Essential AWS CLI Commands: Requirements, Descriptions, and Real-World Use Cases

Introduction

In the world of cloud computing, Amazon Web Services (AWS) has become a dominant platform, offering a wide range of services and tools for developers, businesses, and IT professionals. The AWS Command Line Interface (CLI) is an essential tool that allows users to manage their AWS services and resources from a command line. This document aims to provide a comprehensive guide to some of the most commonly used and highly important AWS CLI commands. Each command will be accompanied by its requirements, a description, and a real-world use case to illustrate its application.

AWS CLI Commands

1. Setting Up AWS CLI

Requirement: AWS CLI installed and configured with AWS credentials.

Command:

```
Sh: aws configure
```

Description: This command configures the AWS CLI with your AWS access key, secret key, region, and output format.

Use Case: Setting up the AWS CLI for the first time to enable subsequent AWS service management via the command line.

---

2. EC2 (Elastic Compute Cloud) Commands

Requirement: Proper IAM permissions to manage EC2 instances.

- **Describe Instances**
  Command:
  ```
  Sh: aws ec2 describe-instances
  ```

  Description: Lists all EC2 instances in your account.

  Use Case: Getting an overview of all running and stopped instances for inventory management or troubleshooting.

- **Launch a New EC2 Instance**
  Command:
  ```
  Sh:
  aws ec2 run-instances --image-id ami-0abcdef1234567890 --count 1 --
  instance-type t2.micro --key-name MyKeyPair --security-groups
  MySecurityGroup
  ```

  Description: Launches a new EC2 instance with specified parameters.

  Use Case: Deploying a new application or server by launching a new EC2 instance.

- **Stop an EC2 Instance**
  Command:
  ```
  Sh: aws ec2 stop-instances --instance-ids i-1234567890abcdef0
  ```

  Description: Stops a running EC2 instance.

  Use Case: Stopping a non-essential instance to save on costs.

---

3. S3 (Simple Storage Service) Commands

Requirement: Proper IAM permissions to manage S3 buckets.

- **List Buckets**
  Command:
  ```
  Sh: aws s3 ls
  ```

  Description: Lists all S3 buckets in your account.

  Use Case: Viewing all S3 buckets to manage data storage and organization.

- **Create a New Bucket**
  Command:
  ```
  Sh: aws s3 mb s3://my-new-bucket
  ```

  Description: Creates a new S3 bucket.

  Use Case: Setting up a new storage bucket for an application or data backup.

- **Upload a File to S3**
  Command:
  ```
  Sh: aws s3 cp myfile.txt s3://my-new-bucket/
  ```

  Description: Uploads a file to a specified S3 bucket.

  Use Case: Backing up files or uploading application assets to S3 for distribution.

---

4. IAM (Identity and Access Management) Commands

Requirement: Proper IAM permissions to manage IAM resources.

- **Create a New User**
  Command:
  ```
  Sh: aws iam create-user --user-name my-new-user
  ```

  Description: Creates a new IAM user.

  Use Case: Adding a new user to your AWS account for managing access and permissions.

- **Attach Policy to User**
  Command:
  ```
  Sh:
  aws iam attach-user-policy --user-name my-new-user --policy-arn
  arn:aws:iam::aws:policy/AmazonS3FullAccess
  ```

  Description: Attaches a specified IAM policy to a user.

  Use Case: Granting specific permissions to a new user, such as full access to S3.

---

5. CloudFormation Commands

Requirement: Proper IAM permissions to manage CloudFormation stacks.

- **Create a Stack**
  Command:
  Sh:
  ```
  aws cloudformation create-stack --stack-name my-stack --template-body file://template.json
  ```

  Description: Creates a CloudFormation stack from a specified template.
  Use Case: Deploying infrastructure as code by creating resources defined in a CloudFormation template.
- **Delete a Stack**
  Command:
  Sh: `aws cloudformation delete-stack --stack-name my-stack`

  Description: Deletes a specified CloudFormation stack.

  Use Case: Removing a set of resources that are no longer needed, cleaning up the environment.

---

6. RDS (Relational Database Service) Commands

Requirement: Proper IAM permissions to manage RDS instances.

- **Describe DB Instances**
  Command:
  Sh: `aws rds describe-db-instances`

  Description: Lists all RDS instances in your account.

  Use Case: Getting an overview of all database instances for monitoring and management.
- **Create a New DB Instance**
  Command:
  Sh:
  ```
  aws rds create-db-instance --db-instance-identifier mydbinstance --allocated-storage 20 --db-instance-class db.t2.micro --engine mysql --master-username admin --master-user-password password
  ```

  Description: Creates a new RDS instance with specified parameters.

  Use Case: Setting up a new database for an application.

7. Lambda Commands

Requirement: Proper IAM permissions to manage Lambda functions.

- **List Functions**
  Command:
  ```
  Sh: aws lambda list-functions
  ```
  Description: Lists all Lambda functions in your account.

  Use Case: Reviewing existing Lambda functions for monitoring or updating.

- **Create a New Function**
  Command:
  ```
  Sh:
  aws lambda create-function --function-name my-function --zip-file
  fileb://function.zip --handler index.handler --runtime nodejs14.x --
  role arn:aws:iam::123456789012:role/lambda-role
  ```

  Description: Creates a new Lambda function from a specified deployment package.

  Use Case: Deploying a serverless function to handle events such as S3 uploads or DynamoDB updates.

8. ECR (Elastic Container Registry) Commands

Requirement: Proper IAM permissions to manage ECR repositories.

- **Create a Repository**
  Command:
  ```
  Sh: aws ecr create-repository --repository-name my-repo
  ```

  Description: Creates a new ECR repository.

  Use Case: Setting up a new container repository to store Docker images for your applications.
- **List Repositories**
  Command:
  ```
  Sh: aws ecr describe-repositories
  ```

  Description: Lists all ECR repositories in your account.

  Use Case: Reviewing available repositories for container image management.

- **Push an Image to ECR**
  Command:
  Sh:
  ```
  # First, authenticate Docker to your Amazon ECR registry.
  aws ecr get-login-password --region region | docker login --username
  AWS --password-stdin aws_account_id.dkr.ecr.region.amazonaws.com

  # Then, push your Docker image.
  docker tag my-image:latest
  aws_account_id.dkr.ecr.region.amazonaws.com/my-repo:latest
  docker push aws_account_id.dkr.ecr.region.amazonaws.com/my-
  repo:latest
  ```

  Description: Authenticates Docker to your Amazon ECR registry and pushes a Docker image to the specified repository.

  Use Case: Storing Docker images in ECR for deployment to ECS or EKS.

---

9. EKS (Elastic Kubernetes Service) Commands

Requirement: Proper IAM permissions to manage EKS clusters.

- **Create a Cluster**
  Command:
  Sh:
  ```
  aws eks create-cluster --name my-cluster --role-arn
  arn:aws:iam::123456789012:role/eksClusterRole --resources-vpc-config
  subnetIds=subnet-abcde0123,subnet-bcdef1234,subnet-cdefg2345
  ```

  Description: Creates a new EKS cluster.

  Use Case: Setting up a new Kubernetes cluster for deploying containerized applications.

- **Update Kubeconfig**
  Command:
  Sh: `aws eks update-kubeconfig --name my-cluster`

  Description: Updates the kubeconfig file with the new cluster information.

  Use Case: Configuring `kubectl` to interact with your EKS cluster.

- **List Clusters**
  Command:
  ```
  Sh: aws eks list-clusters
  ```

  Description: Lists all EKS clusters in your account.

  Use Case: Getting an overview of all Kubernetes clusters managed by EKS.

---

10. Route 53 Commands

Requirement: Proper IAM permissions to manage Route 53 resources.

- **List Hosted Zones**
  Command:
  ```
  Sh: aws route53 list-hosted-zones
  ```

  Description: Lists all hosted zones in your account.

  Use Case: Reviewing DNS configurations and hosted zones for your domains.

- **Create a Hosted Zone**
  Command:
  ```
  Sh:
  aws route53 create-hosted-zone --name example.com --caller-reference
  unique-string
  ```

  Description: Creates a new hosted zone for a domain.

  Use Case: Setting up DNS management for a new domain.

- **Create a Record Set**
  Command:
  ```
  Sh:
  aws route53 change-resource-record-sets --hosted-zone-id
  Z3M3LMPEXAMPLE --change-batch file://change-batch.json
  ```

  Description: Creates a new DNS record set within a hosted zone.

  Use Case: Adding DNS records for routing traffic to your applications.

---

11. CloudWatch Commands

Requirement: Proper IAM permissions to manage CloudWatch resources.

- **List Metrics**
  Command:
  Sh:
  ```
  aws cloudwatch list-metrics
  ```

  Description: Lists available CloudWatch metrics.

  Use Case: Identifying metrics for monitoring and alerting.

- **Create an Alarm**
  Command:
  Sh:
  ```
  aws cloudwatch put-metric-alarm --alarm-name my-alarm --metric-name
  CPUUtilization --namespace AWS/EC2 --statistic Average --period 300 -
  -threshold 70 --comparison-operator GreaterThanThreshold --
  evaluation-periods 2 --alarm-actions arn:aws:sns:us-east-
  1:123456789012:my-sns-topic
  ```

  Description: Creates a CloudWatch alarm that triggers when the specified metric exceeds a threshold.

  Use Case: Monitoring EC2 instance CPU utilization and triggering notifications when usage is high.

- **Get Metric Statistics**
  Command:
  Sh:
  ```
  aws cloudwatch get-metric-statistics --namespace AWS/EC2 --metric-
  name CPUUtilization --start-time 2023-01-01T00:00:00Z --end-time
  2023-01-01T23:59:59Z --period 3600 --statistics Average
  ```

  Description: Retrieves statistics for a specified metric.

  Use Case: Analyzing historical metric data for performance tuning and capacity planning.

---

12. CloudFront (CDN) Commands

Requirement: Proper IAM permissions to manage CloudFront distributions.

- **List Distributions**
  Command:
  ```
  Sh: aws cloudfront list-distributions
  ```

  Description: Lists all CloudFront distributions in your account.

  Use Case: Reviewing distributions to manage content delivery settings.

- **Create a Distribution**
  Command:
  ```
  Sh:
  aws cloudfront create-distribution --origin-domain-name example-bucket.s3.amazonaws.com
  ```

  Description: Creates a new CloudFront distribution with the specified origin.

  Use Case: Setting up a new CDN distribution for a website or application.

- **Invalidate Cache**
  Command:
  ```
  Sh:
  aws cloudfront create-invalidation --distribution-id E1234567890 --paths "/images/*"
  ```

  Description: Creates an invalidation to clear cached content.

  Use Case: Updating cached assets on the CDN to ensure users get the latest content.

Conclusion

The AWS CLI is a powerful tool that provides granular control over AWS services and resources. By mastering these commonly used commands, IT professionals and developers can efficiently manage their AWS environments, automate workflows, and enhance productivity. This document serves as a quick reference guide to some of the most critical AWS CLI commands, complete with their requirements, descriptions, and practical use cases.