22AIE305: CLOUD COMPUTING

HOW TO RUN A DOCKER CONTAINER ON THE CLOUD?

Using a Container Registry

If your container runs locally, the easiest way to bring it to the Cloud is to select a container registry that acts as a centralized location to store your containers. Essentially, you will need to push your container to this registry, whether public or private, so that your image can be distributed from there.

Using Container-as-a-Service

Containers-as-a-Service (CaaS) is a concept that allows companies to directly run their container on the Cloud using any given provider of choice. With CaaS, the infrastructure required to run containers such as orchestration tools e.g Docker Swarm, Kubernetes, OpenStack, etc., as well as cluster management software are non-existent for the user.

ADVANTAGES OF USING CONTAINER-AS-A-SERVICE

Cost reduction: it eliminates the time, effort, and money spent on maintaining secure infrastructure to run your container.

Flexibility: you can easily move from cloud to cloud or even back to your on-premise infrastructure, freeing you from vendor lock-in.

Speed: Since the underlying infrastructure abstracts from it, you can deploy your container quicker.

Overall, CaaS will not only simplify the running process of a software application but also improve overall security around it as most CaaS solutions offer vulnerability scans. Furthermore, you don't have to worry about managing the hardware that will run your container.

KEY CONSIDERATIONS

- Can it operate multi-container applications?
- What networks and storage functions are available?
- Which file formats does it support?
- How is storage achieved?
- Which billing model does it use?

AMAZON ELASTIC CONTAINER SERVICE (AMAZON ECS)

Amazon ECS is a scalable container orchestration platform by AWS designed to run, stop, and manage containers in a cluster environment by using task definition.

Task definition is where you define

- The container to use
- How many containers to run
- How your containers are linked
- What resources your containers use

It is also important to note that AWS ECS also supports mounting EFS volumes.

With that in mind, you have two ways of using ECS, either by using EC2 Instances or Fargate.

AWS FARGATE

AWS Fargate was launched in 2017, and with this model, you don't have to be worried about managing EC2 Instances. AWS Fargate directly manages the underlying servers required to run your container by pre-configuring a cluster for you. You will just need to add your workload to it.
The advantages include:

- ✓ No infrastructure to manage
- ✓ AWS deals with availability and scalability of your container application
- ✓ Fargate Spot, based on similar principles as the Spot instances, AWS mentions a cost reduction of up to 70%.

ECS/FARGATE VS KUBERNETES

Infrastructure Layer	Component	ECS Fargate	Kubernetes
Workload	Deployment Unit Desired State Access Endpoint	Task Service ALB	Pod Deployment Ingress
Control Plane	API Endpoint Scheduler Controller State Management	Frontend Service Capacity Manager Cluster Manager State DB	Kube-apiserver Kube-scheduler Kube-controller etcd
Data Plane	Guest OS Agent Container Runtime Network	Amazon Linux 2 Fargate Agent Docker ENI/VPC	Linux/Windows Kubelet Containerd CN/Kubeproxy

AWS APP RUNNER

Launched in 2021, AWS App Runner facilitates bringing a web application to the cloud without worrying about scaling or the infrastructure associated with it. Essentially, it runs Amazon ECS with Fargate to execute your container but you don't need to set up or configure anything related to Fargate.

It can run in build mode which pulls code from your GitHub repository and builds the application at any commits you might push to your main branch.

Alternatively, it can run in container mode, where you will connect your container registry (only AWS ECR is supported) and point to your image. If you want to see what AWS has planned for App Runner, they outline everything you need to know with their detailed roadmap.

The core advantage of AWS App Runner when it comes to running a Docker container on the Cloud is that it is easy to configure, and provides a simple way to get a web application to run in the Cloud.

CONFIGURATIONS

A registry that supports the Docker Registry HTTP API V2 and presents a valid certificate for HTTPS traffic may be needed.

By default, apps listen for connections on the port specified in the PORT environment variable for the app. If you do not specify a port in the EXPOSE directive, then the app listens on the value of the PORT environment variable as determined by Cloud provider.

When configuring a Docker image for Cloud, you can control the exposed port and the corresponding value of PORT by specifying the EXPOSE directive in the image Dockerfile. If you specify the EXPOSE directive, then the corresponding app pushed to Cloud listens on that exposed port. For example, if you set EXPOSE to 7070, then the app listens for connections on port 7070.

PUSHING APPS TO THE CLOUD

Most Cloud providers support "pushing Apps" from container registries such as Docker Hub, Google Container Registry (GCR), and Amazon Elastic Container Registry (ECR).

You can use any Docker image registry that presents a valid certificate for HTTPS traffic, such as a company-internal Docker registry.

Service provider automatically encrypts all your data at rest to provide an additional layer of security.

Amazon CloudFront is a tool to distribute content globally and improve Apps performance. It is a content delivery network (CDN) service that caches and delivers your content, including web pages, videos, images, and other files, to users globally

GENERAL DEPLOYMENT STEPS

A load balancer, Web server, and database server components should be selected from a library of preconfigured virtual machine images.
Configure each component to make a custom image.
Load balancer is configured accordingly; web server should be populated with the static contents by uploading them to the storage cloud where as the database servers are populated with the dynamic content of the site.
The developer then feeds the custom code into the new architecture making components meet their specific requirements.

DEPLOYMENT ON AZURE CLOUD

1) Step-1

Initially start visual studio in the administrator mode then go to file select new file. Select cloud service from project

types and from template select web cloud service.

In the solution explorer double click on default.aspx.Develop and

press F5 to compile and debug the application.

In the solution explorer, right click on the application and

then click publish.

A publish folder gets opened which contains service package file and cloud service configuration file.

DEPLOYMENT ON AZURE CLOUD CONT'D

2) Step-2

Log in to the windows azure portal using your windows

live id to deploy the application on the cloud

3) Step-3

In the portal, click on the hosted services, storage accounts and CDN

Click new hosted service. Select a subscription that will be used for application.

4) Step-4

Enter the name of the application, enter URL for your application, and then choose a region from the list of regions.

Select deploy to stage environment

DEPLOYMENT ON AZURE CLOUD CONT'D

Step-5

Ensure that Start after successful deployment is checked.

Specify a name for the deployment.

Step-6

For Package location, click the corresponding Browse locally… button, navigate to the folder where your <Your Project Name>.cspkg file is, and select the file.

For Configuration file, click the corresponding Browse locally… button, navigate to the folder where your Service Configuration.cscfg is, and select the file.

DEPLOYMENT ON AZURE CLOUD CONT'D

Step-7

Click OK. You will receive a warning after you click OK because there is only one instance of the web role defined for your application (this setting is contained in the Service Configuration. Cscfg file). For purposes of this walk-through, override the warning by clicking yes, but realize that you likely will want more than one instance of a web role for a robust application.

Heroku is a platform as a service based on a managed container system, with integrated data services and a powerful ecosystem, for deploying and running modern apps. The Heroku developer experience is an app-centric approach for software delivery, integrated with today's most popular developer tools and workflows.

Heroku Runtime

Heroku runs your apps inside dynos — smart containers on a reliable, fully managed runtime environment. Developers deploy their code written in Node.js, Ruby, Java, PHP, Python, Go, Scala, or Clojure to a build system which produces an app that's ready for execution. The system and language stacks are monitored, patched, and upgraded, so it's always ready and up-to-date. The runtime keeps apps running without any manual intervention.

Heroku Developer Experience (DX)

The Heroku Developer Experience is an app-centric approach to software delivery so developers can focus on creating and continuously delivering applications, without being distracted by servers or infrastructure. Developers deploy directly from popular tools like Git, GitHub or Continuous Integration (CI) systems. The intuitive web-based Heroku Dashboard makes it easy to manage your app and gain greater visibility into performance.

Data Services and Ecosystem

Heroku Elements let developers extend their apps with Add-ons, customize their application stack with Buildpacks and jumpstart their projects with Buttons. Add-ons are 3rd party cloud services that developers can use to immediately extend their apps with a range of functionality such as data stores, logging, monitoring and more. Heroku provides three fully-managed data service Add-ons: Heroku Postgres, Heroku Key-Value Store, and Apache Kafka on Heroku.

Heroku Operational Experience (OpEx)

The Heroku Operational Experience is a key component of the platform. It helps developers through troubleshooting and remediation of common issues and customizing their ops experience to quickly identify and address negative trends in their application health. Heroku provides a set of tools to alert you if something goes wrong, or to automatically scale your web dynos if the response time for web requests exceeds a threshold you specify. Application metrics, Threshold Alerting, and Autoscaling are some of the features you get access to with no extra cost.

Security and Compliance

Developers from around the world entrust sensitive data to Heroku, and nothing is more important to us than honoring our custodial commitments to protect this data. Heroku regularly performs audits and maintains PCI, HIPAA, ISO, and SOC compliance to further strengthen our trust with customers. Learn more by visiting our compliance center.


Cost Effective
Serverless
Google Cloud Platform
Highly Scalable
Cloud AI
Custom Machine Types
Big Data Analytics
IoT
Internet of Things
API Platform and Ecosystem

The comparative analysis of the best deployment models

	Public	Private	Community	Hybrid
Ease of setup and use	Easy	Requires IT proficiency	Requires IT proficiency	Requires IT proficiency
Data security and privacy	Low	High	Comparatively high	High
Data control	Little to none	High	Comparatively high	Comparatively high
Reliability	Low	High	Comparatively high	High
Scalability and flexibility	High	High	Fixed capacity	High
Cost-effectiveness	The cheapest	Cost-intensive; the most expensive model	Cost is shared among community members	Cheaper than a private model but more costly than a public one
Demand for in-house hardware	No	Depends	Depends	Depends

DEPLOYING CONTAINER IMAGES

Container is a powerful way for developers to package and deploy application

They are lightweight and provide a consistent, portable software environment for applications to easily run and scale anywhere.

Containers provide high security, reliability, and scalability.

Containers make it easy to package entire applications and move them to the Cloud without needing to make any code changes.

Use containers to build platforms that remove the need for developers to manage infrastructure and standardize how your applications are deployed and managed.

Containers provide process isolation that makes it easy to break apart and run applications as independent components called microservices

CONTAINERD AND RUNC

Containers are an abstraction at the app layer that packages code and dependencies together. Many application instances can be running in containers on a single host without visibility into each others' processes, files, network, and so on.

containerd is an industry-standard container runtime that leverages runc and was created with an emphasis on simplicity, robustness and portability.

containerd is the core container runtime of the Docker Engine.

container image specification and runtime code is known as runc

FREE AMAZON TOOLS FOR DEPLOYMENT

Build container solutions using these product offers from the AWS Free Tier.

PRODUCT	DESCRIPTION	FREE TIER OFFER DETAILS	PRODUCT PRICING
Amazon Elastic Container Registry (ECR) Container Registry	Amazon Elastic Container Registry (Amazon ECR) is a fully managed container registry that makes it easy to store, manage, share, and deploy your container images and artifacts anywhere.	**12 MONTHS FREE** **500 MB** per month of storage for your private repositories	Amazon ECR pricing
Amazon Elastic Container Service (ECS) Container Orchestration 	Amazon ECS is a fully managed container orchestration service that helps you easily deploy, manage, and scale containerized applications. It deeply integrates with the rest of the AWS platform to provide a secure and easy-to-use solution for running container workloads in the cloud.	**ALWAYS FREE** Amazon ECS is always free. You pay for AWS resources (e.g. EC2 instances, EBS volumes, and memory) you create to store and run your application.	Amazon ECS pricing
Amazon Elastic Container Service (ECS) Anywhere Hybrid Deployments	A feature of Amazon ECS that enables you to easily run containers on customer-managed infrastructure, including on your own virtual machines (VMs) and bare metal servers.	**6 MONTHS FREE** **2,200 instance** hours per month	Amazon ECS Anywhere pricing

FREE AMAZON TOOLS FOR DEPLOYMENT

Amazon Lightsail Virtual Private Server	Lightsail is an easy-to-use virtual private server (VPS) that offers you everything needed to build an application or website, plus a cost-effective, monthly plan.	**3 MONTH FREE TRIAL** Offer only applies to one bundle per account. Standard charges apply after first **750 hours** of usage of the selected bundle each month. Try the 3.50/mo, $5/mo, or $10/mo plans for free for three months when using Linux/Unix Try the $8 USD, $12, or $20 plans for free for three months when using Windows Try the $10 USD per node for free for three months when using Containers Try the $15 USD database plans for free for three months when using Databases	Amazon Lightsail pricing
Amazon EC2 Elastic Compute Cloud	EC2 virtual machines gives you control of your server clusters and provide a broad range of customization options.	**12 MONTHS FREE** **750 hours** per month of Linux, RHEL, SLES or Windows t2.micro or t3.micro instance dependent on region	Amazon EC2 pricing

SOME AWS TOOLS

Amazon SageMaker is designed for building, training, and deploying machine learning models at scale

Amazon VPC is used to create and manage virtual private networks (VPNs) on cloud

AWS Redshift provides a fully managed, scalable, and server-less data warehouse, which is optimized for online analytic processing (OLAP) and can handle large-scale data warehousing and analysis workloads.

AWS CloudTrail is used for Auditing and tracking API activity in your AWS account. It records actions taken by users, services, or AWS Management Console, providing visibility into the historical usage of AWS services and resources.

IAM is a service for Identity and Access Management. It enables you to manage user identities and their access to AWS resources securely. IAM helps you control who can access your AWS resources and what actions they can perform.

SOME MORE AWS TOOLS

AWS Lambda is a serverless compute service that allows you to run your code without the need to provision or manage servers. It automatically scales your application in response to incoming requests and charges you only for the compute time consumed.

Amazon SES is an email service commonly used for sending emails to customers

Amazon S3 can be used to host a static website using HTML, CSS, JavaScript, etc. You can configure S3 to serve the static website content directly to users, making it a simple and cost-effective solution for hosting static websites

AWS Cloud Formation is a service that allows you to define your infrastructure as code using templates. It enables you to create, update, and delete resources in a predictable and automated way, making it easier to manage and deploy your infrastructure.

SOME MORE AWS TOOLS

Amazon DynamoDB is a fully managed NoSQL database service that provides fast and predictable performance with seamless scalability. It is designed to handle large amounts of data and can automatically scale up or down based on your application's needs.

AWS CloudWatch is a monitoring and observability service that provides metrics, logs, and events for resources and applications running on AWS. It allows you to collect and track metrics, monitor logs, and set alarms to gain insights into the performance and health of your applications.

AWS Secrets Manager is a service that enables you to securely store and manage secrets such as database credentials, API keys, and encryption keys. It provides an easy and secure way to retrieve secrets programmatically for your applications.

SOME MORE AWS TOOLS

Amazon Virtual Private Cloud (Amazon VPC) allows you to create and manage virtual networks in the AWS cloud. It provides isolated sections of the AWS cloud where you can launch AWS resources in a virtual network that you define.

Amazon QuickSight is a business intelligence service that allows you to analyze and visualize data through interactive dashboards, charts, and reports. It enables you to gain insights from your data and make data-driven decisions.

AWS Amplify is a development platform that provides a set of tools and services for building serverless backend applications. It simplifies the development process by offering features like authentication, APIs, storage, and hosting, allowing developers to focus on building their application logic.

SOME MORE AWS TOOLS

Web metrics is used for load testing and Web performance management

Amazon Kinesis is a platform for real-time streaming of data at a massive scale. It enables you to collect, process, and analyze streaming data such as website clickstreams, IoT telemetry data, and application logs in real-time.

AWS Glue is a fully managed, serverless data integration service that makes it easy to prepare and load data for analytics. It automatically generates code to extract, transform, and load (ETL) data from various sources and catalogs the metadata for easy discovery.

AWS Elastic Beanstalk is a platform as a service (PaaS) offering that makes it easy to deploy, run, and scale web applications. It abstracts the underlying infrastructure details and allows developers to focus on writing code while AWS handles the deployment and management aspects.

ECS, EFS, EMR, ..

AWS Direct Connect enables you to establish a dedicated network connection from your premises to AWS. It provides a more consistent and reliable network experience compared to internet-based connections, which is particularly useful for large data transfers or sensitive workloads.

Amazon Elastic Container Service (ECS) is a fully managed container orchestration service for Docker containers. It allows you to run, manage, and scale containerized applications using familiar Amazon EC2 instances or AWS Fargate.

Amazon Elastic File System (EFS) is a fully managed file storage service that provides scalable and highly available file storage for use with Amazon EC2 instances. It supports multiple EC2 instances accessing the same file system concurrently.

Amazon Elastic MapReduce (EMR) is a fully managed big data processing service

Amazon Relational Database Service (RDS) is a fully managed service that provides resizable and scalable relational databases in the cloud. It supports popular database engines like MySQL, PostgreSQL, Oracle, and SQL Server.