Three Tier Architechture on AWS

8 Services and 2 Databases

Vaibhav 9 August 2024

Agenda

E-commerce Website: The Robot Shop

- To Explain the design of three tier architecture
- What are the different components usually involved
- What is a micro service architecture
- How to write Docker Files
- How to write Kubernetes Resources
- How to write Helm Charts
- Finally to deploy it on an EKS platform

The Robot Shop

Microservice Component Tools

- Node JS(express)
- Java(Spring Boot)
- Python(Flask)
- Golang
- PHP(Apache)
- MongoDB
- Redis
- MYSQL(Maxmind data)
- RabbitMQ
- NGINX
- AngularJS(1.x)

DEMO

Instana Project

Overview

- This is a demo project used to test the Instana service of IBM cloud
- Made changes in the fork to make it easily deployable on EKS

Microservices

- Cart
- Catalogue
- Dispatch
- Payments
- Rating
- Shipping
- User

Create an EKS cluster!
(Takes 10-15 mins)

Should have AWS-CLI,eksctl,kubectl installed

High Level Diagram

The Robot Shop

3 Tier Architecture	3 Layered	Clicks on product	Processes
Frontend	Presentation Layer	Displays The product to user	Displays the UI component
Backend	Logic Layer	Processes the user request	Fetches the UI request
Database	Data Layer	Stores the Data	Stores the Data Which is fetched
Notes			

User Workflow

The Robot Shop

Category	Primary Flow	Secondary Flow	Tertiary Flow
User	Registration	Login	User Information
Catalogue	AI Products / Robots	Accessories	Extended Warranty
Rating	Product Rating	Feedbacks	Complain Tickets
Cart	Primary Orders	Wishlist	Previous Orders
Payment	Payment window	Refunds	Vouchers / Coupons
Shipping	Address Info	Product Returns	Warehouse Info
Order Complete	Order Id	Shipping ID	Tracking ID

Monolithic Vs Micro-service Arch.

The Robot Shop

Difference	Monolithic	Microservices
Deployment	Entire application is deployed as one package.	Each service is deployed independently.
Technology Stack	Typically uses a single technology stack.	Each service can use a different technology stack.
Scalability	Scaling involves replicating the entire application.	Individual services can be scaled independently.
Failure Isolation	A failure in one part can affect the entire application.	Failures are isolated to individual services.
Testing	Easier To Test, Complex Tests	Individual Testing, Integration Testing
Maintenance	Harder to maintain when app grows in complexity	Easier to maintain individual services
Cost	Low initial cost, higher maintenance costs	Higher initial cost, low maintenance cost

Microservice Architechture

The Robot Shop

3 Tier Architecture	UI	Logic	DB
Frontend	Angular (Presentation)	Cart,catalogue,etc	Rabbit Message Queue
Backend	JS	Golang,Python,etc	Catalogue-MYSQL User-Mongo DB
Database	Reddis(Stateful on k8s) (In memory data store)	In memory cache (If site crashes, data lost)	
Notes	Search for In memory cache and in memory data store		

Docker Files

The Robot Shop-shipping docker file

```
#
# Build
#
FROM debian:10 AS build

RUN apt-get update && apt-get -y install maven

WORKDIR /opt/shipping

COPY pom.xml /opt/shipping/
RUN mvn dependency:resolve
COPY src /opt/shipping/src/
RUN mvn package

#
# Run
#
FROM openjdk:8-jdk

EXPOSE 8080

WORKDIR /opt/shipping

ENV CART_ENDPOINT=cart:8080
ENV DB_HOST=mysql

COPY --from=build /opt/shipping/target/shipping-1.0.jar shipping.jar

CMD [ "java", "-Xmn256m", "-Xmx768m", "-jar", "shipping.jar" ]
```

What is Helm

The Robot Shop

- Package Manager for K8s
- Charts: Helm packages are called charts,decribes the set for k8s resources like-(deployements,services,etc)
- Why use helm-
- Simplifies deployment
- Version Control for k8s deployments
- Reusability

Installation and Deployments

The Robot Shop

- Pre-requisites
- EKS Cluster Setup
- OIDC IAM
- ALB Configurations
- EBS CSI Drivers

K8s Deployment and Service Files

The Robot Shop

Aspect	Deployments	Service
Purpose	Manages deployment and ensures the number of pods running	Provides network access to set of pods, balances traffic between them.
Primary Use Case(Defines)	State of an application,the set of replicas, updates, and rollout history	to expose the application inside or outside the cluster and enable communication between microservices.
Update Strategy	Supports rolling updates and rollback of applications.	Does not manage updates; it simply routes traffic to the most current set of healthy pods.
Types	Deployment, ReplicaSet, StatefulSet, DaemonSet, etc.	ClusterIP, NodePort, LoadBalancer, ExternalName.

OICD IAM configuration

The Robot Shop

- EKS is a service on AWS deployed on pods
- Redis->Stateful set->Persistent Volume(EBS Volume)
- EKS has to communicate with EBS service
- K8s services accounts are integrated with AWS IAM
- The pod can now communicate with other services
- Example- EKS(Redis pod) should communicate with EBS via IAM

AWS EBS CSI plugin configuration

The Robot Shop

- Redis is deployed as a stateful set(best practice).
- Redis requires persistent volume(on AWS it is EBS)
- Redis has two component- PVC(persistent volume claim) and SC(storage class)
- AWS helps us via EBS CSI plugin on EKS
- Whenever a PVC is created EBS is automatically created and is attached automatically to the Redis Stateful set

Deployment of Application With Helm

The Robot Shop

- We can either created every docker container and deploy on k8s
- Or can easily use helm and deploy it as image sets
- We can also easily re-use these images.

K8s pods

The Robot Shop

```
~/three-tier-architecture-demo/EKS/helm on master
kubectl get pods -n robot-shop
NAME                            READY   STATUS    RESTARTS   AGE
cart-78dbff49b-dbt9b            1/1     Running   0          4m31s
catalogue-7b4b777975-xgrst      1/1     Running   0          4m32s
dispatch-7d4ff989d7-mdjpx       1/1     Running   0          4m31s
mongodb-b487b86b6-qvp8g         1/1     Running   0          4m31s
mysql-7c9bcd9464-n4nxt          1/1     Running   0          4m32s
payment-7474f4f69f-4kghh        1/1     Running   0          4m31s
rabbitmq-7bc9649444-lm5qh       1/1     Running   0          4m31s
ratings-8c68dd6c5-2nfh2         1/1     Running   0          4m31s
redis-0                         1/1     Running   0          4m31s
shipping-5c899bdb6c-2fpj9       0/1     Running   0          4m32s
user-596968bd87-wwzr4           1/1     Running   0          4m32s
web-6545b6c677-mtjfd            1/1     Running   0          4m32s
```

Ways to expose K8s pods to real world

The Robot Shop

Aspect	Load Balancer Service	Ingress Based Services
Basic Functionality	Directly exposes services via a cloud provider's load balancer.	Provides a single entry point (load balancer) to manage multiple services via HTTP/ HTTPS routing.
Use Case	Best for exposing individual services directly to the outside world.	Best for consolidating access to multiple services under a single external IP, often with path-based or host-based routing.
External IP Address	Assigns an external IP for each Load Balanced Service	Assigns a single external IP for all services managed by Ingress.
Health Checks	Performs health checks at the service level through the cloud provider's load balancer.	Can perform health checks through the Ingress Controller, often more customizable.

Ingress based service

The Robot Shop

- Ingress.yaml
- Creating ingress for the service known as web
- Service port is 8080
- Exposed using the ingress controller

Ingress.yaml

```yaml
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  namespace: robot-shop
  name: robot-shop
  annotations:
    kubernetes.io/ingress.class: alb
    alb.ingress.kubernetes.io/scheme: internet-facing
    alb.ingress.kubernetes.io/target-type: ip
spec:
  rules:
    - http:
        paths:
          - path: /
            pathType: Prefix
            backend:
              service:
                name: web
                port:
                  number: 8080
```

kubectl get pods

```
~/three-tier-architecture-demo/EKS/helm on  master
kubectl get pods -n robot-shop
NAME                          READY   STATUS    RESTARTS   AGE
cart-78dbff49b-dbt9b          1/1     Running   0          24m
catalogue-7b4b777975-xgrst    1/1     Running   0          24m
dispatch-7d4ff989d7-mdjpx     1/1     Running   0          24m
mongodb-b487b86b6-qvp8g       1/1     Running   0          24m
mysql-7c9bcd9464-n4nxt        1/1     Running   0          24m
payment-7474f4f69f-4kghh      1/1     Running   0          24m
rabbitmq-7bc9649444-lm5qh     1/1     Running   0          24m
ratings-8c68dd6c5-2nfh2       1/1     Running   0          24m
redis-0                       1/1     Running   0          24m
shipping-5c899bdb6c-2fpj9     1/1     Running   0          24m
user-596968bd87-wwzr4         1/1     Running   0          24m
web-6545b6c677-mtjfd          1/1     Running   0          24m
```

Deployed!

Stan's Robot Shop

Search

Login / Register

Cart

€953.00

Categories

- Artificial Intelligence
  - Ewooid
  - Stan
  - Watson
- Robot
  - Cybernated Neutralization Android
  - Exceptional Medical Machine
  - Extreme Probe Emulator
  - High-Powered Travel Droid
  - Responsive Enforcer Droid
  - Robotic Mining Cyborg
  - Stan
  - Strategic Human Control Emulator
  - Ultimate Harvesting Juggernaut

Greetings Vaibhav Mishra

Email - vaibhavmishra5g@gmail.com

Order History

Order ID Items Total