

1. Record Nr.	UNINA9910300654603321
Autore	Shackelford Adam
Titolo	Beginning Amazon Web Services with Node.js [[electronic resource] /] / by Adam Shackelford, Adam Shackelford
Pubbl/distr/stampa	Berkeley, CA : , : Apress : , : Imprint : Apress, , 2015
ISBN	1-4842-0653-3
Edizione	[1st ed. 2015.]
Descrizione fisica	1 online resource (249 p.)
Collana	Expert's Voice in Web Development
Disciplina	006.78
Soggetti	Computer programming Software engineering Web Development Software Engineering/Programming and Operating Systems
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	Includes index.
Nota di contenuto	Contents at a Glance; Chapter 1: Getting Started with Amazon Web Services; Understanding the Cloud; The Approach in This Book; Requirements; AWS Account; Domain Registration; SSL Certificate; Code Repository; Download the Sample Project; Local Environment; ExpressJS; Sample Project; Overview; Source Code Organization; Configuration and Startup; Working with the Sample App; Home Route; Users Route; Review-The Order of Things; Example-Working with Parameters; Try It Out; Albums; Photos; Developing with the Sample App; Identity and Access Management; The IAM Dashboard; IAM Users; IAM Groups IAM Managed PoliciesIAM Permissions Editor; Summary; Chapter 2: Working with AWS OpsWorks; Understanding OpsWorks; Allocating Resources; Regions and Availability Zones; Additional IAM Roles; Instance Role; Service Role; The OpsWorks Environment; The OpsWorks Dashboard; Stacks; Amazon Linux; Amazon Machine Images; Instance vs. EBS; Stack Options-Summary; Layers; Creating an OpsWorks Layer; Instances; Apps; Application Source; Deploying from Git; Deploying from Subversion; Deploying from HTTP Archive; Deploying from S3 Archive; Creating your App; Deploying Your App; Summary Chapter 3: OpsWorks Part II: Databases and ScalingRelational Database Service (RDS); Multi-AZ Deployment; Read Replicas; Provisioned IOPS

Storage; DB Security Groups; Creating an RDS Database; Instance Specifications; Settings; Advanced Settings; Database Import; OpsWorks RDS Layer; Environments and Environment Variables; Stack Commands: Backup Scenario; RDS Snapshot; Creating a New RDS Layer; Connect New Database Layer to App; Run Stack Command; Elastic Load Balancing (ELB); Creating a Load Balancer; Define Load Balancer; Configure Health Check; Add EC2 Instances; Add Tags; Review OpsWorks ELB Layer Adding a New Instance; Summary; Chapter 4: CloudFront and DNS Management; CloudFront; Creating the Distribution; Configuring the Distribution; Origin Settings; Default Cache Behavior Settings; Distribution Settings; Price Class; Alternate Domain Names; SSL Certificate; Default Root Object; Distribution Settings-Summary; Distribution Detail View; Origins; Behaviors; Behavior with Query Strings; Deploy Code Changes; Add New Behavior; Caching; Invalidations; Controlling Caching; Testing CloudFront Caching; Cache Statistics; Route 53; Summary

Chapter 5: Simple Storage Service and Content Delivery Using S3 in the Application; Creating an S3 Bucket; Enabling S3 Access in IAM; Storing Credentials; Implementing IAM Roles; Using IAM User Credentials; Adding OpsWorks Environment Variables; Developing with the AWS SDK; Updating Dependencies; Accessing Environment Variables; Handling File Uploads; Updating the Database Schema; Integrating with CloudFront; Creating CloudFront S3 Origin; CloudFront S3 Behavior; Finishing Touches; Absolute URLs; Enhanced Image Caching; Summary;

Chapter 6: Simple Email Service; Introducing Simple Email Service Exploring the SES Dashboard

---

## Sommario/riassunto

Beginning Amazon Web Services with Node.js teaches any novice Node.js developer to configure, deploy, and maintain scalable small to large scale Node.js applications in Amazon Web Services. Hosting a Node.js application in a production environment usually means turning to PaaS hosting, but this approach brings problems. Deploying Node.js directly to AWS solves the problems you encounter in these situations, enabling you to cut out the middle man. You will begin with a basic RESTful web service in Node.js, using the popular Express.js framework, pre-built and ready to run in your local environment. You will be introduced to the most powerful tools in AWS, and learn how to configure your project to take advantage of them. You will be guided through the steps of getting the various key components to work together on AWS. Through code samples using the AWS JavaScript SDK and tutorials in the AWS console, you will gain the knowledge to incorporate secure user authentication, server auto-scaling, a load balancer, CDN, customized caching behavior, and outage monitoring. Node.js is single-threaded, and designed to run high input / high output applications, making it ideal for the cloud environment. If your main task is handling a high volume of requests over HTTP / the web, it will scale very well in proportion to the computing power you allocate to it. Amazon Web Services offers a broad set of services that help organizations move faster, lower costs, and scale applications. Trusted by the largest enterprises and start-ups alike, AWS powers a wide variety of workloads across a broad spectrum. If leveraged properly, you can build a Node.js app on AWS which can automatically power itself up to handle a massive volume of traffic, and then scale back down to a lighter configuration when user demand has dropped. Amazon Web Services offers a broad set of services that help organizations move faster, lower costs, and scale applications. Get started with Node.js and AWS using this book today.

---