Automated Backup of AWS Servers using Lambda functions

Table of Contents

Overview

There are two kinds of backups with respect to Elastic Cloud Compute (EC2) servers:

- Amazon Machine Image (AMI): This is the image of the current server with respect to the Operating system.
- EBS Snapshot: Elastic Block Storage (EBS) volumes are similar to hard drives in a PC which is externally attached to the EC2 instance. Point in time snapshots can be taken for these EBS volumes.

These backups can be taken via the AWS Console or can be automated using various methods such as AWS CLI, AWS SDK, etc.

Overview of proposed system

The proposed system to take the AMI as well as EBS backup is through the use of an AWS service called Lambda.

Proposed system details

Lambda functions

Function Name	**Description**
ami-image-lambda	This lambda function is responsible to take AMI of the instance that is tagged for AMI backup.
ebs-snapshot-lambda	This lambda function is responsible to take snapshots of all the EBS volumes except the root volume(/dev/sda1) attached to the instance tagged for EBS snapshot
store-logs-lambda	This lambda function is responsible to take the CloudWatch logs generated by the above mentioned 2 lambda functions and store them in S3 bucket and also generate a CSV report from the logs.
backup-initiator-lambda	This lambda function will trigger ami-image-lambda and ebs-snapshot-lambda functions and if they are successful then trigger the store-logs-lambda function.
report-mailer-lambda	This lambda function will send email with the generated report as attachment to the registered email addresses.

IAM Roles

1) backup-worker-role

Create a new “service role” for AWS Lambda from Management Console

Build a new IAM policy as follows:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "logs:*"
            ],
            "Resource": "arn:aws:logs:*:*:*"
        },
        {
            "Effect": "Allow",
            "Action": "ec2:Describe*",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateImage",
                "ec2:DeregisterImage",
                "ec2:CreateSnapshot",
                "ec2:DeleteSnapshot",
                "ec2:CreateTags",
                "ec2:ModifySnapshotAttribute",
                "ec2:ResetSnapshotAttribute",
                "iam:Get*"
            ],
            "Resource": [
                "*"
            ]
        }
```

```
    ]
}
```

Attach this role to **ebs-snapshot-lambda** and **ami-image-lambda** function at the time of creation.

2) Cloudwatch-log-role

Create a new “service role” for AWS Lambda from Management Console AmazonS3FullAccess policy is attached to this role.

Custom IAM policy attached to this role:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "logs:CreateLogGroup",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogStream",
                "logs:PutLogEvents",
                "logs:GetLogEvents",
                "logs:DescribeLogStreams",
                "logs:DeleteLogStream"
            ],
            "Resource": [
                "*"
            ]
        }
    ]
}
```

This role is then attached to a new lambda function **store-logs-lambda**, used to move CloudWatch logs to S3.

3) Report-mailer-role

Create a new “service role” for AWS Lambda from Management Console

Build a new IAM policy as follows:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogGroup",
                "logs:CreateLogStream",
                "logs:PutLogEvents"
            ],
            "Resource": "arn:aws:logs:*:*:*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "cloudwatch:PutMetricData"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ses:SendEmail",
                "ses:SendRawEmail"
            ],
            "Resource": "*"
```

```
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:*"
            ],
            "Resource": "*"
        }
    ]
}
```

Attach this role to **report-mailer-lambda** function at the time of creation.

4) Backup-initiator-role

Create a new “service role” for AWS Lambda from Management Console

AWSLambdaFullAccess policy is attached to this role.

Backup Scheduling

The EC2 backup can be scheduled by scheduling a cron job which will run at the specified time of the day.

backup-initiator-lambda function is scheduled to run as a cron job.

Storing logs to S3 and generating report

Following S3 buckets are created to store the logs:

Bucket Name **:** ami-image-cw-logs
Description **:** This bucket is used to store the CloudWatch logs generated from the ami-image-lambda function.

Bucket Name **:** ebs-snapshot -cw-logs
Description **:** This bucket is used to store the CloudWatch logs generated from the ebs-snapshot-lambda function.

Bucket Name : backup-reports
Description : This bucket is used to store the CSV report generated from store-logs-lambda function.

store-logs-lambda function moves the logs from CloudWatch to ami-image-cw-logs and ebs-snapshot -cw-logs buckets and then generate a CSV report and store it to backup-reports.

Sending report via email

Sending of report via email is controlled by another lambda function "report-mailer-lambda". As soon as the report is generated and PUT in desired S3 bucket, the lambda function runs and sends the email with the recently put object as attachment.

Registering Sender as well as recipient with SES

Simple Email Service is an AWS service provided to send emails. To register and verify email addresses with SES, you can refer http://docs.aws.amazon.com/ses/latest/DeveloperGuide/verify-email-addresses.html .

The **report-mailer-lambda** function should be modified accordingly for sender and receiver email IDs.

Tagging the instance

The backup scripts work using the associated tags to the EC2 instance. The tag details are mentioned below:

Tag Name	Possible Values	Description
AMI_Backup	Yes,yes	This tag is to find the instances which needs backup of AMI
AMI_Retention	Positive integer Values(e.g. 4, 7 ,30)	Number of days the AMI image needs to be retained
EBS_Backup	Yes,yes	This tag is to find the instances with EBS volumes attached apart from root volume and needs snapshot of those EBS volumes.
EBS_Snapshot_Retention	Positive integer Values(e.g. 4, 7 ,30)	Number of days the EBS snapshot needs to be retained