DAY-I

An Introduction to

amazon web services™

By P.S.R.Patnaik

Microsoft Specialist: Architecting Microsoft Azure Solutions
Microsoft, License F312-3640,
AWS Certified Solutions Architect - Associate
Amazon Web Services, License 639BWPV1JFRE11CR

AWS Account

AWS Account

Basics of Cloud Computing

The AWS Cloud provides a broad set of infrastructure services, such as computing power, storage options, networking, and databases that are delivered as a utility: on-demand, available in seconds, with pay-as-you-go pricing. From data warehousing to deployment tools, directories to content delivery, over 90 AWS services are available. This allows enterprises, start-ups, small and medium-sized businesses, and customers in the public sector to access the building blocks they need to respond quickly to changing business requirements.

AWS Account

Basics of Cloud Computing . . .

What is Cloud Computing?

amazon web services™

In 2006, Amazon Web Services (AWS) began offering IT infrastructure services to businesses as web services, now commonly known as **cloud computing**. One of the key benefits of cloud computing is the opportunity to replace upfront capital infrastructure expenses with low variable costs that scale with your business.

Today, AWS provides a highly reliable, scalable, low-cost infrastructure platform in the cloud that powers hundreds of thousands of businesses in 190 countries around the world.

AWS Account

Basics of Cloud Computing . . .

Six Advantages of Cloud Computing

- Trade capital expense for variable expense – Instead of having to invest heavily in data centers and servers before you know how you’re going to use them, you can pay only when you consume computing resources, and pay only for how much you consume.

AWS Account

Basics of Cloud Computing . . .

Six Advantages of Cloud Computing

- Benefit from massive economies of scale – By using cloud computing, you can achieve a lower variable cost than you can get on your own. Because usage from hundreds of thousands of customers is aggregated in the cloud, providers such as AWS can achieve higher economies of scale, which translates into lower pay as-you-go prices.

AWS Account

Basics of Cloud Computing . . .

Six Advantages of Cloud Computing

- Stop guessing about capacity – Eliminate guessing on your infrastructure capacity needs. When you make a capacity decision prior to deploying an application, you often end up either sitting on expensive idle resources or dealing with limited capacity. With cloud computing, these problems go away. You can access as much or as little capacity as you need, and scale up and down as required with only a few minutes' notice.

AWS Account

Basics of Cloud Computing . . .

Six Advantages of Cloud Computing

- Increase speed and agility – In a cloud computing environment, new IT resources are only a click away, which means that you reduce the time to make those resources available to your developers from weeks to just minutes. This results in a dramatic increase in agility for the organization, since the cost and time it takes to experiment and develop is significantly lower.

AWS Account

Basics of Cloud Computing . . .

Six Advantages of Cloud Computing

- Stop spending money running and maintaining data centers – Focus on projects that differentiate your business, not the infrastructure. Cloud computing lets you focus on your own customers, rather than on the heavy lifting of racking, stacking, and powering servers.

AWS Account

Basics of Cloud Computing . . .

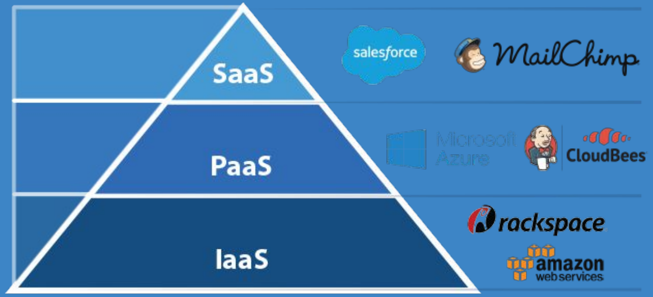
Six Advantages of Cloud Computing

- Go global in minutes – Easily deploy your application in multiple regions around the world with just a few clicks. This means you can provide lower latency and a better experience for your customers at minimal cost.

AWS Account

Basics of Cloud Computing . . .

Types of Cloud Computing

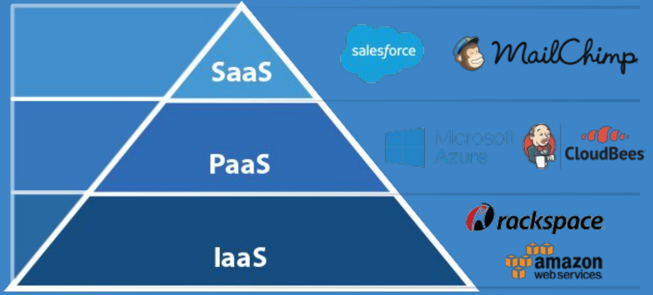


As cloud computing has grown in popularity, several different models and deployment strategies have emerged to help meet specific needs of different users. Each type of cloud service and deployment method provides you with different levels of control, flexibility, and management. Understanding the differences between **Infrastructure as a Service, Platform as a Service, and Software as a Service**, as well as what deployment strategies you can use, can help you decide what set of services is right for your needs.

AWS Account

Basics of Cloud Computing . . .

Cloud Computing Models - IaaS

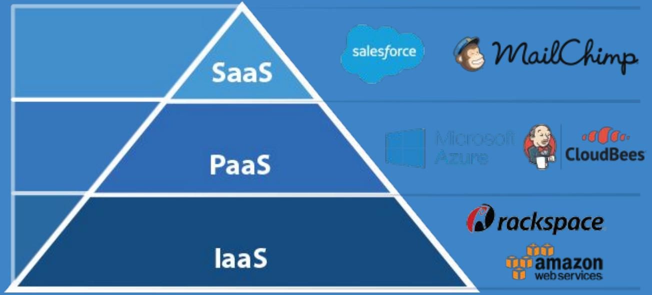


Infrastructure as a Service (IaaS) contains the basic building blocks for cloud IT and typically provide access to networking features, computers (virtual or on dedicated hardware), and data storage space. IaaS provides you with the highest level of flexibility and management control over your IT resources and is most similar to existing IT resources that many IT departments and developers are familiar with today.

AWS Account

Basics of Cloud Computing . . .

Cloud Computing Models - PaaS

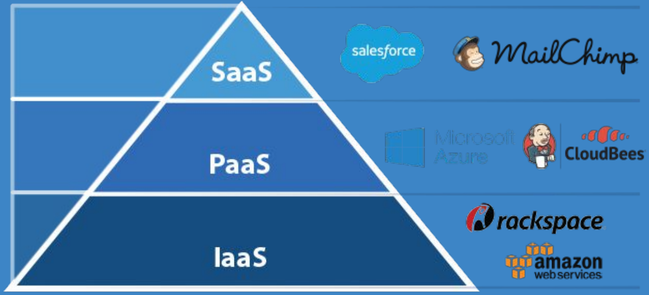


Platform as a Service (PaaS) removes the need for your organization to manage the underlying infrastructure (usually hardware and operating systems) and allows you to focus on the deployment and management of your applications. This helps you be more efficient as you don't need to worry about resource procurement, capacity planning, software maintenance, patching, or any of the other undifferentiated heavy lifting involved in running your application.

AWS Account

Basics of Cloud Computing . . .

Cloud Computing Models - SaaS



Software as a Service (SaaS) provides you with a completed product that is run and managed by the service provider. In most cases, people referring to Software as a Service are referring to end-user applications. With a SaaS offering you do not have to think about how the service is maintained or how the underlying infrastructure is managed; you only need to think about how you will use that particular piece of software. A common example of a SaaS application is web-based email.

AWS Account

Basics of Cloud Computing . . .

Cloud Computing Deployment Models - Cloud

A cloud-based application is fully deployed in the cloud and all parts of the application run in the cloud. Applications in the cloud have either been created in the cloud or have been migrated from an existing infrastructure to take advantage of the benefits of cloud computing. Cloud-based applications can be built on low-level infrastructure pieces or can use higher level services that provide abstraction from the management, architecting, and scaling requirements of core infrastructure.

AWS Account

Basics of Cloud Computing . . .

Cloud Computing Deployment Models - Hybrid

A hybrid deployment is a way to connect infrastructure and applications between cloud-based resources and existing resources that are not located in the cloud. The most common method of hybrid deployment is between the cloud and existing on-premises infrastructure to extend, and grow, an organization's infrastructure into the cloud while connecting cloud resources to the internal system.

AWS Account

Basics of Cloud Computing . . .

Cloud Computing Deployment Models - On premises

The deployment of resources on-premises, using virtualization and resource management tools, is sometimes called the “private cloud.” On-premises deployment doesn’t provide many of the benefits of cloud computing but is sometimes sought for its ability to provide dedicated resources. In most cases this deployment model is the same as legacy IT infrastructure while using application management and virtualization technologies to try and increase resource utilization.

AWS Account

Basics of Cloud Computing . . .

Security and Compliance

Cloud security at AWS is the highest priority. As an AWS customer, you will benefit from a data center and network architecture built to meet the requirements of the most security-sensitive organizations. Security in the cloud is much like security in your on-premises data centers—only without the costs of maintaining facilities and hardware. In the cloud, you don’t have to manage physical servers or storage devices. Instead, you use software-based security tools to monitor and protect the flow of information into and of out of your cloud resources.

AWS Account

Basics of Cloud Computing . . .

Benefits of AWS Security

- Keep Your Data Safe: The AWS infrastructure puts strong safeguards in place to help protect your privacy. All data is stored in highly secure AWS data centers.
- Meet Compliance Requirements: AWS manages dozens of compliance programs in its infrastructure. This means that segments of your compliance have already been completed.

AWS Account

Basics of Cloud Computing . . .

Benefits of AWS Security

- Save Money: Cut costs by using AWS data centers. Maintain the highest standard of security without having to manage your own facility
- Scale Quickly: Security scales with your AWS Cloud usage. No matter the size of your business, the AWS infrastructure is designed to keep your data safe.

AWS Account

Basics of Cloud Computing . . .

Compliance

The IT infrastructure that AWS provides to its customers is designed and managed in alignment with best security practices and a variety of IT security standards. The following is a partial list of assurance programs with which AWS complies:

- SOC 1/ISAE 3402, SOC 2, SOC 3
- FISMA, DIACAP, and FedRAMP
- PCI DSS Level 1
- ISO 9001, ISO 27001, ISO 27018

AWS Account

Introduction to AWS . . .

Amazon Web Services Cloud Platform

AWS consists of many cloud services that you can use in combinations tailored to your business or organizational needs. This section introduces the major AWS services by category. To access the services, you can use the AWS Management Console, the Command Line Interface, or Software Development Kits (SDKs).

AWS Account

Introduction to AWS . . .

Amazon Web Services Cloud Platform . . .

AWS Management Console

Access and manage Amazon Web Services through the AWS Management Console, a simple and intuitive user interface. You can also use the AWS Console Mobile App to quickly view resources on the go.

AWS Account

Introduction to AWS . . .

Amazon Web Services Cloud Platform . . .

AWS Command Line Interface

The AWS Command Line Interface (CLI) is a unified tool to manage your AWS services. With just one tool to download and configure, you can control multiple AWS services from the command line and automate them through scripts.

AWS Account

Introduction to AWS . . .

Amazon Web Services Cloud Platform . . .

Software Development Kits

Software Development Kits (SDKs) simplify using AWS services in your applications with an Application Program Interface (API) tailored to your programming language or platform.

AWS Account
Global Infrastructure

AWS serves over a million active customers in more than 190 countries. The AWS Cloud infrastructure is built around **AWS Regions and Availability Zones**. An AWS Region is a physical location in the world where we have multiple Availability Zones. Availability Zones consist of one or more discrete data centers, each with redundant power, networking, and connectivity, housed in separate facilities. The **AWS Cloud operates 42 Availability Zones within 16 geographic Regions around the world.**

AWS Account

Introduction to AWS . . .

Amazon Web Services Walkthrough . . .

Ref. Link

Services

- Compute Services
- Storage
- Database
- Migration
- Networking and Content Delivery
- Developer Tools
- Management Tools
- Security, Identity, and Compliance
- Analytics
- Artificial Intelligence
- Mobile Services
- Application Services
- Messaging
- Business Productivity
- Desktop & App Streaming
- Internet of Things (IoT)
- Game Development

AWS Account

AWS Benchmarks

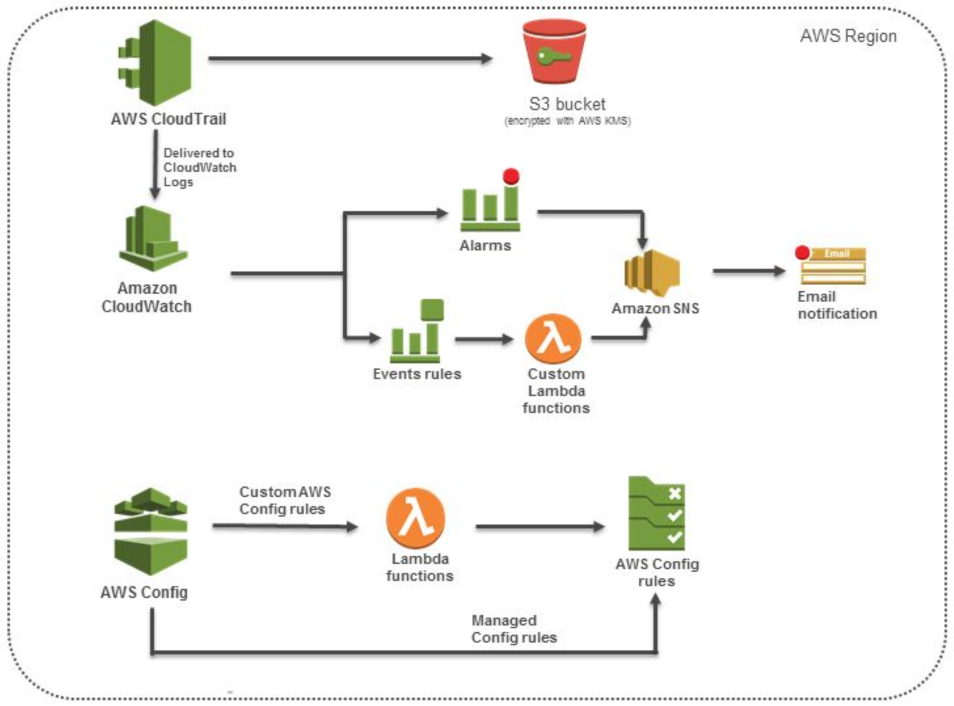
Ref. Link

This Quick Start deploys and configures a standardized architecture for the Center for Internet Security (CIS) AWS Foundations Benchmark. CIS Benchmarks are consensus-based configuration guidelines developed by experts in US government, business, industry, and academia to help organizations assess and improve security. The Quick Start supports the benchmark by creating AWS Config rules, Amazon CloudWatch alarms, and CloudWatch Events rules in your AWS account.

AWS Account
AWS Benchmarks

Ref. Link

- AWS Config rules
- CloudWatch alarms
- CloudWatch Events
- Lambda functions
- AWS CloudTrail
- AWS Config



AWS Account

AWS Benchmarks

Ref. Link

You can build your standardized CSI Benchmark environment by following these steps:

1. Sign in to your AWS account at https://aws.amazon.com.
2. Launch the Quick Start. The deployment takes about 10 minutes. You can choose from two options:
   - Deploy the Quick Start into the AWS GovCloud (US) Region
   - Deploy the Quick Start into another AWS Region
3. Confirm your subscription to email notifications for security configuration changes. Test your deployment by viewing the resources that were created by the Quick Start.

AWS Account

AWS Access

Ref. Link

Following are the steps to access AWS services

1. Create an AWS account.
2. Sign-up for AWS services.
3. Create your password and access your account credentials.
4. Activate your services in credits section.

AWS Account

AWS Account Creation

Ref. Link

Create an AWS account

1. Navigate to AWS.
2. Enter account information.
3. Enter contact information.
4. Enter payment information.
5. Verify your identity.
6. Select a support plan and finish account creation.

AWS Account

AWS Management Console

Ref. Link

After you create an instance with the cPanel & WHM Amazon Machine Image (AMI), you can manage that instance from within the Amazon Web Services (AWS) Management Console.

1. Create and add key pairs
2. Access your instance for the first time
3. Start, stop, reboot, or terminate instances.

AWS Account

Basic Account Management Settings

Ref. Link

- Manage multiple AWS accounts for billing purposes
- Manage multiple AWS accounts for security purposes
- Tag AWS resources
- Receive notifications on approaching AWS service limits
- Most cost-effective instances usage
- Monitor and analyze estimated AWS costs
- Manage Amazon WorkSpaces billing models to optimize costs
- Monitor my AWS account activity in real-time

AWS Account

Introduction to Billing Dashboard & Cost Explorer

Ref. Link

AWS Cost Explorer lets you dive deeper into your cost and usage data to identify trends, pinpoint cost drivers, and detect anomalies.

1. Get started quickly
2. Set time interval and granularity
3. Filter/Group your data
4. Forecast future costs and usage
5. Save your progress
6. Access data programmatically

AWS Account

Setting up Billing Alarm & Budget

Ref. Link

You can monitor your estimated AWS charges using Amazon CloudWatch. When you enable the monitoring of estimated charges for your AWS account, the estimated charges are calculated and sent several times daily to CloudWatch as metric data.

To enable the monitoring of estimated charges

1. Open the Billing and Cost Management console at https://console.aws.amazon.com/billing/home?#.
2. In the navigation pane, choose Preferences.
3. Choose Receive Billing Alerts.
4. Choose Save preferences

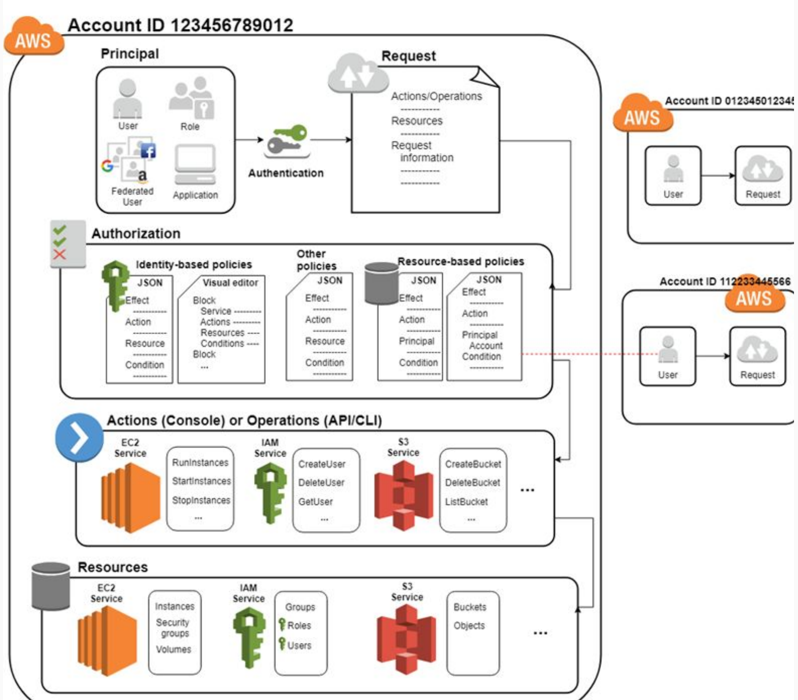
AWS-IAM

AWS IAM

Basics of Authentication & Authorisations

Ref. Link

IAM provides the infrastructure necessary to control **authentication and authorization** for your account.



AWS IAM

Introduction to Identity & Access Management

Ref. Link

For greater security and organization, you can give access to your AWS account to specific users—identities that you create with custom permissions. You can further simplify access for those users by federating existing identities into AWS.

- First-Time Access Only: Your Root User Credentials
- IAM Users
- Federating Existing Users

AWS IAM

Basics of Authentication & Authorisations

Ref. Link

The IAM infrastructure includes the following elements:

- Principal
- Request
- Authentication
- Authorization
- Actions or Operations
- Resources

AWS IAM

Creating and Managing Users & Groups

Ref. Link

As a best practice, do not use the AWS account root user for any task where it's not required. Instead, create a new IAM user for each person that requires administrator access. Then make those users administrators by placing the users into an "Administrators" group to which you attach the Administrator Access managed policy.

- Creating an Administrator IAM User and Group (Console)
- Creating an IAM User and Group (AWS CLI)

AWS IAM

Creating and Managing IAM Policies

Ref. Link

IAM gives you the tools to create and manage all types of IAM policies (managed policies and inline policies).

- Creating IAM Policies
- Validating JSON Policies
- Testing IAM Policies with the IAM Policy Simulator
- Adding and Removing IAM Policies
- Versioning IAM Policies
- Editing IAM Policies
- Deleting IAM Policies
- Reducing Policy Scope by Viewing User Activity

AWS IAM

Roles and its use cases

Ref. Link

Service	Access level	This policy provides:
IAM	Full access	Access to all actions within the IAM service
CloudWatch	**Full:** List	Access to all CloudWatch actions in the List access level, but no access to actions with the Read, Write, or Permissions management access level classification
Data Pipeline	**Limited:** List, Read	Access to at least one but not all AWS Data Pipeline actions in the List and Readaccess level, but not the Write or Permissions management actions
EC2	**Full:** List, Read **Limited:** Write	Access to all Amazon EC2 List and Read actions and access to at least one but not all Amazon EC2 Write actions, but no access to actions with the Permissions management access level classification
S3	**Limited:** Read, Write, Permissions management	Access to at least one but not all Amazon S3 Read, Write and Permissions management actions
codedploy	(empty)	Unknown access, because IAM does not recognize this service
API Gateway	None	No access is defined in the policy
CodeBuild	No actions are defined.	No access because no actions are defined for the service. To learn how to understand and troubleshoot this issue, see My Policy Does Not Grant the Expected Permissions

AWS IAM

Multi-Factor Authentication - [MFA]

Ref. Link

IAM users who are configured with multi-factor authentication (MFA) devices must use their MFA devices to sign in to the AWS Management Console. After the user enters the user name and password, AWS checks the user's account to see if MFA is required for that user.

1. Signing in with a Virtual MFA Device
2. Signing in with a U2F Security Key
3. Signing in with a Hardware MFA Device

AWS IAM

Security Features in IAM

Ref. Link

You can use the AWS Security Token Service (AWS STS) to create and provide trusted users with temporary security credentials that can control access to your AWS resources. Temporary security credentials work almost identically to the long-term access key credentials that your IAM users can use

AWS IAM

Best Practices of IAM

Ref. Link

To help secure your AWS resources, follow these recommendations for the AWS Identity and Access Management (IAM) service.

- lock Away Your AWS Account Root User Access Keys
- Create Individual IAM Users
- Use Groups to Assign Permissions to IAM Users
- Use AWS Defined Policies to Assign Permissions Whenever Possible
- Grant Least Privilege
- Use Access Levels to Review IAM Permissions
- Configure a Strong Password Policy for Your Users
- Enable MFA for Privileged Users
- Use Roles for Applications That Run on Amazon EC2 Instances, etc . . .

AWS-CLI

AWS CLI

Introduction to AWS CLI

Ref. Link

The AWS Command Line Interface (CLI) is a unified tool to manage your AWS services. With just one tool to download and configure, you can control multiple AWS services from the command line and automate them through scripts.

The AWS CLI introduces a new set of simple file commands for efficient file transfers to and from Amazon S3.

AWS CLI
Setting-Up AWS CLI on local machine

Ref. Link

The primary distribution method for the AWS CLI on Linux, Windows, and macOS is pip, a package manager for Python that provides an easy way to install, upgrade, and remove Python packages and their dependencies.

Current AWS CLI Version

The AWS CLI is updated frequently with support for new services and commands. To see if you have the latest version, see the releases page on GitHub.

AWS CLI

Creating Users and groups using AWS CLI

Ref. Link

The CLI stores credentials specified with aws configure in a local file named credentials in a folder named .aws in your home directory. Home directory location varies but can be referred to using the environment variables %UserProfile% in Windows and $HOME or ~ (tilde) in Unix-like systems.

For example, the following commands list the contents of the .aws folder: Linux, macOS, or Unix

```
$ ls  ~/.aws
```

AWS CLI

Creating & Managing Policy using AWS CLI

Ref. Link

You can create an IAM policy or an inline policy using the AWS Command Line Interface (AWS CLI).

To create a customer managed policy (AWS CLI)

Use the following command:

- Create-policy

To create an inline policy for a principal entity (group, user or role) (AWS CLI)

Use one of the following commands:

- put-group-policy
- put-role-policy
- Put-user-policy

AWS CLI

Creating and Managing IAM Roles using AWS CLI

Ref. Link

You can create an IAM policy or an inline policy using the AWS API.

To create a customer managed policy (AWS API)

Call the following operation:

- CreatePolicy

To create an inline policy for a principal entity (group, user, or role) (AWS API)

Call one of the following operations:

- PutGroupPolicy
- PutRolePolicy
- PutUserPolicy

AWS CLI

AWS CLI Command Syntax walkthrough

Ref. Link

The AWS CLI uses a multipart structure on the command line. It starts with the base call to aws. The next part specifies a top-level command, which often represents an AWS service supported in the AWS CLI. Each AWS service has additional subcommands that specify the operation to perform. The general CLI options, or the specific parameters for an operation, can be specified on the command line in any order. If an exclusive parameter is specified multiple times, then only the last value applies.

```
$ aws <command> <subcommand> [options and parameters]
```

AWS-S3

AWS S3

Basics of Storage System

Ref. Link

Amazon Simple Storage Service is storage for the Internet. It is designed to make web-scale computing easier for developers.

Amazon S3 has a simple web services interface that you can use to store and retrieve any amount of data, at any time, from anywhere on the web. It gives any developer access to the same highly scalable, reliable, fast, inexpensive data storage infrastructure that Amazon uses to run its own global network of web sites. The service aims to maximize benefits of scale and to pass those benefits on to developers.

AWS S3

Storage Services provided by AWS

Storage Classes

Bucket Policies

AWS Identity and Access Management

Access Control Lists

Versioning

Operations

AWS S3

Difference Between Object storage and Block Storage

Each Amazon S3 object has data, a key, and metadata. Object key (or key name) uniquely identifies the object in a bucket. Object metadata is a set of name-value pairs. You can set object metadata at the time you upload it. After you upload the object, you cannot modify object metadata. The only way to modify object metadata is to make a copy of the object and set the metadata.

- Object Keys
- Object Metadata

AWS S3

Introduction to Simple Storage Service - S3

- Storage Classes
- Bucket Policies
- AWS Identity and Access Management
- Access Control Lists
- Versioning
- Operations

AWS S3

Use Case & Benefits of using S3

Ref. Link

Benefits

- UNMATCHED DURABILITY, AVAILABILITY, & SCALABILITY,
- MOST COMPREHENSIVE SECURITY & COMPLIANCE CAPABILITIES,
- QUERY IN PLACE,
- FLEXIBLE MANAGEMENT,
- MOST SUPPORTED BY PARTNERS, VENDORS, & AWS SERVICES,
- EASY, FLEXIBLE DATA TRANSFER

Use cases

- BACKUP & RECOVERY,
- DATA ARCHIVING,
- DATA LAKES & BIG DATA ANALYTICS,
- HYBRID CLOUD STORAGE,
- CLOUD-NATIVE APPLICATION DATA,
- DISASTER RECOVERY

AWS S3

Components of S3

Ref. Link

You can select from four different storage classes to store your data in Amazon S3: S3 Standard, S3 Standard-IA, S3 One Zone-IA, and Amazon Glacier. You can learn more about each of these storage classes on the Amazon S3 Storage Classes page. Objects may be automatically moved between storage classes using Lifecycle Management Policies.

AWS S3

Important Properties of S3 bucket

Ref. Link

Create a Bucket – Create and name your own bucket in which to store your objects.

Write an Object – Store data by creating or overwriting an object. When you write an object, you specify a unique key in the namespace of your bucket. This is also a good time to specify any access control you want on the object.

Read an Object – Read data back. You can download the data via HTTP or BitTorrent.

AWS S3

Important Properties of S3 bucket

Deleting an Object – Delete some of your data.

Listing Keys – List the keys contained in one of your buckets. You can filter the key list based on a prefix.

AWS S3

Enabling and Managing Versioning on S3 bucket

Ref. Link

Versioning enables you to keep multiple versions of an object in one bucket. This section describes how to enable object versioning on a bucket. To enable or disable versioning on an S3 bucket

- Sign in to the AWS Management Console and open the Amazon S3 console at https://console.aws.amazon.com/s3/.
- In the Bucket name list, choose the name of the bucket that you want to enable versioning for.
- Choose Properties.
- Choose Versioning.
- Choose Enable versioning or Suspend versioning, and then choose Save.

AWS S3

Managing Logging on S3 Bucket

Ref. Link

To enable server access logging for an S3 bucket

- Sign in to the AWS Management Console and open the Amazon S3 console at https://console.aws.amazon.com/s3/.
- In the Bucket name list, choose the name of the bucket that you want to enable server access logging for.
- Choose Properties.
- Choose Server access logging.
- Choose Enable Logging. For Target, choose the name of the bucket that you want to receive the log record objects. The target bucket must be in the same Region as the source bucket.
- Choose Save.

AWS S3

Creating Life-Cycle Management Rule S3

Ref. Link

A lifecycle configuration is a set of rules that define actions that Amazon S3 applies to a group of objects. There are two types of actions:

- **Transition actions :** Define when objects transition to another storage class. For example, you might choose to transition objects to the STANDARD_IA storage class 30 days after you created them, or archive objects to the GLACIER storage class one year after creating them. There are costs associated with the lifecycle transition requests. For pricing information, see Amazon S3 Pricing.
- **Expiration actions** : Define when objects expire. Amazon S3 deletes expired objects on your behalf. The lifecycle expiration costs depend on when you choose to expire objects. For more information, see Configuring Object Expiration.

AWS S3

Hosting a static-website in S3

Ref. Link

You can host a static website on Amazon Simple Storage Service (Amazon S3). On a static website, individual webpages include static content. They might also contain client-side scripts. By contrast, a dynamic website relies on server-side processing, including server-side scripts such as PHP, JSP, or ASP.NET. Amazon S3 does not support server-side scripting. Amazon Web Services (AWS) also has resources for hosting dynamic websites.

AWS S3

Cross-Region replication in S3

Ref. Link

Cross-region replication (CRR) enables automatic, asynchronous copying of objects across buckets in different AWS Regions. Buckets configured for cross-region replication can be owned by the same AWS account or by different accounts. Cross-region replication is enabled with a bucket-level configuration. You add the replication configuration to your source bucket. In the minimum configuration, you provide the following :

- The destination bucket, where you want Amazon S3 to replicate objects
- An AWS IAM role that Amazon S3 can assume to replicate objects on your behalf
- Additional configuration options are available.

AWS S3

Transfer Accelerator in S3

Ref. Link

Amazon S3 Transfer Acceleration enables fast, easy, and secure transfers of files over long distances between your client and an S3 bucket. Transfer Acceleration takes advantage of Amazon CloudFront's globally distributed edge locations. As the data arrives at an edge location, data is routed to Amazon S3 over an optimized network path.

When using Transfer Acceleration, additional data transfer charges may apply.

AWS S3

MFA and Pre-signed URL in S3

Ref. Link

The following example generates a presigned URL that you can give to others so that they can retrieve an object from an S3 bucket.

When you create a presigned URL for your object, you must provide your security credentials, specify a bucket name, an object key, specify the HTTP method (GET to download the object) and expiration date and time. The presigned URLs are valid only for the specified duration.

Anyone who receives the presigned URL can then access the

AWS S3

Security feature of S3-Encryption, Buclet Policy, Permission etc.

Ref. Link

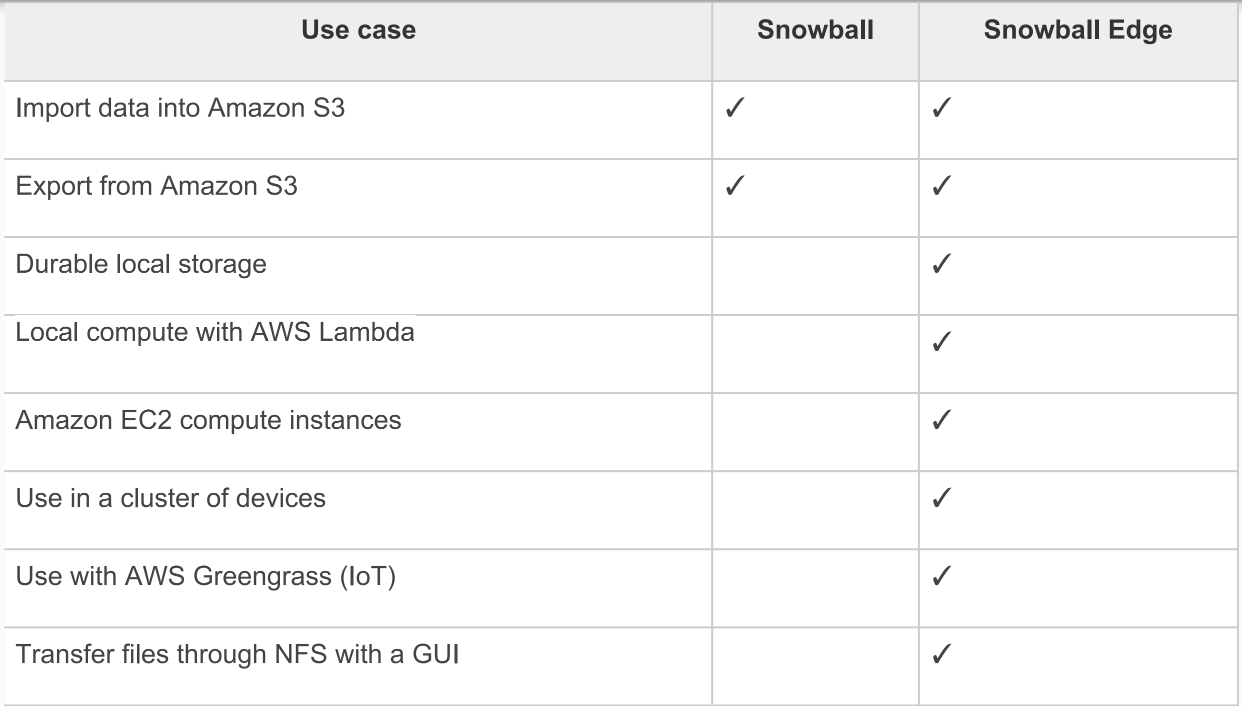
- **Default Encryption** – You can now mandate that all objects in a bucket must be stored in encrypted form without having to construct a bucket policy that rejects objects that are not encrypted.
- **Permission Checks** – The S3 Console now displays a prominent indicator next to each S3 bucket that is publicly accessible.
- **Cross-Region Replication ACL Overwrite** – When you replicate objects across AWS accounts, you can now specify that the object gets a new ACL that gives full access to the destination account.
- **Cross-Region Replication with KMS** – You can now replicate objects that are encrypted with keys that are managed by AWS Key Management Service (KMS).
- **Detailed Inventory Report** – The S3 Inventory report now includes the encryption status of each object. The report itself can also be encrypted.

AWS S3

Snowball - Types & Use Cases

Ref. Link

The Snowball and the Snowball Edge are two different devices.



Use case	Snowball	Snowball Edge
Import data into Amazon S3	✓	✓
Export from Amazon S3	✓	✓
Durable local storage		✓
Local compute with AWS Lambda		✓
Amazon EC2 compute instances		✓
Use in a cluster of devices		✓
Use with AWS Greengrass (IoT)		✓
Transfer files through NFS with a GUI		✓

AWS S3

Managing S3 components using CLI Commands

Ref. Link

Creating Buckets

Use the aws s3 mb command to create a new bucket. Bucket names must be unique and should be DNS compliant. Bucket names can contain lowercase letters, numbers, hyphens and periods. Bucket names can only start and end with a letter or number, and cannot contain a period next to a hyphen or another period.

```
$ aws s3 mb s3://bucket-name
```

Removing Buckets

To remove a bucket, use the aws s3 rb command

```
.$ aws s3 rb s3://bucket-name
```

AWS-EC2

AWS EC2

Basics of Virtual Servers

Ref. Link

Amazon Elastic Compute Cloud (Amazon EC2) provides scalable computing capacity in the Amazon Web Services (AWS) cloud. Using Amazon EC2 eliminates your need to invest in hardware up front, so you can develop and deploy applications faster. You can use Amazon EC2 to launch as many or as few virtual servers as you need, configure security and networking, and manage storage. Amazon EC2 enables you to scale up or down to handle changes in requirements or spikes in popularity, reducing your need to forecast traffic.

AWS EC2

Components of a Virtual Server

You can provision Amazon EC2 resources, such as instances and volumes, directly using Amazon EC2. You can also provision Amazon EC2 resources using other services in AWS. For more information, see the following documentation:

Amazon EC2 Auto Scaling User Guide

AWS CloudFormation User Guide

AWS Elastic Beanstalk Developer Guide

AWS OpsWorks User Guide

AWS EC2

Introduction to Elastic Cloud Compute - EC2

Ref. Link

Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides secure, resizable compute capacity in the cloud. It is designed to make web-scale cloud computing easier for developers.

Amazon EC2's simple web service interface allows you to obtain and configure capacity with minimal friction. It provides you with complete control of your computing resources and lets you run on Amazon's proven computing environment.

AWS EC2

Use cases and important features of EC2

Ref. Link

Instances provide compute power and are the fundamental building blocks. Instances are created by launching an Amazon Machine Image (AMI) on a particular instance type.

Instance Types comprise various combinations of CPU, memory, storage, and networking capacity and give you the flexibility to choose the appropriate mix of resources for your applications.

A vCPU is a virtual Central Processing Unit (CPU). A multicore processor has two or more vCPUs.

AWS EC2

Introduction to AMI - Its Uses

Ref. Link

An Amazon Machine Image (AMI) provides the information required to launch an instance, which is a virtual server in the cloud. You must specify a source AMI when you launch an instance.

An AMI includes the following:

- A template for the root volume for the instance (for example, an operating system, an application server, and applications)
- Launch permissions that control which AWS accounts can use the AMI to launch instances
- A block device mapping that specifies the volumes to attach to the instance when it's launched

AWS EC2

Introduction to Instance and its types

Ref. Link

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instance types comprise varying combinations of CPU, memory, storage, and networking capacity and give you the flexibility to choose the appropriate mix of resources for your applications. Each instance type includes one or more instance sizes, allowing you to scale your resources to the requirements of your target workload.

AWS EC2

Security Groups - Creation & Management

Ref. Link

When you need to set the permissions for an identity in IAM, you must decide whether to use an AWS managed policy, a customer managed policy, or an inline policy. The following sections provide more information about each of the types of identity-based policies and when to use them.

- AWS Managed Policies
- Customer Managed Policies
- Inline Policies
- Choosing Between Managed Policies and Inline Policies
- Deprecated AWS Managed Policies

AWS EC2

Key Pair - Why & How

Ref. Link

Amazon EC2 uses public–key cryptography to encrypt and decrypt login information. Public–key cryptography uses a public key to encrypt a piece of data, such as a password, then the recipient uses the private key to decrypt the data. The public and private keys are known as a key pair.

To log in to your instance, you must create a key pair, specify the name of the key pair when you launch the instance, and provide the private key when you connect to the instance.

AWS EC2

Launching & Connecting to Window Instance

Ref. Link

1. Sign in to the AWS Management Console and open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
2. In the navigation pane, under Instances, choose Instances.
3. Browse to and choose your Windows Server instance in the list.
4. Choose Connect.
5. Choose Get Password.
6. Choose Browse. Browse to and choose the Amazon EC2 instance key pair file associated with the Windows Server Amazon EC2 instance, and then choose Open.
7. Choose Decrypt Password. Make a note of the password that is displayed. You need it in step 10.

AWS EC2

Launching & Connecting to Window Instance . . .

Ref. Link

8. Choose Download Remote Desktop File, and then open the file.
9. If you are prompted to connect even though the publisher of the remote connection can't be identified, proceed.
10. Type the password you noted in step 7, and then proceed. (If your RDP connection client application prompts you for a user name, type Administrator.)
11. If you are prompted to connect even though the identity of the remote computer cannot be verified, proceed.
12. After you are connected, the desktop of the Amazon EC2 instance running Windows Server is displayed.
13. You can now sign out of the running Amazon EC2 instance.

AWS EC2

Launching & Connecting to Linux Instance

Ref. Link

1. (Optional) You can verify the RSA key fingerprint on your running instance by using one of the following commands on your local system (not on the instance).
2. In a command-line shell, change directories to the location of the private key file that you created when you launched the instance.
3. Use the following command to set the permissions of your private key file so that only you can read it.
4. Use the ssh command to connect to the instance.
5. (IPv6 only) Alternatively, you can connect to the instance using its IPv6 address.
6. (Optional) Verify that the fingerprint in the security alert matches the fingerprint that you obtained in step 1.
7. Enter yes.

AWS EC2

Setting up a web server on linux Instance - Hosting a website

Ref. Link

1. Connect to your instance.
2. To ensure that all of your software packages are up to date, perform a quick software update on your instance.
3. Now that your instance is current, you can install the Apache web server, MySQL, and PHP software packages.
4. Start the Apache web server.
5. Use the chkconfig command to configure the Apache web server to start at each system boot.
6. Add a security rule to allow inbound HTTP (port 80) connections to your instance if you have not already done so.
7. Test your web server. In a web browser, type the public DNS address (or the public IP address) of your instance.

AWS EC2

Elastic IP Address

Ref. Link

An Elastic IP address is a static IPv4 address designed for dynamic cloud computing. An Elastic IP address is associated with your AWS account. With an Elastic IP address, you can mask the failure of an instance or software by rapidly remapping the address to another instance in your account.

An Elastic IP address is a public IPv4 address, which is reachable from the internet. If your instance does not have a public IPv4 address, you can associate an Elastic IP address with your instance.

AWS EC2
Placement Group

Ref. Link

You can launch or start instances in a placement group, which determines how instances are placed on underlying hardware. When you create a placement group, you specify one of the following strategies for the group:

Cluster : clusters instances into a low-latency group in a single Availability Zone

Spread : spreads instances across underlying hardware

AWS-EBS

AWS EBS

Introduction to Elastic Block Storage

Ref. Link

Amazon Elastic Block Store (Amazon EBS) provides block level storage volumes for use with EC2 instances. EBS volumes are highly available and reliable storage volumes that can be attached to any running instance that is in the same Availability Zone. EBS volumes that are attached to an EC2 instance are exposed as storage volumes that persist independently from the life of the instance. With Amazon EBS, you pay only for what you use. For more information about Amazon EBS pricing, see the Projecting Costs section of the Amazon Elastic Block Store page.

AWS EBS

Components of EBS

Ref. Link

Amazon Elastic Block Store (Amazon EBS) provides persistent block storage volumes for use with Amazon EC2 instances in the AWS Cloud. Each Amazon EBS volume is automatically replicated within its Availability Zone to protect you from component failure, offering high availability and durability. Amazon EBS volumes offer the consistent and low-latency performance needed to run your workloads. With Amazon EBS, you can scale your usage up or down within minutes all while paying a low price for only what you provision.

AWS EBS

Types of Volumes in EBS

Ref. Link

Amazon EBS provides the following volume types, which differ in performance characteristics and price, so that you can tailor your storage performance and cost to the needs of your applications. The volumes types fall into two categories:

- **SSD-backed volumes** optimized for transactional workloads involving frequent read/write operations with small I/O size, where the dominant performance attribute is IOPS
- **HDD-backed volumes** optimized for large streaming workloads where throughput (measured in MiB/s) is a better performance measure than IOPS

AWS EBS

Creating and Managing EBS Volume

Ref. Link

1. To create an EBS volume using the console
2. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
3. From the navigation bar, select the region in which you would like to create your volume. This choice is important because some Amazon EC2 resources can be shared between regions, while others can't. For more information, see Resource Locations.
4. In the navigation pane, choose ELASTIC BLOCK STORE, Volumes.
5. Choose Create Volume.

AWS EBS

Creating and Managing EBS Volume

Ref. Link

6. For Size (GiB), type the size of the volume.
7. With a Provisioned IOPS SSD volume, for IOPS, type the maximum number of input/output operations per second (IOPS) that the volume should support.
8. For Availability Zone, choose the Availability Zone in which to create the volume. EBS volumes can only be attached to EC2 instances within the same Availability Zone.

AWS EBS

Creating and Managing EBS Volume

Ref. Link

9. (Optional) To create an encrypted volume, select the Encrypted box and choose the master key you want to use when encrypting the volume. You can choose the default master key for your account, or you can choose any customer master key (CMK) that you have previously created using the AWS Key Management Service. Available keys are visible in the Master Key menu, or you can paste the full ARN of any key that you have access to.
10. (Optional) Choose Create additional tags to add tags to the volume. For each tag, provide a tag key and a tag value.
11. Choose Create Volume.

AWS EBS

Creating and Managing Snapshot from EBS Volume

Ref. Link

You can restore an Amazon EBS volume with data from a snapshot stored in Amazon S3. You need to know the ID of the snapshot you want to restore your volume from and you need to have access permissions for the snapshot. For more information on snapshots, see Amazon EBS Snapshots.

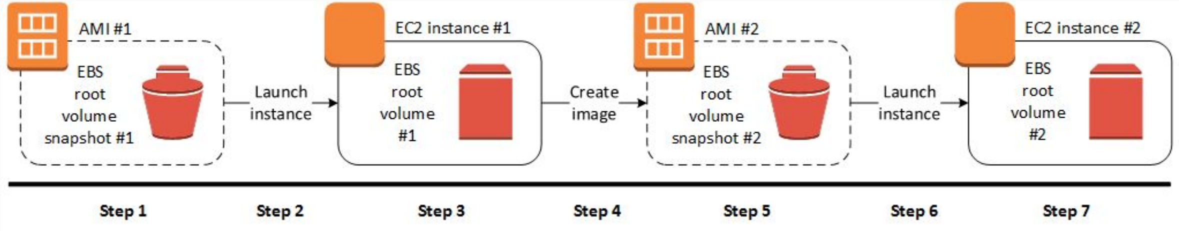
New volumes created from existing EBS snapshots load lazily in the background.

AWS EBS

Creating Image of running EC2 Instance

Ref. Link

You can create an AMI using the AWS Management Console or the command line. The following diagram summarizes the process for creating an Amazon EBS-backed AMI from a running EC2 instance. Start with an existing AMI, launch an instance, customize it, create a new AMI from it, and finally launch an instance of your new AMI. The steps in the following diagram match the steps in the procedure below.



AWS EBS

Migration of My AMI and Snapshot to another Region

Ref. Link

To create a copy of your AMI in another AWS Region, follow these steps:

1. Create an AMI of your EC2 instance:
2. To create a Linux AMI, see Creating Your Own AMI
3. To create a Windows AMI, see Creating a Custom Windows AMI.
4. Copy the AMI of your EC2 instance to another AWS Region:
5. To copy a Linux AMI, see Copying an AMI.
6. To copy a Windows AMI, see Copying an AMI.
7. After the copy operation completes, launch a new EC2 instance from your AMI in the new AWS Region.

AWS EBS

Managing EBS using CLI Commands

Ref. Link

The EB CLI is a command line interface for Elastic Beanstalk that provides interactive commands that simplify creating, updating and monitoring environments from a local repository. Use the EB CLI as part of your everyday development and testing cycle as an alternative to the AWS Management Console.

Once you've installed the EB CLI and configured a repository, you can create environments with a single command:

```
~/my-app$ eb create my-env
```