Here, we are going to monitor the AWS resources that are currently running for an organization.

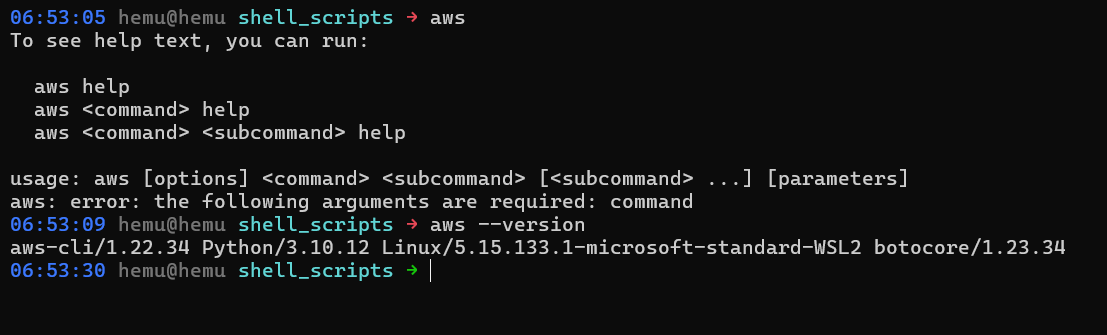
Resources that are under/ not utilized will be shutdown to reduce costs.

**Resources monitored:**

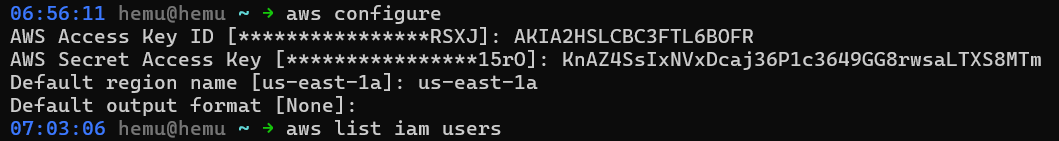
* EC2 instances
* S3 bucket
* Lambda function
* IAM

Everyday at a specific time(6 PM ), this resource utilization report file is sent to reporting dashboard

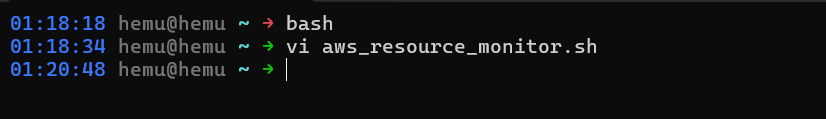
* Since, every day at 6pm you might not be available or may miss time ... to login to ec2 and generate the report, we can automate this task using CRON job
* CRON Job: Linux process wait for the scheduled time and at the specific time mentioned, a shell script is run (or whatever task is given) will be executed and report is generation automatically thus reducing manual work
* So, to achieve this, we are writing a shell script and integrate it with cron job
* First, we install AWS CLI from <https://docs.aws.amazon.com/cli/latest/userguide/getting-started-install.html>



* Now, we authenticate to our AWS account using AWS configure command.
* Access key and secret access keys can be generated for IAM user (IAM best practices) using AWS console.



* Run bash command, to switch to bash shell if there is any other shell selected by default



* Write the script to track AWS resources

#!/bin/bash

#############################################

#Author: Hemali Jayswal

#Date: 24th March

#script to monitor & report the AWS resources

#Resources monitored: AWS s3, AWS EC2, AWS Lambda, AWS IAM users

#############################################

#set -x

echo "list iam users: "

aws iam list-users | jq '.Users[].UserId'

echo "list ec2 instances: "

aws ec2 describe-instances | jq '.Reservations[].Instances[].InstanceId'

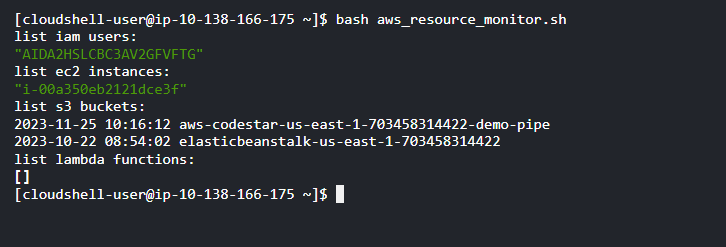
echo "list s3 buckets: "

aws s3 ls

echo "list lambda functions: "

aws lambda list-functions | jq '.Functions'

Output:



Scheduling a CRON job that automatically generates this report

1. Give execute permission to shell script

chmod +x /path/to/your/script.sh

1. Test the script manually whether its working fine without any errors
2. Run crontab -e and open a Nano/ vim text editor to edit crontab
3. Add a new line defining the schedule and the script

minute hour day\_of\_month month day\_of\_week command

use \* for all selecting all values

1. Save (ctrl + O and ENTER) and exit (ctrl + x)

Note: To output the results to a file use below command:

0 \* \* \* \* /path/to/your/script.sh >> /var/log/script.log

**Why can’t we use grep in place of jq?**

jq is json processor so we get the exact intended key by traversing json. But if there are multiple keys at different level of json with same name then grep will give us all the results and it is difficult to understand which key we are actually looking for.