Learning AWS

With the increasing global interest in leveraging cloud infrastructure, AWS Cloud from Amazon offers a cutting-edge platform for architecting, building, and deploying web-scale cloud applications. The variety of features available within AWS can reduce overall infrastructure costs and accelerate the development process for both large enterprises and startups alike.

Beginning with basic cloud concepts, you’ll learn about the various cloud services models and the design implications of multi-tenant applications. You’ll then design, implement, and deploy a multi-tier, scalable, highly-available and secure application on the AWS platform. At every step, we explain the key guiding principles driving real-world production-ready application architectures. Finally, you will learn how to automate your cloud infrastructure, set up operations, application monitoring, and DevOps pipeline.

Who this book is written for
This book is targeted at expert programmers and architects wanting to learn AWS. Some familiarity with Spring, MySQL, and RESTful web services is assumed.

What you will learn from this book
- Multi-tenant design approaches for SaaS applications
- Planning dev, test, staging, and production environments
- Design considerations for scalability and high-availability
- Planning for DR
- The security considerations and implementation of AWS applications
- Production deployment processes and tools
- Monitoring AWS applications using CloudWatch
- Automating infrastructure using CloudFormation

Aurobindo Sarkar
Amit Shah

In this package, you will find:

- The author's biography
- A preview chapter from the book, Chapter 1 'Cloud 101 – Understanding the Basics'
- A synopsis of the book’s content
- More information on Learning AWS
About the Authors

Aurobindo Sarkar is a consulting CTO at BYOF Studios. With a career spanning 22 plus years, he has consulted at some of the leading organizations in the U.S., the UK, and Canada. He specializes in Software as a Service product development, cloud computing, cloud economics, big data analytics, Internet of Things (IoT) platforms, and web-scale architectures. His domain expertise runs across financial services, media, mobile gaming, public and automotive sectors. Aurobindo has been actively working with technology start-ups for over 5 years now. As a member of the top leadership team at various start-ups, he has mentored several founders and CxOs, provided technology advisory services, developed cloud strategies, drawn up product roadmaps, and set up large engineering teams. Aurobindo has an MS (Computer Science) from New York University, M.Tech (Management) from Indian Institute of Science, and B.Tech (Engineering) from IIT Delhi.

Amit Shah has a bachelor's degree in electronics. He is a senior manager at Western Outdoor Interactive. He has been programming since the early ’80s with the first wave of personal computing—initially as a hobbyist and then as a professional. His areas of interest include embedded systems, Internet of Things (IoT), analog and digital hardware design, systems programming, cloud computing, and enterprise architecture. He has been working extensively in the field of cloud computing and enterprise architecture for the past 4 years.
Preface

With an increasing interest in leveraging cloud infrastructure around the world, the AWS cloud from Amazon offers a cutting-edge platform for architecting, building, and deploying web-scale cloud applications through a user-friendly interface. The variety of features available within AWS can reduce overall infrastructure costs and accelerate the development process for both large enterprises and start-ups alike.

Learning AWS covers basic, intermediate, and advanced features and concepts as they relate to designing, developing, and deploying scalable, highly available, and secure applications on the AWS platform. By sequentially working through the steps in each chapter, you will quickly master the features of AWS to create an enterprise-grade cloud application. We use a three-tiered services-oriented sample application for extensive hands-on exercises.

This book will not only teach you about the AWS components, but you will gain valuable information about key concepts such as multi-tenancy, auto scaling, planning, implementing application development environments, addressing application security concerns, and so on. You will also learn how these concepts relate to cost effective architectural decisions while designing scalable, highly available, and secure AWS cloud applications.

A step-by-step approach to cloud application design and implementation is explained in a conversational and easy-to-follow style. Each topic is explained sequentially in the process of creating an AWS cloud application. Detailed explanations of the basic and advanced features of AWS that address the needs of readers with a wide range of real-world experiences are also included. Expert programmers and architects will appreciate the focus on the practice rather than the theory.
What this book covers

Chapter 1, Cloud 101 – Understanding the Basics, describes basic cloud concepts including the public, private, and hybrid cloud models. We explain and compare the Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), and Software-as-a-Service (SaaS) cloud service delivery models. In addition, we explain multi-tenancy models and the challenges they present in design, implementation, and operations.

Chapter 2, Designing Cloud Applications – An Architect’s Perspective, describes familiar and not-so familiar architectural best practices in the cloud context. These include designing a multi-tier architecture and designing for multi-tenancy, scalability, and availability. We will also guide you through the process of estimating your cloud computing costs.

Chapter 3, AWS Components, Cost Model, and Application Development Environments, introduces you to the AWS components—EC2, S3, RDS, DynamoDB, SQS Queues, SNS, and so on. We will discuss strategies to lower your AWS infrastructure costs and their implications on architectural decisions. We will explain the typical characteristics of the Development, QA, Staging, and Production environments on the AWS cloud.

Chapter 4, Designing for and Implementing Scalability, provides guidance on how to define your scalability objectives, and then discusses the design and implementation of specific strategies to achieve scalability.

Chapter 5, Designing for and Implementing High Availability, provides guidance on how to define your availability objectives, discuss the nature of failures, and then discuss the design and implementation of specific strategies to achieve high availability. In addition, we will describe the approaches that leverage the AWS features and services for your Disaster Recovery planning.

Chapter 6, Designing for and Implementing Security, provides guidance on how to define security objectives, explains your security responsibilities, and then discusses the implementations of specific best practices for application security.

Chapter 7, Deploying to Production and Going Live, provides guidance on managing infrastructure, deployments, support, and operations for your cloud application. In addition, we provide some tips on planning your production Go-Live activities.
Cloud 101 – Understanding the Basics

In this chapter, we will introduce you to cloud computing and the key terminologies used commonly by cloud practitioners.

We will briefly describe what public, private, and hybrid clouds are, followed by a description of various cloud service models (offered by the service providers), including the features of Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS).

To help you get started on Amazon Web Services (AWS), we will end the chapter by walking you through the step-by-step process of creating an AWS account, and describing some of the salient features of the AWS dashboard.

This chapter will cover the following points:

• Define cloud computing and describe some of its characteristics
• Describe and compare public, private, and hybrid clouds
• Explain and compare IaaS, PaaS, and SaaS cloud service delivery models
• Steps to create an AWS account
• A brief overview of the AWS management console
What is cloud computing?

Wikipedia defines cloud computing as:

"Cloud computing is internet-based computing in which large groups of remote servers are networked to allow the centralized data storage, and online access to computer services or resources."

The National Institute of Standards and Technology (NIST) gives the following definition of cloud computing:

"Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction."

There are several other broadly accepted definitions of cloud computing. Some explicitly emphasize configurability of the resources, while others include the need for rapid on-demand provisioning of resources, and still others drop the requirement of access via the internet. We define cloud computing as a model that enables the features listed here:

- Users should be able to provision and release resources on-demand
- The resources can be scaled up or down automatically, depending on the load
- The provisioned resources should be accessible over a network
- Cloud service providers should enable a pay-as-you-go model, where customers are charged based on the type and quantum of resources they consume

Some of the implications of choosing to use the cloud for your computing needs are as follows:

- The illusion of infinite processing and storage resources, available on-demand, reduces the need for detailed advance planning and procurement processes.
- The model promotes the use of resources as per customer needs, for example, starting small, and then increasing resources based on an increase in need.
- The development and test environments can be provisioned on a smaller scale than production environment, and enabled only during normal business hours, to reduce costs.
• The staging environment can be provisioned for a short duration to be a replica of the production environment. This enables testing using production configuration (and scale) for improved defect resolution.

• There will be ease of scaling, both vertically and horizontally, in order to better manage spikes in demand and variations due to business cycles or time-of-day reasons, and so on.

• This encourages experimentation, by trying out new ideas and software by quickly provisioning resources, rather than requisition for resources through time-consuming and cumbersome processes.

In addition, there are several key operational and maintenance-related implications, including no hardware maintenance or data center operations required, zero-downtime migrations and upgrades, ease of replacement of unhealthy machines, ease of implementation of high-availability and disaster recovery strategies, and many more.

These and other implications of using cloud services to design scalable, highly available, and secure applications are discussed in-depth in subsequent chapters.

Public, private, and hybrid clouds

Basically, there are three types of clouds in cloud computing, they are public, private, and hybrid clouds.

In a public cloud, third-party service providers make resources and services available to their customers via the internet. The customers' applications and data are deployed on infrastructure owned and secured by the service provider.

A private cloud provides many of the same benefits of a public cloud but the services and data are managed by the organization or a third-party, solely for the customer's organization. Usually, private cloud places increase administrative overheads on the customer but give greater control over the infrastructure and reduce security-related concerns. The infrastructure may be located on or off the organization's premises.

A hybrid cloud is a combination of both a private and a public cloud. The decision on what runs on the private versus the public cloud is usually based on several factors, including business criticality of the application, sensitivity of the data, industry certifications and standards required, regulations, and many more. But in some cases, spikes in demand for resources are also handled in the public cloud.
Cloud service models – IaaS, PaaS, and SaaS

There are three cloud-based service models, IaaS, PaaS, and SaaS. The main features of each of these are listed here:

- Infrastructure as a Service (IaaS) provides users the capability to provision processing, storage, and network resources on demand. The customers deploy and run their own applications on these resources. Using this service model is closest to the traditional in-premise models and the virtual server provisioning models (typically offered by data center outsourcers). The onus of administering these resources rests largely with the customer.

- In Platform as a Service (PaaS), the service provider makes certain core components, such as databases, queues, workflow engines, e-mails, and so on, which are available as services to the customer. The customer then leverages these components for building their own applications. The service provider ensures high service levels, and is responsible for scalability, high-availability, and so on for these components. This allows customers to focus a lot more on their application's functionality. However, this model also leads to application-level dependency on the providers' services.

- In the Software as a Service (SaaS) model, typically, third-party providers using a subscription model provide end-user applications to their customers. The customers might have some administrative capability at the application level, for example, to create and manage their users. Such applications also provide some degree of customizability, for example, the customers can use their own corporate logos, colors, and many more. Applications that have a very wide user base most often operate in a self-service mode. In contrast, the provider provisions the application for the customer for more specialized applications. The provider also hands over certain application administrative tasks to the customer's application administrator (in most cases, this is limited to creating new users, managing passwords, and so on through well-defined application interfaces).

From an infrastructure perspective, the customer does not manage or control the underlying cloud infrastructure in all three service models.

The following diagram illustrates who is responsible for managing the various components of a typical user application across IaaS, PaaS, and SaaS cloud service models. The column labeled User Application represents the main components of a user application stack, while the following columns depict the varying levels of management responsibilities in each of the three service models. The shaded boxes are managed by the service provider, while the unshaded boxes are managed by the user.
The level of control over operating systems, storage, applications, and certain network components (for example, load balancers) is the highest in the IaaS model, while the least (or none) in the SaaS model.

We would like to conclude our introduction to cloud computing by getting you started on AWS, right away. The next two sections will help you set up your AWS account and familiarize you with the AWS management console.

### Setting up your AWS account

You will need to create an account on Amazon before you can use the Amazon Web Services (AWS). Amazon provides a 12 month limited fully functional free account that can be used to learn the different components of AWS. With this account, you get access to services provided by AWS, but there are some limitations based on resources consumed. The list of AWS services is available at [http://aws.amazon.com/free](http://aws.amazon.com/free).
We are assuming that you do not have a pre-existing AWS account with Amazon (if you do, please feel free to skip this section). Perform the following steps:

1. Point your browser to http://aws.amazon.com/ and click on **Create a Free Account**.

   The process to create a brand new AWS account has started. You can sign in using your existing Amazon retail account, but you will have to go through the process of creating an AWS account; the two accounts are different for accounting purposes, even though they share the same common login. Let's take a look at the following screenshot:

   ![Sign In or Create an AWS Account](image)

   2. After creating a new account or using your existing retail Amazon account, select the **I am a returning user and my password is:** option and click on **Sign in using our secure server**. A set of intuitive screens will guide you through multiple screens in order to create an AWS account, these include:
° **Contact Information**: Amazon also uses this information for billing and invoicing. The **Full Name** field is also used by the AWS management console to identify your account, as shown in the following screenshot:

![Contact Information](image)

° **Payment Information**: When you create an AWS account and sign up for services you are required to enter payment information. Amazon executes a minimal amount transaction against the card on file to confirm that it is valid and not reported lost or stolen. This is not an actual charge it merely places the 'X' amount on hold on the card which will eventually drop off. The 'X' amount depends on the country of origin.

° **Identity Verification**: Amazon does a call back via an automated system to verify your telephone number.

° **Support Plan**: You can subscribe to one from the following, **Basic**, **Developer**, **Business**, or **Enterprise**. We recommend subscribing to the Basic plan to start with.

> The Basic plan costs nothing, but is severely limited and hence not recommended for production. It is an excellent way to learn and get familiar with AWS.
Confirmation: On clicking on Launch Management Console you will be requested to login, as shown in the following screenshot:

3. At this stage, you have successfully created an AWS account, and you are ready to start using the services offered by AWS.

The AWS management console

The AWS management console is the central location from where you can access all the Amazon services. The management console has links to the following:

- Amazon Web Services: This is a dashboard view that lists all the AWS services currently available in a specific Amazon region. Clicking on any one of these launches the dashboard for the selected service, as shown in the following screenshot:
• **Shortcuts for Amazon Web Services:** On the console management screen, you can create shortcuts of frequently accessed services via the **Edit** option, as shown in the following screenshot:

![Console Management Screen](image)

• **Account related information:** This allows you to access your account-related data. This includes security credentials needed to access the AWS resources by your application. The **Billing & Cost Management** option gives you real-time information on your current month's billing; this helps in managing costs, as shown in the following screenshot:

![Account Information](image)

• **Amazon regions:** This option allows you to access the AWS in a specific region. In the following screenshot, all the Amazon Web Services are located in the **US East (N. Virginia)** region:

![Regions](image)
• **Support**: You can navigate to the Help, Forums, and support pages:

![Support](image)

• **Service Health**: Launches the health dashboard of all the Amazon Web Services across all regions, and not of your deployed service:

![Service Health](image)

**Summary**

In this chapter, we introduced you to a few cloud computing concepts and terminologies. We described the basic features of public, private, and hybrid clouds. We introduced the main cloud delivery models, namely, IaaS, PaaS, and SaaS. Finally, we listed the steps for creating your AWS account, and described the salient features of the AWS management console.

With the basics out of the way, in the next chapter we will deep dive into the details of how multitenanted cloud applications are different from traditional multi-tiered applications. We will also walk you through creating a sample application (using Spring and MySQL) that will be used to illustrate key cloud application design concepts through the rest of this book.
Where to buy this book

You can buy Learning AWS from the Packt Publishing website.

Alternatively, you can buy the book from Amazon, BN.com, Computer Manuals and most internet book retailers.

Click here for ordering and shipping details.