**Automated AMI Backup**

# **Objectives:**

Creating an AMI which automatically snapshots all associated EBS volumes for that instance. This is to improve the reliability and efficiency to recover the instance. A daily backup schedule is used with a retention period of 7 days for each snapshot. The system also removes the automatically created AMIs and any associated snapshots as per the settings.

**Source**: [AWS-AMI-Creation-Deletion-Github](https://github.com/webdigi/AWS-AMI-Automated-Creation-Deletion)

**Prerequisites:**

1. Access to your **AWS Console Management** (<https://aws.amazon.com/>)
2. Lambda createAMI.js script
3. Lambda deleteAMI.js script
4. IAM role.json script

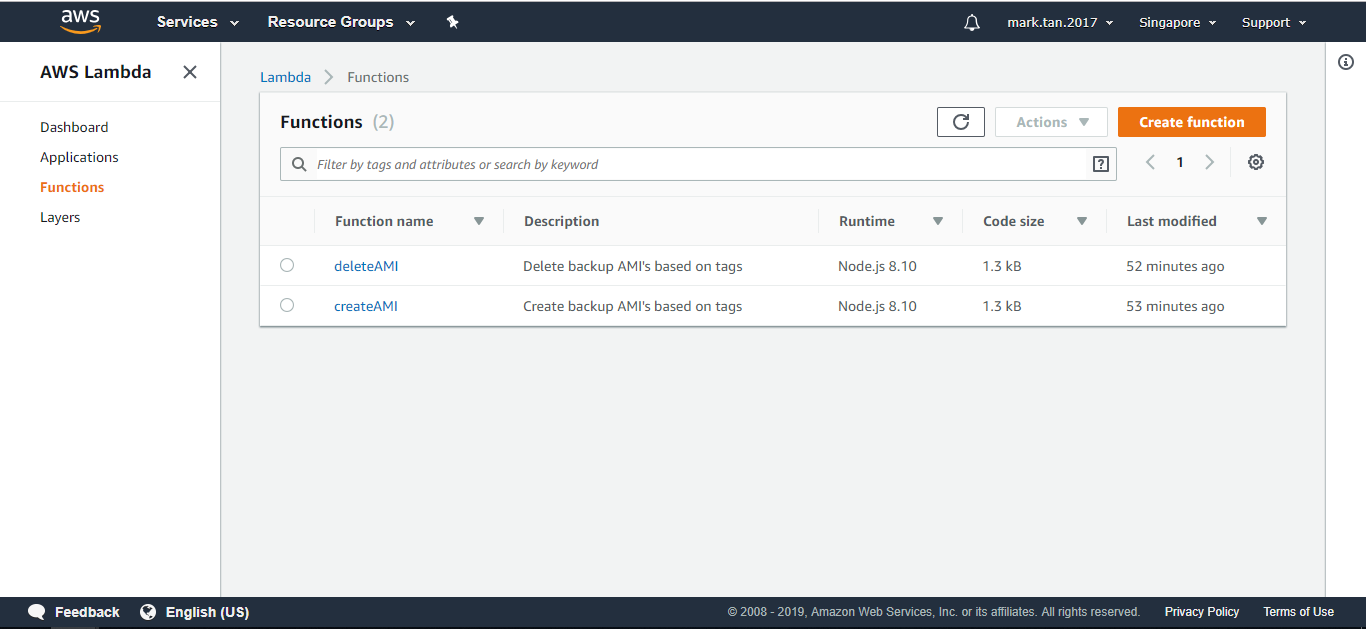
# **Overview:**

● [Task A: Creating Lambda Functions](#_altoubdqyzu)

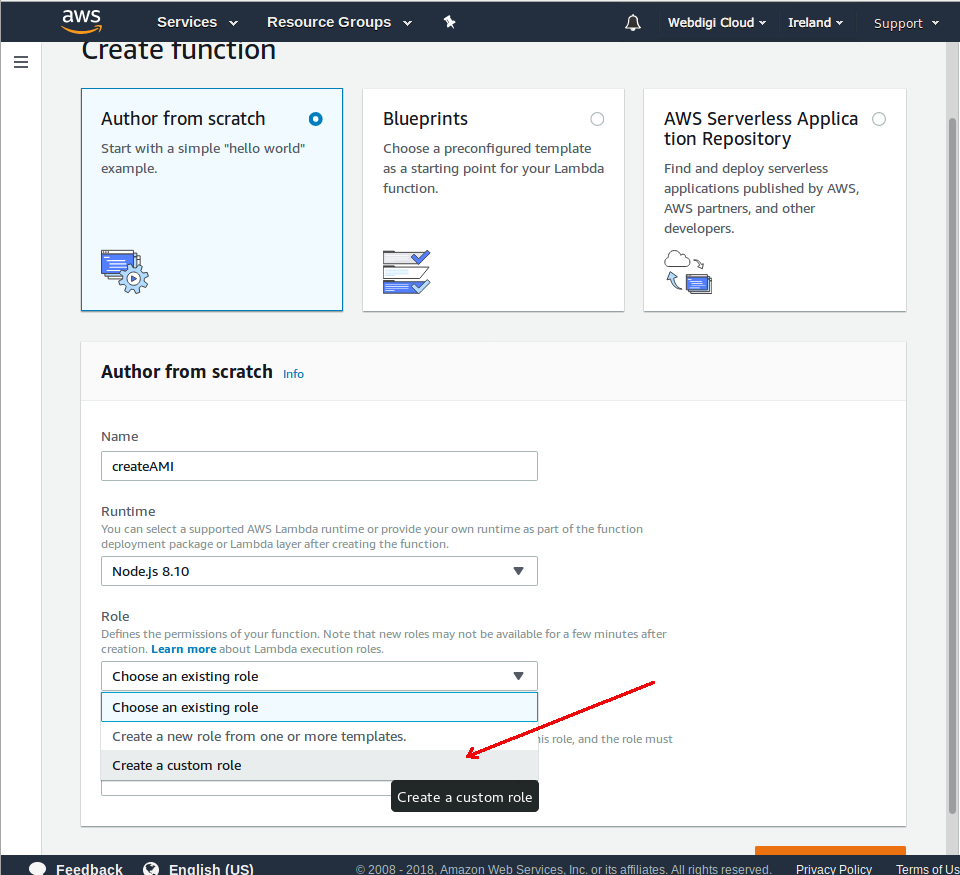
● [Task B: Setting tags for EC2 instances](#_6yul94zgdxx1)

# **Task A: Creating Lambda Function**

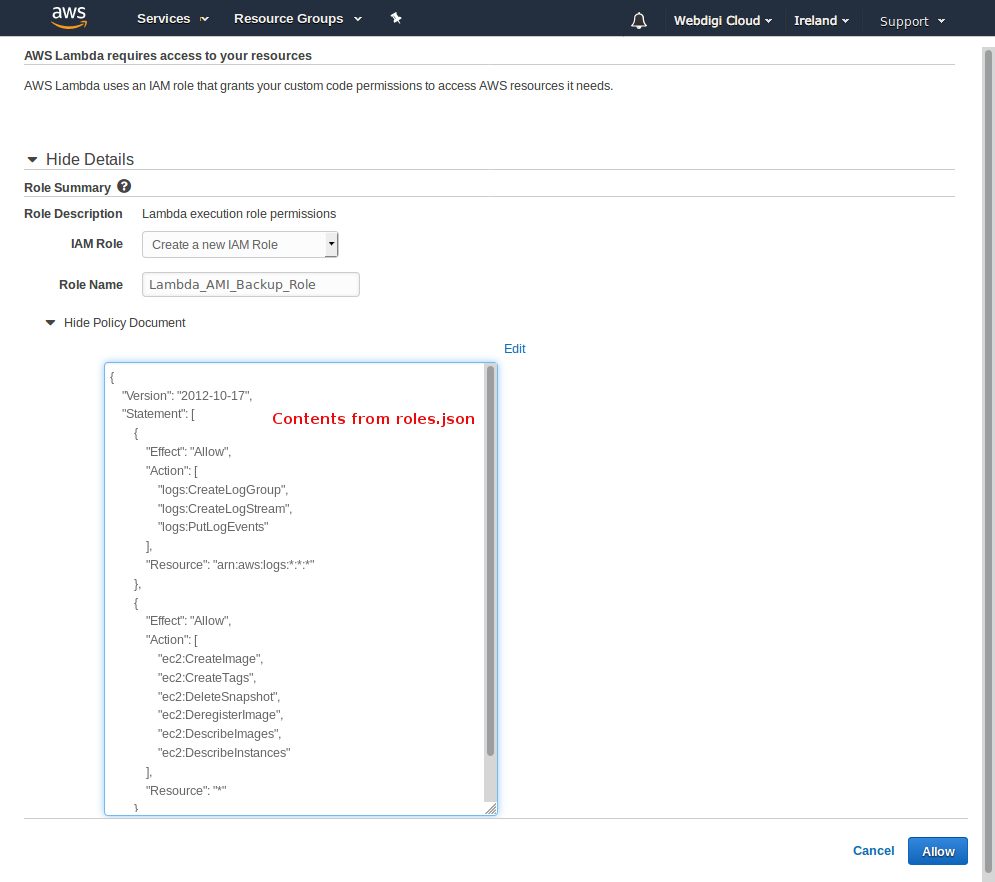
1. Go to the *AWS Lambda Console* and click *Create function*.



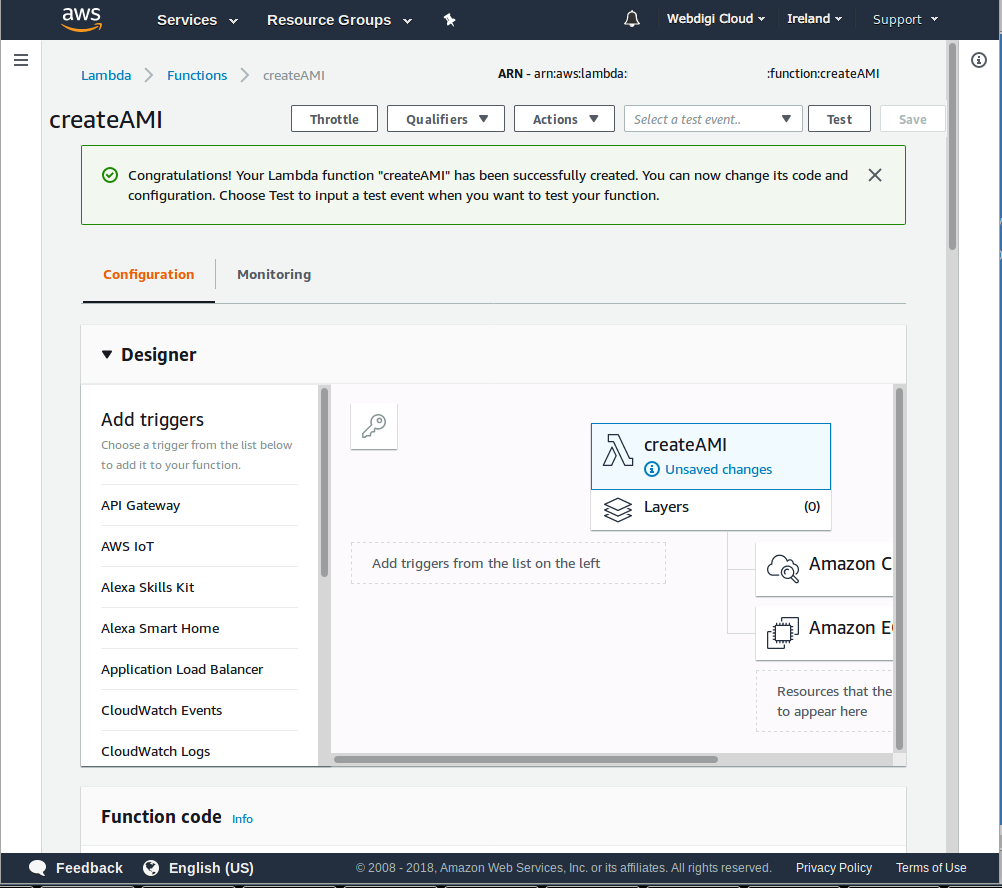
1. Name the function call ‘***createAMI*’** and select *Create a custom role*:



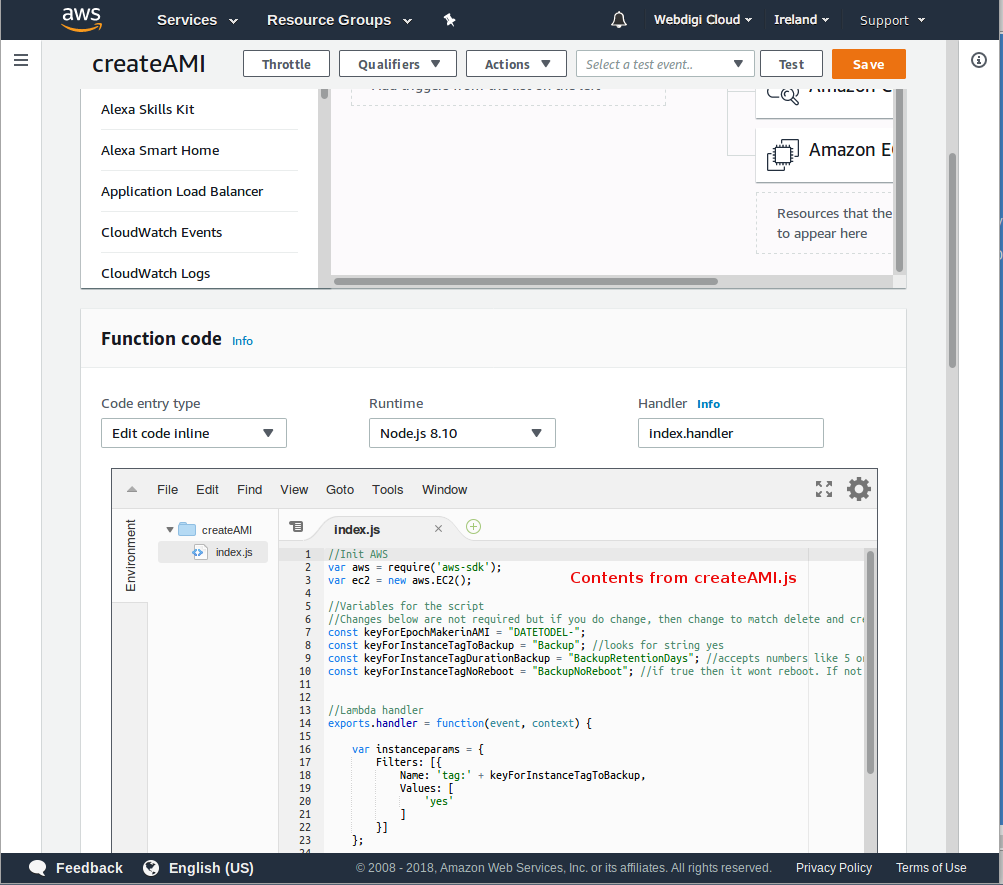
1. Give the custom role a name, ‘**Lambda\_AMI\_Backup\_Role**’, and paste the contents of [roles.json](https://github.com/webdigi/AWS-AMI-Automated-Creation-Deletion/blob/master/roles/roles.json) into the edit box.



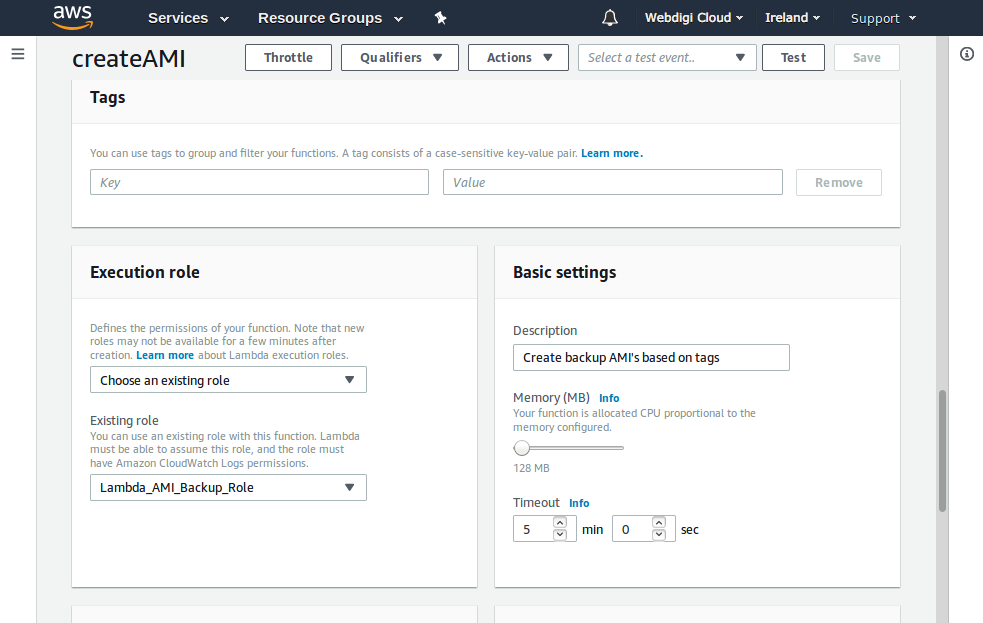
1. Now the function has been created, and you'll be presented with the lambda configuration screen.



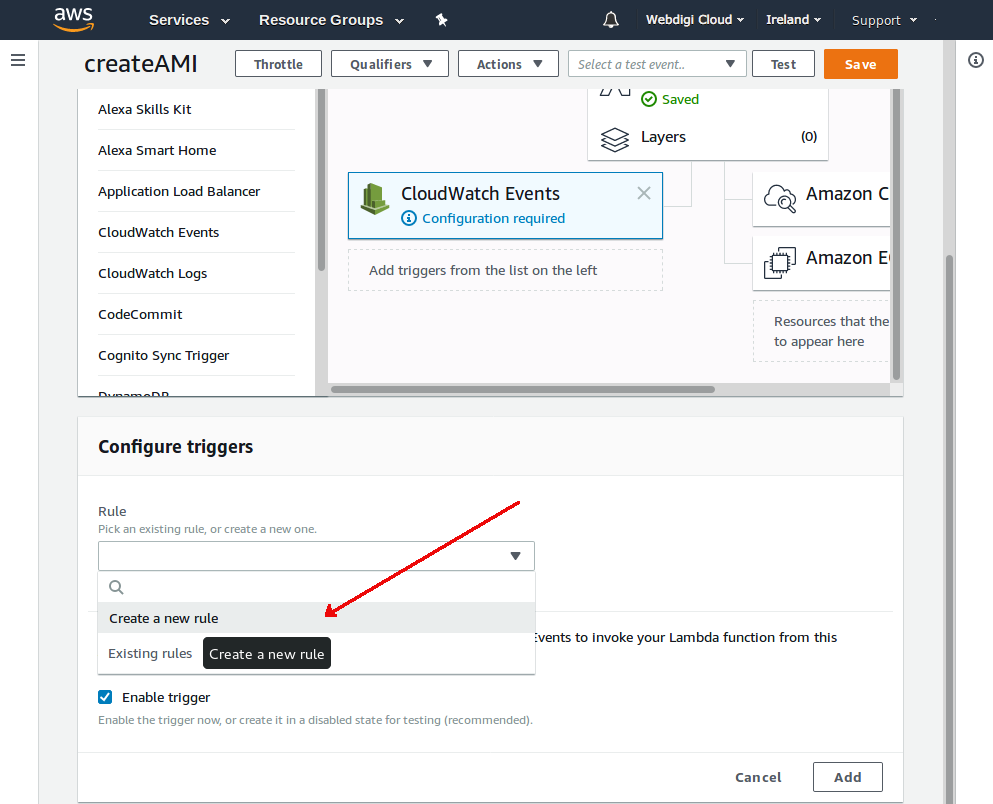
1. Scroll down to the code editor. Replace the stub code with the the contents of **createAMI.js** into the edit box.



1. Scroll down further and modify the *Basic Settings* and provide a *description* and set the timeout to 5 minutes. Click *Save*.

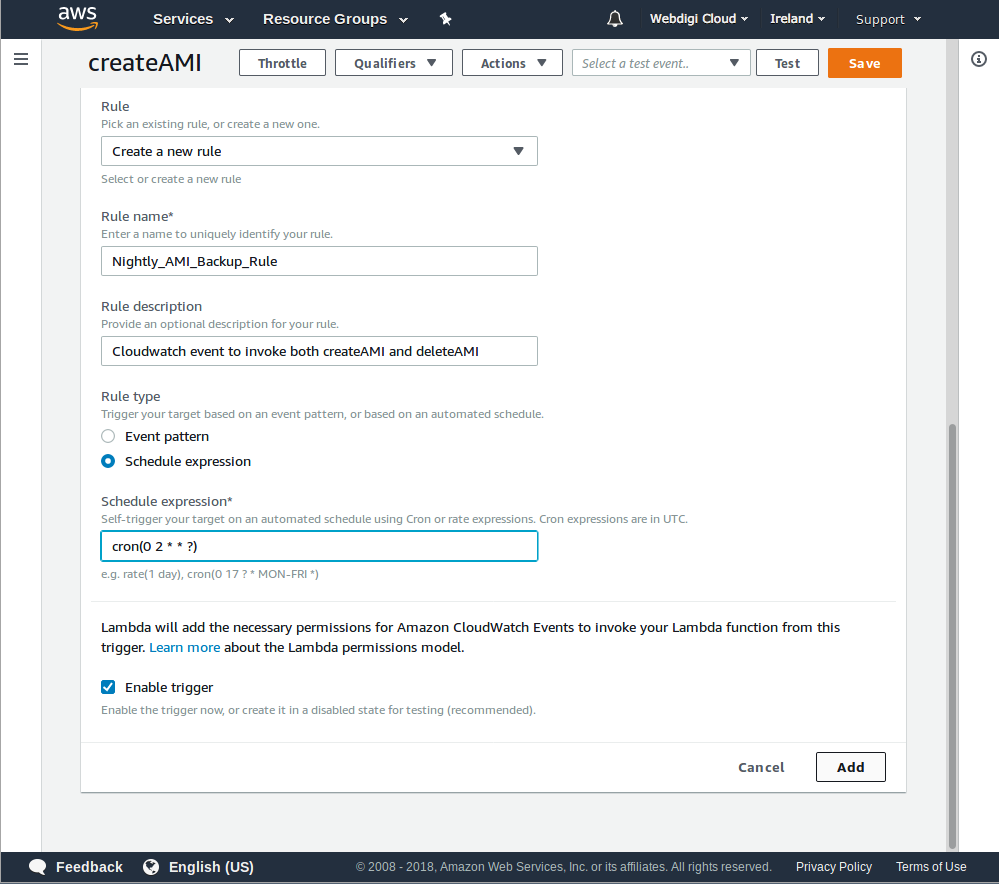


1. Add a trigger by selecting *CloudWatch Event* and *Create a new rule*.



1. Configure the Rule:

* Provide a rule name and description.
  + Rule Name: **Nightly\_AMI\_Backup\_Rule**
  + Rule Description: **Cloudwatch event to invoke both createAMI and deleteAMI**
* Set the event pattern or schedule.
  + The cron expression, ‘**cron(0 15 \* \* ? \*)**’, will run at 11 PM (GMT+8) every day.
* Finish by clicking *Add* at the bottom.



1. Repeat the basic steps as above to create a second function named **deleteAMI**, but this time:

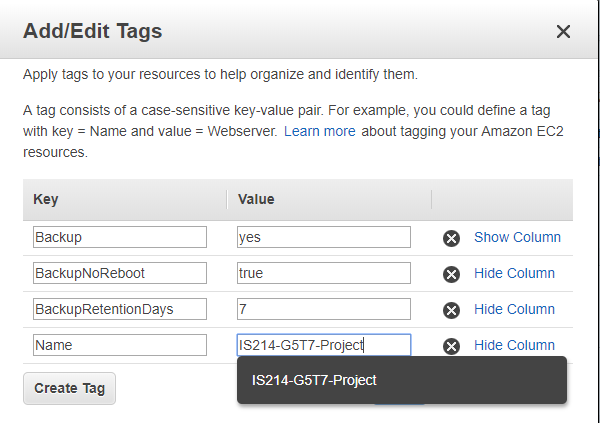
* Use **deleteAMI.js**
* Use the existing *Lambda Role*.
* Use the existing *CloudWatch Rule*.

# **Task B: Setting tags for EC2 instances**

Set the tags on the instances you want backed up.

* **Backup**: yes
* **BackupRetentionDays**: 7
* **BackupNoReboot**: true

**Note**: BackupRetentionDays must be of a positive integer. In this case, we will be using 7. For BackupNoReboot, the tag is optional but a reboot will happen unless the tag is present and value is true.



# 