Contents

[**Compute** 1](#_Toc41114971)

[**Automatically updating security groups with lambda function** 1](#_Toc41114972)

[**Best Practice organizing large serverless applications** 1](#_Toc41114973)

# **Compute**

## **Automatically updating security groups with lambda function**

<https://aws.amazon.com/blogs/compute/automating-security-group-updates-with-aws-lambda/>

When autoscaling in triggered, it transmits “lifecycle hook” event, which can be used for triggering a lambda function which can list out all the public IP address of that EC2 instance and then update the security group to add the new Ips to allow it passthrough the security group.

## **Best Practice organizing large serverless applications**

<https://aws.amazon.com/it/blogs/compute/best-practices-for-organizing-larger-serverless-applications/>

* Organized code into multiple repositories – not too small codebase that are difficult to share and not too large codebase that kills reusability. Use SAM to create a deployment pipeline for each of the organized code repositories.
* User AWS service instead of code libraries – for example instead of routing code written in Flask or node.js use API gateway.
* Use separate organization/AWS account for each developers or group of developers, separated from the production environment. Implement CI/CD for deployment of code from/to multiple account.
* Implement CI/CD to manage future releases.