

AWS Lambda



Run code **without provisioning or managing** servers.
Servers automatically start and stop when needed.
Serverless Functions. Pay per invocation.





Introduction to AWS Lambda

AWS Lambda is a compute service that lets you run code **without provisioning or managing servers.**

Lambda executes your code only when needed and scales automatically to a few to a 1000 lambda functions concurrently in seconds.

You pay only for the compute time you consume - there is no charge when your code is not running.

Lambda is **Cheap**

Lambda is **Serverless**

Lambda **Scales Automatically**

Natively supports 7 runtimes languages:

You can also create your own **custom runtime** environments

1. Ruby
2. Python
3. Java
4. Go
5. Powershell
6. NodeJs
7. C#



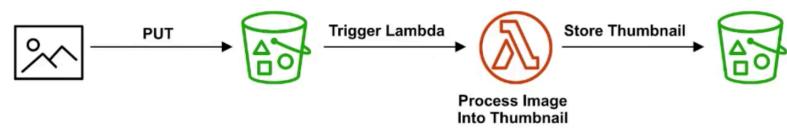


AWS Lambda - Use Cases

Lambda is commonly used to **glue different services together** so the use cases are endless.

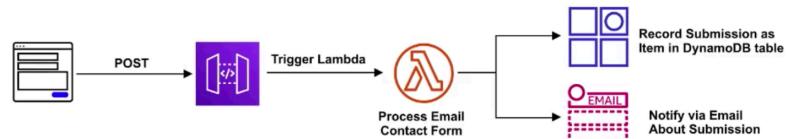
Processing Thumbnails

A web-service allows users to upload their profile photo. They are stored in an S3 bucket. We can setup an Event Trigger which will invoke a Lambda which will process the Profile Photo into a Thumbnail and the store it back in the bucket.



Contact Email Form

A company has a contact email form which submits form data via API Gateway Endpoint. That endpoint triggers a lambda which validates the form data and if valid will save the submission in DynamoDB and send an email notification via SNS to the company





AWS Lambda - Triggers

Lambdas can be **invoked** via the AWS SDK or trigger from other AWS Services.
(This is not a complete list)

 API Gateway api application-services aws serverless	 CloudWatch Logs aws logging management-tools
 AWS IoT aws devices iot	 CodeCommit aws developer-tools git
 Alexa Skills Kit alexa iot	 Cognito Sync Trigger authentication aws identity mobile-services sync
 Alexa Smart Home alexa iot	 DynamoDB aws database nosql
 Application Load Balancer aws load-balancing	 Kinesis analytics aws streaming
 CloudFront aws cdn edge	 S3 aws storage
 CloudWatch Events aws events management-tools	 SNS aws messaging notifications pub-sub push
	 SQS aws queue



SUBSCRIBE



AWS Lambda - Triggers

Partner event sources (powered by Amazon EventBridge)



Datadog

Datadog is the essential monitoring platform for cloud applications.



OneLogin

OneLogin, the leader in Unified Access Management, connects people with technology through a simple and secure login, empowering organizations to access the world.



PagerDuty

PagerDuty helps AWS users automatically turn any signal into the right insight and action.



Saviynt

Saviynt enables enterprises to secure applications, data and infrastructure in a single platform for cloud and enterprise.



Segment

Segment provides the customer data infrastructure that businesses use to put their customers first.



SignalFx

SignalFx, the only real-time cloud monitoring platform for infrastructure, microservices, and applications, collects and analyzes metrics and traces across every component in your cloud environment.



SugarCRM

SugarCRM enables businesses to create extraordinary customer relationships with the most empowering, adaptable and affordable customer relationship management (CRM) solution on the market.



Whispir

Whispir is a cloud based platform that automates, personalises and layers communications using smart workflow technology.



Zendesk

Zendesk makes better customer service experiences for agents, admins, and customers.



SUBSCRIBE



AWS Lambda – Pricing

First **1 million requests** per month are free.

There-after **\$0.20** per additional 1 million requests

400,000 GB seconds free per month

Thereafter **\$0.0000166667** for every GB second

**This price will vary on the amount of memory you allocate

128MB of Memory **X** 30M executed per month **X** 200ms run time per invocation = **\$5.83**



SUBSCRIBE



AWS Lambda - Interface

You choose your runtime

The screenshot shows the AWS Lambda interface with a red arrow pointing from the text "You choose your runtime" to the "Code entry type" dropdown menu. The dropdown menu has four options: "Edit code inline" (selected), "Edit code inline", "Upload a .zip file", and "Upload a file from Amazon S3". Above the dropdown, there is a "Runtime" dropdown set to "Ruby 2.5".

You upload Your code

The screenshot shows the AWS Lambda interface with a red arrow pointing from the text "You upload Your code" to the code editor area. The code editor contains a Ruby script named "lambda_function.rb". The code defines a lambda_handler function that processes event records and pushes them to a DynamoDB table. A red arrow also points from the text "You choose your triggers" to the trigger configuration section below.

You choose your triggers

The screenshot shows the AWS Lambda interface with a red arrow pointing from the text "You choose your triggers" to the "DynamoDB" trigger configuration section. The "DynamoDB" tab is selected, and the "Add trigger" button is visible. To the right, a list of available triggers includes "exapro-events" (selected), "Layers (0)", "Amazon CloudWatch Logs", "Amazon DynamoDB", and "Amazon Kinesis Firehose". A red arrow also points from the text "You grant permissions for outputs via an IAM Role" to the "Resources that the function's role has access to appear here" section.

You grant permissions for outputs via an IAM Role





AWS Lambda - Defaults and Limits

By Default you can have 1000 Lambda running concurrently
(Ask AWS Support for Limit Increase)

/tmp directory can contain up to **500MB**

By Default Lambda run in No VPC. You can set them to by in your own VPC but your lambda will lose internet access

You can set timeout to be a maximum of **15 minutes**

Memory can be set between **128MB** to a Maximum of **3008MB** at an increment of **64MB**

Unreserved account concurrency **1000**

- Use unreserved account concurrency
 Reserve concurrency

Virtual Private Cloud (VPC) [Info](#)

Choose a VPC for your function to access.

No VPC

Timeout [Info](#)

15 min 0 sec

Memory (MB) [Info](#)

Your function is allocated CPU proportional to the memory configured.

3008 MB



SUBSCRIBE

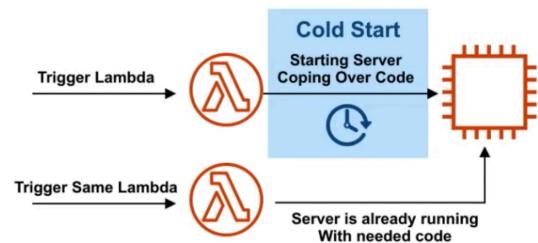


AWS Lambda - Cold Starts

AWS has servers preconfigured (just sitting around turned off) for your runtime environment. When a Lambda is invoked these servers need to be turned on and your code needs to be copied over.

During the time there will be a delay when the function will initially run which is called a **Cold Start**

If the same Lambda is invoked and the server is still running it will use that server again, so there will be little to delay to running that function. This what we call a **Warm Server**



Serverless functions are **cheap** but everything comes with a trade off. Serverless functions Cold Starts can **cause delays in the User Experience**. If your web-application relies on being very responsive, than you want to reconsider Serverless functions.

There are strategies around Cold Starts such as **Pre Warming** which keep servers continuously running. Cloud Providers are always looking for ways to reduce cold starts.



Lambda *CheatSheet*

- **Lambda's** are serverless **functions**. You upload your code and it runs without you managing or provisioning any servers.
- Lambda is **serverless**. You don't need to worry about underlying architecture
- Lambda is a good fit for short running tasks where you don't need to customize the os environment. If you need long running tasks (> 15mins) and a custom OS environment than consider using **Fargate**
- There are **7 runtime language environments** officially supported by Lambda: **Ruby, Python, Java, NodeJs, C#, Powershell and Go**
- You pay per invocation (The **duration** and the amount of **memory** used) rounded up to the nearest 100 milliseconds and you based on amount of requests. First 1M requests per month are free
- You can adjust the duration timeout for up to **15 mins** and memory up to **3008 MB**
- You can trigger Lambdas from the SDK or multiple AWS services eg. S3, API Gateway, DynamoDB
- Lambdas by default run in No VPC. To interact with some services you need to have your Lambda in the same VPC eg. RDS
- Lambda can scale to **1000 of concurrent functions** in seconds. (1000 is the default, you can increase with AWS Service Limit Increase)
- Lambdas have **Cold Starts**. If a function has not been recently been execute there will be a delay



SUBSCRIBE