Amazon CloudFront.***

4. *Amazon CloudFront* retrieves the requested static website assets (e.g., HTML/CSS, JavaScript, images, etc.) from **S3. *Amazon CloudFront*** speeds up content delivery by requesting the content from the Edge location closest to the ***User***.

5. *Amazon CloudFront* sends the assets to the ***Client.*** The login page is displayed in the ***Client***.

6. Assuming the ***User*** has already created an account, he/she enters his/her credentials (e.g., username, email, password) into the ***Client.***

7. *Amazon Cognito* authenticates the ***User*** using the Secure Remote Password (SRP) protocol. If authentication succeeds, Cognito User Pools sends the ***Client*** a set of JSON Web Tokens (JWT). If authentication fails, it sends back a 403 Forbidden message.

8. *User* performs an action on the ***Client*** to view protected data**.**

9. The ***Client*** sends the HTTP request to the ***Amazon API Gateway,*** which exposes our ***AWS Lambda*** function via a private RESTful API. Our ***Amazon API Gateway*** is Edge-optimized to serve our global user base.

10. *Amazon API Gateway* verifies the ***Client's*** JWT (see Step 7) by sending a request to ***Amazon Cognito***. If authorization succeeds, the ***User*** can proceed to Step 11. Otherwise, it sends back a 401 Unauthorized message to the ***Client***.

11. Invoke the ***AWS Lambda*** function via the ***Amazon API Gateway*** RESTful API.

12. *AWS Lambda* connects to the data persistence layer (i.e., ***DynamoDB***) and fetches the requested data. ***AWS Lambda*** is low-cost (i.e., charge is based on RAM allocated for every 100ms of compute time used) and automatically scales with the number of requests.

13. *AWS Lambda* serves the JSON data to the ***Client.***