

NewgenONE OmniDocs RMS

Configuration and Deployment Guide for AWS

Version: 4.0 SP1

Newgen Software Technologies Ltd.

This document contains propriety information of NSTL. No part of this document may be reproduced, stored, copied, or transmitted in any form or by any means of electronic, mechanical, photocopying, or otherwise, without the consent of NSTL.

Table of contents

1	Pref	ace	. 3
	1.1	Revision history	3
	1.2	Intended audience	3
	1.3	Documentation feedback	3
2	Con	figuring AWS Kubernetes Cluster	. 4
	2.1	Creating an IAM user	4
	2.2	Creating a VPC	6
	2.3	Creating a subnets	8
	2.4	Creating an internet gateway	11
	2.5	Creating a route table	12
	2.6	Creating an IAM role	13
	2.7	Creating a security group	14
	2.8	Creating an EKS cluster	14
	2.9	Creating a key pair	16
	2.10	Provisioning Kubernetes worker nodes using cloud formation	17
	2.11	Adding inbound rule in EC2 instance	20
	2.12	Enabling worker node to join EKS cluster	22
	2.13	Running Kubectl from a local machine	24
	2.14	Creating EFS	24
	2.15	Mounting EFS to worker nodes	27
	2.16	Configuring Kubernetes dashboard	27
	2.17	Configuring AWS load balancer controller	27
	2.18	Configuring AWS Elastic Redis Cache	30
	2.19	Registering domain using route-53	35
	2.20	Generating SSL certificate against the registered domain	37
	2.21	Cluster AutoScaler	41
	2.21	.1 Node Group IAM Policy	41
	2.21	2 Updating auto scaling group	42
	2.21	3 Deploying cluster autoscaler	44
	2.21	4 Viewing cluster autoscaler logs	45
	2.22	Setting up CloudWatch container insights	45
3	Dep	loying OmniDocs and RMS containers	46
	3.1	Prerequisites	46
	3.2	Deliverables	46
	3.2.1	Docker Images	47
	3.2.2	Configuration files	47
	3.2.3	YAML files	48
	3.3	Changes in Product's YAML files	49
	3.4	Changes in AWS load balancer controller YAML files	54
	3.5	Changes in configuration files	56
	3.5.1	Prerequisites	56
	3.5.2	OmniDocs+RMS Web Changes	56
	3.5.3	Wrapper changes	60

3.5.4	AlarmMailer changes	61
3.5.5	LDAP changes	62
3.5.6	SSO changes	69
3.5.7	Scheduler changes	69
3.5.8	ThumbnailManager changes	70
3.5.9	TEM changes	71
3.5.10	EasySearch changes	72
3.5.11	WOPI changes	78
3.5.12	OmniScanWeb changes	80
3.5.13	RMS SharePoint Adapter changes	82
3.5.14	Messaging Service changes	84
3.6 De	ploying containers	86
3.7 Cre	ating cabinet and data source	90
3.7.1	Getting started with OSA	90
3.7.2	Registering JTS server	92
3.7.3	Connecting OSA to JTS Server	94
3.7.4	Creating a cabinet	96
3.7.5	Associating a cabinet	101
3.7.6	Creating a data source	105
3.7.7	Registering a cabinet in OmniDocs	118
3.7.8	Registering a cabinet in RMS	120
3.7.9	Creating site and volume	121
3.8 Eas	ySearch Post-Deployment Changes	126
3.9 Om	niScanWeb: Registering a cabinet	
3.10 Cre	ating a Secret manager policy and secrets	
4 Configu	ring AWS CodePipeline for container deployment on EKS	132
4.1 Ov	erview	132
4.2 Arc	hitecture of CICD pipeline	132
4.3 Co	figuring AWS Elastic container registry	
4.4 Pus	h and Pull docker images to or from AWS ECR	135
4.5 Co	figuring AWS CodePipeline	
4.5.1	Creating an IAM Policy and IAM Role	
4.5.2	Creating AWS CodeCommit repository	
4.5.3	Creating AWS CodeBuild project	
4.5.4	Creating AWS CodePipeline	
4.5.4.	1 Configuring AWS CodePipeline for Dev stage	
4.5.4.	2 Configuring notification	157
4.5.4.	3 Configuring AWS CodePipeline for UAT stage	162
4.5.4.	4 Configuring AWS CodePipeline for production stage	174
Appendix		

1 Preface

This guide describes the deployment and configuration of NewgenONE OmniDocs Record Management System (RMS) 4.0 SP1. It includes Docker images and their required configuration files on the AWS Elastic Kubernetes Service (EKS).

1.1 Revision history

Revision Date	Description
April 2024	Initial publication

1.2 Intended audience

This guide is intended for system administrators, developers, and all other users who are looking for information on the deployment of NewgenONE OmniDocs and RMS containers on AWS Kubernetes Services. The reader must have administrative rights on the machine.

1.3 Documentation feedback

To provide feedback or any improvement suggestions on technical documentation, write an email to <u>docs.feedback@newgensoft.com</u>.

To help capture your feedback effectively, share the following information in your email.

- Document Name:
- Version:
- Chapter, Topic, or Section:
- Feedback or Suggestions:

2 Configuring AWS Kubernetes Cluster

This chapter describes the configuration of AWS Kubernetes Service. For procedural details, refer to the below sections.

2.1 Creating an IAM user

For creating an IAM user, configure the AWS Kubernetes Cluster instead of using the Amazon Management Console root user.

Perform the below steps to create an IAM user:

- 1. Sign in to the AWS Management Console using the root user and open the IAM console in Services.
- 2. Select Users and then select Add User in the navigation panel.
- 3. Enter the **username** for the new user. It's a signed-in name for AWS.
- 4. Select the user's access type. You can select programmatic access or access to the AWS Management Console, and both.
 - Select **Programmatic access** to access the API, AWS CLI, or Tools for Windows PowerShell. This creates an access key for each new user. You can view or download the access keys once you reach the final page.
 - Select **AWS Management Console access** to access the AWS Management Console. This creates a password for each new user.
- 5. For the Console password, select any of the following:
 - Auto-generated password: It provides a randomly generated password to each user that meets the account password policy in effect (if any). Once complete, you can view or download the passwords.
 - **Custom password**: The password you entered is assigned to each user.

aws Services	🗸 🛛 Resource Groups 🗸 🖌		∆ vivek_	_kumar 👻 Globa	▼ Support ▼
	rou oun uuu matapio uooro ut onoo n	тато запто вочного суро или ротпьолого, воитт тъто			
	User name*	vivek			
		Add another user			
	Select AWS access type				
	Select how these users will access Al	S. Access keys and autogenerated passwords are provided in the last step. Learn more			
	Access type*	Programmatic access Enables an access key ID and secret access key for the AWS API, CLI, SDK, and other development tools.			
		AWS Management Console access Enables a password that allows users to sign-in to the AWS Management Console.			
	Console password*	Autogenerated password Custom password			
		system123# ✓ Show password			
	Require password reset	 User must create a new password at next sign-in Users automatically get the IAMUserChangePassword policy to allow them to change their own password. 			
	* Required	Cancel Next	t: Permissions		v

Figure 2.1

- 6. Click Next: Permissions. The Set Permissions screen appears.
- 7. Select the Attach existing policies directly and select the Administrator Access policy.

aws Services +	Resource Groups 🗸 🔹		Ѻ vivek_kumar + Global + Support +	
	Cop exis	y permissions from ting user Attach ex directly	sting policies	*
	Filter policies ~ Q Search		Showing 468 results	1
	Policy name 👻 Type	Used as	Description	
	AdministratorAccess Job function	on None	Provides full access to AWS services and	
	AlexaForBusinessD AWS man	aged None	Provide device setup access to AlexaFor	
	AlexaForBusinessF AWS man	aged None	Grants full access to AlexaForBusiness r	
	AlexaForBusinessG AWS man	aged None	Provide gateway execution access to Ale	
	AlexaForBusinessR AWS man	aged None	Provide read only access to AlexaForBusi	
	AmazonAPIGatewa AWS man	aged None	Provides full access to create/edit/delete	
	AmazonAPIGatewa AWS man	aged None	Provides full access to invoke APIs in Am	
	AmazonAPIGatewa AWS man	aged None	Allows API Gateway to push logs to user'	
	Set permissions boundary			
			Cancel Previous Next: Tags	-
🔍 Feedback 🔇 English (US)		© 2008	- 2019, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use	

Figure 2.2

8. Click Next. The user is created successfully.

2.2 Creating a VPC

Perform the below steps to create VPC (Virtual Private Cloud):

- 1. Sign in to the AWS Management Console using the root user and open the VPC in Services.
- 2. Select **Your VPC** and click Create VPC in the navigation pane.
- 3. Select **Resources** to create **VPC Only** in the Create VPC.
- 4. Specify the user-defined VPC name in the Name Tag field.
- 5. Specify the IPv4 CIDR block as 10.0.0/16 and click Create.

PC > Your VPCs > Create VPC	
Create VPC Info	
VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amaz	zon EC2 instances.
VPC settings	
Resources to create Info	
Create only the VPC resource or create VPC, subnets, etc.	
VPC only VPC, subnets, etc.	
Name tag - ontional	
Creates a tag with a key of 'Name' and a value that you specify.	
amit-vpc	
·	
IPv4 CIDR block Info	
 IPv4 CIDR manual input 	
 IPAM-allocated IPv4 CIDR block 	
IPv4 CIDR	
10.0.0/16	
IPv6 CIDR block Info	
No IPv6 CIDR block	
 IPAM-allocated IPv6 CIDR block 	
 Amazon-provided IPv6 CIDR block 	
 IPv6 CIDR owned by me 	
Tenancy, Info	
	-
Derault	•
Tags	
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional valu	e. You can use tags to search and filter
your resources or track your AWS costs.	
Key Value - optional	
Q Name X Q amit-vpc	× Remove
Add new tag	
You can add 49 more tags.	
	Cancel Create VPC

Figure 2.3

6. In the VPC, go to Action and click Edit DNS hostnames.

aws Services	Q Search for services, features, blogs, docs, and more [Alt+S]) 4 0) Mumbai ▼ amit @ 5923-1672-0
New VPC Experience Tell us what you think	Ø You successfully created vpc-09cc0215d4da16fbb / amit-vpc		×
VPC Dashboard EC2 Global View New	VPC > Your VPCs > vpc-09cc0215d4da16fbb		
Filter by VPC:	vpc-09cc0215d4da16fbb / amit-vpc		Actions 🔺
Q Select a VPC			Create flow log
VIRTUAL PRIVATE	Details Info		Edit CIDRs Edit DHCP options set
Your VPCs	VPC ID State DNS hostnames	DN	Edit DNS hostnames
Subnets	□ vpc-09cc0215d4da16fbb ② Available Disabled	Ena	Edit DNS resolution
Internet Gateways Egress Only Internet Gateways	Tenancy DHCP options set Main route table Default dopt-0e582a04e0e91efde rtb-0aa95049697d4b6e4 Default VPC IPvd CIDP IPv6 nool	Mai acl-	Manage middlebox routes Manage tags Delete VPC
DHCP Option Sets	No 10.0.0/16 -	-	, CIDIX
Elastic IPs Managed Prefix Lists Endpoints	Route 53 Resolver DNS Firewall Owner ID rule groups DS 592316720147		
Endpoint Services			

Figure 2.4

- 7. Select the Enable checkbox to enable DNS Hostnames.
- 8. Click Save Changes.

/PC > Your VPCs > vpc-09cc0215d4da16f	bb > Edit DNS hostnames		
Edit DNS hostnames Info			
DNS hostnames Indicates whether instances with public IP addresses	get corresponding public DNS hostnames.		
VPC ID D vpc-09cc0215d4da16fbb	DNS hostnames		
		Cancel	Save changes

Figure 2.5

2.3 Creating a subnets

It contains information on creating three subnets for the availability zone in the Mumbai region for High Availability.

Perform the below steps to create subnets:

- 1. In VPC Dashboard, go to the **Subnets**.
- 2. Go to the **Create Subnet** and select the created VPC using the dropdown.
- 3. In Subnet settings, specify the user-defined subnet name in the **Subnet name** field.
- 4. Select the **ap-south-1a** in the Availability Zone.
- 5. Specify the **10.0.1.0/24** in the IPv4 CIDR block.
- 6. Create two more subnets for other availability zones: **ap-south-1b** and **ap-south-1c** by clicking **Add new subnet**.
- 7. Click Create subnet.

VPC > Subnets > Create subnet
Create subnet Info
VPC
VPCID
vpc-09cc0215d4da16fbb (amit-vpc)
Associated VPC CIDRs
10.0.0/16
Subnet settings Sperify the CIDR blocks and Availability Zone for the submet.
Subnet 1 of 3
Subnet name Create a tag with a key of 'Name' and a value that you specify.
amit-subnet-1A
The name can be up to 256 characters long.
Availability Joine Timbert Choose the porter in which your subnet will reside, or let Amaron choose one for you.
изы Распс (Рилпон) / ар-зоцит-та 🛛 🔻
Q 10.0.1.0/24 X
▼ Tags - optional
Key Value - optional
Q, Name X Q, amit-subnet-1A X Remove
Add new tag You can add 40 more tags.
Remove
Subnet 2 of 3
Subnet name Create a tag with a key of 'Neme' and a value that you specify.
amit-subnet-18
The name can be up to 250 characters long. Availability Zone Infe
Choose the zone in which your subnet will reside, or let Amazon choose one for you. Acia Pacific (Mumbail / ao-south-1b
Pu4 CIDB block, tota
Q 10.0.2.0/24 X
▼ Tags - optional
Key Value - optional
You can add 40 more tags.
Remove
Subnet 3 of 3
Subnet name Create a tay with a key of 'Nama' and a value that you specify.
amit-subnet-1C The name can be up to 256 characters long.
Availability Zone Info
Choose one zone in tench your submet with inside, or set Amazon choose one for you. Adia Pacific (Mumbai) / ap-south-1c
IPv4 CIDR block Info
Q, 10.0.3.0/24 X
▼ Tags - optional
Name X Q, amit-subnet-IC X Remove
Add new tag You can add 40 more tags.
Remove
Add new subnet
Cancel Create subnet

Figure 2.6

- 8. After creating subnets, Edit all subnet settings.
- 9. Select one subnet.
- 10. Go to the **Action** and Edit **subnet settings**.

aws Services	Search for services, features, blogs, docs, and more [Alt+S]	⑦ Mumbai ▼ amit @ 5923-16	72-0147
New VPC Experience Tell us what you think	⊘ You have successfully created 3 subnets: subnet-04569865ba524020d, subnet-049ab9ea6a419f9e4, subnet-05595381b9	98be7c19	×
VPC Dashboard	Subnets (1/3) Info	Actions Create subnet	
EC2 Global View New	Q Filter subnets	View details	0
Filter by VPC:		Create flow log	
Q Select a VPC	Subnet ID: subnet-04569865ba524020d X Subnet ID: subnet-049ab9ea6a419f9e4 X	Edit subnet settings	
_ VIRTUAL PRIVATE	Subnet ID: subnet-05595381b98be7c19 X Clear filters	Edit IPv6 CIDRs	
CLOUD		Edit network ACL association	
Your VPCs	■ Name Subnet ID State VPC	Edit route table association	
Subnets	amit-subnet-1A subnet-04569865ba524020d 🥝 Available vpc-09cc0215d4da16fbb	Edit CIDR reservations	
Route Tables	amit-subnet-1C subnet-05595381b98be7c19 🤗 Available vpc-09cc0215d4da16fbb	Share subnet	
Internet Gateways	□ amit-subnet-1B subnet-049ab9ea6a419f9e4 ⊘Available vpc-09cc0215d4da16fbb	Manage tags	*
Egress Only Internet	4	Dalata subart	•
Gateways	=		
DHCP Option Sets	subnet-04569865ba524020d / amit-subnet-1A		-
Elastic IPs			
Managed Prefix Lists	Details Flow logs Route table Network ACL CIDR reservations Sharing Tags		
Endpoints			•
Endpoint Services			

Figure 2.7

- 11. In the Auto-assign IP settings, Enable auto-assign public IPv4 address.
- 12. Click **Save** to save subnet settings.

20d 🗦 Edit subnet settings
Name
amit-subnet-1A
quest a public IPv4 or IPv6 address for a new network interface in this subnet.
fo
2

Figure 2.8

2.4 Creating an internet gateway

This section explains how to create an Internet Gateway for the **public Route Table**. The creation of the Route Table is described in the section <u>create a route table</u>.

NOTE:

To use a private Route Table, you must create Nat Gateway which cost up to \$40.

Perform the below steps to create an Internet Gateway:

- 1. In VPC Dashboard, go to the Internet Gateways and click Create internet gateway.
- 2. Specify the user-defined name in the Name tag field and click Create.

aws Services - Resource Groups - +	۵	vivek @ 5105-0170-3457 🗸	Mumbai 👻	Support 👻
Internet gateways > Create internet gateway				
Create internet gateway				
An internet gateway is a virtual router that connects a VPC to the internet. To create a new internet gateway specify the name for the gateway below.				
Name tag vivek_internet_gateway				
* Required		Cancel Create		
Feedback 😧 English (US) 0 2008 - 2019, Amazon Internet Services	s Private I	.td. or its affiliates. All rights reserved.	Privacy Polic	y Terms of Use



- 3. Select the created internet gateway.
- 4. Select the Attach to VPC option in the Actions menu.
- 5. Select the created VPC and click **Attach**.

aws ser	rices 🗸 Resource Groups 🗸 🛧	۵	vivek @ 5105-0170-3457 🗸	Mumbai 👻	Support 👻
Internet gateways > Atta	ch to VPC				
Attach to VP	C				
Attach an internet gatewa	y to a VPC to enable communication with the internet. Specify the VPC you would like to attach below.				
VPC*	vpc-002069cbb4dd53364 🗸 🗸				
AWS Command Line	Interface command				
* Required			Cancel Attach		

Figure 2.10

2.5 Creating a route table

Perform the below steps to create the Route table:

- 1. In VPC Dashboard, go to the Route Tables and click Create route tables.
- 2. Specify the user-defined route table name in the Name tag field.
- 3. Select the created VPC and click Create.

aws Services - Res	source Groups 🗸 🔸		۵	vivek @ 5105-0170-3457 🕶	Mumbai 👻
Route Tables > Create route table					
Create route table					
A route table specifies how packets are for	warded between the subnets within your VPC, the intern	et, and your VPN connection.			
Name tag	vivek_route_table	0			
VPC*	vpc-002069cbb4dd53364	C0			
* Required				Cancel Create	



- 4. Select the created route table.
- 5. Go to the Routes tab and click Edit Route.
- 6. To provide internet access to a created route table, add a **new route** and specify **0.0.0/0** in the **Destination** field.
- 7. Select the created Internet gateway in the Target field.
- 8. Click Save routes.
- 9. Select the created route table.
- 10. Go to Subnet Associations and click Edit subnet associations.
- 11. Select the created subnets for all availability zones for the Mumbai region.

aws Services - Res	ource Groups 🗸 🚯	۵	vivek @ 5105-0170-3457 👻	Mumba
Route Tables > Edit subnet associations				
Edit subnet associatio	ns			
Bouto tobio	4b 02b0407794bf79a65 (uluulu asuto tabla)			
Koule lable				
Associated subnets	subnet-065fc1d0079ee9286 subnet-02b4e1422db90c20b subnet-0334d5a8e34d15ec8			
			٥	
	Q Filter by attributes or search by keyword	K	< 1 to 3 of 3 > >	
	Subnet ID VIPv4 CIDR VIPv6 CIDR Current Route 1	Table		
	subnet-065fc1d0079ee9286 vivek_sub 10.0.3.0/24 - Main			
	subnet-0334d5a8e34d15ec8 vivek_sub 10.0.1.0/24 - Main			
* Required			Gancel Save	
Second Address Proc			Sale Sale	

Figure 2.12

2.6 Creating an IAM role

Before creating a Kubernetes Cluster, you must create an IAM role that Kubernetes can assume to create AWS resources.

For example, when a load balancer is created, Kubernetes assumes the role to create an Elastic Load Balancing load balancer in your account. You can create this one time only and can be used for multiple EKS clusters.

Perform the below steps to create an IAM Role:

- 1. Go to IAM Dashboard.
- 2. Go to the **Roles** and click **Create role**.
- 3. Select **EKS** from the list of services.
- 4. Select EKS Cluster to Allows access to other AWS service resources that are required to operate clusters managed by EKS for your use case.
- 5. Click Next: Permissions.

Sele	ct your use case
EKS Allows I	EKS to manage clusters on your behalf.
EKS - C	Cluster
Allows a	access to other AWS service resources that are required to operate clusters managed by EKS.
EKS - C	Connector
Allows a	access to other AWS service resources that are required to connect to external clusters
EKS - F	Fargate pod
Allows a	access to other AWS service resources that are required to run Amazon EKS pods on AWS Fargate.
EKS - F	Fargate profile
Allows I	EKS to run Fargate tasks.
EKS - N	Nodegroup
Allow E	IKS to manage nodegroups on your behalf.

Figure 2.13

- 6. Click Next: Tags.
- 7. Click the Next: Review.
- 8. Specify the user-defined role name given under review and then click Create role.

aws	Services	Q Search for services, features, blogs, docs, and more	[Alt+S]	\$ Ø	Global 🔻 vi	vek.kumar @ 2731-6361-0351 🔻
		Create role		1 2	3 4	
		Review				
		Provide the required information below and review	this role before you create it.			
		Role name*	vivek_role1			
			Use alphanumeric and '+=,.@' characters. Maximum 64 character	ters.		
		Role description	Allows access to other AWS service resources that are EKS.	e required to operate clusters ma	anaged by	
			Maximum 1000 characters. Use alphanumeric and '+=,.@' chara	acters.		
		Trusted entities	AWS service: eks.amazonaws.com			
		Policies	AmazonEKSClusterPolicy			
		Permissions boundary	Permissions boundary is not set			
		No tags were added.				
		* Required		Cancel Previous	Create role	
Feedbad	k English (US) י	,	© 2022, An	mazon Web Services, Inc. or its affilia	ates. Privacy	Terms Cookie preferences

Figure 2.14

2.7 Creating a security group

Creating a security group is required for the EKS cluster.

Perform the below steps to create a Security Group:

- 1. In the VPC Dashboard, go to the Security Groups and click Create security group.
- 2. On the Create security group tab, specify the user-defined security group name and description.
- 3. Select the created VPC and click **Create**.

2.8 Creating an EKS cluster

Before creating the EKS Cluster, you must sign in to the AWS Management Console using an IAM user instead of using a root user for EKS Cluster creation.

Perform the below steps to create an AWS Kubernetes Cluster:

1. Go to EKS Service and click the Next step.





- 2. Specify the following in the Create Cluster:
 - **Cluster name**: Enter the User-defined name.
 - Kubernetes version: Select default that is, 1.21.
 - Role name: Select the created IAM role.
- 3. Click Next.

onfigure cluster	
Cluster configuration Info	
Name - Not editable after creation. Enter a unique name for this cluster.	
EKSCluster	
Kubernetes version Info	
Select the Kubernetes version for this cluster.	
1.21	
Cluster Service Role Info - Not editable after creation.	
Select the IAM Role to allow the Kubernetes control plane to manage AWS To create a new role, go to the IAM console.	S resources on your behalf.

Figure 2.16

- 4. Specify the following in the Networking info:
 - VPC: Select the created VPC.
 - Subnets: Select all the subnets of the Mumbai region.
 - Security groups: Select the created security group.
 - Cluster endpoint access: Enable both Private access and Public access.
- 5. Once all the details are specified, click **Next**.

VPC Info Select a VPC to use for your EKS Cluster re- To create a new VPC, go to the VPC consol	ources.
vpc-002069cbb4dd53364 vivek_v	pc 🔻 🖸
Subnets Info Choose the subnets in your VPC where the Fo create a new subnet, go to the correspo	control plane may place elastic network interfaces (ENIs) to facilitate communication with your clus nding page in the VPC console.
Select subnets	• C
subnet-065fc1d0079ee9286 🗙	subnet-02b4e1422db90c20b 🗙
subnet-0334d5a8e34d15ec8 🗙	
Security groups Info Choose the security groups to apply to the fo create a new security group, go to the o Select security groups	EKS-managed Elastic Network Interfaces that are created in your worker node subnets. orresponding page in the VPC console.
country groups Info hoose the security groups to apply to the o create a new security group, go to the o Select security groups sg-049392491607b5e43 X	EKS-managed Elastic Network Interfaces that are created in your worker node subnets. presponding page in the VPC console.
Security groups Info Choose the security groups to apply to the for create a new security group, go to the or Select security groups sg-049392491607b5e43 X	EKS-managed Elastic Network Interfaces that are created in your worker node subnets. orresponding page in the VPC console.
Security groups Info Choose the security groups to apply to the or reate a new security group, go to the o Select security groups sg-049392491607b5e43 × Sg-049392491607b5e43 × Cluster endpoint access Info Configure access to the Kubernetes API ser Public	EKS-managed Elastic Network Interfaces that are created in your worker node subnets. prresponding page in the VPC console.
Security groups Info Choose the security groups to apply to the or reate a new security group, go to the Select security groups sg-0493992491607b5e43 X Sg-0493992491607b5e43 X Cluster endpoint access Info Configure access to the Kubernetes API ser Public The duster endpoint is accessible from Public and private	EKS-managed Elastic Network Interfaces that are created in your worker node subnets. prresponding page in the VPC console. wer endpoint. outside of your VPC. Worker node traffic will leave your VPC to connect to the endpoint.
Security groups Info Choose the security groups to apply to the or serve a new security group, go to the Select security groups of the sg-049392491607b5c43 X Cluster endpoint access Info Configure access to the Kubernetes API ser Public The cluster endpoint is accessible from Public endprivate The cluster endpoint is accessible from	EKS-managed Elastic Network Interfaces that are created in your worker node subnets.
Security groups Info Choose the security groups to apply to the or create a new security group to the Select security groups sg-049392491607b5e43 X Cluster endpoint access Info Configure access to the Kubernetes API ser Public The duster endpoint is accessible from Public and private The duster endpoint is accessible from Pirvate	EKS-managed Elastic Network Interfaces that are created in your worker node subnets. Interpretending page in the VPC console. Interpretending page page page page page page page pag

Figure 2.17

- 6. On the Configure tab, click **Next**.
- 7. In the Review tab, click **Create** and create a page.

NOTE:

Ensure that the cluster status is **ACTIVE**.

2.9 Creating a key pair

Perform the below steps to create a key pair:

- 1. Go to EC2 Dashboard.
- 2. Click Key Pairs.
- 3. Click Create Key Pair. The Create Key Pair dialog appears.
- 4. Specify the Key pair name and click Create. A <KeyPair Name>.pem gets downloaded.

NOTE:

For the SSH connection, you must keep the key pair name. Convert this *.pem* file to *.ppk* for SSH connection through the local machine.

2.10 Provisioning Kubernetes worker nodes using cloud formation

Perform the below steps to provide provision Kubernetes worker nodes using Cloud Formation:

 Download the latest version of the AWS CloudFormation template. curl -o amazon-eks-nodegroup.yaml <u>https://raw.githubusercontent.com/awslabs/amazon-eks-</u> ami/master/amazon-eks-nodegroup.yaml

NOTE:

To download the latest YAML file, refer to the below link: https://docs.aws.amazon.com/eks/latest/userguide/launch-workers.html

- 2. Open the AWS CloudFormation console.
- 3. Go to Create Stack under With new resources (standard).
- 4. To **Specify a template**, select **Upload a template file** and then select **Select** file. Select the *amazon-eks-nodegroup.yaml* file that you downloaded earlier and then click **Next**.

Prerequisite - Prepare template	
Prepare template Every stack is based on a template. A template is a JSON Template is ready	or YAML file that contains configuration information about the AWS resources you want to include in the stack. Use a sample template Create template in Designer
Specify template	
Template source Selecting a template generates an Amazon S3 URL when	it will be stored.
Template source Selecting a template generates an Amazon 53 URL when Amazon 53 URL Upload a template file Choose file Amazon amazon-eks-nodegroup, JSON or YAML formatted file	it will be stored.

Figure 2.18

- 5. On the Specify stack details, specify the following details:
 - Stack name Select a stack name for the AWS CloudFormation stack.
 - **ClusterName** Enter the name used for creating the Amazon EKS cluster. This name must exactly match the name as per the given name.
 - **ClusterControlPlaneSecurityGroup** Select the SecurityGroups of EKS Cluster.

- **NodeGroupName** Enter a name for your node group. Later, this name identifies the Auto Scaling node group created for your worker nodes.
- **NodeAutoScalingGroupMinSize** Enter the minimum number of nodes that your worker node Auto Scaling group can scale in them.
- NodeAutoScalingGroupDesiredCapacity Enter the desired number of nodes to scale your created stack.
- **NodeAutoScalingGroupMaxSize** Enter the maximum number of nodes that your worker node Auto Scaling group can scale out in them.
- **NodeInstanceType** Select an instance type for your worker nodes.
- NodelmageldSSMParam This is a pre-populated optimized Amazon Linux AMI ID for a Kubernetes version. Change the Kubernetes minor version supported with EKS Cluster. For example, earlier you had created an EKS Cluster with v1.21. You must use the same version here as shown below:

/aws/service/eks/optimized-ami/1.21/amazon-linux-2/recommended/image_id

- **NodeImageId** This is an optional field. If you are using your custom AMI, then enter a node AMI ID otherwise leave it blank.
- **NodeVolumeSize** Specify a node volume size for your nodes, in GiB.
- **KeyName** Enter the name of an Amazon EC2 SSH key pair that you can use to connect using SSH into your worker nodes after they launch.
- 6. After specifying the above details, click **Next**.

pecify stack details		
Stack name		
Stack name		
EKSClusterStack		
Stack name can include letters (A-Z and a-z), numbers (0-9), and dashes (-).		
Parameters Parameters are defined in your template and allow you to inout custom values when you create or undate a stack.		
EKS Cluster		
Lustername The cluster name provided when the cluster was created. If it is incorrect, nodes will not be able to join the cluster.		
EKSCluster		
ClusterControlPlaneSecurityGroup The security group of the cluster control plane.		
sg-0c47be37edb8fb278		•
Worker Node Configuration		
NodeGroupName		
Jnique identifier for the Node Group.		
VedeAutoCralingGroupMinGing		
VodeAutoScalingGroupMinSize Minimum size of Node Group ASG.		
2		
NodeAutoScalingGroupDesiredCapacity Desired capacity of Node Group ASG.		
2		
NodeAutoScalingGroupMaxSize		
Maximum size of Node Group ASG. Set to at least 1 greater than NodeAutoScalingGroupDesiredCapacity.		
NodelnstanceTyne		
EC2 instance type for the node instances		
t3.medium		
NodelmageldSSMParam		
WVS systems Manager Parameter Store parameter or the APH ID for the worker node instances. Lhange this value to match the version of /aws/service/eks/optimized-ami/1.23/amazon-linux-2/recommended/image_id	r Kubernetes you ar	e using.
vodeimageid Optional) Specify your own custom image ID. This value overrides any AWS Systems Manager Parameter Store value specified above.		
Enter String		
NodeVolumeSize		
100		
NodeVolumeType		
EBS volume type for nodes		
942		
KeyName The EC2 Key Pair to allow SSH access to the instances		
ekscluster		•
BootstrapArguments		
Arguments to pass to the bootstrap script. See files/bootstrap.sh in https://github.com/awslabs/amazon-eks-ami Enter String		
false		•
Worker Network Configuration		
vpcia The VPC of the worker instances		
vpc-09cc0215d4da16fbb		•
Subnets The subnets where workers can be created		
Select List <aws::ec2::subnet:id></aws::ec2::subnet:id>		
subnet-04569865ba524020d X subnet-05595381b98be7c19 X subnet-049ab9ea6a419f9e4 X		

Figure 2.19

- 7. Click **Next** and configure stack options.
- 8. In the Review tab, review all the specified details and select the checkbox I acknowledge that AWS CloudFormation might create IAM resources.
- 9. Click Create stack. Ensure that the creation status becomes CREATE_COMPLETE now.

) The following resource(s) require capabilities: [AWS::IAM::Role]			
This template contains Identity and Access Management (IAM) resources that might Check that you want to create each of these resources and that they have the minim	provide entities a	ccess to make changes to yo hissions. Learn more	ur AWS account.
since that you many to create each of anote resources and that they have the minim	ann reganed pern		
✓ I acknowledge that AWS CloudFormation might create IAM resources.			
I acknowledge that AWS CloudFormation might create IAM resources.	10 <u>2</u> 0	Par. 0	
✓ I acknowledge that AWS CloudFormation might create IAM resources.	Brenieure	Contratory and	

NOTE:

Once your stack creation is complete, select it on the console, and select the Outputs tab. Record the **NodeInstanceRole** for the node group created earlier. Keep this NodeInstanceRole to configure your Amazon EKS worker nodes.

2.11 Adding inbound rule in EC2 instance

Perform the below steps to add the Inbound rule in an EC2 instance:

- 1. Go to VPC Dashboard.
- 2. Select **Security Groups** and select the security group mapped with EKS Node/EC2 Instance.
- 3. Go to the Inbound Rules tab and click Edit rules.
- 4. Click Add a new Rule and specify 22 in the Port Range field.
- 5. Select **My IP** or any other IP range from where you want to access the ssh of worker nodes in the Source field.

	Port Range (i)	Source (i)		Description (i)	
All	0 - 65535	Custom •	sg-012d5121ea84105de	Allow node to communic	۲
TCP	1025 - 65535	Custom •	sg-00f6101793274434f	Allow worker Kubelets	8
TCP	443	Custom •	sg-00f6101793274434f	Allow pods running ext	8
TCP	22	My IP 🔹	115.111.224.194/32	e.g. SSH for Admin Desktop	8
	All TCP TCP TCP	All 0 - 65535 TCP 1025 - 65535 TCP 443 TCP 22	All 0 - 65535 Custom • TCP 1025 - 65535 Custom • TCP 443 Custom • TCP 22 My IP •	All 0 - 65535 Custom sg-012d5121ea84105de TCP 1025 - 65535 Custom sg-00f6101793274434f TCP 443 Custom sg-00f6101793274434f TCP 22 My IP 115.111.224.194/32	All 0 - 65535 Custom sg-012d5121ea84105de Allow node to communic TCP 1025 - 65535 Custom sg-0016101793274434f Allow worker Kubelets TCP 443 Custom sg-0016101793274434f Allow pods running ext TCP 22 My IP 115.111.224.194/32 e.g. SSH for Admin Desktop

Figure 2.21

6. Connect this EC2 instance from the local machine using the default user name **ec2-user** and SSH key pair as created earlier.



Figure 2.22

2.12 Enabling worker node to join EKS cluster

Perform the following to enable the Worker Node/EC2 instance to EKS Cluster:

- 1. Connect to the worker node through the Putty.
- 2. Install or Update AWS Cli on the worker node by using the below URL: https://docs.aws.amazon.com/cli/latest/userguide/getting-started-install.html

NOTE:

AWS cli version 2.x is required. Check the AWS cli version using the below command: aws --version

- 3. Install the Kubectl on the worker node by using the below URL: <u>https://docs.aws.amazon.com/eks/latest/userguide/install-kubectl.html</u>
- 4. Execute the below command to the worker node: aws configure
- 5. After the above command is executed, specify the following details:
 - Access Key ID: Provide the Access key ID of the user which is used to create the EKS Cluster.
 - Secret Access Key: Provide the SecretKey ID of the user which is used to create the EKS Cluster.
 - **Region**: ap-south-1
 - Output: JSON

P ec2-user@ip-10-0-2-166:~	_	\times
<pre>[ec2-user@ip-10-0-2-166 ~]\$ aws configure AWS Access Key ID [None]:</pre>		^



6. Now, execute the below command:

aws eks --region <RegionName> update-kubeconfig --name <ClusterName>

For example,

aws eks --region ap-south-1 update-kubeconfig --name EKSCluster

7. To download the configuration map, execute the below command:

curl -o aws-auth-cm.yaml <u>https://amazon-eks.s3.us-west-</u> 2.amazonaws.com/cloudformation/2020-10-29/aws-auth-cm.yaml

NOTE:

To download the latest S3 URL, refer to the below link: https://docs.aws.amazon.com/eks/latest/userguide/launch-workers.html

- 8. After the above command gets executed, a file *aws-auth-cm.yaml* is downloaded to the worker node.
- 9. Open this file in the edit mode and replace the <ARN of instance role (not instance profile)> snippet with the **NodeInstanceRole** value recorded in the procedure.
- 10. Save the file.



Figure 2.24

11. Execute the below command on the worker node.

kubectl apply -f aws-auth-cm.yaml

Check the status of your nodes and wait for them to reach the Ready status.

kubectl get nodes --watch

2.13 Running Kubectl from a local machine

Before running the kubectl commands from your local machine, ensure that you have the following prerequisites:

- kubectl: https://docs.aws.amazon.com/eks/latest/userguide/install-kubectl.html.
- aws-cli: <u>https://docs.aws.amazon.com/cli/latest/userguide/cli-chap-install.html.</u>
- Execute the below command to the worker node: aws configure
- Once the above command gets executed, specify the following details:
 - Access Key ID: Provide the Access key ID of the user which is used to create the EKS Cluster.
 - Secret Access Key: Provide the SecretKey ID of the user which is used to create the EKS Cluster.
 - Region: ap-south-1
 - output: JSON
- Execute the below command: aws eks --region <RegionName> update-kubeconfig --name <Cluster_Name>

```
For example,
aws eks --region ap-south-1 update-kubeconfig --name EKSCluster
```

• Execute kubectl commands from our machine.

```
For example, kubectl get pods
```

2.14 Creating EFS

Perform the below steps to create an AWS Elastic File System (EFS) storage:

- 1. Go to EFS Service and select Create file system.
- 2. In the **Name optional**, specify the user-defined name for your file system that is, **omnidocs-efs**.
- 3. In the Virtual Private Cloud (VPC), select the created VNC for your EKS cluster.
- 4. In the Availability and Durability, select Regional.
- 5. Click Create.

Create file system	>	<
Create an EFS file system with serv	ice recommended settings. Learn more 🖸	
Name - optional Name your file system.		
omnidocs-efs		
Name must not be longer than 256 characters: + - = : /	acters, and must only contain letters, numbers, and these	
Virtual Private Cloud (VPC) Choose the VPC where you want EC2 inst vpc-0205fcf47041ec2df	tances to connect to your file system. Learn more 🔀	
OmnidocsUATVPC	·	
Availability and Durability Choose Regional (recommended) to crea Zone to create a file system using One Zo	te a file system using regional storage classes. Choose One one storage classes. Learn more 🏹	
• Regional Stores data redundantly across multiple AZs	One Zone Stores data redundantly within a single AZ	
	Cancel Customize Create	
	Figure 2.25	

6. Open the created EFS and switch to the **Access Point** tab and select **Create access point**.

aws Services v	Q Search for services, features, marketplace products, and docs	[Alt+S] D A [®] omnidocs@ngdev ▼ Mumbai ▼ Support ▼
Elastic File System $\qquad imes$	omnidocs-efs (fs-df1d280e)	Delete Attach
File systems Access points	General	Edit
AWS Backup 🖸 AWS DataSync 🖸 AWS Transfer 💈 Documentation 💈	Performance mode General Purpose Throughput mode Bursting Lifecycle policy 30 days since last access Availability zone Regional Metered size Monitoring Tags File system policy Access	Automatic backups
	Access points (0) Q Search access points by name or ID Name Access point ID	C View details Delete Create access point <
Feedback English (US) 🔻	© 2008 - 2021, Amaz	No resources on Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

Figure 2.26

- 7. In the Name optional, specify the user-defined name that is, omnidocs-efs-accesspoint.
- 8. In the **Root directory path**, use **/mnt/efs** [This directory path must exist on EC2 worker nodes, else, create it if does not exist].

- 9. In the **POSIX user –optional**, specify **1000** in the User ID, Group ID, and Secondary group IDs textboxes.
- 10. In the **Root directory creation permissions optional**, specify **1000** in the Owner user ID and Owner group ID. And keep the default POSIX permission **0755**.
- 11. Click Create access point.

NOTE:

The Worker node's Security group must be added to the EFS Network.

shared datasets. Learn more	entry point most of a state system and money is concered intended, appreciation acce
Details	
Details	
File system Choose the file system to which your access	point is associated.
Q, fs-082ceed3f363f7872	
Name - optional	
Maximum of 256 Linicode letters, whitespare	e and numbers plus + = = - /
Dent directory with antional	
Connections use the specified path as the file	e system's virtual root directory Learn more 🔀
/mnt/efs	
Example: "/foo/bar"	
POSIX user - optional The full POSIX identity on the access point th	hat is used for all file operations by NFS clients. Learn more 🗹
User ID	
POSIX user ID used for all file system operati	ions using this access point.
Accepts values from 0 to 4294967295	
Group ID	
POSIX group ID used for all file system opera	ations using this access point.
1000	
Accepts values from 0 to 4294967295	
Secondary group IDs Secondary POSIX group IDs used for all file s	system operations using this access point.
1000	
A comma-separated list of valid POSIX group	a IDs
Root directory creation perm	nissions - optional ot directory with these permissions if the directory does not already exist. Learn more P
Owner user ID Owner user ID for the access point's root dire	actory, If the directory does not already exist.
1000	
Accepts values from 0 to 4294967295	
Owner group ID	
owner group ID for the access point's root di	rectory, if the directory does not already exist.
Accepts values from 0 to 4294967295	
POSIX nermissions to products the each disc	tory path
755	and beau
An octal number representing the file's mode	e bits.
• Tags - optional	
Add tags to associate key-value pairs t	to your resource. Learn more 🔀
The last	Technical antipad
ray key	Tag value - optional
O Enter key	Kemove tag
Q Enter key	
Q Enter key Add tag	

Figure 2.27

2.15 Mounting EFS to worker nodes

For mounting the Elastic File System (EFS) with Worker nodes, install the **amazon-efs-utils** using the below command:

sudo yum install -y amazon-efs-utils

Add the below line to the */etc/fstab* file on each Worker Node:

```
fs-8241f853.efs.ap-south-1.amazonaws.com:/ /mnt/efs efs
_netdev,tls,accesspoint=fsap-0bbac155fbd3ad350 0 0
```

Where,

fs-8241f853.efs.ap-south-1.amazonaws.com =Elastic File System DNS Name /mnt/efs=Existing directory structure of EC2 instance [Create if does not exist] fsap-0bbac155fbd3ad350= Attached Access Point to the EFS

Execute the below command:

sudo mount -a

NOTE:

You must mount EFS to all the running worker nodes.

2.16 Configuring Kubernetes dashboard

Use the below URL to configure the Kubernetes Dashboard: <u>https://docs.aws.amazon.com/eks/latest/userguide/dashboard-tutorial.html</u>

Once Kubernetes Dashboard is configured, execute the below command:

kubectl proxy

Use the below URL to open the Kubernetes Dashboard: <u>http://localhost:8001/api/v1/namespaces/kubernetes-dashboard/services/https:kubernetes-dashboard:/proxy/#!/login</u>

2.17 Configuring AWS load balancer controller

The AWS Load Balancer Controller manages AWS Elastic Load Balancers for a Kubernetes cluster. It creates an application load balancer when you create a Kubernetes ingress. The Ingress resource configures the ALB to route HTTP or HTTPS traffic to different pods within the cluster.

Perform the below steps to configure the AWS Load Balancer Controller:

1. Create an IAM OIDC provider and associate it with your cluster using the below commands:

```
eksctl utils associate-iam-oidc-provider \
    --region <region-code> \
    --cluster <cluster-name> \
    --approve
```

NOTE:

If you don't have the eksctl version 0.25.0 or later installed, then complete the installation using the below URL: <u>https://docs.aws.amazon.com/eks/latest/userguide/eksctl.html#installing-eksctl</u>

2. Download an IAM policy for the AWS Load Balancer Controller that allows it to make calls to AWS APIs on your behalf using the below command:

curl -o iam policy.json https://raw.githubusercontent.com/kubernetes-sigs/awsload-balancer-controller/v2.3.1/docs/install/iam policy.json

NOTE:

To get the latest iam-policy, refer to the below link: <u>https://docs.aws.amazon.com/eks/latest/userguide/aws-load-balancer-controller.html</u>

 Create an IAM policy called AWSLoadBalancerControllerIAMPolicy using the downloaded policy.

aws iam create-policy --policy-name AWSLoadBalancerControllerIAMPolicy -- policy-document file://iam policy.json

4. Create an IAM role for the AWS Load Balancer Controller and attach the role to the service account created in the further steps.

Perform the below steps to create the IAM role:

- 1. Open the IAM console and select Create Roles.
- 2. In the Select type of trusted entity section, select Web identity.
- 3. In the Select a web identity provider, specify the following:
 - i. In the **Identity provider**, select the URL for your cluster.
 - ii. In the **Audience**, select *sts.amazonaws.com*.
 - iii. Click Next: Permissions.
- 4. In the Attach Policy section, select the policy AWSLoadBalancerControllerIAMPolicy
- 5. Specify the role name as AmazonEKSLoadBalancerControllerRole and then select Create Role.
- 6. After the role is created, select the role in the console to open it for editing.
- 7. Select the **Trust relationships** tab and select the **Edit trust policy**.
 - i. Edit the OIDC provider suffix and change it from **aud** to: **sub**.
 - ii. Replace *sts.amazonaws.com* with the service account ID given in the quotes below. (This service account id is created in further steps).

"system:serviceaccount:kube-system:aws-load-balancer-controller"

iii. The resulting line in policy must be as follows:

```
"oidc.eks.region-
code.amazonaws.com/id/EXAMPLED539D4633E53DE1B716D3041E:sub":
"system:serviceaccount:SERVICE ACCOUNT NAMESPACE:SERVICE ACCOUNT NAME"
```

For example,

```
"oidc.eks.ap-south-
1.amazonaws.com/id/C9D4F2E6E31D3880DCE2BEFEA007C4CB:sub": "
"system:serviceaccount:kube-system:aws-load-balancer-controller"
```

iv. Select Update policy to finish.

NOTE:

Take note of the Role ARN of the newly created role AmazonEKSLoadBalancerControllerRole.

 Create a Kubernetes service account named aws-load-balancer-controller in the kube-system namespace. To create the same, save the following contents to a file that's named as aws-loadbalancer-controller-service-account.yaml, replacing the created role ARN.

```
apiVersion: v1
kind: ServiceAccount
metadata:
    labels:
        app.kubernetes.io/component: controller
        app.kubernetes.io/name: aws-load-balancer-controller
        name: aws-load-balancer-controller
        namespace: kube-system
        annotations:
            eks.amazonaws.com/role-arn:
arn:aws:iam::273163610351:role/AmazonEKSLoadBalancerControllerRole
```

NOTE:

To download the latest service account YAML contents, refer to the below link: <u>https://docs.aws.amazon.com/eks/latest/userguide/aws-load-balancer-controller.html</u>

9. Execute the following command to create a Kubernetes Service Account:

kubectl apply -f aws-load-balancer-controller-service-account.yaml

10. Install the cert-manager using the following command:

kubectl apply --validate=false -f https://github.com/jetstack/certmanager/releases/download/v1.5.4/cert-manager.yaml

11. Download the controller specification using the below command:

```
curl -Lo v2_4_1_full.yaml https://github.com/kubernetes-sigs/aws-load-
balancer-controller/releases/download/v2.4.1/v2 4 1 full.yaml
```

- 12. Make the following edit in v2_4_1_full.yaml file.
 - a. Delete the kind: ServiceAccount section of the file.
 - b. Replace **your-cluster-name** in the **Deployment spec** section of the file with the name of your cluster.

For example,

c. Apply the file.

kubectl apply -f v2_4_1_full.yaml

NOTE:

If a user is facing issues like no matches for kind **IngressClassParams** in version **elbv2.k8s.aws/v1beta1** then execute the below command to fix this issue:

```
sudo yum install git -y
kubectl apply -k "github.com/aws/eks-charts/stable/aws-load-balancer-
controller//crds?ref=master"
kubectl apply -f v2_4_1_full.yaml
```

13. Use the below command to verify the status of the AWS Load Balancer Controller:

kubectl get deployment -n kube-system aws-load-balancer-controller

14. Use the below command to check the logs of the AWS Load Balancer Controller:

kubectl logs deployment.apps/aws-load-balancer-controller -n kube-system

2.18 Configuring AWS Elastic Redis Cache

To configure the AWS Elastic Redis, perform the below:

- 1. Sign in to the AWS Management Console and open the ElastiCache console.
- 2. Select Get Started Now. If you already have an available cluster, select Create.
- 3. For the Cluster engine, select Redis.
- 4. Complete the Redis settings section as follows:
 - i. Cluster creation method Configure and create a new cluster.
 - ii. Cluster mode Disabled.



Figure 2.28

- iii. Name Enter a user-defined name.
- iv. **Description (optional)** Enter any description.
- v. Location AWS Cloud
- vi. Multi-AZ Enabled

newgen-redis-cache	
The name is required, can have up to 40 characters, and must be consecutive hyphens. Valid characters: A-Z, a-z, 0-9, and - (hyphe	rgin with a letter. It should not end with a hyphen or contain two en).
Description - optional	
newgen-redis-cache	
ocation	
Location	iises.
.ocation hoose whether to host the cluster in the AWS Cloud or on prem	iises.
.ocation Choose whether to host the cluster in the AWS Cloud or on prem	nises.
Location	iises.
Location Choose whether to host the cluster in the AWS Cloud or on prem	On premises
Location Choose whether to host the cluster in the AWS Cloud or on prem Location AWS Cloud Use the AWS Cloud for your ElastiCache instances.	nises. On premises Create your ElastiCache instances on an Outpost (through AWS Outposts). You need to create a subnet ID on an
Location Choose whether to host the cluster in the AWS Cloud or on prem- Location AWS Cloud Use the AWS Cloud for your ElastiCache instances.	nises. On premises Create your ElastiCache instances on an Outpost (through AWS Outposts). You need to create a subnet ID on an Outpost first.
Location Choose whether to host the cluster in the AWS Cloud or on premised Location AWS Cloud Use the AWS Cloud for your ElastiCache instances.	nises. On premises Create your ElastiCache instances on an Outpost (through AWS Outposts). You need to create a subnet ID on an Outpost first.
Location Location Location AWS Cloud Use the AWS Cloud for your ElastiCache instances. Aulti-AZ	nises. On premises Create your ElastiCache instances on an Outpost (through AWS Outposts). You need to create a subnet ID on an Outpost first.
Location Choose whether to host the cluster in the AWS Cloud or on prem Location AWS Cloud Use the AWS Cloud for your ElastiCache instances. Multi-AZ	 On premises Create your ElastiCache instances on an Outpost (through AWS Outposts). You need to create a subnet ID on an Outpost first.

Figure 2.29

- vii. Engine version Compatibility Select the latest version.
- viii. **Port** Accept the default port.
- ix. **Parameter group** Accept the default parameter group.
- x. Node type: Select the node type that you want to use for this cluster.
- xi. **Number of replicas** Select the number of nodes you want to provide as a provision for this this cluster.

Engine version Version compatibility	of the Redis engine that will run on your nodes.
6.2	•
Port	
The port number that	nodes accept connections on.
6379	
default.redis6.x	▼
Node type	
The type of node to be	e deployed and its associated memory size.
cache.r5.large	_
	Up to 10 Gigabit network performance
13.07 GiB memory	
13.07 GiB memory	
13.07 GiB memory	



xii. Create subnet group

- a. Name Type a unique name
- b. **Description** Type any description.
- c. **VPC ID** Select a VPC in which EKS cluster is created.
- d. **Subnets** Select all subnets.
- e. Availability zones placement: Accept as default that is, No Preference.
- f. Click Next.

Subnet group settings A subnet group is a collection of subnets (typi Cloud (VPC) environment.	ically private). Designate a subnet group for your clu	isters running in an Amazon Virtual Private
Subnet groups O Choose existing subnet group	• Create a new su	ibnet group
Name newgen-redis-subnet-group The name is required, can have up to 255 char consecutive hyphens. Valid characters: A-7, a-	racters, and must begin with a letter. It should not o	nd with a hyphen or contain two
Description - <i>optional</i>		
VPC ID The identifier for the VPC environment where vpc-0205fcf47041ec2df For Multi-AZ high availability me table below.	e your cluster is to run. ode, choose IDs for at least two subnets fror	Create VPC
Selected subnets (2)		Manage
Availability Zone	Subnet ID	\bigtriangledown CIDR block \bigtriangledown
ap-south-1a	subnet-0048aa992f3464134	10.0.2.0/24
ap-south-1b	subnet-0defa4d35f35b9415	10.0.1.0/24
Tags for subnet group		
Availability Zone placements Use the following fields to configure placeme	ents for Availability Zones.	
Availability Zone placements HA mode - Globally, distribute AZs to maximi for within-shard HA. Low latency mode - For No preference	ize AZ spread across shard masters. At the second le fast writes, put all shard masters in the same AZ.	vel, spread nodes within a shard across AZs
		•

Figure 2.31

- xiii. Under the **Advanced settings** section as follows:
 - a. Disable the Encryption at rest.
 - b. Disable the Encryption in transit.
 - c. Select EKS Worker node's in security group and click Next.

dvanced settings Info		
Security Use the following section to configure networ	k security and data security for your cluster.	
Encryption at rest		
Enable Enables encryption of data stored on disk.		
Encryption in transit		
 Enable Enables encryption of data that moves be 	tween the service and client.	
Selected security groups (1) A security group acts like a firewall that control	ols network access to your clusters.	Manage
Group ID [2]	Name	
sg-07f6e9607be371a77	EKSCluster-NodeSecurityGroup-NFFGAYHU0A3A	

Figure 2.32

5. Review the settings and click **Create** to launch your Redis cluster.

2.19 Registering domain using route-53

AWS ALB Ingress Controller creates an Application Load Balancer and routes the incoming requests to the target Kubernetes services according to the host-based routing rules. Host-based routing is a capability of ALB that redirects the user requests to the right service based on the request-host header.

For example, we can set the rules as below:

- If URL is *ibpsportal.aws.co.in* then redirect to the iBPS Portal container.
- If URL is *ibpsbam.aws.co.in* then redirect to the iBPS BAM container.

To support host-based routing, you must register a domain and create a new record. Perform the below steps to Record Set for each host path:

1. Register a domain using the AWS Route-53 service. Open the route53 service and go to **domain registration** in the Domain section.
2. Once the domain is registered, it creates a Hosted Zone. Click the newly created Hosted Zone list and then click **Go to Record Sets**.

Dashboard 4 Hosted zones	The new Route 53 console experience is now available We have redesigned dashboard and hosted zones pages to make them easier to use. Try out the new console. Starting in August, we will gradually deprecate the old console.	×
Health checks Fraffic flow Traffic policies	Create Hosted Zone Go to Record Sets Delete Hosted Zone	c
Policy records	Q. Search all fields X All Types V K Clisplaying 1 to 1 out of 1 Hosted Zones >>	Hosted Zone Details
Domains	Domain Name - Type - Record Set Count - Comment Hosted Zone ID -	Domain Name: omnidocs-aws.com. Type: Public Hosted Zone
Registered domains	omnidocs-aws.com. Public 3 HostedZone created by Route5 Z02575523ON85FD2ENHGL	Hosted Zone ID: Z02575523ON85FD2ENHGL
Resolver /PCs nbound endpoints Dutbound endpoints Rules Jse the new console Jse the new console		Comment: VotesdeZone created by Route53 Registrar Comment: VotesdeZone created by Route53 Registrar Name Servers *1 no.1221 nuesdns.24 cog no.528 auxidns.24 cod no.5207 nuesdns.24 cod no.5207 nuesdns.2507 nue

Figure 2.33

3. Click Create Record Set. The Create Record Set dialog appears:

aws Services	v Resource Groups v 🛠	🗘 vivek @ viveik 👻 Global 👻 Support 👻
Dashboard 4 Hosted zones Health checks	The new Route 53 console experience is now available We have redesigned dishboard and hosted zones pages to make them easier to use. Try out the new console. Starting in August, we will gradually deprecate the old console.	x
Traffic flow Traffic policies	Back to Hosted Zones Create Record Set Import Zone File Delete Record Set Test Record Set	200
Policy records	C Record Set Name X Any Type V Aliases Only Weighted Only	Create Record Set
Domains	Signal State	Name: omnidocs-aws.com.
Registered domains	Name Type Value	Type: A – IPv4 address
Pending requests	omnidocs-aws.com. A ALIAS dualstack.a23d8625f164b4c19bcf8014f34fe3 No -	
Resolver VPCs Inbound endpoints	ns-1221.awsdns-24.org. ns-168.awsdns-18.com. ns-2007.awsdns-58.co.u.k. ns-873.awsdns-20.net.	TTL (Seconds): 300 Tm Sm In 13 Value: See example below 1 Ded address Enter molitich addresse
Outbound endpoints Rules	omnidocs-aws.com. SOA ns-1221.awsdns-24.org. awsdns-hostmaster.amazo 90	on separate lines. Example: 192.02.235 198.51.100.234
Use the new console Starting in August, we will gradually deprecate the old console.		Routing Policy: Simple Route 53 responds to queries based only on the values in this record. Learn More Create
🗨 Feedback 🥥 English (US) © 2008 - 2020, Amazon Internet Services Privat	e Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Figure 2.34

- 4. Enter the following details in the Create Record Set to create a new RecordSet.
 - i. Name: Enter the user-defined name.
 - ii. **Type**: Select type as *A IPv4-address*.
 - iii. Alias: Select alias as Yes.

iv. Alias Target: Select the alias target as Load Balancer.

Import Zone File Delete F	Record Set	Test Reco	ord Set)	C 🕈
Aliases Only Weighted (Only ord Sets 》》	Create R Name: Type:	A-IPv	t 14 address	omnidocs-aws.com
ack.a23d8625f164b4c19bcf8014f34fe3	No	Alias: (• Yes) No	
dns-24.org. ns-19.com. dns-58.co.uk. ns-20.net.	-	You can a - CloudF - Elastic - ELB loa	also type ront distri Beanstall ad balanc	— <i>S3 website endpo</i> No Targets Available — <i>ELB Application k</i>	oints — ≩ load balancers —
dns-24.org. awsdns-hostmaster.amazo	-	- S3 web	site endr	— ELB Classic load	balancers —
ack.a23d8625f164b4c19bcf8014f34fe3	No	- VPC er - API Ga	ndpoint: e teway cu	a23d8625f164b4c19 — ELB Network load	bcf8014f34fe385-10737 d balancers —
ack.a23d8625f164b4c19bcf8014f34fe3	No	west-2.an - Global Learn Mo	nazonaw Accelerat pre	No Targets Available	÷ .
		Routing Route 53	Policy: responds	Simple to queries based only or	• the values in this record.
	•	Learn Mo	ore	Creat	te

Figure 2.35

Use this RecordSet as a host path for Ingress Controller. Thus, ALB is registered with a Domain name.

2.20 Generating SSL certificate against the registered domain

This section explains how to generate an SSL certificate against the registered domain.

Prerequisite:

You must have a registered domain in Route53.

Perform the below steps to generate an SSL Certificate:

- 1. Go to the Certificate Manager given under the Services.
- 2. Click Request a certificate.

aws Services ▼						Ş	vivek @ viveik 🔻 Mumbai 🔻	Support 🔻		
Certificates Certificate Manager	Certificates	domain names from your certifi	ates into public certificate transpare	ncy (CT) loas when renew	ing certificates. You	can opt out of CT logging	1. Learn more	Θ		
Private certificate authority Private CAs	Success You deleted the certific	ale						×		
	Request a certificate Limport a certificate									
						«	< Viewing certificates 1	to 2 > >>		
	Name 👻	Domain name 👻	Additional names	Status 👻	Туре 👻	In use? 👻	Renewal eligibility 👻			
	□ ▶ -	-	-	Issued	Imported	No	Ineligible			
	□ ► od-aws			Issued	Imported	No	Ineligible			
						«	< Viewing certificates 1	to 2 > >>		

Figure 2.36

3. To **Request a public certificate**, select the type of certificate for ACM to provide.

Services •	¢	vivek @ viveik 🔻	Mumbal 🔻	Support 🔻
Choose Import a certificate to import an existing certificate instead of requesting a new one. Learn more.				
Request a certificate				
Choose the type of certificate for ACM to provide.				
		Cancel	Request a	certificate

Figure 2.37

- 4. Add your registered domain name like *omnidocs-aws.com*.
- 5. Add another name to this certificate as *.<DOMAIN_NAME> like *.omnidocs-aws.com)

aws Services •	٩	vivek @ viveik 🔻	Mumbai 🔻	Support 🔻
Request a certificat	te			
Step 1: Add domain names Step 2: Select validation method	AWS Certificate Manager logs domain names from your certificates into public certificate transparency (CT) logs when renewing certificates. You can opt out of CT lo	gging. Learn more		
Step 3: Add tags Step 4: Review Step 5: Validation	You can use AWS Certificate Manager certificates with other AWS Services.			
	Add domain names			0
	Type the fully qualified domain name of the site you want to secure with an SSL/TLS certificate (for example, www.example.com). Use an asterisk (*) to request a wildca same domain. For example .* example com protects www.example.com, site example.com and images example.com.	d certificate to protec	t several sites	in the
	Domain name* *At least one domain name is required			
	omnidocs-aws.com			
	* omnidocs-aws.com			
	Add another name to this certificate			
	You can add additional names to this certificate. For example, if you're requesting a certificate for "www.example.com", you might want to add the name "example.com" so that customers c	an reach your site by eith	her name. Learn i	more.
			Cancel	Next

Figure 2.38

6. Select a validation method: Email validation.

lidation method						
Choose how AWS Certificate Manager (ACM) validates your certificate request. Before we issue your certificate, we need to validate that you own or control the doma certificate. ACM can validate ownership by using DNS or by sending email to the contact addresses of the domain owner.						
NS validation hoses this option if you have or can obtain permission to modify the DNS configuration for the domains in your certificate request. Learn mo mail validation	vre.					
hoose this option if you do not have permission or cannot obtain permission to modify the DNS configuration for the domains in your certific	ate request. Learr	n more.				
	Ilidation method AWS Certificate Manager (ACM) validates your certificate request. Before we issue your certificate, we need to validate that you own or cor M can validate ownership by using DNS or by sending email to the contact addresses of the domain owner. NS validation hoose this option if you have or can obtain permission to modify the DNS configuration for the domains in your certificate request. Learn me mail validation hoose this option if you do not have permission or cannot obtain permission to modify the DNS configuration for the domains in your certific	Ilidation method AWS Certificate Manager (ACM) validates your certificate request. Before we issue your certificate, we need to validate that you own or control the domains f M can validate ownership by using DNS or by sending email to the contact addresses of the domain owner. NS validation Mose this option if you have or can obtain permission to modify the DNS configuration for the domains in your certificate request. Learn more. mail validation hoose this option if you do not have permission or cannot obtain permission to modify the DNS configuration for the domains in your certificate request. Learn	Ilidation method AWS Certificate Manager (ACM) validates your certificate request. Before we issue your certificate, we need to validate that you own or control the domains for which you are re- XM can validate ownership by using DNS or by sending email to the contact addresses of the domain owner. NS validation hoose this option if you have or can obtain permission to modify the DNS configuration for the domains in your certificate request. Learn more. mail validation hoose this option if you do not have permission or cannot obtain permission to modify the DNS configuration for the domains in your certificate request. Learn more. Cancel	Ilidation method AWS Certificate Manager (ACM) validates your certificate request. Before we issue your certificate, we need to validate that you own or control the domains for which you are requesting the M can validate ownership by using DNS or by sending email to the contact addresses of the domain owner. NS validation house this option if you have or can obtain permission to modify the DNS configuration for the domains in your certificate request. Learn more. mail validation house this option if you do not have permission or cannot obtain permission to modify the DNS configuration for the domains in your certificate request. Learn more. Cancel Previous		

Figure 2.39

7. On the Add Tags, click **Review**.

aws Services ▼	vivek @ viveik 🔻	Mumbai 🔻	Support 🔻
Step 1: Add domain names Step 2: Select validation method Step 3: Add tags Step 4: Review Step 5: Validation Tag Name Value Tag Name Value	Cancel Pre	vious	eview

Figure 2.40

8. Click **Confirm and request**.

aws Services ▼		¢	vivek @ viveik 🔻	Mumbai 🔻	Support 🔻
Request a certificate	e				
Step 1: Add domain names Step 2: Select validation method	Review				
Step 3: Add tags	Domain name				
Step 5: Validation	The names you want to secure with an SSL/TLS certificate.				
	Domain name ormridocs-aws.com Additional name *omridocs-aws.com				
	Validation method				
	The method AWS uses to validate your certificate request. Validation method Email				
	Can	cel	Previous	onfirm and re	quest

Figure 2.41

- 9. Click **Continue**. An approval mail is sent to the below recipients of the registered domain.
 - Registrant Contact
 - Administrative Contact
 - Technical Contact

aws services ▼ Request a certifica	Ç wek@wek★ Martai★	Support 🔻
Step 1: Add domain names Step 2: Select validation method Step 3: Add tegs	Request in progress A certificate request with a status of Pending validation has been created. Further action is needed to complete the validation and approval of the certificate.	
Step 4: Review Step 5: Validation	Validation We will send email to the registered owner of each domain listed below. To validate control of the domain, the owner of the domain or an authorized representative must go to the Amazon certificate approverbate and approve the request. Further instructions are provided in the body of the email.	Ø
	omnidocs-aws.com *.omnidocs-aws.com If you or an authorized representative did not receive the email we sent, or if you want to learn more, click the help icon (?) above.	
		ntinue

Figure 2.42

dws Services ▼	Registered	domains > omr	idocs-aws.com		\$ ·	vivek @ viveik 🔻 Global 🔻	Support ▼
Dashboard 4 Hosted zones	Edit contacts	Manage DNS Delete of	Iomain				
Health checks Traffic flow Traffic policies	Domain Registration date @	omnidocs-aws.com 2020-04-28	Transfer lock 🖲 Authorization code	Disabled (enable) Get code	Name servers	ns-1221.awsdns-24.org ns-673.awsdns-20.net ns-158.awsdns-19.com	
Policy records Domains	Expiration date	2021-04-28 (extend) Disabled (enable)	Domain name status code 🕀 Tag 🤁	addPeriod ok View and manage tags for your domains using Tag editor	DNSSEC status 🖲	ns-2007.awsdns-58.co.uk Add or edit name servers Disabled Manage keys	
Registered domains Pending requests Resolver VPCs Inbound endpoints Outbound endpoints Rules Query logging	Registrant contact Vivek Pandey vivek, kumar@newge +91.9654451860 SouthEx Part1, New ID New Delhi DL 110003 IN	Verified n.co.in Delhi	Administrative cor Vivek Pandey vivek_kumar@newg +91.9654451860 SouthEx Part1, New New Delhi DL 11000 IN	tact ien.co.in Deihi 13	Technical contact Vivek Pandey vivek, Yumar@newgen.ci +91.9654451860 SouthEX Pant1, New Delt New Delhi DL 110003 IN	o.in hi	

Figure 2.43

10. Once all the recipients approve the certificate, the certificate status gets changed from **Pending Validation** to **issue.**

aws Services ▼									¢	vivek @ viveik 🔻	Mumbai 🔻	Support 🔻
Certificates		Cert	tifica	tes								0
Contracto manager		AWS	S Certi	ficate Manager logs	domain names from your certifica	ates into public certificate transp	arency (CT) logs when	renewing certificates. You	can opt out of CT log	iging. Learn more		
Private certificate authority Private CAs	1	Requ	iest a c	certificate	nport a certificate Action	s •					0 1	• •
									4	< Viewing certif	ficates 1 to 3	3 > »
				Name 👻	Domain name 👻	Additional names	Status 👻	Type 👻	In use? 👻	Renewal eligi	bility 👻	
	[•	-	omnidocs-aws.com	*.omnidocs-aws.com	Issued	Amazon Issued	No	Ineligible		
			•	100			Issued	Imported	No	Ineligible		
			•	od-aws	2	ъ.	Issued	Imported	No	Ineligible		
									4	< Viewing certif	ficates 1 to 3	\$ > >>

Figure 2.44

2.21 Cluster AutoScaler

The Cluster AutoScaler requires additional IAM and resource tagging considerations that are given in the following subsections:

2.21.1 Node Group IAM Policy

Create an IAM policy with the following JSON scripts and attach it to the Worker Node's IAM Role.

```
"autoscaling:DescribeAutoScalingInstances",
    "autoscaling:DescribeLaunchConfigurations",
    "autoscaling:DescribeTags",
    "autoscaling:SetDesiredCapacity",
    "autoscaling:TerminateInstanceInAutoScalingGroup",
    "ec2:DescribeLaunchTemplateVersions"
],
    "Resource": "*",
    "Effect": "Allow"
}
```

2.21.2 Updating auto scaling group

Perform the below steps to update the Auto Scaling Group:

- 1. Create an AMI of any worker node.
- 2. Go to the Auto Scaling Groups and click the created autoscaling group for this EKS Cluster.



Figure 2.36

3. Click the attached Launch Template.

Details Activity Automatic scaling I	nstance management Monitoring	g Instance refresh		
Group details				E
Desired capacity 2 Minimum capacity 2 Maximum capacity 3		Auto Scaling group name EKSCluster-NodeGroup-1X Date created Mon Aug 31 2020 18:37:52 Amazon Resource Name (A arn:aws:autoscaling:ap-sou 52ebf239de0b:autoScaling	(272UXQVEIE GMT+0530 (India Standard Time) RN) ht-1:678035612169:autoScalingGroup:357a5c32-3ea5-47 GroupName/EKSCluster-NodeGroup-1XY272UXQVEIE	7bb-bdd7
Launch template				E
Launch template NodeLaunchTemplate_3jfx1a4czh1W 🔀 It-Obecda36adf1a86ea	AMI ID ami-03a7c4b686e21c90a		Instance type m5.xlarge	
Version	Security groups		Security group IDs	
Latest	-		sg-07f6e9607be371a77	

4. Select the created Launch Template and select the Modify Template (Create new version) option from the Actions menu.

EC2 > Launch templates			
Launch templates (1/1) Info	;	Actions Create launch tem	plate
Q Filter by tags or properties or search by keyword		Launch instance from template	Ó
search: It-Obecda36adf1a86ea X Clear filters		Modify template (Create new version)	
		Delete template	
Launch template ID ∇ Launch template name ∇ Default version ∇	La	Delete template version	Create ti
It-Obecda36adf1a86ea NodeLaunchTemplate_3jfx1a4czh1W	2		2020-08
			÷

Figure 2.47

- 5. Select the created AMI and click **Create template version**.
- 6. Go back to the Auto Scaling Group and click **Edit** given in the right panel.

Launch template		Edit
Launch template NodeLaunchTemplate_3jfx1a4czh1W 🔀 It-Obecda36adf1a86ea	AMI ID ami-03a7c4b686e21c90a	Instance type m5.xlarge
Version Latest	Security groups - Key pair pame	Security group IDs sg-07f6e9607be371a77 Storane (volumes)



7. Select the Latest version and click Update.

Edit EKSCluster-NodeGroup-1XY272UXQVEIE Info			
Launch template Info		Switch to launch configuration	
Launch template Choose a launch template that contains the in security groups. NodeLaunchTemplate_3jfx1a4czh1W	stance-level settings, such as the Amazon Mach	ine Image (AMI), instance type, key pair, and	
Create a launch template 🖸 Version Latest (2)			
Create a launch template version 🗹			
Description	Launch template	Instance type	
OmnidocsUAT2-EKS-cluster	NodeLaunchTemplate_3jfx1a4czh1W C lt-Obecda36adf1a86ea	m5.xlarge	

Figure 2.49

8. The Cluster AutoScaler requires the following tags on your node group that is Auto Scaling groups so it can be auto-discovered.

Кеу	Value
k8s.io/cluster-autoscaler/ <cluster-name></cluster-name>	owned
k8s.io/cluster-autoscaler/enabled	True

2.21.3 Deploying cluster autoscaler

Perform the below steps to deploy the Cluster AutoScaler:

1. Deploy the Cluster Autoscaler to your cluster using the below command:

```
kubectl apply -f
https://raw.githubusercontent.com/kubernetes/autoscaler/master/cluster-
autoscaler/cloudprovider/aws/examples/cluster-autoscaler-autodiscover.yaml
```

2. Add the cluster-autoscaler.kubernetes.io/safe-to-evict annotation to the deployment using the below command:

```
kubectl -n kube-system annotate deployment.apps/cluster-autoscaler cluster-
autoscaler.kubernetes.io/safe-to-evict="false"
```

3. Edit the Cluster AutoScaler deployment using the below command:

kubectl -n kube-system edit deployment.apps/cluster-autoscaler

• Edit the cluster-autoscaler container command to replace <YOUR CLUSTER NAME> with your cluster's name, and add the following options:

- --balance-similar-node-groups
- --skip-nodes-with-system-pods=false
- 4. Save and close the file to apply the changes.

For Example:

spec:
containers:
- command:
/cluster-autoscaler
v=4
stderrthreshold=info
cloud-provider=aws
skip-nodes-with-local-storage=false
expander=least-waste
node-group-auto-discovery=asg:tag=k8s.io/cluster-
<pre>autoscaler/enabled,k8s.io/cluster-autoscaler/<your cluster="" name=""></your></pre>
balance-similar-node-groups
skip-nodes-with-system-pods=false

5. Open the below URL in a web browser and find the latest Cluster AutoScaler version that matches your cluster's Kubernetes major and minor versions.

For example, if your cluster's Kubernetes version is 1.21 then find the latest Cluster AutoScaler release that begins with 1.21. Record the semantic version number (1.21.n) for that release to use in the further step.

https://github.com/kubernetes/autoscaler/releases.

- 6. Set the Cluster AutoScaler image tag to the version that you have recorded.
- 7. Use the below command and replace 1.21.n with your value. kubectl -n kube-system set image deployment.apps/cluster-autoscaler clusterautoscaler=us.gcr.io/k8s-artifacts-prod/autoscaling/clusterautoscaler:v1.21.n

2.21.4 Viewing cluster autoscaler logs

Once you have deployed the Cluster Autoscaler, you can view the logs and verify that it is monitoring your cluster load.

You can view your Cluster AutoScaler logs using the below command: kubectl -n kube-system logs -f deployment.apps/cluster-autoscaler

2.22 Setting up CloudWatch container insights

Container Insights is a fully managed CloudWatch service and is used to collect, aggregate, and summarize metrics and logs of containerized applications deployed on ECS or EKS service. The metrics include the utilization of resources such as CPU, memory, disk, and network.

Container insights also provide diagnostic information, such as container restart failures, to help you isolate issues and resolve them quickly. The metrics that Container Insights collects are available in CloudWatch automatic dashboards

Perform the below steps to set up the CloudWatch container insights:

- 1. Attach the below policy to the IAM role of your worker nodes:
 - CloudWatchAgentServerPolicy

```
2. Execute the below command to deploy container insights on the EKS cluster:
    curl https://raw.githubusercontent.com/aws-samples/amazon-cloudwatch-
    container-insights/latest/k8s-deployment-manifest-templates/deployment-
    mode/daemonset/container-insights-monitoring/quickstart/cwagent-fluentd-
    quickstart.yaml | sed
    "s/{{cluster_name}}/CLUSTER_NAME/;s/{{region_name}}/REGION_NAME/" | kubectl
    apply -f -
```

In the above command, change the **CLUSTER_NAME** and **REGION_NAME** as required.

3 Deploying OmniDocs and RMS containers

This chapter describes the deployment of OmniDocs and RMS containers on AWS. Refer to the below sub-sections for procedural details.

3.1 Prerequisites

To deploy OmniDocs and RMS containers, the AWS Elastic Kubernetes Service must already be configured, and its Worker nodes must be in the ready state.

NOTE:

Refer to the Configuration of AWS Kubernetes Cluster for the configuration of AWS Elastic Kubernetes Service.

3.2 Deliverables

Newgen has isolated the product suite into multiple Docker containers to enable the independent scalability of each Docker container. This separation is done based on the product's usability. At a broad level, Web components and EJB components are isolated for deployment in separate container instances. Web components is deployed on the underlying web server JBoss WebServer 5.7.x. EJB components is deployed on the underlying application server JBoss EAP 7.4.x. Newgen has released multiple Docker images for the different product suites along with some configuration files for data persistence, YAML files for deployment, and some documentation for end-to-end configurations and deployments.

The followings are the list of deliverables:

- Docker Images
- <u>Configuration Files</u>
- YAML Files

3.2.1 Docker Images

The following 7 Docker images are delivered for the initial product deployment:

- OmniDocs+RMS Web Components
- OmniDocs Web Service Components
- OmniDocs+RMS EJB Components
- OmniDocs Add-on Services (Wrapper, AlarmMailer, Scheduler, ThumbnailManager and LDAP)
- Docker Images for EasySearch (Only one is required based on the infrastructure availability)
 - EasySearch (Apache Manifold and ElasticSearch [freeware software])
 Or
 - EasySearch (Apache Manifold only [freeware software] with AWS managed ElasticSearch)
- Text Extraction Manager or Full-Text Search (TEM/FTS)
- OmniScan Web Components
- RMS SharePoint Adapter
- Messaging Service

NOTE:

These Docker images can be delivered to a private Docker repository like AWS ECR (Elastic Container Registry) or in the form of compressed files that can be shared over the FTP or similar kind of media.

3.2.2 Configuration files

Configuration files are dynamic in nature and data is written at runtime. Database details in configuration files such as *Server.xml* and *standalone.xml* are written at runtime. These types of files must be kept outside the container to persist the data. Here, AWS EFS is used to persist configuration files.

The following configuration files are shared for OmniDocs and RMS Docker images:

- OmniDocs11.0Web
- TEM11.0
- OmniDocs11.0Ejb
- OmniscanWeb6.0

- OD11.0Services
- EasySearch11.0

- RMS SharePoint Adapter
- MessagingService

3.2.3 YAML files

YAML files stands for "YAML Ain't Markup Language". It is a human-readable object configuration file that is used to deploy and manage the objects on the Kubernetes cluster. In other words, it is a manifest file that contains the deployment descriptor of Kubernetes containers. You can execute YAML files using "kubectl apply –f <YAMLFile>" or use these files in AWS CodePipeline to deploy the containers.

The following configuration files has shared for OmniDocs and RMS Docker images:

- OmniDocs11.0Web.yml
- OmniDocs11.0Web_Services.yml
- OmniDocs11.0EJB.yml
- OmniDocs11.0Services.yml
- EasySearch11.0.yml
- EasySearch11.0_ApacheOnly.yml
- OmniDocs11.0MessagingService.yml

- TEM11.0.yml
- OmniScanWeb6.0.yml
- AWS_ALB-IngressController.yml
- buildspec.yml
- buildspec_EasySearch.yml
- RMSSharePointAdapter.yml

Here's an example of a YAML file:



Figure 3.1

AWS_ALB-IngressController.yml is used for the ingress controller. An ingress controller is an object running inside the Kubernetes cluster that is used to manage the host-based routing rules. For

example, you can set the host-based routing rules like if the URL is *omnidocs.newgendocker.com* then the ingress controller redirects the user request to OmniDocs+RMS WEB containers and if the URL is *omniscan.newgendocker.com* then it redirects the user request to the OmniScanWEB containers.

Buildspec.yml is used in AWS CodePipeline for deploying the containers on AWS Elastic Kubernetes Service.

NOTE:

You can store the above YAML files in AWS CodeCommit Repo that is used by AWS CodePipeline.

3.3 Changes in Product's YAML files

The changes in the Product's YAML files are:

 Name: In the OmniDocs11.0Web.yml file, od110web is given as the default name of Kubernetes objects - deployment, replicas, container, and service. You can change this name as per your choice. While changing the name, ensure that this name is not more than 13 letters in length and must contain small letters only. For example,

```
apiVersion: apps/v1
kind: Deployment
metadata:
#Name should not be more than 13 letters
name: od110web
spec:
replicas: 1
selector:
    matchLabels:
        name: od110web
strategy:
type: RollingUpdate
rollingUpdate:
    maxSurge: 1
    maxUnavailable: 0
template:
    metadata:
    labels:
        name: od110web
    spec:
    containers:
    - name: od110web
```

Figure 3.2

- **Replica:** In the *OmniDocs11.0Web.yml* file, the default replica is given as **1**. That means only one container is created after the deployment. You can increase this number as per our choice.
- Image: In the *OmniDocs11.0Web*.yml file, update the **image** location. By default, the below value is given:

image: REGISTRY_ID.dkr.ecr.REGION.amazonaws.com/REPOSITORY_NAME:IMAGE_TAG

Here:

- REGISTRY_ID is the AWS account ID where AWS ECR (Elastic Container Registry) is created.
- **REGION** is the AWS Region where AWS ECR (Elastic Container Registry) is created.
- **REPOSITORY_NAME** is the Omnidocs WEB Docker image name.
- **IMAGE_TAG** is a Docker image's tag name that you want to deploy.

As AWS CodePipeline is used to deploy Docker images, you do not need to update the image in the YAML file as AWS CodePipeline updates these values at runtime.

• SecurityContext: In the *OmniDocs11.0Web.yml* file, SecurityContext [runAsNonRoot: true] is defined. It means the OmniDocs11.0Web container can never be run with root privileges. If any container tries to run with the root user, then Kubernetes stops its deployments.



• **Resource request and limit:** In the *OmniDocs11.0Web.yml* file, resource request and resource limit parameters are defined. The **request** parameter specifies the minimum required resources to run the particular container and the **limit** parameter specifies the maximum resource limit that a container can use. In other words, a running container is not allowed to use more than the resource limit you set.

requests:			
memory: 1024Mi			
cpu: 800m			
limits:			
memory: 2048Mi			
cpu: 1000m			
5. 0.4			



Here, 1000m CPU = 1 Core CPU

The above-specified limit is the minimum required resource to run a container. If users are increasing, then you must increase the limit range accordingly.

 VolumeMounts and Volume: Volume mounts and volumes are used to persist the data outside the container so that whenever the container terminates due to any reason our data is always persisted. In the OmniDocs11.0Web.yml file, we have persisted configuration files or folders and log files.

```
volumeMounts:
- name: omnidocsconfig
mountPath: /Newgen/jws-5.7/tomcat/bin/Newgen/NGConfig

    name: omniscanweb

mountPath: /Newgen/jws-5.7/tomcat/bin/omniscanweb
- name: context-xml
mountPath: /Newgen/jws-5.7/tomcat/conf/context.xml
- name: web-xml
mountPath: /Newgen/jws-5.7/tomcat/conf/web.xml
- name: jboss-ejb-client-properties
 mountPath: /Newgen/jws-5.7/tomcat/lib/jboss-ejb-client.properties
- name: tomcat-logs
 mountPath: /Newgen/jws-5.7/tomcat/logs
 subPathExpr: $(POD NAME)
- name: omnidocs-logs
 mountPath: /Newgen/jws-5.7/tomcat/bin/Newgen/NGLogs
 subPathExpr: $(POD NAME)
- name: ng-temp
 mountPath: /Newgen/jws-5.7/tomcat/bin/Newgen/NGTemp
 subPathExpr: $(POD NAME)
- name: system-report
 mountPath: /Newgen/jboss-eap-7.4/bin/SystemReports
```

Figure	3.5
--------	-----

In volumeMounts, **mountPath** is a path inside the container that is being mounted. Here, mountPath cannot be changed as this structure is predefined in a Docker container. The **name** is a user-defined name that must be matched with the name specified in volumes.





In volumes, the hostPath mounts a file or directory from the worker node's file system into the container. This path must exist in the worker node's file system. The hostPath can be a file path or folder path, you just need to define its type whether it is a File or Directory. In this YAML file, some hostPath contains dynamic values whose value gets updated at runtime.

• **Ports:** In the *OmniDocs11.0Web.yml* file, containerPort is specified as **8080**. That means only port 8080 is exposed outside the container and no other port is accessible from outside.



• **ReadinessProbe:** The kubelet uses the readiness probe to know when a container is ready to start accepting traffic. Until unless the readiness probe is not succeeded, the container does not serve the user requests.

```
readinessProbe:
httpGet:
   path: /omnidocs/web
   port: 8080
# Intial delay is set to a high value to have a better
# visibility of the ramped deployment
initialDelaySeconds: 30
periodSeconds: 5
```

Figure 3.8

Here, until unless *ip:port/omnidocs/web* is not accessible through a browser, the container does not accept the user request.

• LivenessProbe: Docker containers have healing power, if an application running inside the container gets down due to any reason or becomes unresponsive then Kubernetes restarts the application automatically inside the container. This feature is known as LivenessProbe in Kubernetes.





• Environment variable: In the *OmniDocs11.0Web.yml* file, the JAVA_OPTS parameter is defined that is used to set the heap size in the WEB container dynamically.

```
- name: JAVA_OPTS
value: "-XX:+UseContainerSupport -XX:+DisableExplicitGC -XX:InitialRAMPercentage=75.0
-XX:MaxRAMPercentage=75.0"
```

```
Figure 3.10
```

Ensure '-**XX:MaxRAMPercentage**' is a parameter through which you can provide the available memory to use as a max heap size to JVM. In the above example, 75% of total memory is allocated as heap size.

NOTE:

You can use the above guidelines to update other YAML Files that are as follows:

- OmniDocs11.0Web_Services.yml
- OmniDocs11.0EJB.yml
- OmniDocs11.0Services.yml
- EasySearch11.0.yml
- EasySearch11.0_ApacheOnly.yml
- TEM11.0.yml
- OmniScanWeb6.0.yml
- RMSSharePointAdapter.yml
- OmniDocs11.0MessagingService.yml

3.4 Changes in AWS load balancer controller YAML files

The AWS Load Balancer Controller creates an Application Load Balancer and routes the incoming requests to the target Kubernetes services according to the host-based routing rules. Host-based routing is a capability of ALB that redirects the user requests to the right service based on the request-host header

For example, you can set the rules as below:

- If URL is *omnidocs.newgendocker.com*, then redirect to the OmniDocs+RMS Web container.
- If URL is *omniscan.newgendocker.com*, then redirect to the OmniScanWeb container.

NOTE:

To support the host-based routing, we must register a domain, create a new RecordSet in Route-53 for each host-path and generate the SSL certificate against the registered domain. Refer to the <u>Configuration of AWS Kubernetes Cluster</u> section for the configuration of AWS ALB Ingress Controller, Route-53, and Certificate Manager.

- Once AWS ALB Ingress is configured, RecordSet is created in Route-53, and an SSL certificate is generated. You must deploy the Ingress controller along with its ruleset using the YAML file.
- Before deploying the same you need to update some settings in the AWS_ALB-IngressController.yml file.
- To access your application using both the HTTP and HTTPS protocols, ensure that the below annotation is added:

alb.ingress.kubernetes.io/listen-ports: '[{"HTTPS":443}, {"HTTP":80}]'

```
annotations:
```

```
kubernetes.io/ingress.class: alb
alb.ingress.kubernetes.io/scheme: internet-facing
alb.ingress.kubernetes.io/listen-ports: '[{"HTTPS":443}, {"HTTP":80}]'
alb.ingress.kubernetes.io/certificate-arn: <AWS Certificate ARN>
alb.ingress.kubernetes.io/ssl-policy: ELBSecurityPolicy-TLS-1-2-Ext-2018-06
```

Figure 3.11

- To access your application using the HTTPS protocol only, update the annotation as below: alb.ingress.kubernetes.io/listen-ports: '[{"HTTPS":443}]'
- Update the SSL certificate ARN generated from AWS Certificate Manager.
- If you want to open both the HTTP and HTTPS protocols and whenever the calls come to the HTTP, it redirects to HTTPS, then make sure the below annotation is added: alb.ingress.kubernetes.io/ssl-redirect: '443'
- Update the subnets and security groups associated with the Kubernetes Worker nodes.
 alb.ingress.kubernetes.io/subnets: subnet-0a37ea6be439259a0, subnet-0e3a0a6a7d3887eca, subnet-09ee1bc2c393de555
 alb.ingress.kubernetes.io/security-groups: sg-0f4f4504892233a90
- In the above step, there are multiple host-based rules defined.
 - omnidocs.newgendocker.com [Specified as a record set in Route-53] If the host URL is omnidocs.newgendocker.com, then it redirects the user request to the od110web container's service which is running on port 8080. Here, od110web is the name of the OmniDocs+RMS Web container.
 - omnidocswebservices.newgendocker.com [Specified as a record set in Route-53] If the host URL is omnidocswebservices.newgendocker.com, then it redirects the user request to the od110websvc container's service which is running on port 8080. Here, od110websvc is the name of the OmniDocs Web Service container.
 - omnidocsconsole.newgendocker.com [Specified as a record set in Route-53] If the host URL is omnidocsconsole.newgendocker.com, then it redirects the user request to the od110ejb container's service which is running on port 9990. Here, od110ejb is the name of the OmniDocs+RMS EJB container.
 - > apachemanifold.newgendocker.com [Specified as a record set in Route-53]

If the host URL is *apachemanifold.newgendocker.com* then it redirects the user request to the easysearch11 container's service which is running on port 8345. Here, easysearch11 is the name of the EasySearch container.

omniscan.newgendocker.com [Specified as a record set in Route-53] If the host URL is omniscan.newgendocker.com, then it redirects the user request to the omniscan web container's service which is running on port 8080. Here, omniscanweb is the name of the OmniScan Web container.

- Odmsgservice.newgendocker.com [Specified as a record set in Route-53] If the host URL is odmsgservice.newgendocker.com, then it redirects the user request to the odmessage container's service which is running on port 8080. Here, odmsgservice is the name of the OD Message Service container.
- In this YAML file, you can change the host URL, ServiceName, ServicePort, and the name **name: alb-ingress** as per your choice.
- After making the required changes as per our choice, you can deploy the Ingress controller by executing this YAML file using the below command: kubectl apply -f AWS_ALB-IngressController.yml

NOTE:

To execute the above command, kubectl must be configured on your local server. Refer to the <u>Configuration of AWS</u> <u>Kubernetes Cluster</u> section to run kubectl from your local machine.

3.5 Changes in configuration files

The section includes the following sub-sections:

3.5.1 Prerequisites

The Prerequisites are as follows:

- All the configuration files must be copied to the worker node's hostPath location defined in the YAML files and that hostPath must be mounted to the AWS EFS.
- The RedisCache server is already configured.
- A valid wildcard certificate and the domain are already configured.
- SSL or TLS must be configured for the Application's URL.

NOTE:

- Refer to the Mount EFS to Worker Nodes section to mount the EFS to hostPath.
- By default, the applications run on HTTPS only. If you want to run with HTTP protocol then some additional setting is required. For more information, refer to the Docker Troubleshooting Guide.

3.5.2 OmniDocs+RMS Web Changes

The changes in OmniDocs+RMS Web are:

 Update the OmniDocs+RMS EJB container name [Defined in OmniDocs11.0EJB.yml file] in NGOClientData.xml, RMAppConfig.xml and RMClientData.xml files in between the <endPointURL></endPointURL> tags located inside the OmniDocs11.0Web\Newgen\NGConfig\ngdbini folder at the mapped location on the Worker node.



Figure 3.12

Here, **od110ejb** is the name of the OmniDocs+RMS EJB container.

 Update the OmniDocs+RMS EJB container name [Defined in OmniDocs11.0EJB.yml file] in IS.ini file in between the <endPointURL ></endPointURL > tags located inside the OmniDocs11.0Web\Newgen\NGConfig folder at the mapped location on the Worker node. For example,



Figure 3.13

 Update the OmniDocs+RMS EJB container name [Defined in OmniDocs11.0EJB.yml file] in *jboss-ejb-client.properties* file located inside the OmniDocs11.0Web folder at the mapped location on the Worker node. For example,

```
remote.connectionprovider.create.options.org.xnio.Options.SSL_ENABLED=false
remote.connections=default
remote.connection.default.host=odl10ejb
remote.connection.default.port = 8080
remote.connection.default.username=site
remote.connection.default.password=create
remote.connection.default.password=create
remote.connection.default.connect.options.org.xnio.Options.SASL_POLICY_NOANONYMOUS=false
```

Figure 3.14

Here, **od110ejb** is the name of the OmniDocs+RMS EJB container.

- Update the **RMSEnabled=Y** in *eworkstyle.ini* file located at *OmniDocs11.0Web/Newgen/NGConfig/ngdbini/odwebini* folder at mapped location.
- Update the AWS Elastic Redis cache's configuration endpoint in *redisson.yaml* file against the singleServerConfig or clusterServersConfig. If redis cache is SSL enabled, then use the *rediss://<endpoint url>:port* and if SSL is not enabled, then use *redis://<endpoint url>:port*. This file *redisson.yaml* is located inside the OmniDocs11.0Web folder at the mapped location on the Worker node.

```
ingleServerConfig:
 idleConnectionTimeout: 10000
 connectTimeout: 10000
 timeout: 3000
 retryAttempts: 3
 retryInterval: 1500
 password: null
 subscriptionsPerConnection: 5
 clientName: null
                                                    active mazonaws.com:6379"
 address: "redis://#
                          100
 subscriptionConnectionMinimumIdleSize: 1
 subscriptionConnectionPoolSize: 50
 connectionMinimumIdleSize: 24
 connectionPoolSize: 64
 database: 0
 dnsMonitoringInterval: 5000
threads: 16
nettyThreads: 32
podec: !<org.redisson.codec.MarshallingCodec> {}
transportMode: "NIO"
Reference: https://github.com/redisson/redisson/wiki/2.-Configuration#26-single-instance-mode
CLUSTER ---
CLUSTER clusterServersConfig:
CLUSTER idleConnectionTimeout: 10000
CLUSTER connectTimeout: 10000
CLUSTER timeout: 3000
CLUSTER retryAttempts: 3
CLUSTER retryInterval: 1500
CLUSTER failedSlaveReconnectionInterval: 3000
CLUSTER failedSlaveCheckInterval: 60000
CLUSTER password: null
CLUSTER
          subscriptionsPerConnection: 5
CLUSTER clientName: null
CLUSTER loadBalancer: !<org.redisson.connection.balancer.RoundRobinLoadBalancer> {}
CLUSTER
        subscriptionConnectionMinimumIdleSize: 1
```



- Open the *web.xml* file in edit mode located inside the OmniDocs11.0Web folder at the mapped location on the Worker node.
- Search for filter httpHeaderSecurity and update the <param-value></param-value> tag's value with OmniDocs URL without context name against <param-name> antiClickJackingUri</param-name>.

```
<filter>
<filter-name>httpHeaderSecurity</filter-name>
<filter-class>org.apache.catalina.filters.HttpHeaderSecurityFilter</filter-class>
<async-supported>true</async-supported>
<init-param>
<param-name>antiClickJackingOption</param-name>
<param-value>ALLOW-FROM</param-value>
</init-param>
<init-param>
<param-name>antiClickJackingUri</param-name>
<param-name>antiClickJackingUri</param-name>
<param-name>antiClickJackingUri</param-name>
<param-name>antiClickJackingUri</param-name>
<param-value>omnidocs.newgendocker.com</param-value>
</init-param>
</init-param>
</init-param>
</init-param>
```



- Open the *web_svc.xml* file in edit mode located inside the OmniDocs11.0Web folder at the mapped location on the Worker node.
- Search for filter-class <filter-class>org.apache.catalina.filters.CorsFilter</filter-class> and update the <param-value></param-value> tag's value with OmniDocs URL with protocol against <param-name> antiClickJackingUri</param-name>



Figure 3.17

3.5.3 Wrapper changes

Update the OmniDocs+RMS EJB container name [Defined in *OmniDocs11.0EJB.yml* file] in NGOClientData.xml, *RMAppConfig.xml and RMClientData.xml* files in between the *<endPointURL></endPointURL>* tags file located inside the *OD11.0Services/Wrapper/ngdbini* folder at the mapped location on the Worker node.



Figure 3.18

Here, **od110ejb** is the name of the OmniDocs+RMS EJB container.

3.5.4 AlarmMailer changes

Prerequisite:

- The cabinet is created and associated with the running containers. If the cabinet is not created, then refer to <u>Cabinet and Data Source Creation</u> section.
- SecretManager policy must be created and attached to the Worker Node IAM role. If not created, then refer to the Creating secret manager policy and secrets section.

The changes in AlarmMailer are as follows:

 Update the OmniDocs+RMS EJB container name [Defined in OmniDocs11.0EJB.yml file] in *IS.ini* in between the *<endPointURL></endPointURL>* tags file located inside the *OD11.0Services* or *AlarmMailer* folder at the mapped location on the Worker node. For example,

<endPointURL>http://od110ejb:8080/callbroker/execute/GenericCallBroker</endPointURL> Here, **od110ejb** is the name of the OmniDocs+RMS EJB container.

 Update the OmniDocs+RMS EJB container name [Defined in OmniDocs11.0EJB.yml file] in NGOClientData.xml in between the <endPointURL></endPointURL> tags file located inside the OD11.0Services/AlarmMailer/ngdbini folder at the mapped location on the Worker node. For example,

- 3. Update the below settings in the *Alarm.ini* file located inside the *OD11.0Services/AlarmMailer* folder at the mapped location on the Worker node.
 - Update the OmniDocs URL without context name in between the <webservername></webservername> tag.
 For example, <webservername>omnidocs.newgendocker.com</webservername>
 Here, omnidocs.newgendocker.com is the host path defined in the AWS_ALB-IngressController.yml file.
 - ii. Leave the WebServerPort as blank if OmniDocsWEB URL does not contain a port.For example, <webserverport></webserverport>
 - iii. Update the OmniDocs cabinet name in between <cabinetname></cabinetname> tag.
 For example, <cabinetname>ecmsuite</cabinetname>
 Here, ecmsuite is the OmniDocs cabinet name that gets created.
 - iv. Update the OmniDocs supervisor group's user in between the <user></user> tag.
 For example, <user>supervisor</user>
 - V. Update the OmniDocs supervisor group's user password in between the
 <password></password> tag. Ensure that this password must be in an encrypted format.

For example, <password>:X-D;U:T-C;P-C;p5-C;b:d:</password>

- 4. If Secret Vault is used to store sensitive information, then refer below points:
 - i. Update the values in below tags that is located inside <general> tag in Alarm.ini located inside the OD11.0Services/AlarmMailer folder at the mapped location on the Worker node.

<keyvault>true</keyvault> <secret>AlarmMailerPSequence</secret> Here: AlarmMailerPSequence is Secret name that is already created.

Update base64 Encoded string of https://<OmniDocs+RMS Web Host URL>/Security
 [Present in AWS_ALB-IngressController.yml file] in SecretManager.ini file.
 For Example,

If Security war context URL is as following: https://rms40.newgendocker.com/Security

Then, EndPointURL property in *SecretManager.ini* must be as following: EndPointURL=aHR0cHM6Ly9ybXM0MC5uZXdnZW5kb2NrZXIuY29tLINIY3VyaXR5

3.5.5 LDAP changes

Prerequisite:

- The cabinet is created and associated with the running containers. If the cabinet is not created, then refer to the <u>Cabinet and Data Source Creation</u> section.
- SecretManager policy must be created and attached to the Worker Node IAM role. If not created, then refer to the Creating a Secret Manager policy and secrets section.

The changes in LDAP are as follows: (For On_Prem Active Directory)

- Ensure that the LDAP Domain server is configured, and a private tunnel is created between the Kubernetes worker nodes and the LDAP Domain server.
- Update the OmniDocs+RMS EJB container name [Defined in OmniDocs11.0EJB.yml file] in *NGOClientData.xml* in between the *<endPointURL></endPointURL>* tags file located inside the *OD11.0Services/ODAuthMgr/ngdbini* folder at the mapped location on the Worker node. For example,

<endPointURL>http://od110ejb:8080/callbroker/execute/GenericCallBroker</endPointURL> Here, **od110ejb** is the name of the OmniDocs+RMS EJB container.

• Update the cabinet name and domain name in the Idap.ini and Hook.ini file located inside the *OD11.0Services/ODAuthMgr* folder at the mapped location.

```
#
#Tue Nov 26 11:34:40 IST 2013
DISPort=1999
DISIPAddress=127.0.0.1
Log4j_properties_file=jtshook_log4j.properties
Encoding=UTF-8
PROTOCOL=1dap Hook.ini
LOGOUTTIME=15000
DIRECTORYSERVICE=ActiveDS
REACTUI=true
# Default domain name to add user For multidomain LDAP
DEFAULTDOMAIN=eco.com
ecmsuite=eco.com
```





Figure 3.20

Here, **ecmsuite** is the cabinet name and *eco.com* is the domain name.

- Update the same cabinet name and domain name in the *ldap.ini* and *Hook.ini* file located inside the *OmniDocs11.0Web\Newgen\NGConfig* folder at the mapped location.
- Update the OD11.0Services container's service name [Defined in respective YAML file] in *Idap.ini* and *Hook.ini* file located inside the *OmniDocs11.0Web\Newgen\NGConfig* folder at the mapped location.

```
#
#Tue Nov 26 11:34:40 IST 2013
DISPort=1999
DISIPAddress=odl10services
Log4j_properties_file=jtshook_log4j.properties
Encoding=UTF-8
PROTOCOL=1dap Hook.ini
LOGOUTTIME=15000
DIRECTORYSERVICE=ActiveDS
REACTUI=true
# Default domain name to add user For multidomain LDAP
DEFAULTDOMAIN=eco.com
ecmsuite=eco.com
```





Figure 3.22

Here, **od110services** is the service name of the OD11.0Services container.

 Set <Display> as true for LDAP in AdminMenuOptions.xml located inside OmniDocs11.0Web/Newgen/NGConfig/ngdbini/Custom/CABINETNAME folder at mapped location.

```
<SSALink>
    <LinkName>Ldap</LinkName>
    <LinkDescription>LdapDescription</LinkDescription>
    <JspName>/ldap/config.jsp</JspName>
    <Display>true</Display>
    <IconURL></IconURL>
</SSALink>
```

Figure 3.23

- 5. If Secret Vault is used to store sensitive information, then refer below points:
 - Update the values in below tags that are located inside *<ServerInfo>* tag in *Server.xml* located inside the *OmniDocs11.0Ejb/ngdbini* folder at the mapped location on the Worker node.

<secretName>LDAP</secretName> <secretManager>Y</secretManager> Here: LDAP is Secret name that is already created.

• Update **base64 Encoded** string of *https://<OmniDocs+RMS Web Host URL>/Security* [Present in *AWS_ALB-IngressController.yml* file] in *SecretManager.ini* file located inside *OmniDocs11.0Ejb/ngdbini* folder and *OmniDocs11.0Web/ngdbini* folder.

For Example, If Security war context URL is as following: https://rms40.newgendocker.com/Security/ Then, EndPointURL property in SecretManager.ini must be as following: EndPointURL=aHR0cHM6Ly9ybXM0MC5uZXdnZW5kb2NrZXIuY29tL1N1Y3VyaXR5

- Update the below properties in KeyVault.properties file located inside *OmniDocs11.0Web/ngdbini folder*.
 - Region (Its value must be region of the AWS account. For example, ap-south-1)
 - KeyVaultType (Its value must be the key vault type. For example., AWS)

The changes in LDAP are as follows: (For Azure Active Directory)

 Update the OmniDocs+RMS EJB container name [Defined in OmniDocs11.0EJB.yml file] in *NGOClientData.xml* in between the *<endPointURL></endPointURL>* tags file located inside the *OD11.0Services/ODAuthMgr/ngdbini* folder at the mapped location on the Worker node. For example,

<endPointURL>http://od110ejb:8080/callbroker/execute/GenericCallBroker</endPointURL> Here, **od110ejb** is the name of the OmniDocs+RMS EJB container.

• Update the cabinet name, domain name, and directory service as **AzureAD** in the Hook.ini file located inside the *OD11.0Services/ODAuthMgr* folder at the mapped location.

```
DISPort=1999

DISIPAddress=127.0.0.1

Log4j_properties_file=jtshook_log4j.properties

Encoding=UTF-8

PROTOCOL=1dap Hook.ini

LOGOUTTIME=15000

DIRECTORYSERVICE=AzureAD

REACTUI=true

# Default domain name to add user For multidomain LDAP

DEFAULTDOMAIN=eco.com

ecmsuite=eco.com
```

Figure 3.24

• Update the cabinet name and domain name in the *ldap.ini* file located inside the *OD11.0Services* or *ODAuthMgr* folder at the mapped location.

# #Wed Dec 23 10:53:14 GMT+05:30 2009 DISPort=1999	
DISIPAddress=127.0.0.1 Encoding=UTF-8 LDAP.ini	
LogOutTime=15000 IsMakerChecker=N	
# Default domain name to add user For multidomain LD. ecmsuite=eco.com	AP

Figure 3.25

Here, **ecmsuite** is the cabinet name and *eco.com* is the domain name.

• Update the directory service as **AzureAD** in the DIS.xml file located inside the **OD11.0Services** or **ODAuthMgr** folder at the mapped location.





- Update the same cabinet name and domain name in the *ldap.ini* and *Hook.ini* file located inside the **OmniDocs11.0Web\Newgen\NGConfig** folder at the mapped location.
- Update the OD11.0Services container's service name [Defined in respective YAML file] in Idap.ini and Hook.ini file located inside the **OmniDocs11.0Web\Newgen\NGConfig** folder at the mapped location.
- Update the directory service as **AzureAD** in Hook.ini and config.ini located inside the **OmniDocs11.0Web\Newgen\NGConfig** folder at the mapped location.

```
DISPort=1999

DISIPAddress=odll0services

Log4j_properties_file=jtshook_log4j.properties

Encoding=UTF-8

PROTOCOL=ldap

LOGOUTTIME=15000

DIRECTORYSERVICE=AzureAD

REACTUI=true

# Default domain name to add user For multidomain LDAP

DEFAULTDOMAIN=eco.com

ecmsuite=eco.com
```









Figure 3.29

Here, **od110services** is the service name of the OD11.0Services container.

 Set <Display> as true for Idap in AdminMenuOptions.xml located inside OmniDocs11.0Web/Newgen/NGConfig/ngdbini/Custom/CABINETNAME folder at mapped location.

For example,

```
<SSALink>

<LinkName>Ldap</LinkName>

<LinkDescription>LdapDescription</LinkDescription>

<JspName>/ldap/config.jsp</JspName>

<Display>true</Display>

<IconURL></IconURL>

</SSALink>
```

Figure 3.30

3.5.6 SSO changes

Prerequisite:

The cabinet is created and associated with the running containers. If the cabinet is not created, then refer to <u>Cabinet and Data Source Creation</u> section.

The changes in SSO are as follows:

- Update the <Host-Path URL of OmniDocsWeb container> at the place of ibps5aurora.newgendocker.com in mapping.xml file located inside the OmniDocs11.0Web/Newgen/NGConfig/ngdbini/SSOConFig folder.
- Update the **CabinetName** in *mapping.xml* file located inside the *OmniDocs11.0Web/Newgen/NGConfig/ngdbini/SSOConFig* folder.
- Configure the CabinetName=DomainName in *sso.ini* file located inside the *OmniDocs11.0Web/Newgen/NGConfig/ngdbini/SSOConFig* folder.
- ecmsuite=eco.com

3.5.7 Scheduler changes

Prerequisite:

The cabinet is created and associated with the running containers. If the cabinet is not created, then refer to <u>Cabinet and Data Source Creation</u> section.

The changes in Scheduler are as follows:

Update the OmniDocs+RMS EJB container name [Defined in OmniDocs11.0EJB.yml file] in IS.ini in between the <endPointURL></endPointURL> tags file located inside the OD11.0Services or Scheduler folder at the mapped location on the Worker node. For example,

- Update the OmniDocs+RMS EJB container name [Defined in OmniDocs11.0EJB.yml file] in NGOClientData.xml, RMAppConfig.xml and RMClientData.xml in between the <endPointURL></endPointURL> tags file located inside the OD11.0Services/Scheduler/ngdbini folder at the mapped location on the Worker node. For example, <endPointURL>http://od110ejb:8080/callbroker/execute/GenericCallBroker</endPointURL> Here, od110ejb is the name of the OmniDocs+RMS EJB container.
- Update the OD11.0Services container's service name [Defined in respective YAML file] in *SchedulerConf.ini* file located at **OD11.0Services** or **Scheduler** folder at the mapped location. For example: **schedulerIpAddress=od110services**
- Update the OD11.0Services container's service name [Defined in respective YAML file] in eworkstyle.ini file located at OmniDocs11.0Web/Newgen/NGConfig/ngdbini/Custom/<CABINETNAME> folder at mapped location.

For example: schedularLocation=od110services

3.5.8 ThumbnailManager changes

Prerequisite:

The cabinet is created and associated with the running containers. If the cabinet is not created, then refer to <u>Cabinet and Data Source Creation</u> section.

The changes in ThumbnailManager are as follows:

Update the OmniDocs+RMS EJB container name [Defined in OmniDocs11.0EJB.yml file] in IS.ini in between the <endPointURL></endPointURL> tags file located inside the OD11.0Services or ThumbnailManager folder at the mapped location on the Worker node. For example,

<endPointURL>http://od110ejb:8080/callbroker/execute/GenericCallBroker</endPointURL> Here, od110ejb is the name of the OmniDocs+RMS EJB container.

 Update the OmniDocs+RMS EJB container name [Defined in OmniDocs11.0EJB.yml file] in NGOClientData.xml in between the <endPointURL></endPointURL> tags file located inside the OD11.0Services/ThumbnailManager/ngdbini folder at the mapped location on the Worker node.

For example,

• Update the cabinet name, supervisor group's user name, and password in *ThumnailConfig.xml* located inside the **OD11.0Services** or **ThumbnailManager** folder at the mapped location on the Worker node.

```
<cabinets><cabinet><cabinetname>ecmsuite</cabinetname><jtsip>127.0.0.1
</jtsip><jtsport>3333</jtsport><user>supervisor</user><password>:X-D;U:T-C;P-C;p5-C;b:
</password><BatchSize>10</BatchSize><priority>1</priority><encoding>UTF-8
</encoding></cabinet></cabinets>
```

```
Figure 3.31
```

3.5.9 TEM changes

Prerequisite:

The cabinet is created and associated with the running containers. If the cabinet is not created, then refer to <u>Cabinet and Data Source Creation</u> section.

The changes in TEM are as follows:

 Update the OmniDocs+RMS EJB container name [Defined in OmniDocs11.0EJB.yml file] in IS.ini and NGOClientData.xml in between the <endPointURL></endPointURL> tags file located inside the TEM11.0 folder at the mapped location on the Worker node. For example, <endPointURL>http://od110ejb:8080/callbroker/execute/GenericCallBroker</endPointURL>

- Update the cabinet name in filename FTSServer-CABINETNAME-1.properties. For example: FTSServer-**ecmsuite**-1.properties [ecmsuite is the cabinet name].
- Update the OmniDocs+RMS EJB container name [Defined in OmniDocs11.0EJB.yml file] in FTSServer-ecmsuite-1.properties.
- Update the OmniDocs supervisor group's user name.
- Update the OmniDocs supervisor group's user password. Ensure this password must be in an encrypted format.

```
ServerAddress=od110ejb
SiteId=1
UserName=supervisor
Password=:X-D;U:T-C;P-C;p5-C;b:d:
PollTime=10
OCRPath=tesseract
DocumentCount=1000
Language=eng
SleepTime=15
```


3.5.10 EasySearch changes

Prerequisite:

The cabinet is created and associated with the running containers. If the cabinet is not created, then refer to <u>Cabinet and Data Source Creation</u> section.

The changes in EasySearch are as follows:

- Docker Images for EasySearch (Only one is required based on the infrastructure availability)
 - EasySearch (Apache Manifold and ElasticSearch [freeware software])
 Or
 - > EasySearch (Apache Manifold only [freeware software] with AWS managed ElasticSearch)

The EasySearch container consists of Apache Manifold and ElasticSearch [both are freeware software]. You have another option as well where you can use ElasticSearch ManagedService on AWS. In such a case, the EasySearch container consists of Apache Manifold only [freeware software]. Hence, there are two Docker images for EasySearch:

- EasySearch (Apache Manifold and ElasticSearch)
- EasySearch (Apache Manifold only)

Based on the EasySearch Docker images, configuration changes are done accordingly.

EasySearch (Apache Manifold and ElasticSearch)

- Update the EasySearch container name [Defined in *EasySearch11.0.yml* file] against the
 ESServerIPAddress key in the *ESconfig.ini* file located inside the
 EasySearch11.0\ESConfigurator\conf folder at the mapped location on the Worker node.
 For example, ESServerIPAddress=easysearch11 [Where easysearch11 is the container name].
- Update the Database details in the *ESconfig.ini* file located inside the *EasySearch11.0**ESConfigurator**conf* folder at the mapped location on the Worker node.
 - ESClusterName=CABINETNAME_cluster
 - OdDBIPAddress=DBIP
 - OdDBPort=DBPORT
 - OdCabinetName=CABINETNAME
 - OdDBUserName=DBUSER
 - OdDBPassword=DBPASSWORD in encrypted format
 - OdDBType=sqlserver | oracle | postgres

```
ESServerIPAddress=easysearch11
ESServerTCPPort=9300
ESServerHttpPort=9200
ESProtocol=http
ESClusterName=ecmsuite_cluster
OdDBIPAddress=omnidocs-aurorards-db-instance-1-restore
OdDBPort=5432
OdCabinetName=ecmsuite
OdDBUserName=postgres
OdDBUserName=postgres
OdDBPassword=:X-D;Y-D;L-C;N-C;VSJ-C;4T-C;r
OdDBType=postgres
MCFIPAddress=127.0.0.1
```

Figure 3.33

- Update the cabinet name in the CrawlerConfig.xml file located inside the EasySearch11.0\apache-manifoldcf-2.19\example folder at the mapped location on the Worker node.
- Update the OmniDocs supervisor group's user name.
- Update the OmniDocs supervisor group's user password. Ensure this password must be in an encrypted format.

```
<cabinets>
<cabinets>
<cabinetame>ecmsuite</cabinetname>
<jtsip>127.0.0.1</jtsip>
<jtsport>3333</jtsport>
<user>supervisor</user>
<password>:X-D;U:T-C;P-C;p5-C;b:d</password>
<StopPhraseFlag>N</StopPhraseFlag>
<StopPhrases>
<StopPhrase>Newgen Software Technologies</StopPhrase>
<StopPhrase>omnidocs 11.0</StopPhrase>
</StopPhrases>
<Pages>ALL</Pages>
</cabinet>
</cabinets>
```



• Update the cabinet name in the *elasticsearch.yml* file located inside the *EasySearch11.0\elasticsearch-7.17.4\config* folder at the mapped location on the Worker node. For example,



Where, **ecmsuite** is the cabinet name.

• Update the cabinet name and EasySearch container name [Defined in *EasySearch11.0.yml* file] in the *EasySearch.xml* file located inside the *OmniDocs110Ejb\ngdbini* folder at the mapped location on the Worker node.

```
<ElasticSearch>
  <ElasticSearch>
  <ESIPAddress>easysearch11</ESIPAddress> <!--Adress of ElasticSearch Server -->
  <ESPort>9200</ESPort> <!--Port ElasticSearch Server running on -->
  <ESClusterName>ecmsuite_cluster</ESClusterName> <!-- Name of your cluster -->
  <ESUserName></ESUserName> <!-- Mandatory value for basic authentication -->
  <ESSecret></ESSecret>  <!-- Mandatory value for basic authentication -->
  <ESProtocol>http</ESProtocol> <!-- Set https for SSL/TLS configuration -->
  <ESTrustStorePath></ESTrustStorePath>  <!--Value optional for SSL/TLS configuration-->
  <ESTrustStoreSecret></ESTrustStoreSecret> <!--Value optional for SSL/TLS configuration-->
  </ElasticSearch>
```

Figure 3.36

Here, **easysearch11** is the container name and **ecmsuite** is the cabinet name.

 Update the EnableEasySearch=Y and EasySearch container name [Defined in EasySearch11.0.yml file] in the eworkstyle.ini file located inside the OmniDocs11.0Web\ Newgen\NGConfig\ngdbini\Custom\CABINET_NAME folder at the mapped location on the Worker node.



Figure 3.37

Here, **easysearch11** is the container name.

EasySearch (Apache Manifold Only)

In the case of the EasySearch (Apache Manifold only) container, use the AWS ElasticSearch Managed Service. Also, ensure that the Elastic Managed Service is already created before further processing.

- Update the EasySearch container name [Defined in EasySearch11.0_ApacheOnly.yml file] against the ESServerIPAddress key in the ESconfig.ini file located inside the EasySearch11.0\ESConfigurator\conf folder at the mapped location on the Worker node. For example, ESServerIPAddress=easysearch11 [Where easysearch11 is the container name].
- Update the Database details in the *ESconfig.ini* file located inside the *EasySearch11.0**ESConfigurator**conf* folder at the mapped location on the Worker node.
 - ESClusterName=CABINETNAME_cluster
 - OdDBIPAddress=DBIP
 - OdDBPort=DBPORT
 - OdCabinetName=CABINETNAME
 - OdDBUserName=DBUSER
 - OdDBPassword=DBPASSWORD in encrypted format
 - OdDBType=sqlserver | oracle | postgres

```
ESServerIPAddress=easysearch11
ESServerTCPPort=9300
ESServerHttpPort=9200
ESProtocol=http
ESClusterName=ecmsuite_cluster
OdDBIPAddress=omnidocs-aurorards-db-instance-1-restore
OdDBPort=5432
OdCabinetName=ecmsuite
OdDBUserName=postgres
OdDBPassword=:X-D;Y-D;L-C;N-C;VSJ-C;4T-C;r
OdDBType=postgres
MCFIPAddress=127.0.0.1
```



- Update the UseAWSElasticSearch=true in the *ESconfig.ini* file located inside the *EasySearch11.0\ESConfigurator\conf* folder at the mapped location on the Worker node.
- Update the AWSESDomainURL, AWSAccessKey, AWSSecretKey, and AWSRegion in the *ESconfig.ini* file located inside the *EasySearch11.0\ESConfigurator\conf* folder at the mapped location on the Worker node. For example,

```
UseAWSElasticSearch=true
AWSESDomainURL=<u>https://vpc-testdomain1.ap-south-1.es.amazonaws.com</u>
AWSAccessKey=AKTABGAGGATALDGUS7DN
AWSSecretKey=LewegingesenK2TestastAlusterThatGUK4sbyy
AWSRegion=ap-south-1
```

Figure 3.39

Update the cabinet name in the *CrawlerConfig.xml* file located inside the
 EasySearch11.0\apache-manifoldcf-2.19\example folder at the mapped location on the
 Worker node.

```
<cabinets>
<cabinets>
<cabinetname>ecmsuite</cabinetname>
<jtsip>127.0.0.1</jtsip>
<jtsport>3333</jtsport>
<user>supervisor</user>
<password>:X-D;U:T-C;P-C;p5-C;b:d</password>
<StopPhraseFlag>N</StopPhraseFlag>
<StopPhrases>
<StopPhrase>Newgen Software Technologies</StopPhrase>
<StopPhrase>omnidocs 11.0</StopPhrase>
</stopPhrases>
</stopPhrases>
</cabinets>
```

Figure 3.40

 Update the cabinet name and EasySearch container name [Defined in EasySearch11.0_ApacheOnly.yml file] in the EasySearch.xml file located inside the OmniDocs11.0Ejb\ngdbini folder at the mapped location on the Worker node. For example,

Figure 3.41

Here, **easysearch11** is the container name and ecmsuite is the cabinet name.

- Update the **<AWSElasticSearch value= "true">** in the *EasySearch.xml* file located inside the *OmniDocs11.0Ejb\ngdbini* folder at the mapped location on the Worker node.
- Update the AWSESDomainURL, AWSAccessKey, AWSSecretKey, and AWSRegion in the EasySearch.xml file located inside the OmniDocs11.0Ejb\ngdbini folder at the mapped location on the Worker node.

For example,



Figure 3.42

 Update the EnableEasySearch=Y and EasySearch container name [Defined in EasySearch11.0.yml file] in the eworkstyle.ini file located inside the OmniDocs11.0Web\ Newgen\NGConfig\ngdbini\Custom\CABINET_NAME folder at the mapped location on the Worker node.

For example,



Figure 3.43

Here, **easysearch11** is the container name.

3.5.11 WOPI changes

 Update the OmniDocsEJB container name [Defined in OmniDocs11.0EJB.yml file] in NGOClientData.xml file in between the <endPointURL></endPointURL> tags located inside the OmniDocs_WOPI\Newgen\NGConfig\ngdbini folder at the mapped location on the Worker node.

xml version="1.0"?
<clientinfo></clientinfo>
<providerurl></providerurl>
<jndiservername></jndiservername>
<jndiserverport></jndiserverport>
<contextsuffix></contextsuffix>
<wildflyusername></wildflyusername>
<wildflypassword></wildflypassword>
<jndicontextfactory></jndicontextfactory>
<clientlookupname>ejb:omnidocs_ejb/omnidocs_ejb/NGOClientServiceHandlerBean!com.newgen.omni.jts.txn.NGOClientServiceHandlerHome</clientlookupname>
<adminlookupname>ejb:omnidocs_ejb/omnidocs_ejb/NGOAdminServiceHandlerBean!com.newgen.omni.jts.txn.NGOAdminServiceHandlerHome</adminlookupname>
<urlpackageprefix></urlpackageprefix>
<endpointurl>http://rms40ejb:8080/callbroker/execute/GenericCallBroker</endpointurl>
<mlparamname>InputXML</mlparamname>
<headerkey></headerkey>
<headervalue></headervalue>

Figure 3.44

Here, **rms40ejb** is the name of the OmniDocsEJB container.

 Update the OmniDocsEJB container name [Defined in OmniDocs11.0EJB.yml file] in IS.ini file in between the <endPointURL ></endPointURL > tags located inside the OmniDocs_WOPI\Newgen\NGConfig folder at the mapped location on the Worker node. For example,

<encoding>UTF-8</encoding>
<logpath>Replication.log</logpath>
<smstimeout>60000</smstimeout>
<smsreadinterval>30000</smsreadinterval>
<smsretrycount>5</smsretrycount>
<smsgeneratelog>true</smsgeneratelog>
<isjndi>true</isjndi>
<pre><endpointurl>http://rms40ejb:8080/callbroker/execute/GenericCallBroker</endpointurl></pre>
<xmlparamname>InputXML</xmlparamname>
<headerkey></headerkey>
<headervalue></headervalue>
<providerurl></providerurl>

Figure 3.45

• Update the WOPI_SOURCE, RMS_REDIRECTURL and CABINETNAME in *WOPIConfiguration.ini* file located inside the *OmniDocs_WOPI\Newgen\NGConfig\AddInsConfig* folder at the mapped location on the Worker node.



Figure 3.46

Where,

https://wopi.newgendocker.com is host URL of WOPI container. https://rms40.newgendocker.com is Host URL of RMS WEB container. **rmspostgres27feb** is cabinet name.

- Open the *web.xml* file in edit mode located inside the *OmniDocs_WOPI* folder at the mapped location on the Worker node.
- Search for filter-class <filter-class>org.apache.catalina.filters.CorsFilter</filter-class> and update the <param-value></param-value> tag's value with OmniDocs URL against <paramname> antiClickJackingUri</param-name> and * against <paramname>cors.allowed.origins</param-name>

<filter></filter>
<filter-name>httpHeaderSecurity</filter-name>
<filter-class>org.apache.catalina.filters.HttpHeaderSecurityFilter-class></filter-class>
<async-supported>true</async-supported>
<init-param></init-param>
<pre><pre>aram-name>antiClickJackingOption</pre>/param-name></pre>
<pre><pre>ram-value>ALLOW-FROM</pre></pre>
<init-param></init-param>
<pre><param-name>antiClickJackingUri</param-name></pre>
<pre><pre><pre>cparam-value>omnidocs11alpine2.newgendocker.com</pre></pre></pre>
<pre><filter></filter></pre>
<filter-name>CorsFilter</filter-name>
<filter-class>org.apache.catalina.filters.CorsFilter</filter-class>
<init-param></init-param>
<pre><param-name>cors.allowed.origins</param-name></pre> /param-name>
<pre><pre><pre></pre></pre></pre>
<init-param></init-param>
<pre><pre><pre>cors.allowed.methods</pre>/param-name></pre></pre>
<pre><pre><pre><pre><pre><pre><pre><pre></pre></pre></pre></pre></pre></pre></pre></pre>
<init-param></init-param>
<pre><pre>cors.allowed.headers</pre>/param-name></pre>
<pre><pre>cparam-value>Content-Type,X-Requested-With,accept,Origin,Access-Control-Request-Method,Access-Control-Request-Headers,Access-Control-Allow-Origin</pre></pre>

Figure 3.47

• Add the CSPHeaderAllowedDomains tag in the *eworkstyle.ini* file located inside the *OmniDocs11.0Web/Newgen/NGConfig/ngdbini/odwebini* folder at the mapped location on the Worker node.

• Add the WOPIOfficeExtensionSuppport and WOPIOfficeExtensionSuppportURL tag in the *eworkstyle.ini* file located inside the *OmniDocs11.0Web/Newgen/NGConfig/ngdbini/Custom/CABINET_NAME* folder at the mapped location on the Worker node.

WOPIOfficeExtensionSuppport=doc,docx,DOCX,DOC,xls,xlsx,XLSX,XLS,ppt,pptx,PPTX,PPT,wopitest,WOPITEST,wopitestx,WOPITESTX WOPIOfficeExtensionSuppportURL=https://wopi.newgendocker.com

3.5.12 OmniScanWeb changes

The changes in OmniScanWeb are as follows:

Update the AWS Elastic Redis cache's configuration endpoint in *redisson.yaml* file against the singleServerConfig or clusterServersConfig. If redis cache is SSL enabled, then use the *rediss://<endpoint url>:port* and if SSL is not enabled, then use *redis://<endpoint url>:port*. This *redisson.yaml* file is located inside the OmniScanWeb6.0 folder at the mapped location on the Worker node.

```
singleServerConfig:
 idleConnectionTimeout: 10000
 connectTimeout: 10000
 timeout: 3000
 retryAttempts: 3
 retryInterval: 1500
 password: null
 subscriptionsPerConnection: 5
 clientName: null
 address: "redis://
                                         .com:6379"
 subscriptionConnectionMinimumIdleSize: 1
 subscriptionConnectionPoolSize: 50
 connectionMinimumIdleSize: 24
 connectionPoolSize: 64
 database: 0
 dnsMonitoringInterval: 5000
hreads: 16
hettyThreads: 32
odec: !<org.redisson.codec.MarshallingCodec> {}
transportMode: "NIO"
Reference: https://github.com/redisson/redisson/wiki/2.-Configuration#26-single-instance-mode
CLUSTER ---
CLUSTER clusterServersConfig:
CLUSTER idleConnectionTimeout: 10000
CLUSTER connectTimeout: 10000
CLUSTER timeout: 3000
CLUSTER retryAttempts: 3
CLUSTER retryInterval: 1500
CLUSTER failedSlaveReconnectionInterval: 3000
CLUSTER failedSlaveCheckInterval: 60000
CLUSTER password: null
CLUSTER subscriptionsPerConnection: 5
CLUSTER clientName: null
CLUSTER loadBalancer: !<org.redisson.connection.balancer.RoundRobinLoadBalancer> {}
CLUSTER
        subscriptionConnectionMinimumIdleSize: 1
```



3.5.13 RMS SharePoint Adapter changes

Prerequisite:

The cabinet must be created and associated with the running containers. To create a cabinet, refer to the <u>Creating cabinet and data source</u> section.

The changes in RMS SharePoint Adapter are as follows:

- Update the cabinet name in filename RMS-SPServer-CABINETNAME-1.properties located inside SharePointAdapter/properties folder at the mapped location on the worker node. For example, RMS-SPServer-ecmsuite-1.properties [ecmsuite is the cabinet name].
- 2. Change the required changes in *RMS-SPServer-CABINETNAME-1.properties* file. Following is the sample of the Properties file:

```
AutoArchiveActive=Y
OnlineSharePointSite=Y
RMSCabinetName =spcab2
RMSServerAddress=omnidocs.newgendocker.com
RMSServerPort=443
RMSServerProtocols=https
RMSUserName=supervisor2
RMSPassword=:X-D;U:T-C;P-C;p5-C;b:d:u:SJDE
sharePointUserDomain=vwm55.onmicrosoft.com
sharePointHostIPAddress=vwm55.onmicrosoft.com
shraePointPort=443
sharePointProtocols=https
sharePointSiteAbsoluteURL=
sharePointUser=spadmin@vwm55.onmicrosoft.com
sharePointSequence=:F:0s-C;6-C;G-C;a:8:y-D;W:Z-C;4P-C;a-C;k:0
sharePointSite=newSite
SharePointReconciliationReports=Y
SharePointReportsLibrary=/sites/newSite/SharePointRMSReport/
MaxFailerCount=3
```

Description of the Properties keys:

Keys	Description
AutoArchiveActive	Y signifies this properties file is in used whereas N signifies this file is not in use.
OnlineSharePointSite	Y – Signifies online method is working.
Ommeshareromtsite	N – Signifies offline method is working.
RMSCabinetName	Name of the RMS cabinet.
RMSServerAddress	RMS web container host URL.
RMSServerPort	RMS web container port that is, 443 = if SSL enabled or 80 = if SSL Disabled.
RMSServerProtocols	RMS server protocol that is, https = if SSL enabled or http = if SSL Disabled.
RMSUserName	RMS supervisor's group user name.
RMSPassword	RMS supervisor's group user encrypt password from alarm mailer.
sharePointUserDomain	SharePoint domain name.
sharePointHostIPAddress	SharePoint site server address.
shraePointPort	SharePoint site server port.
sharePointProtocols	SharePoint site server protocols.
sharePointSiteAbsoluteURL	SharePoint site absolute URL.
sharePointUser	SharePoint admin user name.
sharePointSequence	SharePoint encrypt password from alarm mailer.
sharePointSite	SharePoint site name.
SharePointReconciliationReports	Y or N.
SharePointReportsLibrary	SharePoint Reconciliation Reports absolute path.
MaxFailerCount	Maximum failure count.

```
AutoArchiveActive=Y
OnlineSharePointSite=Y
RMSCabinetName=rms23dec
RMSServerAddress=omnidocs.newgendocker.com
RMSServerPort=443
RMSServerProtocols=https
RMSUserName=supervisor2
RMSPassword=:137-C;q;J-D;lX-C;6
sharePointUserDomain=vwm55.onmicrosoft.com
sharePointHostIPAddress=vwm55.onmicrosoft.com
shraePointPort=443
sharePointProtocols=https
sharePointSiteAbsoluteURL=
sharePointUser=spadmin@vwm55.onmicrosoft.com
sharePointSequence=:X-D;Y-D;N-C;VSJ-C;4T-C;r
sharePointSite=SharpeointRMssite
SharePointReconciliationReports=Y
SharePointReportsLibrary=/site/SharpeointRMssite/SharePointRMSReport/
MaxFailerCount=3
```

3. Update **site name**, **tenantID**, **ClientID** and **clientSecret** in file SharePointConfig.ini located inside the *OmniDocs11.0Web\Newgen\NGConfig\AddInsConfig* folder at the mapped location on the Worker node.

For Example,

Figure 3.50

NOTE:

The TenantID, ClientID and clientSecret must be in encrypted format.

3.5.14 Messaging Service changes

Prerequisite:

A cabinet must be created. To create a cabinet, refer to the <u>Creating cabinet and data source</u> section.

The changes in Messaging Service are as follows:

 Update the Message Service Hosted URL in hostHeaderWhitelist in the application.properties file located inside the *MessagingService* folder at the mapped location on the Worker node.
 For Example –





 Update the Database details in the ServerInfo.json file located inside the MessagingService/omniconf folder at the mapped location on the Worker node. For Example –

```
"cabinet_name": {
    "userName": "postgres",
```

```
"password": ":X-D;Y-D;L-C-C;VSJ-C;4T-C;s",
        "maxDBConnections": 50,
        "minDBConnections": 5,
        "querytimeout": 1000,
        "cabinetType": "B",
        "databaseType": "POSTGRES",
        "port": 5432,
        "service": "",
        "comment": "postgres",
        "driverName": "org.postgresql.Driver",
        "driverURL": "jdbc:postgresql://auroxxx-xxxxxxxx-xx.cluster-
cvxxxxxxekwxu.ap-south-1.rds.amazonaws.com:5432/cabinet name",
        "tnsname": "",
        "idleTimeout":600000,
        "maxLifetime":1800000,
        "connectionTimeout": 300000,
        "autoCommit": true
```

Parameters	Description
cabinet_name	The name of the cabinet. Provide the cabinet name in lowercase letters.
userName	Database username.
Password	The password of the above user in encrypted form. Encrypt the password as follows:
	For Windows:
	Encrypt the password using the <i>encryptPassword.bat</i> file located at <i>OD11\Common</i> Services for J2EE\AlarmMailer\encryptPassword.bat.
	For Linux:
	Go to the <i>encryptPassword.sh</i> file located at <i>OD11\Common Services for</i> <i>J2EE\AlarmMailer\encryptPassword.sh</i> and provide full rights using the below command for password encryption:
	chmod 777 encryptPassword.sh.
maxDBConnections	The maximum number of allowed database connections.
minDBConnections	The minimum number of allowed database connections.
querytimeout	It defines the timeout period of the responding database in minutes.
	Example – 6
cabinetType	It defines the type of cabinet.
	Example - B or I or D
databaseType	It defines the type of database cabinet.
	Example – POSTGRES
port	Provide the port number of the database server.
service	NA
comment	It is used to provide related comments at the time of cabinet creation.
driverName	The name of the class that implements a protocol JDBC for a database connection.
	Example: "org.postgresql.Driver

Parameters	Description
driverURL	It contains information about the database to connect and other configuration properties.
	Example:
	"jdbc:postgresql:// <ip address="" database="" host="" name="" of="" or="" server="" the="">:<port database="" number="" of="" server="" the="">/<cabinet name="">"</cabinet></port></ip>
tnsname	It is the configuration file that contains network service names mapped to connect descriptors for the local naming method, or net service names mapped to the listener
	protocol addresses.
idleTimeout	It controls the maximum amount of time that a connection is allowed to sit idle in the pool. The minimum allowed value is 10000ms (10 seconds). Default: 600000 (10 minutes).
maxLifetime	It controls the maximum lifetime of a connection in the pool. A value of 0 indicates no maximum lifetime (infinite lifetime), subject of course to the idleTimeout setting. The minimum allowed value is 30000ms (30 seconds). Default: 1800000 (30 minutes).
connectionTimeout	It controls the maximum number of milliseconds that a client will wait for a connection from the pool. The lowest acceptable connection timeout is 250 ms. Default: 30000 (30 seconds).
autoCommit	It controls the default auto-commit behavior of connections returned from the pool. It is a boolean value. Default: true.

3.6 Deploying containers

Perform the below steps to deploy the containers:

 You can deploy the containers on AWS Elastic Kubernetes Service from our local machine by executing the below command or you can deploy them using AWS CodePipeline. However, it recommends deploying the containers using AWS CodePipeline for better traceability. kubectl apply -f <YAML_File>

```
For example,
kubectl apply -f OmniDocs11.0Web.yml
```

```
NOTE:
```

- To execute the above command, kubectl must be configured on your local server. Refer to the <u>Configuration</u> <u>of AWS Kubernetes Cluster</u> section to run kubectl from your local machine.
- To deploy the containers using AWS CodePipeline, AWS CodePipeline must be configured. Refer to the <u>Configuration of AWS CodePipeline</u> section.
- In AWS CodePipeline, a separate Release pipeline is created for each Docker image like OmniDocs11.0Web, OmniDocs11.0WebService, OmniDocs11.0EJB, OmniDocs11.0Services, EasySearch11.0, TEM11.0, and OmniScanWeb6.0.
 For Example

aws Services 🔻	9	Q Search for service	es, features, marketplace products, and docs	[Alt+S]	∑ ♣ [*] omnidocs @ ngdev ▼ Mumbai ▼ Supp
Developer Tools X CodePipeline	Develop	er Tools 〉 CodePipe	line > Pipelines		
	Pipe	elines Info	C 🖓 Notify 💌	View history Release change	Delete pipeline Create pipeline
Source + CodeCommit	٩				< 1 > @
Build • CodeBuild		Name	Most recent execution	Latest source revisions	Last executed
Deploy • CodeDeploy		OmniDocs1015e rvices-DevStage	⊘ Succeeded	Source – 7fb6f314: ok AWS-ECR-Registry – sha256:1:	9 days ago
 Pipeline CodePipeline Getting started 		TEM101- DevStage	⊘ Succeeded	Source – 7fb6f314: ok AWS-ECR-Registry – sha256:d:	9 days ago
Pipelines		OmniDocs101EJ B-DevStage	⊘ Succeeded	Source - 7fb6f314: ok AWS-ECR-Registry - sha256:5:	9 days ago
Settings		OmniDocs101W eb-DevStage	⊘ Succeeded	AWS-ECR-Registry – sha256:9: Source – 7fb6f314: ok	9 days ago
Q, Go to resource		OmniScanWeb5 0-DevStage	⊘ Succeeded	AWS-ECR-Registry – sha256:8: Source – 7fb6f314: ok	10 days ago
E Feedback		EasySearch101- DevStage	⊘ Succeeded	AWS-ECR-Registry – sha256:b: Source – 7fb6f314: ok	9 days ago
		OmniDocs101W eb-ProdStage	⊘ Succeeded	AWS-ECR-Registry – sha256:d: Source – 7fb6f314: ok	14 days ago
		OmniDocs101W eb-UATStage	⊘ Succeeded	Source - 7fb6f314: ok	9 days ago



- 3. Trigger the Release Pipeline to deploy the required Docker containers.
- 4. Once the deployment is done, deployed containers can be visible from the Kubernetes Dashboard. Refer to the <u>Configuration of AWS Kubernetes Cluster</u> to configure the Kubernetes Dashboard.

🛞 kubernetes		Q Se	arch							+	٠	θ
≡ Workloads > Pods												
Cluster	Pods										÷	
Cluster Roles Namespaces	Name 🕇		Namespace	Labels	Node	Status	Restarts	CPU Usage (cores)	Memory Usage (bytes)	Age		
Nodes Persistent Volumes	easysearch101-5d9f	8d6458-ns8j9	default	app: easysearch101 pod-template-hash: 5d9f8d6458	ip-10-0-2-168.ap- south- 1.compute.interna	Running I	0	-	-	13.days		:
Storage Classes	ibps5siejb-55f55949	6-f5rt9	default	app: ibps5slejb pod-template-hash: 55f559496	ip-10-0-2-168.ap- south- 1.compute.interna	Running	0	-		a.month		:
Namespace default	ibps5siweb-698f8dfc	c96-bcw2d	default	app: ibps5siweb pod-template-hash: 698f8dfc96	ip-10-0-1-186.ap- south- 1.compute.interna	Running	0		-	.7.days		:
Overview	ibps5siweb-698f8dfc	c96-sitvm	default	app: ibps5siweb pod-template-hash: 698f8dfc96	ip-10-0-2-168.ap- south- 1.compute.interna	Running	0			a.month		:
Cron Jobs	odeserver-8d547b8	45-75ck5	default	app: nodeserver pod-template-hash: 8d547b845	ip-10-0-2-168.ap- south- 1.compute.interna	Running	0			2.days		:
Deployments	od101ejb-58658d484	4d-nt8fk	default	app: od101ejb pod-template-hash: 58658d484d	ip-10-0-1-186.ap- south- 1.compute.interna	Running	0		-	a.montb		:
Pods Replica Sets	od101services-648f6	6bbff6-lgdm7	default	app: od101services pod-template-hash: 648f6bbff6	ip-10-0-2-168.ap- south- 1.compute.interna	Running	0			38.seconds		:
Replication Controllers Stateful Sets	od101web-65786d4	dcc-5fgpn	default	app: od101web pod-template-hash: 65786d4dcc	ip-10-0-1-186.ap- south- 1.compute.interna	Running	0			23.days		:
Discovery and Load Balancing	omsejb-559555dd8c	⊱nkc9j	default	app: omsejb pod-template-hash: 559555dd8c	ip-10-0-1-186.ap- south- 1 compute interna	Running	0	-	-	19.days		:



5. Update the container's replica set from **1** (default value) to any other number in YAML files, then that number of containers is listed in Kubernetes Dashboard.

6. Increase the replica set from Kubernetes Dashboard from the **Deployments** menu on the leftpanel.

For Example,

🛞 kubernetes		Q Search				+ 🌲	Θ
Cluster Cluster Roles	Deployments					₹	•
Namespaces Nodes	Name od101services	default	Labels	Pods	Age ↑ 8.minutes	Images newgencicdpiepline.azurecr.io/od10.1ser vices.latest	:
Persistent Volumes Storage Classes	easysearch101	Scale a resource			.13.days	newgencicdpiepline.azurecr.io/easysearc h10.1:latest	:
Namespace	omsservice	deployment od101web w	vill be updated to reflect the	desired replicas count.	.15.days	newgencicdpiepline.azurecr.lo/omsservic e:1.3	:
default ~	ondeserver nodeserver	Desired replicas	Actual replicas		a.month	newgencicdpiepline.azurecr.lo/nodeserve r:latest	:
Overview	omsweb				.a.month	newgencicdpiepline.azurecr.io/omsweb: 1.3	:
Workloads	od101web	This action is equivale replicas=2	entto:kubectl scale -n defa	ault deployment od101web	a.month	newgencicdpiepline.azurecr.io/omnidocs 10.1web:dist3	:
Cron Jobs Daemon Sets	omsejb	Scale Cancel			.a.month	newgencicdpiepline.azurecr.io/omsejb:1. 1	:
Deployments	od101ejb	oerauit	app: od tu tejo	17.1	a.month	newgencicdpiepline.azurecr.io/omnidocs 10.1ejb:patch2hf05	:
Jobs Pods	ibps5siweb	default	app: ibps5siweb	2/2	2.months	newgencicdpiepline.azurecr.io/ibps5servi ceinstanceweb:latest	÷
Replica Sets	ibps5siejb	default	app: ibps5siejb	1/1	3.months	newgencicdpiepline.azurecr.io/ibps5servi ceinstanceejb:latest	:
Replication Controllers Stateful Sets						1 - 10 of 10 < < >	>1
Discovery and Load Balancing							



- 7. But, when you redeploy the containers from AWS CodePIpeline, the replica set increased from Kubernetes gets overwritten by the replica set defined in the YAML file.
- 8. In any case, if you need to restart the container, then you have two options. Either redeploy the container from AWS CodePipeline which launches the new container by following up the rolling update feature of Kubernetes or execute the restart command from the Kubernetes pod's shell.
- 9. To execute the restart command from the Kubernetes pod's shell, follow the below steps:
 - i. Open the Kubernetes Dashboard and list out all the deployed pods.
 - ii. Click **Pod** that you want to restart.
 - iii. Click **Exec into pod** icon given on upper-right panel.

🛞 kubernetes	Q Search		+	۵	θ
	od101web-65786d4dcc-5fgpn	=	Ð	1	
Cluster	Metadata		ixec into pod		
Cluster Holes Namespace Persistent Volumes Storage Classes Namespace default	Name od101web-65786d4dcc-5fgpn default Jul 20, 2020 23 days 6fe68e70-ca42-11ea-9d3b-0a4cc3ea1b34 Labels app: od10web pod-template-hash: 65786d4dcc Annotations Rubernetes lo/psp: eks.prtvilleged				
Overview	Resource information				
Workloads Cron Jobs	Node ip-10-0-1-186.ap-south-1.compute.internal Running 10.0.1.221 Burstable 0				
Daemon Sets Deployments	Conditions				



- > After that pod's shell terminal opens.
- Execute the below command to restart the container: restartjws.sh
- > The restart command is different for each container. Refer to the below table:

Container Name	Restart Command
OmniDocs11.0Web, OmniDocs11.0WebService	restartjws.sh
OmniDocs11.0EJB	restartjboss.sh
OmniDocs11.0Services	restartalarm.sh, restartauthmgr.sh,
	restartscheduler.sh, restart thumbnail.sh,
	restartwrapper.sh
EasySearch11.0(With ElasticSearch)	restarteasysearch.sh
EasySearch11.0(Without ElasticSearch)	restartapache.sh
TEM11.0	restarttem.sh
OmniScanWeb6.0	restartjws.sh
RMSSharePointAdapter	Restartsharepointadapter.sh
Messaging Service	Restartservice.sh

10. Once the EasySearch11 container is deployed (Whether With ElasticSearch or Without ElasticSearch), execute the below command in Kubernetes pod's shell for the 1st time to configure the Apache Manifold jobs. After that in subsequent deployments, this execution is not required.

runESConfigurator.sh

3.7 Creating cabinet and data source

Prerequisites:

- OmniDocs+RMS Web, OmniDocs+RMS EJB, and OmniDocsServices are already deployed.
- ALB Ingress Controller is already configured and deployed using the AWS_ALB-IngressController.yml file.
- S3 bucket is already created to store the PN files. PN files are encrypted files that contain all the added, uploaded, and scanned documents by Newgen products.

Once the above prerequisites are fulfilled, refer the below sections to create the Cabinet and Data Source.

- Getting started with OSA
- <u>Registering JTS server</u>
- <u>Connecting OSA to JTS Server</u>
- Creating a cabinet
- <u>Associating a cabinet</u>
- Creating a data source
- <u>Registering a cabinet in OmniDocs</u>
- <u>Registering a cabinet in RMS</u>
- Creating Site and Volume

3.7.1 Getting started with OSA

Perform the below steps to start the OSA:

- Since the container is a CLI-based deployment you can't launch any GUI-based application inside the container. But you must use the OSA to create a cabinet that is a GUI-based application. In such a case, deploy OSA to some GUI-based machine either on a local server or on an EC2 instance. Also, add an inbound rule in the Kubernetes worker node's security group to allow OSA to communicate with the OmniDocs11.0Services container deployed on that worker node.
- 2. Once OSA is deployed on a machine, navigate to the OSA folder on that machine and double Click RunAdmin.bat (For Windows) or RunAdmin.sh (For Linux) to start OSA.
- 3. When the application is launched. The Login dialog box appears.

<u>U</u> ser		System	
<u>P</u> assword			
	OK	Cancel	

- 4. Select the user as **System** and specify the password as **system**.
- 5. Click **OK** to log in. After the successful login, the OSA screen appears displaying the list of registered services.

erver Heln	Administration				0000		×
- Servers	Services						
	Select Service	<all></all>	-				
		Server		Location			S
		Stop	Register	Unregister	Connect	Dis	
	Start				the second se		conne
	Start						conne

Figure 3.57

3.7.2 Registering JTS server

Perform the below steps to register the JTS Server:

1. To register the JTS server, click **Register** button. The **Register New Server** dialog box appears.

inter the details	5		
<u>Server</u> Type	JTS		-
IP Address			
Admin <u>P</u> ort			
	ОК	Cancel	



- 2. Select the JTS and specify the public IP address of the Kubernetes Worker node on which the OmniDocs11.0Services (Wrapper, AlarmMailer, THN, and so on) container is deployed. For example, suppose there are two worker nodes in the Kubernetes cluster and after deploying the OmniDocs11.0Services container, it gets deployed to the 1st worker node then specify the IP address of the 1st worker node. But in a case, 2 replicas are deployed on the OmniDocs11.0Services container, one on each worker node, in that case, specify the IP address of any worker node.
- 3. Specify the Admin port of Wrapper service running inside the OmniDocs11.0Services container. Since Wrapper is running inside the container with Admin port 9996 but that Admin port cannot be accessed directly. Kubernetes generates a random port (aka NodePort) for each port running inside the container that is exposed outside the container for public use. To get this NodePort either from Kubernetes Dashboard or by executing the below command from your local machine:

```
kubectl get svc <OmniDocs11.0Services container name>
For example,
```



Figure 3.59

Here, **Wrapper Admin port 9990** is exposed outside the container and Kubernetes has generated a random port 31370 as a NodePort. This NodePort keeps changing whenever you redeploy the container.

Register New S	erver		\times
Enter the details	5		
Server Type	JTS		-
IP Address	35.154.27.	245	
Admin Port	31370		
	ок	Cancel	

Figure 3.60

4. Click **OK** to register the JTS Server.

3.7.3 Connecting OSA to JTS Server

Perform the below steps to connect the OSA to the JTS Server:

1. Once the JTS Server is registered, it is displayed in the list in a disconnected state.

- Servers	Services					
Jan dervers						
	Select Service	e <all></all>	•			
		Server		Location		Sta
	JTS		35.154.27.2	45 : 31370	Disconne	cted
	Start	Stop	<u>R</u> egister	Unregister	Connect	<u>)</u> isconnec

Figure 3.61

- 2. Select the registered JTS Server and click **Connect**. Once JTS is connected, the **Manage** button gets enabled.
- 3. Click **Manage** button, after clicking on the Manage button, an entry of the connected JTS server along with its IP Address is displayed on the upper-left panel in the repository view.
- 4. Select the JTS from the repository view. The list of already created and associated cabinets, appears.

@ OmniDocs Service Adminis	tration			- 0	×
Server JTS Help					
♀ Bervers ♀ □ JTS (35.154.27.245 : 1) ↓ □ ↓ Locks ↓ □ ↓ Locks ↓ ↓ ↓ Users	Server Information IP Address Encoding Log IV Transaction Connect	s 35 154 27 245 1 UIF-8 V ion V Xml V Error Socke	Ser Soc <u>k</u> etTimeor t Error Log Size (MB) 10	verPort 3333	Adm <u>B</u> at
	Cabinet Transaction Pool	Start	Stop Disconnect Ea	<u>C</u> reate	
	CabinetName	MinDBConnections	MaxDBConnections	CabinetType	and a set
	ecmsuitesqi	10	10	Both Document Database and Ima	mssqi
	Cabinet Operations Associate Djssociate	Compile SP Property	Test Dejete Uniock	Enable <u>T</u> race Upgrad	e <u>L</u> icense
	•				•
Status Cabinet List fetched successfully					Ready

Figure 3.62

3.7.4 Creating a cabinet

Perform the below steps to create a cabinet:

For MSSQL:

1. Click **Create**. The Create Cabinet dialog box appears.

Create Cabinet (35.154.27.2	45 : 31370)	Х
Cabinet Type		
O Document datab	ase 🔘 Image Server database 💿 Both	
Database Type		
MSSQL / Amazon	RDS Oracle OPostgres OAzure	
MSSQL Information		
Device Size (N	1B) 5 Log Size (MB) 5	
Cabinet information		
Cabinet Name		
Server Name		
<u>U</u> ser name		
Password		
Database <u>P</u> ath	.mdf	
CD Key		
Security Level	Object Level 💌	
Password Algorithm	PC1 💌	
	Enable <u>F</u> TS	
Status		
Julus		
	OK Cancel	

Figure 3.63

- 2. Select the cabinet type that needs to be created from the Cabinet Type area. The Cabinet can be a **Document database**, an **Image server database**, or both.
- 3. Select the database option from the Database Type section.
- 4. Specify the initial database size in the **Device Size** textbox and specify the initial log size in the **Log Size** textbox. Else, continue with the default values.
- 5. Specify the following cabinet information:
 - Specify the cabinet name in the **Cabinet Name** textbox.
 - Specify the server name (name of the machine where the MS SQL server is running) in the **Server I.P.** textbox.
 - Specify the username in the **User name** textbox.
 - Specify the password in the **Password** textbox.
 - Specify the CD key in the **CD Key** textbox.
 - Select the **Enable FTS** checkbox.

NOTE:

In the case of MSSQL if the Database port is not equal to 1433 (Default port) update the database port in the *DatabaseDriver.xml* file located inside the OmniDocs11.0Ejb/ngdbini folder at the mapped location on the worker node before creating the cabinet.

Create Cabinet (35.154.27.2	45 : 31370)	>
Cabinet Type		
O Document datab	ase 🔘 Image Server database 🔘 Both	
Database Type		
MSSQL / Amazon	RDS Oracle O Postgres O Azure	
MSSQL Information		
Device Size (N	IB) 5 Log Size (MB) 5	
Cabinet information		
C <u>a</u> binet Name	ecmsuite	
Server Name	10.0.1.43	
<u>U</u> ser name	applogin	
Pass <u>w</u> ord	•••••	
Database <u>P</u> ath	ecmsuite.mdf	
CD Key	28GQI0YDOyA0iokFMtD~q8old6izYz0v6ek	1M
Security Level	Object Level	•
Password Algorithm	PC1	•
	✓ Enable <u>F</u> TS	
Status		
	OK Cancel	

Figure 3.64

- 6. Click **OK** to create the cabinet. The Cabinet created successfully dialog appears. **For Aurora PostgreSQL:**
- 1. Click **Create**. The Create Cabinet dialog box appears.

Create Cabinet (13.127.66.151 :	30896)	×
Cabinet Type		
O Document d	atabase 🔘 Image Server database 💿 Both	
Database Type		
MSSQL / Amazon RD	S 🔾 Oracle 🖲 Postgres 🔾 Azure 🔾 Oracle	RAC
Postgres Information		
	Po <u>r</u> t 5432	
Cabinet information		
Cabinet Name		
<u>S</u> erver I.P.		
<u>U</u> ser name		
Pass <u>w</u> ord		
Database <u>P</u> ath	PGDATA	
CD <u>K</u> ey		
Security <u>L</u> evel	Object Level	-
Password Algorithm	PC1	-
	1	
Status		
	OK Cancel	

Figure 3.65

- 2. Select the cabinet type that needs to be created from the Cabinet Type area. The Cabinet can be a **Document database**, an **Image server database**, or both.
- 3. Select the database option from the Database Type section.
- 4. Specify the port number if default port 5432 is not used.
- 5. Specify the following cabinet information:
 - Specify the cabinet name in the **Cabinet Name** textbox.
 - Specify the Aurora PostgreSQL server name in the Server I.P. textbox.
 - Specify the username in the **User name** textbox.
 - Specify the password in the **Password** textbox.
 - Specify the CD key in the **CD Key** textbox.

Create Cabinet (13.127.66.151 :	30896)	×
Cabinet Type		
O Document d	atabase 🔘 Image Server database 💿 Both	
Database Type		
MSSQL / Amazon RD	S 🔾 Oracle 🖲 Postgres 🔾 Azure 📿 OracleRAC	
Postgres Information		
	Port 5432	
Cabinet information		
Cabinet Name	ibpsaurora12dec]
<u>S</u> erver I.P.	uster-cv4updtekwxu.ap-south-1.rds.amazonaws.com	n
<u>U</u> ser name	postgres	
Password	•••••]
Database <u>P</u> ath	PGDATA]
CD <u>K</u> ey	CK1ESV4IWITS0X2CM1HOLLS. W0052nZ-WmyYiuOJZ	ā
Security <u>L</u> evel	Object Level 🗸	-
Password Algorithm	PC1 -	-
Status		
	OK Cancel	

Figure 3.66

6. Click **OK** to create the cabinet. The Cabinet created successfully dialog box appears.

3.7.5 Associating a cabinet

Perform the below steps to associate the cabinet:

For MSSQL:

- 1. Click **Stop** to enable the Associate button.
- 2. Click Associate. The Associate a Cabinet dialog appears with the following tabs:
 - i. Database tab: Select the database type.
 - ii. **Cabinet properties tab:** Specify the cabinet details that you have specified during cabinet creation.

	Cabinet p	roperties	Connection		
Specify th xists. Also net.	e new cabin specify the	et name an User name	d the server name and password for	where the ca accessing th	binet e is cabi
C <u>a</u> binet N	ame	ecms	uite		
Map this	cabinet to				
Docu	ment databa	se	✓ Image Serve	r database	
<u>S</u> erver Na	me	10.0.1	.43		
<u>S</u> erver Na Us <u>e</u> r nam	me 2	10.0.1 applo	.43		
<u>S</u> erver Na Us <u>e</u> r nam <u>P</u> assword	me	10.0.1 applog	.43 gin		

Figure 3.67

iii. Connection tab: Specify the maximum and the minimum number of connections that the JTS must maintain with the database, specify the query time out for the selected cabinet in the Query timeout text box and specify the refresh interval time for connection.

Database	Cabinet propertie	s Connection		
Specify the o this cabin	number of databas net. Also specify the	se connection that o e query timeout peri	an be made available t od for this cabinet.	
Ma <u>x</u> imur	n connection	25		
Minimum	connection	10		
Query tin	neout	0	second(s)	
<u>R</u> efresh I	nterval	60	Minutes	

3. Click **Done** to associate the selected cabinet. Once the cabinet is associated successfully, it appears with the list.

W OmniDocs Service Admin Server JTS Help	istration) X
o Q Servers	Server Information				
	IP Addr Encodi	ng UTF-8 💌	S Soc <u>k</u> etTime	erverP <u>o</u> rt 3333 eout(mins) 30	Adr <u>B</u> at
	✓ Transaction Conne	ction 🗹 Xml 🗹 Error 🗌 Soci	Error Log Size (MB) 10	Log Count 10	
	Cabinet Transaction Pool	MinDBConnections	Stop Disconnect MaxDBConnections	Edit <u>C</u> reate CabinetType	
	ecmsuitesql	10	25	Both Document Database and Ima	mssql
	ecmsuite17july	10	10	Both Document Database and Ima	mssql
	ecmsuite	10	25	Both Document Database and Ima	mssql
	Cabinet Operations				
	<u>A</u> ssociate Dissociate	Compile SP Property	T <u>e</u> st Delete <u>U</u> nlo	ck Enable <u>T</u> race Upgrad	de <u>L</u> icense
		III (•
Status	- [<u></u>				

Figure 3.69

For Aurora PostgreSQL:

- 1. Click **Stop** to enable the Associate button.
- 2. Click **Associate**. The Associate a Cabinet dialog appears with the following tabs:

- i. Database tab: Select the database type.
- ii. **Cabinet properties tab:** Specify the cabinet details that you have specified during cabinet creation.

atabase Gabine	et properties Connection
Specify the new ca xists. Also specify net.	binet name and the server name where the cabinet e the User name and password for accessing this cabi
C <u>a</u> binet Name	ibpsaurora12dec
Map this cabinet to	D
✓ Doc <u>u</u> ment data	abase 🗾 Image Server database
<u>S</u> erver I.P.	odtekwxu.ap-south-1.rds.amazonaws.com
<u>S</u> erver I.P. Us <u>e</u> r name	odtekwxu.ap-south-1.rds.amazonaws.com

Figure 3.70

iii. Connection tab: Specify the maximum and the minimum number of connections that the JTS must maintain with the database. Also, specify the query time out for the selected cabinet in the Query timeout text box and specify the refresh interval time for connection.

Uatabase	Cabinet propert	ies Connectio	n
Specify th	e number of datab	ase connection th	nat can be made available t
o this cab	inet. Also specify	the query timeout	period for this cabinet.
Maximu	m connection	25	
		14.00	
<u>M</u> inimu	m connection	10	
<u>M</u> inimu <u>Q</u> uery ti	m connection meout	0	second(s)

Figure 3.71

3. Click **Done** to associate the selected cabinet. Once the cabinet is associated successfully, it appears with the list.

Ø OmniDocs Service Adminis	stration		_		×
Server JTS Help					
	Server Information				
∲- <u>□</u> JTS (13.127.66.151 : 3	IP Addres Encodin	s 13.127.66.151 a UTF-8 💌			Soc <u>k</u> e
	✓ Transaction Connect	tion 🖌 Xml 🖌 Error 🗌 Socke	t Erro	or Log Size	(MB) 1
		Start	Stop	Disconn	ect
	Cabinet Transaction Pool				
	CabinetName	M	axDBConne	ctions	
	ibpsaurora12dec	10	25		
	Cabinet Operations				
	<u>A</u> ssociate D <u>i</u> ssociate	Compile SP Property	Test	Delete	
	•				•
Status Start JTS successful				Re	ady

Figure 3.72

3.7.6 Creating a data source

Perform the below steps to create the data source:

For MSSQL:

- Open the<Host-Path URL of OmniDocs+RMS EJB container> like http://ecmsuiteconsole.aws.co.in as defined in the AWS_ALB-IngressController.yml file. It automatically redirects to the JBoss EAP 7.4 Admin console.
- 2. Enter the newgen as username and password system123# respectively to login to the Admin console. After a successful login, the Red Hat JBoss Enterprise Application Platform screen appears.



Figure 3.73

- 3. Go to the **Subsystems** in the Configuration tab.
- 4. Go to the Datasources and Drivers. Then, click Datasources.

Red Hat JBoss Enterprise Application	n Platform		🌲 🌲 newgen 🗸
Homepage Deployments	Configuration Runtime Patching	Access Control	
Configuration	Subsystem (29) Datasou	rces & Drivers Datasource	© C Datasources
Subsystems (2) >	Fiter by: name or subtitle Datasour	ces 4 , Filter by: name, xa,/disabled.	Add Datasource 5 Add XA Datasource and XA
Interfaces > Socket Bindings > Paths System Properties	Batch jBeret Core Management Datasources & Drivers 3 > Deployment Scanners Discovery EE EJB 10 Infinispan > JCA	ers > © ExampleOS © ecmsuite17july © ecmsuitesql	 Non-XA datasources are used for applications which do not use transactions, or applications which use transactions are distributed across multiple databases. XA datasources are used by applications whose transactions are distributed across multiple databases. XA datasources introduce additional overhead.
			3.0.10.Final 🥕 Tools \land 😽 Settings

Figure 3.74

- 5. Click Plus + icon and select **Add Datasource**. The Add Datasource dialog appears.
- 6. For MSSQL Database Server, select Microsoft SQLServer and click Next.

Add Datasource					×
Choose Template	Attributes	JDBC Driver	Connection	Test Connection	Review
Channe and af the available		du add a dataan oo ah			
Choose one of the preder	ined templates to quici	dig add a datasource or cho	ose "Custom" to specify	your own settings.	
PostgreSQL					
MySQL					
Oracle					
Microsoft SQLServer					
O IBM DB2					
Sybase					

Figure 3.75

- 7. Provide a DataSource Name and JNDI Name.
 - Name: Enter the OmniDocs cabinet name that is cabinet name.
 - JNDI Name: java:/same as OmniDocs cabinet name
- 8. Click Next.

Add Datasource					×
Choose Template At	tributes	JDBC Driver	Connection	Test Connection	Review
Help					
Name * ecmsuite JNDI Name * java:/ecmsuite					
	Required fields are marked with •				

Figure 3.76

- 9. Select JDBC Driver Name.
- 10. For MSSQL, select sqljdbc42.jar.
- 11. Clear Drive Module Name and Driver Class Name textboxes.
- 12. Click Next.

Add Datasource					×
Choose Template Attrib	outes	JDBC Driver	Connection	Test Connection	Review
Help	colidad2 io	-			
Driver Module Name	sqijabc42.ja	r			
Driver Class Name					
	Required field	ds are marked with *			

Figure 3.77

13. Provide the following Connection Setting details and click Next:

- Connection URL: jdbc:sqlserver://MSSQL_Server_IP:MSSQL_Server_Port;databaseName=CABINET_NAME
- UserName: Enter the SQL Server User Name
- Password: Enter the SQL Server Password
- Security Domain: Keep this blank.
| Choose Template Attributes IDBC Driver Connection Test Connection Review | |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--|
| | |
| The second secon | |
| Connection URL //dbc:sqlserver://10.0.1.43:1522;databaseName=ecmsuite | |
| User Name applogin 🛷 | |
| Password | |
| Security Domain | |

Figure 3.78

14. Click **Next** on the **Test Connection** page.

Add Datasource					>
Choose Template	Attributes	JDBC Driver	Connection	Test Connection	Review
On this page you can ter Please note that testing • If you press Test • If you go back ar bindings once th • If you cancel the If you choose to continu	st the connection of you the connection change: Connection for the fir nd change settings, this e datasource has been ewizard, the datasource without testing the co	ir datasource. is the semantics of this wiza st time, the datasource is c will modify the newly creat created. will be removed again. Thi nnection, the datasource w	rd: reated in advance. ed datasource. Please no s might require a reload ill be created after finish	ote that you cannot change the of the server.	name and JNDI
		Test Con	nection		

15. Click **Finish.** After the creation of the datasource, a success message appears.

Add Datasource				×
Choose Template Attrik	utes JDBC Driver	Connection	Test Connection	Review
Help	ermsuite			
JNDI Name	java:/ecmsuite			
Connection URL	jdbc:sqlserver://10.0.1.43:1522;dat	abaseName=ecmsuite		
Driver Name	sqljdbc42.jar			
User Name	•••••• @			
Password	•••••• @			



- 16. Click View Datasource to view the created datasource. The created datasource appears in the list of Datasource.
- 17. Click **View** against the datasource. A screen appears with the attributes of the datasource appears.
- 18. Click Edit link.

Red Ha	t JBoss Enterprise App	lication Platform				ී Reload Required 룾	🛓 newgen 🗸
« Back	/ Configuration ⇒ Su	ubsystems / Subsyste	n \Rightarrow Datasourc Drivers \checkmark /	Datasources & Drivers ⇒ Datasources /			# C
ecms A JDBc da Attribute	SUITE (enabled) ata-source configuration s Connection Pro	ool Security Cred	ntial Reference Validation 7	Timeouts Statements / Tracking			
		Datasource Clas	com.microsoft.sqlserver.jdbc	.SQLServerDataSource			
		Driver Clas					
		Driver Nam	sqljdbc42.jar				
		JNDI Nam	java:/ecmsuite				
		Statistics Enable	false				

Figure 3.81

19. Clear the **Datasource Class** textbox and click **Save**.

Red Hat JBoss Enterprise Application Platform		ව Reload Required 🌲	🛔 newgen 🗸
« Back / Configuration \Rightarrow Subsystems / Subsystem	Datasourc Drivers 👻 / Datasources & Drivers -> Datasources / Datasource -> ecmsuite ->		# C
CCMSUITE (enabled) A JDBC data-source configuration Attributes Connection Pool Security Credential	Reference Validation Timeouts Statements/Tracking		
1 Help			
Datasource Class Driver Class Driver Name = JNDI Name = Statistics Enabled	sigleb:42.jar jaca/consulte Image: Construction of the second se	Cancel	Save

Figure 3.82

- 20. After that restart the OmniDocs+RMS EJB container.
- 21. Once the OmniDocs+RMS EJB container is restarted, open the JBossEAP Admin console once again.
- 22. Go to the **Subsystems** in the Configuration tab.
- 23. Go to the Datasources and Drivers. Then, click Datasources.
- 24. Select the created data source and click **Test connection** from the dropdown list.

Red Hat JBoss Enterprise Application Platform 🏾 D Reload Required 🦧 🛓 newgen 🗸									
Homepage Deployments	Configuration Runtime Patc	hing Access Control							
Configuration	Subsystem (29)	Datasources & Drivers	Datasource 💿 🖉	ecmsuite					
Subsystems (2)	Fiter by: name or subtitle	Datasources	Filter by: name, xa,/disabled, deployment	Datasource					
Interfaces >	Batch JBeret	JDBC Drivers >	ExampleDS ermsuite View X	The datasource ecmsuite is enabled. Disable					
Socket Bindings >	Core Management		Disable	Main Attributes					
Daths	Datasources & Drivers (3) >		ecmsult Test Connection	Main Attributes					
System Properties	Deployment Scanners		ecmsuitesql	JNDi Name: javaJecmsuite Driver Name: sqljdbc42.jar					
	Discovery			Connection URL: jdbc:sqlserver://10.0.1.43:1522;databaseName=ecm					
	EE			Enabled: true					
	EJB			Statistics Enabled: false					
	10								
	Infinispan >								
	JCA								
	JMX								

Figure 3.83

On the successful data connection, a success message appears.

Red Hat JBoss Enterprise Application	Red Hat JBoss Enterprise Application Platform 🖉 🕹 review 🗸										
Homepage Deployments	Configuration Runtime Pate	hing Access Control									
Configuration	Subsystem (29)	Datasources & Drivers	Datasource ©~ 2	ecms Successfully tested connection for datasource ecmsuite. X							
Subsystems >	Fiter by: name or subtitle	Datasources >	Filter by: name, xa,/disabled, deployment	Datasource							
Interfaces >	Batch JBeret	IDBC Drivers	⊘ ExampleDS	O The datasource ecmsuite is enabled. Disable							
Contrat Dia dia se	Core Management		⊘ ecmsuite enabled View ~								
Socket Bindings >	Datasources & Drivers >		⊘ ecmsuite17juły	Main Attributes							
Paths	Deployment Scanners		⊘ ecmsuitesql	JNDI Name: java:/ecmsuite Driver Name: solidbed2 lar							
System Properties	Discovery			Connection URL: jdbc:sqlserver://10.0.1.43:1522;databaseName=ecm							
	EE			Enabled: true							
	EJB			Statistics Enabled: false							
	10										
	Infinispan >										
	JCA										
	јМХ 🗸										
				3.0.10.Final 🏄 Tools 🔨 😋 Settings							



25. Add the below connection pool setting and idle-connection-timeout setting inside the created DataSource in *standalone.xml* file located inside the **OmniDocs11.0Ejb** or **configuration** folder at the mapped location on the worker node.

For example,

```
Cdatasource jndi-name="java:/auroraod23oct1" pool-name="auroraod23oct1">
                 <connection-url>idbc:
              postgresgl://omnidocs
                                                                                                                                                                                                                                        wwww.ore-cluster.cluster-cv4updtekwxu.ap-south-1.rds.amazonaws.com:5432/auroraod23oct1
                                                                        -url>
                <driver>postgresql-42.2.18.jar</driver>
              <pool>
                              ~min-pool-size>0</min-pool-size>
<initial-pool-size>0</initial-pool-size>
<max-pool-size>600</max-pool-size>
<flush-strategy>Gracefully</flush-strategy>
               </pool>
               <security>
                                <user-name>postFres</user-name>
<password>state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fille:state="fil
              </security>
<validation>
              <timeout>
            <idle-timeout-minutes>5</idle-timeout-minutes>
</timeout>
</datasource
```

Figure 3.85

26. Restart the **OmniDocs+RMS EJB** container once again.

For Aurora PostgreSQL:

- Open the<Host-Path URL of OmniDocs+RMS EJB container> like http://ecmsuiteconsole.aws.co.in as defined in the AWS_ALB-IngressController.yml file. It automatically redirects to the JBossEAP 7.4 Admin console.
- 2. Specify the newgen as username and system123# as password respectively to login to the Admin console.



Figure 3.86

- 3. Go to the **Subsystems** in the Configuration tab.
- 4. Go to the Datasources and Drivers. Then, click Datasources.
- 5. Click Plus + icon and select Add Datasource. The Add Datasource dialog appears.



Figure 3.87

6. For the Aurora PostgreSQL Database Server, select **PostgreSQL** and click **Next**.

Add Datasource					>
Choose Template	Attributes	JDBC Driver	Connection	Test Connection	Review
Choose one of the pred	efined templates to quic	kly add a datasource or ch	oose "Custom" to specify	your own settings.	
○ Custom					
○ H2					
PostgreSQL					
O MySQL					
O Oracle					
O Microsoft SQLServer					
O IBM DB2					



- 7. Provide a DataSource Name and JNDI Name.
 - Name: Enter same as OmniDocs cabinet name.
 - JNDI Name: java:/same as OmniDocs cabinet name.
- 8. Click Next.
- 9. Select JDBC Driver Name.
- 10. For Aurora PostgreSQL, select postgresql-42.5.0.jar.
- 11. Clear Drive Module Name and Driver Class Name textboxes and click Next.
- 12. Provide the following Connection Setting details and click Next:
 - Connection URL:

jdbc:postgresql://AuroraPostgrSQL_Server_IP:AurorPostgreSQL_Server_Port/CABINET_NAM E

- UserName: AuroraPostgreSQL Server User Name
- Password: AuroraPostgreSQL Server Password
- Security Domain: Keep this blank.

Add Datasource					
Choose Template A	ttributes	JDBC Driver	Connection	Test Connection	Review
Help Connection UR	Jdbc:po	stgresql://omnidocs-aurora	ards-db.cluster-cv4updtel	kwxu.ap-south-1.rds.amazonaw	rs.com:5432/ibps
Help Connection UR User Nam	jdbc:po:	stgresql://omnidocs-aurora	ards-db.cluster-cv4updtel	kwxu.ap-south-1.rds.amazonaw	rs.com:5432/ibps
Telp Connection UR User Nam Passwor	L jdbc:po: e ······	stgresql://omnidocs-aurora	ards-db.cluster-cv4updtel	kwxu.ap-south-1.rds.amazonaw	rs.com:5432/ibps

Figure 3.89

- 13. Click Next on the Test Connection page.
- 14. Click **Finish.** After the creation of the datasource, a success message appears.

Add Datasource					×
Choose Template Att	ributes	JDBC Driver	Connection	Test Connection	Review
Help Name	ibpsauro	ra12dec			
JNDI Name	java:/ibps	saurora12dec			
Connection URL	jdbc:post 1.rds.am	gresql://omnidocs-aurorar azonaws.com:5432/ibpsau	ds-db.cluster-cv4updtekv rora12dec	vxu.ap-south-	
Driver Name	postgres	ql-9.3-1102.jdbc4.jar			
User Name	•••••	• •			
Password	••••••	•••• @			
				Cancel	< Back Finish

Figure 3.90

- 15. Click **View Datasource** to view the created datasource. The created datasource appears in the list of **Datasource**.
- 16. Click **View** against the datasource. A screen appears with the attributes of the datasource.
- 17. Click Edit link and clear the Datasource Class textbox.

Red Hat	JBoss Enterprise	Application Platfo	orm				
« Back	Configuration =	Subsystems /	Subsystem \Rightarrow	Datasourc Drivers 🗸	Datasour	ces & Drivers \Rightarrow Datasources	/ Datasource ⇒ ibpsaurora12dec ∨
ibpsa A JDBC da	IUTOTA12de ta-source configurat	C (enabled)					
Attributes	Connection	Pool Securit	y Credential	Reference Validation	Timeouts	Statements / Tracking	
🖌 Edit 🕤	Reset ⑦ Help						
		Data	source Class	org.postgresql.ds.PGSim	pleDataSource		
			Driver Class				
			Driver Name	postgresql-9.3-1102.jdbc	4.jar		
			JNDI Name	java:/ibpsaurora12dec			
		Statis	tics Enabled	false			

Figure 3.91

18. Click **Save**. After that restart the OmniDocs+RMS EJB container.

Red Hat	t IBoss Enterprise	Application Platf	form							🜲 👗 newgen 💊
« Back	Configuration	⇒ Subsystems /	Subsystem =	Datasourc Drivers 🗸	/ Datasources & Drivers	⇒ Datasources	/ Datasource ⇒ ibpsaurora12	lec ~		# C
ibpsa Ajdbc de	aurora12de ata-source configura	2C (enabled) tion								
Attributes	Connection	Pool Securi	ty Credential	Reference Validation	Timeouts Statements /	Tracking				
		Data	asource Class							
			Driver Class							
		D	river Name *	postgresql-9.3-1102.jdbc	4.jar					
			JNDI Name *	java://bpsaurora12dec						
		Stati	stics Enabled	% OFF						
				Required fields are market	d with •					
									Cancel	Save
									3.0.10.Final	F Tools ^ OG Setting

Figure 3.92

- 19. Once the OmniDocs+RMS EJB container is restarted, open the JBossEAP Admin console once again.
- 20. Go to the **Subsystems** in the Configuration tab.
- 21. Go to the **Datasources and Drivers**. Then, click Datasources.
- 22. Select the created data source and click **Test connection** from the dropdown list.

Red Hat JBoss Enterprise Applicatio	n Platform			🌲 🛔 newgen ~
Homepage Deployments	Configuration Runtime Pate	thing Access Control		
Configuration	Subsystem (29)	Datasources & Drivers	Datasource 🕑 🗸 🕽	ibpsaurora12dec
Subsystems 👩 >	Fiter by: name or subtitle	Datasources (4) >	Filter by: name, xa,/disabled, deployment	Datasource
Interfaces >	Batch JBeret	JDBC Drivers >	O ExampleDS	O The datasource ibpsaurora12dec is enabled. Disable
Socket Bindings >	Core Management		ibpsaurora12dec View View View	
Paths	Datasources & Drivers 🔞 🔅		Test Connection 6	INDENime: Java/Ibosaurora12der
System Properties	Deployment Scanners			Driver Name: postgresql-9.3-1102.jdbc4.jar
un mont the tit safe of the same deep	Discovery			Connection URL: jdbc:postgresql://omnidocs-aurorards-db.cluster-cv
	EE			Enabled: true
	EJB			Statistics Enabled: false
	10			
	Infinispan >			
	JCA			
	јмх			
				3.0.10.Final 🖌 Tools 🔿 🕫 Settings

Figure 3.93

Red Hat JBoss Enterprise Applica	ation Platform			🦧 🔹 newgen 👻
Homepage Deployments	Configuration Runtime Pa	tching Access Control		
Configuration	Subsystem (29)	Datasources & Drivers	Datasource ③- ♡	ib Successfully tested connection for datasource ibpsaurora12dec. X
Subsystems	> Fiter by: name or subtitle	Datasources >	Filter by: name, xa,/disabled, deployment	Datasource
Interfaces	Batch	IDBC Drivers	⊘ ExampleDS	O The datasource ibpsaurora12dec is enabled. Disable
	Core Management	percenter	⊘ ibpsaurora12dec View ✓	
Socket Bindings	> Datasources & Drivers >			Main Attributes
Paths				JNDI Name: java://bpsaurora12dec
System Properties	Deployment scanners			Driver Name: postgresql-9.3-1102.jdbc4.jar
	Discovery			Connection URL: jdbc:postgresql://omnidocs-aurorards-db.cluster-cv
	EE			Enabled: true
	EJB			Statistics Enabled: false
	10			
	Infinispan >			
	JCA			
	јМХ	•		
				3.0.10.Final 🥕 Tools 🗠 😋 Settings

On a successful data connection, a success message appears.



23. Add the below connection pool setting and idle-connection-timeout setting inside the created DataSource in *standalone.xml* file located inside the **OmniDocs11.0Ejb** or **configuration** folder at the mapped location on the worker node.

```
<pool>
        <min-pool-size>0</min-pool-size>
        <initial-pool-size>0</initial-pool-size>
        <max-pool-size>600</max-pool-size>
        <flush-strategy>Gracefully</flush-strategy>
</pool>
</timeout>
        <idle-timeout-minutes>5</idle-timeout-minutes>
</timeout>
```

For example,



Figure 3.95

24. Restart the **OmniDocs+RMS EJB** container once again.

3.7.7 Registering a cabinet in OmniDocs

Perform the below steps to register a cabinet:

 Register the cabinet for OmniDocs Admin using the following URL: http://<Host-Path URL of OmniDocsWeb container>/omnidocs/admin/main/registration/registration.jsp For example, http://ecmsuite.aws.co.in/omnidocs/admin/main/registration/registration.jsp

Jun Docs"		
	Registration	
	Cabinet List	
	Select Cabinet ~	
	Site List	
	Select Site 🗸	
	Username	
	supervisor2	
	Password	
	Register as	
	🗆 Admin 🔍 Web 💌 Both	
	Cancel Register	

Figure 3.96

All the created cabinets get auto populated in the **Cabinet List** dropdown list.

- 2. Select the required cabinet, select the associated site, and specify the Username and Password.
- 3. Select the Register as **Both** and click **Register.** After successful registration, a confirmation message appears.

MIN Docs"		
		Cabinet Registration Successful
	Registration	
	Cabinet List	
	auroraod23oct1 v	
	Site List	
	auroraod23oct1site v	
	Username	
	supervisor	
	Password	
	Register as	
	O Admin 🛛 Web 💿 Both	
	Cancel Register	

Figure 3.97

3.7.8 Registering a cabinet in RMS

Perform the below steps to register a cabinet in RMS:

 Configure the cabinet for RMS using the following URL: http://<Host-Path URL of OmniDocsWeb container>/rms/config For example, http://ecmsuite.aws.co.in/rms/config

Configuration Select Cabinet A Username A Password	Configuration Select Cabinet ^ Username D Password Configure	Z	² newgen OmniDocs RM:
 Select Cabinet Username Password Configure 	 Select Cabinet Username Password Configure 	on	figuration
A Username Password Configure	 A Username ▲ Password Configure 	-	Select Cabinet 👻
Password Configure	Password Configure	Å	Username
		₿	Password
			Figure 3.98

All the created cabinets get auto populated in the Cabinet List dropdown list.

2. Select the required cabinet, specify the **Username** and **Password** and click **Configure**.

🏄 newgen OmniDocs RMS
Configuration
🗧 rmsazure04jan 🔻
₼
Configure
Eiguro 2 00

After successful registration of cabinet, a confirmation message appears.

3.7.9 Creating site and volume

Perform the below steps to create site and volume:

 Login to the OmniDocs Admin using the following URL: http://<Host-Path URL of OmniDocsWeb container>/omnidocs/admin For example,

http://ecmsuite.aws.co.in/omnidocs/admin

Highly Scalable Architecture	(ANN Docs
	Omniloss: Enterprise Content Management (ECM) platform enables end-to-end management of large volume of documents and digital content, right from capture to disposition. It seamlessly integrates with other enterprise applications to manage unstructured content and enables digital transformation of your organization.
	Login
	Password
	ecmsuite 👻
	Cogin Forgot Password?
0 • • • •	NEWGEN

Figure 3.100

2. After a successful login, click Sites link under Administration.

🚫 NEWGEN	OmniDocs ADMIN DES	БКТОР			0
오 Administrati	ion	Configure		Personalize	
 Cabinet Details Applications Folders Users Groups Roles 	 DataClasses Global Indexes Keywords Sites Volumes Manage Audit Logs 	OmniProcess Web API	Search Dashboard	Color and Accessibility The Color and Accessibility The Repository View Tool E Custom Operations Multil	ng Page Configuration 3ar Ingual Definition
Management Ecense Management	• Service Management • Trash Management				

Figure 3.101

3. Click +Add. The Add Site dialog appears.

Add Site		×
SMS Site	Site*	
Hadoop Site		
Amazon S3 Site	Site Address*	
HCP Site		
MS Azure Site	Port No*	
	Cancel Sav	e

Figure 3.102

- 4. Click Amazon S3 Site.
- 5. Specify the user-defined site name, **Access Key**, and **Secret Key** that have rights to the S3 bucket.
- 6. Click Save.

Add Site	×
SMS Site	Site*
Hadoop Site	auroraod23oct1site
Amazon S3 Site	Role Based
HCP Site	Region*
MS Azure Site	
	Access Key*
	AKIAJENJGPC2TBE3W5EA
	Secret Key*
	Cancel Save



() N	IEWGEN	OmniDocs	ADMIN DESKTOP		0
Lo Administration	1 Home >	Administration- Sites	5		
©	Sites	+ Add	auroraod23oct1site		
OmniProcess	auroraod23oct	1site			
Search				Site*	
WebAPI				auroraod23oct1site	
Personalize				Region*	
Dashboard					
오. Management				Access Key* AKIAIENIGPC2TBE3W5EA	
				Secret Key*	
					Modify

The added Site appears under Sites in the left pane.

Figure 3.104

7. Go back to the **Home** page.

Administration Cabinet Details • Cabinet Details
 Cabinet Details DataClasses Global Indexes Global Indexes Global Indexes Global Indexes Keywords Stes Stes Volumes Management
Management

Figure 3.105

8. Select **Volumes**. The Volumes screen appears.

	NEWGEN	OmniDocs	ADMIN DESKTOP					0	
Lo Administration	1 Home >	Administration- Vol	umes						
٥	Volumes	+ Add	Name your new volume here*						
OmniProcess ⊕ Search				Home Site	select a site		· · · · · · · · · · · · · · · · · · ·	~	
WebAPI				Default Path*	select a path			-	
O Personalize				Volume Block Size (MB)	50			•	
Dashboard				Encryption	● No Encryption 〇) Default 256-bit	○ Custom Encryption		
C Management				Encryption Class Name					
				Replication Type	Immediate			•	
			Delete						Add

Figure 3.106

- 9. Specify the following details:
 - Home Site: Select the newly created Site name.
 - Default Path: Select the S3 bucket in which you want to store PN files.
 - Volume Name: Specify the user-defined volume name.

10. Click **Add**.

	IEWGEN	OmniDocs	ADMIN DESKTOP				0
	1 Home >	Administration-	/olumes				
© OmniProcess	Volumes	+ Add	auroraod23oct1vol				Run Compaction Replicate
् Search				Home Site	auroraod23oct1site	~	
er WebAPI				Default Path*	SMS:od10devnew	~	
A Personalize				Volume Block Size (MB)	50	~	
Dashboard				Encryption	No Encryption O Default 256-bit	O Custom Encryption	
 Management				Encryption Class Name			
				Replication Type	Immediate	~	
			Delete				Add

Figure 3.107

The added volume appears under Image Volumes in the left panel.



Figure 3.108

11. Go back to the **Home** screen.

Administration Cabinet Details • Applications • Cabinet Details • Applications • Global Indexes • Sidoal Indexes • Sidoal Indexes • Stass • Stass • Stass • Stass • Outmass • Stass • Stass • Volumes • Nanage Audit Logs	🔘 NEWGEN	OmniDocs ADMIN DE		0 0			
 Cabinet Details DataClasses Global Indexes Global Indexes Global Indexes Keywords Stas Stas Stas Volumes Management Service Management Trash Management 	Administrat	tion	Configure		Personalize		
Management • Service Management • Report Management • Service Management • License Management • Trash Management	Cabinet Details Applications Folders Users Groups Roles	DataClasses Global Indexes Keywords Sites Volumes Manage Audit Logs	OmniProcess Web API	Search Dashboard	Color and Accessibility The Landing Page Configura Repository View Tool Bar Custom Operations Multilingual Definition	ion	
Report Management Service Management Ucense Management Trash Management	Manageme	nt					
	Report Management License Management	Service Management Trash Management					

Figure 3.109

12. Click Cabinet Details.

13. Select the added volume from the Default Image Volume using the dropdown

14. Click **Save**. The Site and Volume are now created successfully.

() N	IEWGEN OmniDocs ADMIN DESKTOP		0 0			
Ģ	Home Administration- Cabinet Details					
©	Cabinet Details			2	'ପ୍ର'	1
OmniProcess ① Search	Cabinet Name auroraod23oct1	Cabinet Type postgres	Created Date and Time 08/11/2020 04:46			
ی WebAPI Personalize	Inherit Ownership	Remove the Rights of Supervisor (Rights once removed will not be restored again)	Separate User/ Group Privileges (Once enabled, can't be disabled)			
Dashboard Og Management	Chable Maker Checker Functionality (Once enabled, can't be disabled)	Enable Data Security Functionality (Once enabled, can't be disabled)	Enable User Access Report			
	Key Management Service None	Default Imaging Volume auroraod23octtvol	Auto Versioning			
	Enable Two Factor Authentication	Two Factor Authentication Class Name	Enable Multilingual			
				Cancel	Sa	ave

Figure 3.110

15. Log in to the OmniDocs Web using the below URL to start. http://<Host-Path URL of OmniDocsWeb container>/omnidocs/web For example: http://ecmsuite.aws.co.in/omnidocs/web

3.8 EasySearch Post-Deployment Changes

Perform the below steps to do EasySearch post-deployment changes:

 Login to the ApacheManifold Admin using the following URL: <Host-Path URL of ApacheManifold>/mcf-crawler-ui/login.jsp For example,

http://ecmsuiteapache.aws.co.in/mcf-crawler-ui/login.jsp



Figure 3.111

- 2. Log in with the following credentials:
 - User ID: admin
 - Password: admin
- 3. After a successful login, click **Jobs** tree showing in the left panel.
- 4. Click Status and Job Management. The below job list appears:
 - <CABINET_NAME>_Document
 - <CABINET_NAME>_Folder
- 5. Start both the jobs.
- 6. Once both the jobs started, the Job's status appears as **Running.**

Apache ManifoldCF™	Document Ingestion										
MAIN NAVIGATION	Status of Jobs										
亘 Outputs <											
👗 Authorities 🛛 <	Action	Name	Status	Start Time	End Time	Documents	Active	Processed			
	► Restart 💉 Restart minimal 🔲 Pause 🔲 Abort	ecmsuite_Document	Running	7/29/20 5:07:54 PM		1	1	1			
	► Restart 🖌 Restart minimal 🚺 Pause 🔲 Abort	ecmsuite_Folder	Running	7/30/20 11:34:17 AM		1	1	1			
⊒ Jobs ~											
III List all Jobs	C Refresh										
Status and Job Managemen	·										
Status Reports <											
D History Reports <											
i Miscellaneous											
	Convictet© 2010-2018 The Anaphe® Software Foundation							Version 2			

Figure 3.112

3.9 OmniScanWeb: Registering a cabinet

Perform the below steps to register the cabinet in OmniScanWeb:

 Open the OmniScanWeb using the following URL: http://<Host-Path URL of OmniScanWeb container>/omniscanweb For example,

https://omniscan.newgendocker.com/omniscanweb

- 2. Click Register New Cabinet link on the OmniScan Web login screen.
- Specify the Server URL as given below: http://<Host-Path URL of OmniDocsWeb container>/NGServlet/servlet/ExternalServlet For example,

https://omnidocs.newgendocker.com/NGServlet/Servlet/ExternalServlet

4. Specify the **OmniDocs EJB** container name for AppServer IP or Server URL, 8080 for AppServer Port, and JBOSSEAP for AppServer Type.

Login	Register Cabinet							
1 Connect	2 Register							
Server URL								
https://omnidocs1	1alpine.newgendocker.com/NGServlet/servlet/ExternalS							
AppServer IP								
od110ejb								
AppServer Port								
8080								
AppServer Type								
JBOSSEAP	~							
	Connect							
对 newgen OmniScan								
	© Powered by Newgeo Softwares							

Figure 3.113

- 5. Click Connect.
- 6. Select the **Cabinet Name**, **Site ID**, and **Volume ID** from the list.

← Login	Register Cabinet						
⊘ Connect	2 Register						
Cabinet Name							
odpostgres19nov	1	~					
Site ID							
odpostgres19nov	odpostgres19novsite						
Volume ID							
odpostgres19nov	~						
	Register						
(I)	Powered by Newgen Softwares						
	Figure 3.114						

7. Click **Register**. The registered cabinet appears in the **Cabinet Name** list on the login screen. Now you can log into OmniScan Web.

NOTE:

Ensure that the **OmniScan_Template_Repository** folder is already created in OmniDocs before logging into OmniScan Web.

3.10 Creating a Secret manager policy and secrets

To create IAM Policy and Role, perform the below steps:

1. Create an IAM Policy with the name SecretMgr_Policy with the following permissions:

```
"Version": "2012-10-17",
"Statement": [
    {
        "Sid": "VisualEditor0",
        "Effect": "Allow",
        "Action": [
            "secretsmanager:DescribeSecret",
            "secretsmanager:PutSecretValue",
            "secretsmanager:CreateSecret",
            "secretsmanager:DeleteSecret",
            "secretsmanager:CancelRotateSecret",
            "secretsmanager:ListSecretVersionIds",
            "secretsmanager:UpdateSecret",
            "secretsmanager:GetRandomPassword",
            "secretsmanager:GetResourcePolicy",
            "secretsmanager:GetSecretValue",
            "secretsmanager:StopReplicationToReplica",
            "secretsmanager:ReplicateSecretToRegions",
            "secretsmanager:RestoreSecret",
            "secretsmanager:RotateSecret",
            "secretsmanager:UpdateSecretVersionStage",
            "secretsmanager:RemoveRegionsFromReplication",
            "secretsmanager:ListSecrets"
        ],
        "Resource": "*"
    }
]
```

- 2. Add this policy to the Worker node IAM Role.
- 3. Create a Secret for Alarm Mailer

Refer to the below steps to create Secrets for Alarm Mailer:

- Open the AWS Secret Manager console.
- Click Store a new secret.
- Select Other type of secret.

- Add Key/ value mentioned below:
 - a. CabinetName_Username
 - b. CabinetName_Password

NOTE:

Update the **CabinetName** with your Cabinet Name.

Here values are:

CabinetName_Username : supervisor group's username CabinetName_Password : supervisor group's password

🥦 CodeCommit 🛛 🔉 Support 👩 Elastic C	Container Registry 🛛 🔯 CloudWatch 📑 RDS 🛛 🧕 EC2 💆 Route 53				
Step 2 Configure secret	Secret type Info				
Step 3 - <i>optional</i> Configure rotation	Credentials for Amazon RDS Credentials for Amazon database DocumentDB database				
Step 4 Review	Credentials for other database Other type of secret API key, OAuth token, other.				
	Key/value pairs Info				
	Key/value Plaintext				
	odpostgres19dec_Username Remove				
	odpostgres19dec_Password Add row Add row				

Figure 3.115

- Click Next.
- Enter Secret name is AlarmMailerPSequence.
- Click Next and Store to save this Secret.

4. Create Secret for LDAP:

Follow Below steps to create Secrets for LDAP

- Open the AWS Secret Manager console.
- Click on **Store a new secret.**
- Select Other type of secret.
- Add Key/ value mentioned below:
 - a. ODUsername
 - b. ODPassword

- c. DomainUserName
- d. DomainPassword
- e. DistinguishedName

Here values are:

ODUsername: supervisor group's username ODPassword: supervisor group's username DomainUserName: Active Directory Domain username DomainPassword: Active Directory Domain password DistinguishedName: Active Directory Distinguished username

Credentials for Amazon RDS database	Credentials for Amazon DocumentDB database	 Credentials for Amazon Redshift cluster
Credentials for other database	• Other type of secret API key, OAuth token, other.	
Key/value pairs Info		
Key/value Plaintext		
ODUsername	S	Remove
ODPassword		Remove
DomainUserName		Remove
DomainPassword	sys	Remove
DistinguishedName	administration administration	re.net Remove
+ Add row		

Figure 3.116

- Click **Next**.
- Enter Secret name is LDAP.
- Click Next and Store to save this Secret.

4 Configuring AWS CodePipeline for container deployment on EKS

This chapter describes the configuration of AWS CodePipeline for container deployment on Elastic Kubernetes Service (EKS).

4.1 Overview

The Build Pipeline and Release Pipeline are separated into two parts. Build Pipeline is done through the Jenkins server which can be installed on an on-premises machine or a cloud machine. Using the AWS CodePipeline cloud service, you can manage the Release pipeline. In this architecture, three stages are created that is, Dev, UAT, and Production and in each stage, deployment is quite different. You can have some more stages depending on the requirements. This document describes the configuration of the AWS CodePipeline for container deployment on EKS.

4.2 Architecture of CICD pipeline



Figure 4.1

- The Newgen representative builds the product's base Docker images on the company's onpremises servers using Jenkins.
- As soon as the Dev team commits the code to the source code repository, the Jenkins
 pipeline gets triggered. It pulls the code > compiles them > prepares the build artifacts >
 creates Docker images and pushes the newly created Docker images to the AWS Elastic
 Container Registry.

- As soon as any Docker image is pushed to the AWS Elastic Container Registry, AWS CodePipeline triggers the deployment to the Dev environment. Here, you can configure the performance testing as well as security testing of the application. In Addition, you can perform manual testing as required.
- UAT and Production deployments are based on approval and are available on-demand. To deploy to the UAT environment, you need to trigger the UAT deployment. Upon deployment trigger, an approval mail is sent to the project manager or the concerned team. As soon as the project manager approves the go-ahead, UAT deployment gets started automatically.
- Production deployment is also based on approval, but it is multi-level approval. To deploy a production environment, you require the approval of all stakeholders, and the production environment doesn't get triggered automatically on receiving all the approvals. A manual intervention mail is sent to the engineer who is supposed to deploy to production with a checklist. During deployment, all the checklist points get verified before performing the production deployment. In case any point of the checklist is not covered, then deployment to the production gets rejected.

4.3 Configuring AWS Elastic container registry

This section explains how to configure the AWS Elastic Container Registry.

Perform the below steps to configure AWS Elastic Container Registry:

- 1. Open the Amazon ECR console at <u>https://console.aws.amazon.com/ecr/repositories</u>.
- 2. From the navigation bar, select the Region to create your repository.
- 3. In the navigation pane, select **Repositories**.
- 4. On the Repositories page, select **Create repository.**
- 5. Select **Private** in the Visibility settings.
- 6. Enter a unique name for a repository that is, **omnidocs10.1 web** in the **Repository Name**.
- 7. For **Tag immutability**, do not enable the tag mutability setting for the repository. Repositories are configured with immutable tags that prevent image tags from getting overwritten.
- 8. Enable the image scanning setting for the repository for Scan on push. Repositories that are configured to scan on push start an image scan whenever an image is pushed.
- 9. Keep the Encryption settings as default.
- 10. Select Create repository.

Repusito	ies > Create repository
eate re	pository
General set	ings
isibility settin	s Info
Private Access is mar	ig setting for the repository policy permissions.
Public Publicly visib ^l	e and accessible for image pulls.
Repository nam Provide a concise	e name. A developer should be able to identify the repository contents by the name.
57803561216	0.dkr.ecr.ap-south-1.amazonaws.com/ omnidocs10.1web
15 out of 256 cha	acters maximum (2 minimum). The name must start with a letter and can only contain lowercase letters, numbe res. and forward slashes.
Disabled	
Once a	epository is created, the visibility setting of the repository can't be changed.
 Once a 	epository is created, the visibility setting of the repository can't be changed. settings
Once a Once a Once a Once a	epository is created, the visibility setting of the repository can't be changed. settings sh to have each image automatically scanned after being pushed to a repository. If disabled, each image scan m o get scan results.
Once a Once a Once a Once a	epository is created, the visibility setting of the repository can't be changed. settings sh to have each image automatically scanned after being pushed to a repository. If disabled, each image scan m o get scan results. settings
Once a Mage scan Scan on push manually started Enabled Encryption KMS encryptio You can use AWS settings. Disabled	epository is created, the visibility setting of the repository can't be changed. settings the to have each image automatically scanned after being pushed to a repository. If disabled, each image scan m o get scan results. settings key Management Service (KMS) to encrypt images stored in this repository, instead of using the default encrypti
 Once a mage scan can on push inable scan on punanually started Enabled Encryption KMS encryption MS encryption Disabled The KM 	epository is created, the visibility setting of the repository can't be changed. settings sh to have each image automatically scanned after being pushed to a repository. If disabled, each image scan m o get scan results. settings setings settings settings settings settings settings set
 Once a Image scan Scan on push inable scan on punanually started Enabled Encryption KMS encryption MS encryption Disabled The KM 	epository is created, the visibility setting of the repository can't be changed. settings th to have each image automatically scanned after being pushed to a repository. If disabled, each image scan m o get scan results. settings key Management Service (KMS) to encrypt images stored in this repository, instead of using the default encrypti e encryption settings cannot be changed or disabled after the repository is created.

NOTE:

AWS ECR repositories can also be created while pushing Docker images to the AWS ECR using the AWS CLI.

4.4 Push and Pull docker images to or from AWS ECR

This section describes how to push and pull docker images from AWS ECR.

Prerequisites: Ensure that you have installed the latest version of the AWS CLI and Docker.

Following are the steps to push, and pull Docker images to or from AWS ECR:

- **Authentication**
- Push ٠
- Pull •

Authentication:

- 1. Before you push or pull the Docker images, you need to authenticate the Docker client to the AWS ECR.
- Execute the below command to configure the AWS Accesskey and AWS SecretKey of the IAM user that has the rights to push or pull Docker images to AWS ECR: aws configure set aws_access_key_id <AWS_AccessKey> aws configure set aws secret access key <AWS SecretKey>
- 3. Execute the below command to retrieve the authentication token and authenticate the Docker client:

```
aws ecr get-login-password --region <AWS Region> | docker login --username AWS
--password-stdin <AWS AccountID>.dkr.ecr.<AWS Region>.amazonaws.com
```

Push:

- 1. Before pushing the docker images to AWS ECR, you must create a repository to store them in.
- 2. Execute the below command to create a new repository if already not created:

```
aws ecr describe-repositories --repository-names <RepositoryName> || aws ecr
create-repository --repository-name <RepositoryName> --image-scanning-
configuration scanOnPush=true
```

3. Execute the below command to push the Docker images from your local machine to AWS ECR: Docker tag <ImageName>:<ImageTag>

```
<AWS AccountID>.dkr.ecr.<AWS Region>.amazonaws.com/<ImageName>:<ImageTag>
docker push
```

<AWS AccountID>.dkr.ecr.<AWS Region>.amazonaws.com/<ImageName>:<ImageTag>

NOTE:

- Docker images might be shared in the form of a compressed tar file. As compressed Docker images cannot be used directly, first you need to decompress them in a Docker image form, and then you can use it. In such a case, the client needs to perform the following:
 - Download the compressed Docker image file.
 - Convert the compressed file into a Docker image using the Docker Load command. Example: docker load –i C:\DockerImages\omnidocs110web.tar
 - Re-tag the images with your own registry and push them up.
- To push any local Docker images to a repository, it is mandatory to first tag that image. We can also configure these commands in Jenkins to execute them automatically.

```
4. Use the below batch scripts to configure the 'Push Docker images to AWS ECR' in Jenkins:
  @echo off
  set AWS AccountID=678035612169
  set AWS Region=ap-south-1
  set ImageName=omnidocs11.0web
  set ImageTag=sp1
  set BuildNumber=%ImageTag%-build-%BUILD NUMBER%
  aws configure set aws access key id %AWS AccessKey%
  aws configure set aws secret access key %AWS SecretKey%
  aws ecr get-login-password --region %AWS Region% | docker login --username AWS
  --password-stdin %AWS AccountID%.dkr.ecr.%AWS Region%.amazonaws.com
  aws ecr describe-repositories --repository-names %ImageName% || aws ecr
  create-repository --repository-name %ImageName% --image-scanning-configuration
  scanOnPush=true
  docker tag %ImageName%:%ImageTag%
  %AWS AccountID%.dkr.ecr.%AWS Region%.amazonaws.com/%ImageName%:%ImageTag%
  docker push
  %AWS AccountID%.dkr.ecr.%AWS Region%.amazonaws.com/%ImageName%:%ImageTag%
  docker tag %ImageName%:%ImageTag%
  %AWS AccountID%.dkr.ecr.%AWS Region%.amazonaws.com/%ImageName%:%BuildNumber%
  docker push
  %AWS AccountID%.dkr.ecr.%AWS Region%.amazonaws.com/%ImageName%:%BuildNumber%
```



Figure 4.3

Pull:

1. Execute the below command to pull the Docker images from AWS ECR:

```
docker pull
```

<AWS_AccountID>.dkr.ecr.<AWS_Region>.amazonaws.com/<ImageName>:<ImageTag>

2. Use the below **batch** scripts to configure the **Pull Docker images from AWS ECR** in Jenkins:

```
Figure 4.4
```

4.5 Configuring AWS CodePipeline

To configure the AWS CodePipeline, follow the below subsections:

- Creation of IAM Policy and IAM Role
- <u>Creation of AWS CodeCommit Repository</u>
- <u>Creation of AWS CodeBuild Project</u>
- <u>Creation of AWS CodePipeline</u>

4.5.1 Creating an IAM Policy and IAM Role

Perform the below steps to create IAM Policy and Role:

1. Create an IAM policy with the name EKS-cluster-access with the following permissions:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": "eks:DescribeCluster",
            "Resource": "*"
        }
    ]
}
```

2. Create another IAM policy with the name **code-build-service-policy** with the following permissions:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Resource": [
                "arn:aws:logs:ap-south-1:678035612169:log-group:*"
            ],
            "Action": [
                "logs:CreateLogGroup",
                "logs:CreateLogStream",
                "logs:PutLogEvents"
            1
        },
        {
            "Effect": "Allow",
            "Resource": [
                "arn:aws:s3:::codepipeline-ap-south-1-*"
            ],
            "Action": [
                "s3:PutObject",
                "s3:GetObject",
                "s3:GetObjectVersion",
                "s3:GetBucketAcl",
                "s3:GetBucketLocation"
```

```
},
{
    "Effect": "Allow",
    "Resource": [
        "arn:aws:codecommit:ap-south-1:678035612169:*"
    ],
    "Action": [
        "codecommit:GitPull"
    ]
}
]
```

NOTE:

The policy **code-build-service-policy** is created for the AWS Region **ap-south-1** only. If you want to create this IAM role for other regions, then update the region in the JSON policy file. Use your AWS account ID as the place of **678035612169** in the above JSON policy file.

3. Create an IAM role with the name **genesis-codebuild-eks** and attach the policy **EKS-clusteraccess** and **code-build-service-policy** created in the previous step. It must be applied for the **codebuild** service. This is required for the CodeBuild role to authenticate with the EKS cluster.

NOTE:

CodeBuild role has permission to authenticate the cluster but doesn't have the requisite RBAC access to do any other action on the cluster. Due to the reason that when an Amazon EKS cluster is created, the IAM entity user or role that creates the cluster is automatically granted system masters permissions in the cluster's RBAC configuration. To grant additional AWS users or roles the ability to interact with your cluster, you must edit the aws-auth ConfigMap within Kubernetes.

4. Execute the below command to open the **aws-auth ConfigMap** in edit mode: kubectl edit configmap aws-auth -n kube-system

5. Add the following under data.mapRoles:

```
    rolearn: <ARN of the created IAM role for CodeBuild>
        username: <Name of the created IAM role for CodeBuild>
        groups:
            - system:masters
```

For example,

```
- rolearn: arn:aws:iam::678035612169:role/genesis-codebuild-eks
    username: genesis-codebuild-eks
    groups:
        - system:masters
```

6. The final aws-auth ConfigMap must look somewhat like this:

```
apiVersion: v1
kind: ConfigMap
metadata:
```

```
name: aws-auth
namespace: kube-system
data:
mapRoles: |
    - rolearn: arn:aws:iam::678035612169:role/EKSCluster-NodeInstanceRole-
IFRJ044RCC90Q
    username: system:node:{{EC2PrivateDNSName}}
    groups:
        - system:bootstrappers
        - system:nodes
        - rolearn: arn:aws:iam::678035612169:role/genesis-codebuild-eks
        username: genesis-codebuild-eks
        groups:
                - system:masters
```



Figure 4.5

The CodeBuild role has the requisite RBAC access.

4.5.2 Creating AWS CodeCommit repository

Perform the below steps to create AWS CodeCommit Repository:

- 1. Open the AWS CodeCommit console at: http://console.aws.amazon.com/codesuite/codecommit/home
- 2. Select Create repository.
- 3. For the **Repository name**, enter a unique name for your repository, that is, **Genesis-CodeCommit-Repository**.
- 4. Select Create.

Developer Tools > CodeCommit > Repositories > Create repository								
Create repository								
Create a secure repository to store and share your code. Begin by typing a repository name and a description for your repository. Repository names are included in the URLs for that repository.								
Repository settings								
Repository name								
Genesis-CodeCommit-Repository								
100 characters maximum. Other limits apply.								
Description - optional								
created for AWS CodePipeline								
1,000 characters maximum								
Tags Add								
Cancel Create								

Figure 4.6

- 5. Upload all the YAML files shared with the Release Package to the created AWS CodeCommit repository:
 - AWS_ALB-IngressController.yml
 - buildspec.yml
 - buildspec_EasySearch.yml
 - OmniDocs11.0Web.yml
 - OmniDocs11.0Web_Services.yml

- OmniDocs11.0EJB.yml
- OmniDocs11.0Services.yml
- EasySearch11.0.yml
- EasySearch11.0_ApacheOnly.yml
- TEM11.0.yml
- OmniScanWeb6.0.yml
- RMSSharePointAdapter.yml
- OmniDocs11.0MessagingService.yml

NOTE:

The YAML file is a human-readable object configuration file that is used to deploy and manage the objects on the Kubernetes cluster. In other words, it is a manifest file that contains the deployment descriptor of Docker images.

aws Services	Q Search for	r services,	features, b	ologs, docs, and more [Alt+	5]	۶.	¢	0	Mumbai 🔻	omnido	ocs @ ngde	ev ▼
Developer Tools	×	•	ß	OMS_EJB_QA.yaml							•	(i)
CodeCommit			D	OMS_EJB.yml								
▼ Source • CodeCo	ommit		ß	OMS_HighCharts.yml								
Getting started			ß	OMS_Mail.yml								
Repositories			ß	OMS_Tracker.yml								
Pull requests			ß	OMS_WEB_QA.yaml								
Commits			D	OMS_WEB.yml								
Branches Git tags			ß	OMS_WhatsApp.yml								
Settings			ß	OMS_Wrapper_Auth.yaml								
Approval rule ter	nplates		ß	OXDocumentConverter.yml								
Artifacts • Code.	Artifact		ß	OXEngine.yml								
▶ Build • CodeBuil	d		ß	OXExtractionServices.yml								

Figure 4.7

4.5.3 Creating AWS CodeBuild project

This section explains how to create AWS Code Build Project.

NOTE:

Use the upcoming steps as a reference to configure the Release Pipeline for the below Docker Images:

- OmniDocs 11.0 Web
- OmniDocs 11.0 Web_Services
- OmniDocs 11.0 EJB
- OmniDocs 11.0Services
- EasySearch11.0
- TEM11.0
- OmniScanWeb6.0
- RMS SharePoint Adapter
- Messaging Service

Perform the below steps to create AWS Code Build Project:

- 1. Open the AWS CodeBuild console at: http://console.aws.amazon.com/codesuite/codebuild/home
- Select Create build project.
 Specify the following. Once done, select Create build project.
- 3. Specify the following in the Project configuration.
 - i. Enter a unique name for your CodeBuild project that is, **OmniDocs101Web** in the **Project Name**.
 - ii. (Optional) Enter a description of the build project to help other users understand the project.
 - iii. (Optional) Select **Enable Build badge**. Build badge provides an embeddable, dynamically generated image (badge) that displays the status of the latest build for a project.
 - iv. Restrict the concurrent build limit to start the project to 1.
 - v. Keep the other settings as default.
| Project configuration |
|---------------------------------------------------------------------------------------------------------------------------------------|
| Project name |
| OmniDocs101Web |
| A project name must be 2 to 255 characters. It can include the letters A-Z and a-z, the numbers 0-9, and the special characters - and |
| Description - optional |
| Created for EKS Deployment |
| |
| Build badge - optional |
| ✓ Enable build badge |
| Enable concurrent build limit - optional
Limit the number of allowed concurrent builds for this project. |
| Restrict number of concurrent builds this project can start |
| Concurrent build limit |
| 1 |
| The concurrent build limit must be greater than 0 and less than the account build limit. |
| Additional configuration
tags |



- 4. Specify the following in the **Source**.
 - i. Select the AWS CodeCommit in the **Source provider**.
 - ii. Select the existing AWS CodeCommit repository **Genesis-CodeCommit-Repository** created in the **Creation of AWS CodeCommit Repository**.
 - iii. Select **Branch** as main.
 - iv. Keep the other settings as default.

Source	Add source
Source 1 - Primary	
Source provider	
AWS CodeCommit	▼
Repository	
Q Genesis-CodeCommit-Repository	×
Choose the source version reference type that contains your source co Branch Git tag Commit ID	Gemmit ID antional
Choose a branch that contains the code to build.	Choose a commit ID. This can shorten the duration of your build.
main 🔻	Q
Source version Info	
refs/heads/main	
7fb6f314 ok	
 Additional configuration Git clone depth, Git submodules 	

Figure 4.9

- 5. Specify the following in the **Environment**:
 - i. Select the Managed image in the Environment image.
 - ii. Select Amazon Linux 2 in the Operating system.
 - iii. Select **Standard** in the **Runtime(s)**.
 - iv. For Image, select 'aws/codebuild/amazonlinux2-x86_64-standard:3.0'.
 - v. Select Always use the latest image for this runtime image for the image version.
 - vi. Select the Existing service role **genesis-codebuild-eks** created in the **Creation of IAM Policy and IAM Role** in the **Service role**.

Don't select the checkbox Allow AWS CodeBuild to modify this service role so it can be used with this build project. Also, keep the other settings as default.

Environment image	
• Managed image Use an image managed by AWS CodeBuild	Custom image Specify a Docker image
Operating system	
Amazon Linux 2	▼
The programming language runtimes are now includ recommended for new CodeBuild projects created in for details 2.	ed in the standard image of Ubuntu 18.04, which is the console. See Docker Images Provided by CodeBuil
Standard	•
Image	
aws/codebuild/amazonlinux2-x86_64-standard:3.0	•
mage version	
Always use the latest image for this runtime version	•
Privileged Enable this flag if you want to build Docker images or w elevated privileges Service role	ant your builds to get
Create a service role in your account	Existing service role Choose an existing service role from your account
Role ARN	
Q arn:aws:iam::678035612169:role/genesis-codebuild-e	xs X
Allow ANAC CodeDuild to medify this comites related it of	a be used with this build project

Figure 4.10

- 6. Specify the following in the **Buildspec**.
 - i. Select the Use a buildspec file in the Build specification.
 - ii. Specify *buildspec.yml* in the **Buildspec name optional**.

By default, CodeBuild looks for a file named buildspec.yml in the source code root directory. If your buildspec file uses a different name or location, enter its path from the source root here (for example, buildspec-two.yml or *configuration/buildspec.yml*).

Buildspec	
Build specifications	
• Use a buildspec file Store build commands in a YAML-formatted buildspec file	 Insert build commands Store build commands as build project configuration
Buildspec name - optional By default, CodeBuild looks for a file named buildspec.yml in the sour location, enter its path from the source root here (for example, builds	ce code root directory. If your buildspec file uses a different name or pec-two.yml or configuration/buildspec.yml).
buildspec.yaml	

Figure 4.11

- 7. Batch configuration: Leave with default settings.
- 8. Artifacts: Leave with default settings.
- 9. Specify the following in the Logs:
 - i. Select the CloudWatch logs optional.
 - ii. Specify the Group name the same as the CodeBuild project name that is., **OmniDocs101Web**.
 - iii. For the Stream name, specify the codebuild.

Logs
CloudWatch
Cloudwatch logs - optional Checking this option will upload build output logs to CloudWatch.
Group name
OmniDocs101Web
Stream name
codebuild
S3
S3 logs - optional Checking this option will upload build output logs to \$3.
Cancel Create build project

Figure 4.12

10. Select Create build project.

NOTE:

The same CodeBuild project can be used for all types of stages like Dev, UAT, and Production or any other stage as per the business requirement.

4.5.4 Creating AWS CodePipeline

The three stages are configured that is, Dev, UAT, and Production, and at each stage, the deployment is quite different. You can have some more stages depending on the requirements.

NOTE:

Use the following steps as a reference to configure the Release Pipeline for the below Docker Images.

- OmniDocs11.0Web
- OmniDocs11.0Web_Services
- OmniDocs11.0EJB
- OmniDocs11.0Services
- EasySearch11.0
- TEM11.0
- OmniScanWeb6.0
- RMS SharePoint Adapter
- Messaging Service

Dev Stage: As soon as any Docker Image is pushed to the AWS Elastic Container Registry, AWS CodePipeline triggers the deployment to the Dev environment.

UAT Stage: UAT and Production deployments are approval based and they are called on-demand. To deploy to the UAT environment, triggers the UAT deployment. Once deployment is triggered, an approval mail to the concerned team. Upon receiving approval, UAT deployment gets started automatically.

Production Stage: Production Deployment is also based on approval based but it is multi-level approval. To deploy to a production environment, you require the approval of multiple stakeholders but deployment for the production environment is not get triggered automatically. A manual intervention mail is sent to the engineer who is supposed to deploy to production with a checklist. During the process, if the checklist points are not covered then the deployment to production gets rejected.

4.5.4.1 Configuring AWS CodePipeline for Dev stage

Perform the below steps to configure the Dev Stage:

- 1. Open the AWS CodePipeline console at: http://console.aws.amazon.com/codesuite/codepipeline/home
- 2. Select the Create pipeline.

Specify the required details in the following steps. Once complete, select **Create pipeline** at the Review step:

- 3. Select pipeline settings:
 - i. Enter a unique name for your pipeline, that is, **OmniDocs101Web-DevStage** for the Pipeline name.
 - ii. Select New service role for Service role.
 - iii. Select the checkbox Allow AWS CodePipeline to create a service role so it can be used with this new pipeline.
 - iv. Keep the other settings as default and click **Next**.

hoose pipeline settings Info	
Pipeline settings	
Pipeline name Enter the pipeline name. You cannot edit the pipeline name after it is o	reated.
OmniDocs101Web-DevStage	
No more than 100 characters	
Service role	
• New service role Create a service role in your account	Choose an existing service role from your account
Role name	
AWSCodePipelineServiceRole-ap-south-1-OmniDocs101W	eb-DevStage
 Type your service role name Allow AWS CodePipeline to create a service role so it can pipeline 	be used with this new
 Advanced settings 	
	Cancel Next

Figure 4.13

- 4. Specify the following in the Add source stage:
 - i. Select the AWS CodeCommit for the Source provider.
 - ii. Select the existing **AWS CodeCommit repository** 'Genesis-CodeCommit-Repository' created in the Creation of AWS CodeCommit Repository for the **Repository name**.
 - iii. Select Main for the Branch name.
 - iv. Select the recommended option Amazon CloudWatch Events for Change detection options.

NOTE:

Amazon CloudWatch Events creates a CloudWatch event rule. As soon as the changes are done in the integrated AWS CodeCommit repository, it triggers the pipeline. But do not trigger the AWS CodePipeline whenever there is a change in

the CodeCommit repository. The pipeline must be triggered whenever you push a new image to the container registry like AWS ECR.

Refer to the following sections for the configuration of AWS ECR with CodePipeline to disable the CloudWatch event rule once the pipeline is created.

v. Keep the other settings as default and click **Next**.

Source			
Source provider This is where you stored your input artifacts for your pipeline. Choose	the provider and then provide the cor	nection details.	
AWS CodeCommit	▼		
Repository name Choose a repository that you have already created where you have pu	shed your source code.		
Q Genesis-CodeCommit-Repository	×		
Branch name Choose a branch of the repository			
Q main	×		
Change detection options Choose a detection mode to automatically start your pipeline when a Amazon CloudWatch Events (recommended)	change occurs in the source code.		
Use Amazon CloudWatch Events to automatically start my pipeline when a change occurs	Use AWS CodePipeline to che	ck periodically for c	hanges
Output artifact format Choose the output artifact format.			
• CodePipeline default AWS CodePipeline uses the default zip format for artifacts in the pipeline. Does not include git metadata about the repository.	 Full clone AWS CodePipeline passes me that allows subsequent action supported for AWS CodeBuild 	tadata about the re ns to do a full git clo d actions.	pository one. Only
	5l	Destina	
	Cancel	Previous	Next



- 5. Specify the following in the Add build stage:
 - i. Select the AWS CodeBuild in the Build provider.
 - ii. Select a **region** in which you want to create your pipeline.
 - iii. Select the existing CodeBuild project **OmniDocs101Web** created in the **Creation of AWS CodeBuild Project**.
 - iv. Create the below environment variables for Environment variables (optional):

Name	Value	Туре
AWS_DEFAULT_REGION	ap-south-1	Plaintext
AWS_CLUSTER_NAME	Omnidocs-uat2	Plaintext

YAML_FILE	OmniDocs11.0Web.yml	Plaintext
CODE_PIPELINE_EXECUTION_ID	#{codepipeline.PipelineExecutionId}	Plaintext

- **AWS_DEFAULT_REGION**: Specify the region where the AWS EKS cluster is created.
- **AWS_CLUSTER_NAME:** Specify the name of the EKS cluster for the Dev stage.
- **YAML_FILE**: Specify the name of the YAML file that is stored in the AWS CodeCommit repository and that is used to deploy the **OmniDocs10.1Web** container for the Dev stage.
- **CODE_PIPELINE_EXECUTION_ID:** This variable is just created for logging purposes so that you can track the build-id and its initiated pipeline.
- v. For **Build type**, select **Single build** and Click **Next**.

This is the tool of your build proj	ect. Provide build artifact details like	operating system, build spec file	e, and ou	tput file names.
AWS CodeBuild		•		
Region				
Asia Pacific (Mumbai)		•		
Project name Choose a build project that you h and then return to this task. Q OmniDocs101Web	ave already created in the AWS Code	Build console. Or create a build	project ir	the AWS CodeBuild consol
Environment variables - opti Choose the key, value, and type f CodePipeline. Learn more 🔁 Name	onal or your CodeBuild environment varia Value	bles. In the value field, you can	reference	variables generated by
AWS_DEFAULT_REGION	ap-south-1	Plaintext		Remove
AWS_CLUSTER_NAME	Omnidocs-uat2	Plaintext 🔻		Remove
YAML_FILE	OmniDocs10.1Web.yml	Plaintext •		Remove
CODE DIRELINE EXECUT	#{codepipeline.Pipeline	Plaintext		Remove
CODE_FIFELINE_EXECU				
Add environment varia	ble			
Add environment varia	ble			

Figure 4.15

6. In the Add deploy stage: Skip the deploy stage.

Deploy - optional
Deploy provider Choose how you deploy to instances. Choose the provider, and then provide the configuration details for that provider.
Cancel Previous Skip deploy stage Next



7. In the Review, select Create pipeline.

Step 4: Add deploy stage	
Deploy action provider	
Deployment stage No deploy	
	Cancel Previous Create pipeline
	Figure 4.17

NOTE:

As soon as you create the pipeline, it starts the first pipeline execution. This execution failed as expected if you have not yet integrated the AWS ECR into the pipeline. You need to do the same.

Perform the below steps to integrate AWS ECR into the AWS CodePipeline:

- 1. Open the created pipeline **OmniDocs101Web-DevStage** in **Edit** mode.
- 2. Select the Edit source stage.
- 3. Select + Add action.

Developer Tools > CodePipeline > Pipelines > OmniDocs101Web-DevStage > Edit OmniDocs101Web-DevStage						
Editing: OmniDocs101Web-DevStage	Delete Cancel Save					
Edit: Source	Cancel Done					
+ Add action group						
Source ③ AWS CodeCommit + Add action						
+ Add action group						

Figure 4.18

- 4. In the Edit action panel, specify the unique Action name. that is., AWS-ECR-Registry
- 5. Select the Amazon ECR in **Action provider**.
- 6. Select the **omnidocs10.1web** Docker image that needs to deploy to the Dev stage in the Repository name.

- 7. In **Image tag optional**, select the image tag that you want to use to set up the continuous deployment trigger.
- 8. In **Variable namespace optional**, specify the unique namespace, that is, **AWS-ECR**. This is required to use its output variable in the following sections.
- 9. In **Output artifacts**, specify the unique variable name that is, **SourceArtifact1**. SourceArtifact is already used by AWS CodeCommit action.
- 10. Click Done.

Edit action	×
Action name Choose a name for your action	
AWS-ECR-Registry	
No more than 100 characters	
Action provider	
Amazon ECR 🔹	
Repository name Choose an Amazon ECR repository as the source location.	
Q omnidocs10.1web X C	
Image tag - <i>optional</i> Choose the image tag that triggers your pipeline when a change occurs in the image repository.	
Q patch3hf19-alpine-openjdk X C	
If an image tag is not selected, defaults to latest	
Variable namespace - optional Choose a namespace for the output variables from this action. You must choose a namespace if you want to use the variables this action produces in your configuration. Learn more 🗹	
AWS-ECR	
Output artifacts Choose a name for the output of this action.	
SourceArtifact1	
No more than 100 characters	
	Cancel Done

Figure 4.19

- 11. Click **Done** on the Edit source stage.
- 12. Select the Edit build stage.
- 13. Click **Edit** in the AWS CodeBuild action.

Edit: Build		
+ Add action group		
Build AWS CodeBuild	© ×	+ Add action
+ Add action group		

Figure 4.20

14. Add three new environment variables as given in the table below:

Name	Value	Туре
IMAGE_REGISTRY_ID	#{AWS-ECR.RegistryId}	Plaintext
IMAGE_REPOSITORY_NAME	#{AWS-ECR.RepositoryName}	Plaintext
IMAGE_TAG	#{AWS-ECR.ImageTag}	Plaintext

Here, **AWS-ECR** is the name of the variable namespace, created in the Amazon ACR action.

Environment variables - optional Choose the key, value, and type for your CodeBuild environment v	ariables. In the value field, you can reference variables generated by	CodePipeline. Learn more 🔀	
Name	Value	Туре	
AWS_DEFAULT_REGION	ap-south-1	Plaintext 🔻	Remove
AWS_CLUSTER_NAME	Omnidocs-uat2	Plaintext v	Remove
YAML_FILE	OmniDocs10.1Web.yml	Plaintext v	Remove
CODE_PIPELINE_EXECUTION_ID	#{codepipeline.PipelineExecutionId}	Plaintext 💌	Remove
IMAGE_REGISTRY_ID	#{AWS-ECR.RegistryId}	Plaintext v	Remove
IMAGE_REPOSITORY_NAME	#{AWS-ECR.RepositoryName}	Plaintext 🔻	Remove
IMAGE_TAG	#{AWS-ECR.ImageTag}	Plaintext 🔻	Remove
Add environment variable			

Figure 4.21

- 15. Click **Done** on the edit action panel.
- 16. In the Edit build stage, click **Done**.
- 17. Click **Save** to save the pipeline.

De	Developer Tools > CodePipeline > Pipelines > OmniDocs101Web-DevStage > Edit OmniDocs101Web-DevStage						
E	Editing: OmniDocs101Web-DevStage						
	Edit: Source						Edit stage
	Source AWS CodeCommit	٤	AWS-ECR-Registry Amazon ECR	٩			

Figure 4.22

Once the AWS ECR is integrated into the pipeline, it creates a new CloudWatch event rule that acts as a deployment trigger. Now, whenever you push the new Docker image with the same image tag name that is defined in the Amazon ECR action in the source stage, it triggers the pipeline.

As described in the **Add Source Stage**, the AWS CodeCommit action creates a new CloudWatch event rule, and it triggers the AWS CodePipeline whenever there is a change in the CodeCommit repository in which the CloudWatch event rule is disabled once the pipeline is created.

Perform the below steps to disable the CloudWatch event rule created against the AWS CodeCommit action:

- 1. Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/
- 2. In the **Events** tab, click **Rules** on the navigation panel.
- 3. Search the rule created against the AWS CodeCommit repository Genesis-CodeCommit-Repository created in the Creation of AWS CodeCommit Repository.

aws Services V	Q Search for services, feat	ures, marketplace products, and docs	[Alt+S]	∑ 🚓 omnidocs@ngdev ▼ Mumbai ▼ Support ▼
CloudWatch ×	Rules > codepipeline-Genesi-m	ain-476342-rule		Actions -
Dashboards	Summary			
 Alarms 	ARN 1 am:aws:events:ap-south-1:678035612169:rule/	codepipeline-Genesi-main-476342-rule		
In alarm () Insufficient data () OK () Billing () Log groups Insights () W Logs Log groups Insights () W Aetrics Exponer Streams () Wer () Streams () Event Buses Event Buses Service Map Traces ()	Event pattern ① { **sorce*:[**sorce*:[**sorce*:[**conce*::[**conce*::[**conce*::[**conce*::[**sorce*::[*sorce*::[*sorc	81:Gewesis-CoseConnit-Reportiony* 9:Start your pipeline when a change occ	curs in the AWS CodeCommit source repository and branch. De	leting this may prevent changes from being detected in that pipeline. Read
 Container Insights New 	-			
Resources Performance monitoring	Filter:			$\ll~<$ Viewing 1 to 1 of 1 Targets $>~\gg$
Lambda Insights New	Type Name	Input	Role	Additional parameters
Performance monitoring	CodePipeline OmniDocs101Web-DevSta	00 Matched event	cwe-role-ap-south-1-OmniDocs101Web-DevStage	
Synthetics Canaries Contributor Insights Settings				
Feedback English (US) 🔻			© 2008 - 2021, Amazon Internet Services Private Ltd. or its affiliate	s. All rights reserved. Privacy Policy Terms of Use Cookie preferences
		Figure 4	1.23	

4. Select the rule, go to the **Actions** menu, and select **Disable** or **Delete**. It does not trigger the pipeline whenever any change is done in the AWS CodeCommit repository.

Rules route events from your AWS resources for processing by selected targets. You can create, edit, and delet	e rules
Create rule Actions -	
Status All Delete	
Statu Disable	
CW_SQL-POC_START	
CW_SQL-POC_STOP	
ECRTrigger	
codepipeline-Genesi-main-476342-rule	
codepipeline-omnido-patchh-134251-rule	
startec2instance	
Stopec2instance	

Figure 4.24

5. Click **Release change** to trigger the pipeline manually.

aws Services 🔻		Q Search for ser	vices, features, marketplace products, and docs	[Alt+S]	∑ 🗘 omnidocs @ ngdev ▼ Mumbe	ii 🔻 Support 🔻
Developer Tools CodePipeline	×	Developer Tools > CodePipeline > F	ipelines > OmniDocs101Web-DevStage	↓ Notify ▼ Edit Stop execution	Clone pipeline Release change	Î
Source • CodeCommit		Onnibots to tweb-b	evolage			-
 Artifacts • CodeArtifact Build • CodeBuild 		Source Succeeded Pipeline execution ID: 34b7a42e-4a51-47a5	-b89d-da5fe79e323f			
 Deploy • CodeDeploy Pipeline • CodePipeline Getting started 		Source () AWS CodeCommit () Succeeded - 43 minutes ago 7hb6/\$14	AWS-ECR-Registry Amazon ECR B Didn't Run No executions yet			
Pipelines Pipeline History Settings Settings		7fb6f314 Source: ok				0
Q Go to resource		Build Failed Pipeline execution ID: 34b7a42e-4a51-47a5	-b89d-da5fe79e323f		Retry	
age - sources		Build () AWS CodeBuild () Failed - 42 minutes ago Details				
		7fb6f314 Source: ok				-
Feedback English (US) 🔻				© 2008 - 2021, Amazon Internet Services Private Ltd. or its affiliates. All	I rights reserved. Privacy Policy Terms of Use	Cookie preferences

Figure 4.25

4.5.4.2 Configuring notification

This section contains the configuration of notification using the AWS **SNS topic**, to notify the recipient(s) about the pipeline execution status whether it succeeded or failed. The following are the steps to configure notifications:

- 1. Create an SNS topic
- 2. Create a subscription to the SNS topic
- 3. Create a Lambda function
- 4. Create a CloudWatch event rule

Perform the below steps to create an SNS topic:

- 1. Sign in to the Amazon SNS console https://console.aws.amazon.com/sns/home
- 2. Select the **Region** to create your repository on the navigation panel.
- 3. Select **Topics** in the left navigation panel.
- 4. Select Create topic.
- 5. By default, the console creates a FIFO topic, select **Standard**.
- 6. Enter the Name for the topic such as **SNSTopic1** in the **Details** section.
- 7. In **Display name optional**, use display name such as **DevOps Admin**.
- 8. Select Create topic.

Details	
ype Info opic type cannot be modified after topic is created	
 FIFO (first-in, first-out) Strictly-preserved message ordering Exactly-once message delivery High throughput, up to 300 publishes/second Subscription protocols: SQS 	 Standard Best-effort message ordering At-least once message delivery Highest throughput in publishes/second Subscription protocols: SQS, Lambda, HTTP, SMS, email, mobile application endpoints
lame	
SNSTopic1	
Maximum 256 characters. Can include alphanumeric characters, hy Display name - optional fo use this topic with SMS subscriptions, enter a display name. Onl	phens (-) and underscores (_).

Figure 4.26

Perform the below steps to create a subscription to the SNS topic:

- 1. In the left navigation pane, select **Subscriptions**.
- 2. Click Create subscription. The Create Subscription screen appears.

- 3. Select the **Topic ARN**.
- 4. Select **Email** for **Protocol**.
- 5. In **Endpoint**, enter an email address to receive notifications.
- 6. Select the Create subscription.

Create subscription		
Details		
Topic ARN		
Q arn:aws:sns:us-east-1:678035612169:SNSTopic1 X		
Protocol The type of endpoint to subscribe		
Email		
Endpoint An email address that can receive notifications from Amazon SNS.		
vivek_kumar@newgen.co.in		
③ After your subscription is created, you must confirm it. Info		
 Subscription filter policy - optional This policy filters the messages that a subscriber receives. Info 		
Redrive policy (dead-letter queue) - optional Send undeliverable messages to a dead-letter queue. Info		
	Cancel	Create subscription



- Check your email inbox and select Confirm subscription in the email from AWS Notifications. The sender ID is usually <u>no-reply@sns.amazonaws.com</u>. The Amazon SNS opens in a web browser and displays a subscription confirmation with your subscription ID.
- 8. Create more subscriptions and attached them to the same topic that you created to send emails to multiple recipients.
- 9. Once your subscription is created, click **Confirm**.

Perform the below steps to create a Lambda function:

- 1. Open the function page on the lambda console: https://console.aws.amazon.com/lambda/home
- 2. Select Create function.
- 3. In the Function name, specify the unique function name such as lambda-fuction1.
- 4. In **Runtime**, select **python 3.8** or the latest version.
- 5. Keep the other settings as default and select the **Create function**.

aws	Services 🔻	Q Search for services, features, marketplace products, and docs [Alt+S]	∑ 🗘 omnidocs @ ngdev ▼ Mumba	i 🔻 Support 🔻
≡	Create function Info			(
	Choose one of the following options to create your funct	ion.		
	Author from scratch • • • • • • • • • • • • • • • • • • •	Use a blueprint O Build a Lambda application from sample code and configuration presets for common use cases.	Browse serverless app repository Deploy a sample Lambda application from the A Serverless Application Repository.	ws
	Basic information			
	Function name Enter a name that describes the purpose of your function.			
	lambda-function1			
	Use only letters, numbers, hyphens, or underscores with no spaces.			
	Runtime Info Choose the language to use to write your function. Note that the or	onsole code editor supports only Node.js, Python, and Ruby.		
	Python 3.8	▼		
	Permissions Info			
	By default, Lambda will create an execution role with permissions t	o upload logs to Amazon CloudWatch Logs. You can customize this default role later when adding triggers.		
	Change default execution role			
	Advanced settings			
			Cancel Create funct	ion
Feedback	English (US) 🔻	© 2008 - 2021, Amazon Internet Services Private Ltd. or its af	ffiliates. All rights reserved. Privacy Policy Terms of Use	Cookie preferences

Figure 4.28

- 6. Under the Code tab, select lambda_function.py.
- 7. Replace the default code snippet using the below code snippet and select **Deploy**:

```
import json
import boto3
sns = boto3.client('sns')
pipeline sns map = {
    "pipeline1": "sns arn 1",
    "pipeline2": "sns arn 1"
    "pipeline3": "sns_arn_2"
}
def lambda handler(event, context):
    detail = event['detail']
   pipeline = detail['pipeline']
   execution id = detail['execution-id']
   state = detail['state']
    sns1 = pipeline sns map[pipeline]
    subject = "Pipeline " + pipeline + " has " + state
   message = "Pipeline name : " + pipeline + " has " + state + " with
   execution id : " + execution_id
    # print(message)
    response = sns.publish(
        TopicArn = sns1,
        Message= message,
        Subject=subject
    )
    return(response)
```

8. In the above code snippet, update **pipeline name(s)** in pipeline1, pipeline2, and pipeline3 as well as update the **SNS topic(s)** ARN at the place of sns_arn_1, sns_arn_2.

NOTE:

You can use the same SNS topic for all the pipelines or different SNS topics for each pipeline. Add an entry for each newly created pipeline and its associated SNS topic to use them.

c	ode source Info			
-	File Edit Find View Go	Tools	Window Test Deploy Changes not deployed	
Q	Go to Anything (Ctrl-P)	٦	lambda_function × 🕀	
Environment	 lambda-fuction1 - / ** lambda_function.py 	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25	<pre>import json import boto3 sns = boto3.client('sns') pipeline_sns_map = { "OmnIDocs101Web-DevStage": "arn:aws:sns:ap-south-1:678035612169:SNSTopic1", "pipeline2": "sns_arn_1" "pipeline3": "sns_arn_2" } def lambda_handler(event, context): detail = event['detail'] pipeline = detail['pipeline'] execution_id = detail['execution-id'] state = detail['state'] sns1 = pipeline_sns_map[pipeline] subject = "Pipeline " + pipeline + " has " + state message = "Pipeline nee : " + pipeline + " has " + state + " with execution # print(message) response = sns.publish(TopicArn = sns1, Message=message, Subject=subject) return(response)</pre>	id : " + execution_id

Figure 4.29

9. Go to the **Configuration** tab and select **Permissions**.

10. Select the created **IAM role** for this lambda function. The IAM role **Summary** screen appears.

Code Test Monitor	Configuration Aliases Versions		
General configuration	Execution role		
Triggers			
Permissions	lambda-fuction1-role-bt5m62bg		
Destinations			
Environment variables	Resource summary		

Figure 4.30

- 11. Select the Add inline policy.
- 12. In Service, select the SNS.
- 13. In Actions, select Publish.
- 14. Select All resources in Resources.
- 15. Select the **Review policy**.

aws	Services	• Q	Search for services, features, marketplace products, and docs [Alt+S]	ocs @ ngdev ▼	Global 🔻	Support 🔻
		✓ SNS (1 action)	Clone	Remove		•
		Service Actions	SNS Write Publish			
Documentation		✓ Resources close	Specific All resources As a best practice, define permissions for only specific resources in specific accounts. Alternatively, you can grant least privusing condition keys. Learn more	lege		
		 Request conditions 	Specify request conditions (optional)			
			Add additional pr	ermissions		
		Character count: 117 of 10,240. The current character count includes character	for all inline policies in the role: lambda-fuction 1-role-bt5m62bg.	Review policy		v
Feedback	English (US	S) v	© 2008 - 2021, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privace	Policy Terms	of Use Co	ookie preferences

Figure 4.31

- 16. Specify the policy name such as **sns-lambda-policy**.
- 17. Select Create policy.

Perform the below steps to create a CloudWatch event rule:

- 1. Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/
- 2. In the Events tab, select the **Rules** on the navigation pane.
- 3. Select the Create rule.
- 4. Select the **Event Patten** radio button in the Event Source.
- 5. Select the Service Name as CodePipeline using the dropdown.
- 6. Select the Event Type as CodePipeline Pipeline Execution State Change using the dropdown.
- 7. Select the **Specified state(s)** and select **FAILED** and **SUCCEEDED** states.
- 8. Click +Add target* given on the upper-right.
- 9. Select the Lambda function as a target.
- 10. For **Function***, select the existing function name **lambda-function1**.
- 11. Select the **Configure details** given at the lower right.

aws Services ▼		Q Search for services, features, marketplace products, and docs	[Alt+S] ∑ 🗘 omnidocs@ingdev ▼ Mumbal ▼ Support ▼
CloudWatch	×	Step 1: Create rule	
Dashboards		Create rules to invoke Targets based on Events happening in your AWS environment.	
In alarm Insufficient data OK Billing ▼ Logs Log groups Insights	0	Event Source Build or customize an Event Pattern or set a Schedule to invoke Targets. Event Pattern Schedule Event Pattern to match events by service Service Name CodePipeline	Targets Select Target to invoke when an event matches your Event Pattern or when schedule is triggered. Lambda function
Metrics Explorer Streams New Events Rules Event Buses		Event Type CodePipeline Pipeline Execution State Change Any state Any state Specific state(s) FAILED × SUCCEEDED	Configure input Add target*
ServiceLens Service Map Traces Container Insights Resources Performance monitoring Lambda Insights New Performance monitoring Synthetics Canaries Contributor Insights Settings		<pre> • Event Pattern Preview Copy to clipboard Edit { "source": ["aws.codepipeline"], "detail-type": ["CodePipeline Pipeline Execution State Change"], "detail": { "fAILED", "FAILED", " "fAILED", " " } } } </pre>	
 Favorites + Add a dashboard 		Show sample event(s) Required	Cancel Configure details
Feedback English (US) 🔻	·		0 2008 - 2021, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

Figure 4.32

12. Specify the Unique Rule Name and Description on the Rule definition tab.

4.5.4.3 Configuring AWS CodePipeline for UAT stage

UAT deployments are based on approval and are available on-demand. To deploy to the UAT environment, you need to trigger the UAT deployment. Once deployment is triggered, an approval mail is sent to the project manager or the concerned team. Upon receiving the approval, the UAT deployment gets started automatically.

Perform the below steps to configure the UAT Stage:

- 1. Open the AWS CodePipeline console at: http://console.aws.amazon.com/codesuite/codepipeline/home
- 2. On the Welcome tab, select the Create pipeline.
- 3. Specify the required details in the following steps. Once complete, select **Create pipeline** at the Review step:

Select the pipeline settings and specify the following:

- i. Enter a unique name for your pipeline that is, **OmniDocs101Web-UATStage** in the **Pipeline name**.
- ii. Select the **New service** role in the Service role.

- iii. Select the checkbox Allow AWS CodePipeline to create a service role so it can be used with this new pipeline.
- iv. Keep the other settings as default and click **Next**.

Choose pipeline settings Info
Pipeline settings
Pipeline name Enter the pipeline name. You cannot edit the pipeline name after it is created.
OmniDocs101Web-UATStage
No more than 100 characters
Service role
New service role Create a service role in your account Existing service role Choose an existing service role from your account
Role name
AWSCodePipelineServiceRole-ap-south-1-OmniDocs101Web-UATStage
Type your service role name
Allow AWS CodePipeline to create a service role so it can be used with this new pipeline
Advanced settings
Cancel Next

Figure 4.33

- 4. Add source stage and specify the following:
 - i. Select the AWS CodeCommit in the Source provider.
 - ii. Select the existing AWS CodeCommit repository **Genesis-CodeCommit-Repository** created in **Creation of AWS CodeCommit Repository** in **Repository name**.
 - iii. Select the Main in the Branch name.
 - iv. Select the recommended option Amazon CloudWatch Events in the Change detection options.

Amazon CloudWatch Events creates a CloudWatch event rule. Once any changes are done in the integrated AWS CodeCommit repository, it triggers the pipeline. But do not trigger the AWS CodePipeline whenever there is a change in the CodeCommit repository. The pipeline must be triggered whenever you push a new image to the container registry like AWS ECR. Refer to the following sections for the configuration of AWS ECR with CodePipeline to disable the CloudWatch event rule once the pipeline is created. v. Keep the other settings as default and click **Next**.

Source	
Source provider This is where you stored your input artifacts for your pipeline. Choose	e the provider and then provide the connection details.
AWS CodeCommit	▼
Repository name Choose a repository that you have already created where you have pu	ished your source code.
Q Genesis-CodeCommit-Repository	×
Branch name Choose a branch of the repository	
Q main	×
Change detection options Choose a detection mode to automatically start your pipeline when a Amazon CloudWatch Events (recommended) Use Amazon CloudWatch Events to automatically start my	change occurs in the source code. AWS CodePipeline Use AWS CodePipeline to check periodically for changes
pipeline when a change occurs	eserves cours parte to creak periodically for changes
Output artifact format Choose the output artifact format.	
• CodePipeline default AWS CodePipeline uses the default zip format for artifacts in the pipeline. Does not include git metadata about the repository.	 Full clone AWS CodePipeline passes metadata about the repository that allows subsequent actions to do a full git clone. Only supported for AWS CodeBuild actions.
	Cancel Previous Next



- 5. Add the build stage and specify the following:
 - i. Select the **AWS CodeBuild** in the **Build provider** using the dropdown.
 - ii. Select a **Region** in which you want to create your pipeline using the dropdown.
 - Select the existing CodeBuild project OmniDocs101Web created in the Creation of AWS CodeBuild Project.
 - iv. For Environment variables optional, create the below environment variables:

Name	Value	Туре
AWS_DEFAULT_REGION	ap-south-1	Plaintext
AWS_CLUSTER_NAME	Omnidocs-uat2	Plaintext
YAML_FILE	OmniDocs10.1Web.yml	Plaintext
CODE_PIPELINE_EXECUTION_ID	#{codepipeline.PipelineExecutionId}	Plaintext

- **AWS_DEFAULT_REGION**: Specify the region where the AWS EKS cluster is created.
- AWS_CLUSTER_NAME: Specify the EKS cluster name created for the UAT stage.
- **YAML_FILE**: Specify the YAML file name that is stored in the AWS CodeCommit repository and that is used to deploy the OmniDocs10.1Web container for UAT Stage.
- **CODE_PIPELINE_EXECUTION_ID:** This variable is just created for logging purposes so that you can track the build-ID and its initiated pipeline.
- v. In **Build type**, select the Single build and click **Next**.

Build - optional						
Build provider This is the tool of your build proje	ct. Provide build artifact details like	operating system, build spec file, a	and output file names.			
AWS CodeBuild		•				
Region						
Asia Pacific (Mumbai)		•				
Project name Choose a build project that you h and then return to this task. Q OmniDocs101Web	ave already created in the AWS Code	Build console. Or create a build pr	oject in the AWS CodeBuild console			
Environment variables - opti Choose the key, value, and type for CodePipeline. Learn more	onal or your CodeBuild environment varia	bles. In the value field, you can ref	erence variables generated by			
Name	Value	Туре				
AWS_DEFAULT_REGION	ap-south-1	Plaintext v	Remove			
AWS_CLUSTER_NAME	/S_CLUSTER_NAME Omnidocs-uat2 P		Remove			
YAML_FILE	OmniDocs10.1Web.yml	Plaintext v	Remove			
CODE_PIPELINE_EXECU	#{codepipeline.Pipeline	Plaintext 🔻	Remove			
Add environment variable						
Build type						
Single build Triggers a single build. Batch build Triggers multiple builds as a single execution.						
	Car	ncel Previous	Skip build stage Next			

Figure 4.35

6. In the Add deploy stage, skip the deploy stage.

Deploy - optional	
Deploy provider Choose how you deploy to instances. Choose the provider, and then provide the configuration details for that provider.	
Cancel Previous Skip deploy stag	je Next
Cancel Previous Skip deploy stag	je Next

7. In the **Review**, select **Create pipeline**.

Step 4: Add deploy stage				
Deploy action provider				
Deployment stage No deploy				
	Cancel Previous Create pipeline			
F	igure 4.37			

NOTE:

Once the pipeline is created, it starts its first pipeline execution. This execution fails as expected when you have not yet integrated the AWS ECR into the pipeline. You must perform the same.

Perform the below steps to integrate AWS ECR into the AWS CodePipeline:

- 1. Open the created pipeline **OmniDocs101Web-UATStage** in **Edit** mode.
- 2. Select the Edit source stage.
- 3. Select + Add action.

Developer Tools > CodePipeline > Pipelines > OmniDocs101Web-UATStage > Edit OmniDocs101Web-UATStage Editing: OmniDocs101Web-UATStage Delete Cancel					
Edit: Source			Cancel Done		
+ Add action group					
Source () AWS CodeCommit	+ Add action				
+ Add action group					

Figure 4.38

- 4. In the Edit action panel, specify the unique Action name. that is, AWS-ECR-Registry.
- 5. Select the Amazon ECR in the Action provider.
- 6. Select the **omnidocs10.1web** in the **Repository name**. It enables the Docker image to deploy to the Dev stage.

- 7. In **Image tag optional**, select the image tag that you want to use to set up the continuous deployment trigger.
- 8. In **Variable namespace optional**, specify the unique namespace that is, AWS-ECR. This is required while using its output variable in the following sections.
- 9. In **Output artifacts**, specify the unique variable name. that is, **SourceArtifact1**. SourceArtifact is already used by AWS CodeCommit action.
- 10. Click Done.

Edit action	×
Action name Choose a name for your action AWS-ECR-Registry No more than 100 characters Action provider Amazon ECR Repository name Choose an Amazon ECR repository as the source location. Q omnidocs10.1web X Cign mage tag - optional Choose a namespace for the output variables from this action. You must choose a namespace if you want to use the variables this action produces in your configuration. Learn more Cignose a namespace for the output of this action. Output artifacts Choose a name for the output of this action. SourceArtifact1 No more than 100 characters	
	Cancel Done

Figure 4.39

- 11. Click **Done** on the Edit source stage.
- 12.Select the Edit build stage.
- 13. Click the **Edit** icon for AWS CodeBuild action.

Edit: Build			Cancel	Delete
+ Add action group]			
Build AWS CodeBuild	۵ × ک	+ Add action		
+ Add action group]			

Figure 4.40

14. Add the three new environment variables given in the table below:

Name	Value	Туре
IMAGE_REGISTRY_ID	#{AWS-ECR.RegistryId}	Plaintext
IMAGE_REPOSITORY_NAME	#{AWS-ECR.RepositoryName}	Plaintext
IMAGE_TAG	#{AWS-ECR.ImageTag}	Plaintext

Where, **AWS-ECR** is the name of the variable namespace, created in Amazon ACR action.

Environment variables - optional Choose the key, value, and type for your CodeBuild environment	variables. In the value field, you can reference variables generated by	CodePipeline. Learn more		
Name	Value	Туре		
AWS_DEFAULT_REGION	ap-south-1	Plaintext	•	Remove
AWS_CLUSTER_NAME	Omnidocs-uat2	Plaintext	▼	Remove
YAML_FILE	OmniDocs10.1Web.yml	Plaintext	▼	Remove
CODE_PIPELINE_EXECUTION_ID	#{codepipeline.PipelineExecutionId}	Plaintext	▼	Remove
IMAGE_REGISTRY_ID	#{AWS-ECR.RegistryId}	Plaintext	▼	Remove
IMAGE_REPOSITORY_NAME	#{AWS-ECR.RepositoryName}	Plaintext	▼	Remove
IMAGE_TAG	#{AWS-ECR.ImageTag}	Plaintext	▼	Remove
Add environment variable				

Figure 4.41

- 15. On the Edit action panel, click **Done**.
- 16. Click **Done** on the Edit build stage. Since this is the UAT stage and it must be an **approval-based** pipeline.
- 17. Select the + Add stage in between the Source stage and Build stage.

aws Services v	Q Search for services, features, marketplace products, and docs [Alt+S]	∑ 🔎 omnidocs@ngdev ▼ Mumbai ▼ Support		
Developer Tools X CodePipeline	Developer Tools > CodePipeline > Pipelines > OmniDocs101Web-UATStage > Edit OmniDocs101Web-UATStage	Delete Cancel Sam		
Source • CodeCommit	Editing. On induction web-OAT stage	Detete		
Artifacts • CodeArtifact	Edit: Source	Edit stage		
Build • CodeBuild				
Deploy • CodeDeploy	Source AWS-ECR-Registry			
▼ Pipeline • CodePipeline				
Getting started	+ Add stage			
Pipelines				
Pipeline				
Settings	Edit: Build	Edit stage		
Settings				
P Settings	Build O			
Q Go to resource	ANS COULDING			
🖽 Feedback				
	+ Add stage			

Figure 4.42

- 18. Specify the Stage name such as Approval.
- 19. Click +Add action group under the Approval stage.

Edit: Approval		Cancel Delete Done
+ Add action group		
	Figure 4.43	

- 20. Specify the unique action name such as Approval-for-UAT in the Action name.
- 21. Select the Manual approval in the Action provider.
- 22. Select the ARN of SNSTopic1 created in **Configuration of Notification** in the **SNS topic ARN – optional.**
- 23. Specify a comment to display for the reviewer in the email notifications or the console in **Comments-optional**. For Example, provide your approval for UAT deployment.
- 24. Keep the other settings as default and click **Done**.

Edit action	×
Action name Choose a name for your action	
Approval-for-UAT	
No more than 100 characters	
Action provider	
Manual approval 💌	
Configure the approval request.	
SNS topic ARN - optional	
Q arn:aws:sns:ap-south-1:678035612169:SNSTopic1 X	
URL for review - optional Type the URL you want to provide to the reviewer as part of the approval request. The URL must begin with 'http://' or 'https://'.	
Comments - optional Comments you type here display for the reviewer in email notifications or the console.	
Please provide your approval for UAT deployment.	
Variable namespace - optional Choose a namespace for the output variables from this action. You must choose a namespace if you want to use the variables this action produces in your configuration. Learn more 🖸	
	Cancel Done

Figure 4.44

- 25. Click **Done** on the Approval stage.
- 26. Click **Save** in the upper-right to save the pipeline.

Developer Tools > CodePipe	line > Pi	pelines > OmniDocs101W	eb-UATSta	ge > Edit OmniDocs101Web-UATStage					
Editing: OmniDocs101Web-UATStage									
Edit: Source						Edi	t stage		
Source AWS CodeCommit	١	AWS-ECR-Registry Amazon ECR	١						
				Figure 4.45					

Upon integration of the AWS ECR into the pipeline, this adds the target into the existing CloudWatch event rule that acts as a deployment trigger. Now, whenever you push the new Docker image with the same image tag name that is defined in the Amazon ECR action in the source stage, it triggers the pipeline. But as per the UAT deployment specification, UAT deployments are based on approval and are available on demand. Disable the continuous deployment of the UAT pipeline.

Perform the below steps to remove the target from the existing CloudWatch event rule created against AWS ECR action:

- 1. Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/
- 2. Select **Rules** in the **Events**. Search the rule created against the Amazon ECR with the same image tag name defined in the source stage.

aws Services	•		Q Search for services, features, r	narketplace products, and docs	[Alt+S]	[2] 🚓 omnidocs@ngdev ▼ Mumbai ▼ Support ▼
CloudWatch	×	Rules > codepip	eline-omnido-patcl	nh-186837-rule		Actions -
Dashboards	_	Summary				
▼ Alarms		ARN () arn:aws:events	ap-south-1:678035612169:rule/codepi	peline-omnido-patchh-186837-rule		
In alarm	0	Event pattern Ø			*	
Insufficient data	0	"source": [
ОК	0	-aus.ecc-				
Billing		"detail": {				
▼ Logs		"action-ty	pe": [
Log groups		1,				
Insights		"inage-tag	n: E			
 Metrics 		-patch3n	+19-alpine-openjdk-uat"			
Explorer		"repositor	y-name": [
Streams New		"omnidoc	s10.lweb"			
▼ Events		"result":	t			
Rules		"SUCCESS	-			
Event Buses		Sector Eaching				
 ServiceLens 		Status Enabled				
Service Map		Description Amazon Cloud	Natch Events rule to automatically start	your pipeline when a change occurs	in the Amazon ECR image tag. Deleting this may prevent cha	anges from being detected in that pipeline. Read more:
Traces		Monitoring Show metrics fr	arriazon.com/codepipeline/lates/userg	uiderpipelines-about-starting.ntmi		
 Container Insights 	New					
Resources		largets				
Performance monitoring	- 1	Filter:				$\ll~<$ Viewing 1 to 1 of 1 Targets $>~\gg$
 Lambda Insights 	New	Туре	Name	Input	Role	Additional parameters
Performance monitoring		CodePipeline	OmniDocs101Web-UATStage	Matched event	cwe-role-ap-south-1-OmniDocs101Web-UATStage	
 Synthetics 						
Canaries						
Contributor Insights						
Settinos						
Feedback English (US) 🔻					All rights reserved. Privacy Policy Terms of Use Cookie preferences

Figure 4.46

3. Select the rule, go to the Actions menu and select Disable/Delete.

Create rule	Actions -
Status All	Edit Delete
Stat	a Disable
0	CW_SQL-POC_START
0	CW_SQL-POC_STOP
0	ECRTrigger
0	codepipeline-Genesi-main-476342-rule
0	codepipeline-omnido-patchh-134251-rule
•	codepipeline-omnido-patchh-186837-rule
0	pipeline-execution-status
0	startec2instance
0	stopec2instance

Figure 4.47

Now, it does not trigger the pipeline whenever you push the Docker image to the AWS ECR.

NOTE:

As described in the Add source stage, the AWS CodeCommit action creates a new CloudWatch event rule, and it triggers the AWS CodePipeline whenever there is a change in the CodeCommit repository. You can disable the CloudWatch event rule once the pipeline is created.

Perform the below steps to disable the CloudWatch event rule created against AWS CodeCommit action:

- 1. Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/
- 2. Select Rules under the Events tab given on the navigation panel.
- 3. Search the rule created against the AWS CodeCommit repository Genesis-CodeCommit-Repository created in the creation of the AWS CodeCommit Repository.

aws Services	•		Q Search for services, features, r	marketplace products, and docs	[Alt+S]	∑ 🗘 omnidocs@ngdev 🔻 Mumbai ▼ Support ▼
CloudWatch	×	Rules > code	pipeline-Genesi-main	-476342-rule		Actions -
Dashboards		Summary				
 Alarms 		ARN () arn:aws	events:ap-south-1:678035612169:rule/codepi	peline-Genesi-main-476342-rule		
In alarm	0	Event pattern 0			A.	
Insufficient data	0	"sour	ce": [
ОК	0	-aw	s.codecommit"			
Billing		"deta	il-type": [
▼ Logs		"Co	deCommit Repository State Change"			
Log groups		reso	unces": [
Insights		"an	n:aws:codecommit:ap-south-1:678035612169:Gen	esis-CodeCommit-Repository"		
 Metrics 		J, "deta	il": {			
Explorer		"ev	ent": [
Streams New			referenceCreated", referenceUpdated"			
Events		1,				
Rules		"re	ferenceType": [
Event Buses		Status Fachica	or and the			
 ServiceLens 		Status Enabled				
Service Map		Description Amazon	CloudWatch Events rule to automatically start	your pipeline when a change occurs	in the AWS CodeCommit source repository and branch. Dele	ting this may prevent changes from being detected in that pipeline. Read
Traces		Monitoring Show m	etrics for the rule	aserguide pipelines-about-starting.m	um -	
 Container Insights 	New	T				
Resources		largets				
Performance		Filter:				《 Viewing 1 to 2 of 2 Targets > >>
monitoring						
Performance	New	Туре	Name	Input	Role	Additional parameters
monitoring		CodePipeline	OmniDocs101Web-DevStage	Matched event	cwe-role-ap-south-1-OmniDocs101Web-DevStage	
 Synthetics 		CodePipeline	OmniDocs101Web-UATStage	Matched event	cwe-role-ap-south-1-OmniDocs101Web-UATStage	
Canaries						
Contributor Insight	5					
Settinas	· ·					
Feedback English (U	5) 🔻				© 2008 - 2021, Amazon Internet Services Private Ltd. or its affiliates.	All rights reserved. Privacy Policy Terms of Use Cookie preferences



4. Select the rule, go to the Actions menu, and select Disable or Delete.



Figure 4.49

Now, it does not trigger the pipeline whenever you make any changes in the AWS CodeCommit repository.

NOTE:

Add an entry in the Lambda function lambda_function1 created in **Configuration of AWS CodePipeline for UAT Stage**, for each newly created pipeline and its associated SNS topic that you can use. This is required to notify the recipient(s) about the pipeline execution status whether it succeeded or failed.

5. Add an entry of the pipeline **OmniDocs101Web-UATStage** in the lambda function **lambda_function1**.

C	ode source Info	
-	File Edit Find View Go	Tools Window Test Deploy Changes deployed
Q	Go to Anything (Ctrl-P)	■ lambda_function × ⊕
onment	lambda-fuction1 - / lambda_function.py	<pre>import json import boto3 sns = boto3.client('sns')</pre>
Enviro		<pre>fpipeline_sns_map = { "OmnIDocs101Web-DevStage": "arn:aws:sns:ap-south-1:678035612169:SNSTopic1", "OmnIDocs101Web-UATStage": "arn:aws:sns:ap-south-1:678035612169:SNSTopic1", "pipeline3": "sns_arn_2" } def lambda_handler(event, context): detail = event['detail'] pipeline = detail['pipeline'] execution_id = detail['execution-id'] state = detail['state'] subject = "Pipeline "+ pipeline + " has " + state message = "Pipeline name : " + pipeline + " has " + state + " with execution id : " + execution_id # print(message) response = sns.publish(TopicArn = sns1, Message= message, Subject=subject / return(response) </pre>

Figure 4.50

6. Click Release change to trigger the pipeline manually.

aws Services ▼	Q. Search for services, features, marketplace products, and docs [Alt+S]	∑ 🚓 omnidocs @ ngdev ▼ Mumbai ▼	Support 🔻
Developer Tools CodePipeline	Developer Tools > CodePipeline > Pipelines > OmniDocs101Web-UATStage	Clone nineline Release channe	(J
Source • CodeCommit	-		
 Artifacts - CodeArtifact 			
Build • CodeBuild			
Deploy • CodeDeploy	Source () AWS-ECR-Registry ()		
▼ Pipeline CodePipeline	AWS CodeCommit Amazon ECR 2		
Getting started	7fb6f314 No executions yet		
Pipelines			0
History	7fb6f314 Source: ok		0
Settings	Disable transition		<u> </u>
Q Go to resource	O Approval Didn't Run		
🛱 Feedback	Approval-for-UAT Manual approval O Didn't Run No executions yet		
	Disable transition		-
Feedback English (US) 🔻	© 2008 - 2021, Amazon Internet Services Private Ltd. or its affiliates. All	rights reserved. Privacy Policy Terms of Use	Cookie preferences

Figure 4.51

4.5.4.4 Configuring AWS CodePipeline for production stage

The production deployment is based on approval, but it has a multi-level approval system. To deploy a production environment, you require the approval of all stakeholders. Once approval is received from all the stakeholders, the deployment to the production environment is not triggered automatically. Manual intervention mail is sent to the engineer who is supposed to deploy to production with a checklist. If all the checklist points are not covered, then the deployment to production gets rejected.

Production deployment has a multi-level approval system. To support this multi-level approval, you must create multiple SNS topics in reference to **create an SNS topic** and **create a subscription to the SNS topic** with different subscriptions.

Perform the below steps to configure the Production Stage

- 1. Open the AWS CodePipeline console at: http://console.aws.amazon.com/codesuite/codepipeline/home
- Click Create pipeline given on the Welcome tab.
 Specify the required details in the following steps. Once complete, click Create pipeline at the Review step:
- 3. Specify the following on the Select pipeline settings:
 - i. Enter a unique name for your pipeline that is, **OmniDocs101Web-ProdStage** in the **Pipeline name**.
 - ii. Select the New Service Role in the Service role.

- iii. Select the checkbox Allow AWS CodePipeline to create a service role so it can be used with this new pipeline.
- iv. Keep the other settings as default and click **Next**.

Choose pipeline settings Info	
Pipeline settings	
Pipeline name Enter the pipeline name. You cannot edit the pipeline name after it is created.	
OmniDocs101Web-ProdStage	
No more than 100 characters	
Service role	
New service role Create a service role in your account Existing service role from your account Choose an existing service role from your account	count
Role name	
AWSCodePipelineServiceRole-ap-south-1-OmniDocs101Web-ProdStage	
Type your service role name	
Allow AWS CodePipeline to create a service role so it can be used with this new pipeline	
Advanced settings	
Cancel	Next



- 4. Specify the following on the Add source stage:
 - i. Select the AWS CodeCommit in the Source provider.
 - ii. Select the existing AWS CodeCommit repository **Genesis-CodeCommit-Repository** created in the Configuration of Notification in the **Repository name**.
 - iii. Select main in the Branch name.
 - iv. In Change detection options, select the recommended option Amazon CloudWatch Events.

Amazon CloudWatch Events creates a CloudWatch event rule. As soon as you make any change in the integrated AWS CodeCommit repository, it triggers the pipeline. But you do not want to trigger the AWS CodePipeline whenever there is a change in the CodeCommit repository. The pipeline must be triggered whenever you push a new image to the container registry like AWS ECR. You can see the configuration of AWS ECR with CodePipeline in the following sections. So, you can disable the CloudWatch event rule once the pipeline is created.

v. Keep the other settings as default and click Next.

Source	
Source provider This is where you stored your input artifacts for your pipeline. Choose	the provider and then provide the connection details.
AWS CodeCommit	▼
Repository name Choose a repository that you have already created where you have pus	shed your source code.
Q Genesis-CodeCommit-Repository	×
Branch name Choose a branch of the repository	
Q main	×
Change detection options Choose a detection mode to automatically start your pipeline when a detection mode to automatically start your pipeline when a detection of the start start when a detection of the start	change occurs in the source code.
Use Amazon CloudWatch Events to automatically start my pipeline when a change occurs	Use AWS CodePipeline to check periodically for changes
Output artifact format Choose the output artifact format.	
• CodePipeline default AWS CodePipeline uses the default zip format for artifacts in the pipeline. Does not include git metadata about the repository.	 Full clone AWS CodePipeline passes metadata about the repository that allows subsequent actions to do a full git clone. Only supported for AWS CodeBuild actions.
	Cancel Previous Next



- 5. Specify the following in the Add build stage:
 - i. In the **Build provider**, select **AWS CodeBuild**.
 - ii. In the **Region**, select a region in which you want to create your pipeline.
 - iii. In the Project name, select the existing CodeBuild project **OmniDocs101Web** created in the creation of the AWS CodeBuild Project.
 - iv. In the Environment variables optional, create the below environment variables:

Name	Value	Туре
AWS_DEFAULT_REGION	ap-south-1	Plaintext
AWS_CLUSTER_NAME	Omnidocs-uat2	Plaintext
YAML_FILE	OmniDocs10.1Web.yml	Plaintext
CODE_PIPELINE_EXECUTION_ID	#{codepipeline.PipelineExecutionId}	Plaintext

- **AWS_DEFAULT_REGION**: Specify the region where the AWS EKS cluster is created.
- **AWS_CLUSTER_NAME:** Specify the name of the EKS cluster created for the PROD stage.
- **YAML_FILE**: Specify the name of the YAML file that is stored in the AWS CodeCommit repository and is used to deploy the OmniDocs10.1Web container for the PROD Stage.
- **CODE_PIPELINE_EXECUTION_ID:** This variable is just created for logging purposes so that you can track the build-id and its initiated pipeline.
- 6. For **Build type**, select the Single build and click **Next**.

Build - optional				
Build provider This is the tool of your build proje	ct. Provide build artifact details like	operating system, build spec file, a	and output file nan	nes.
AWS CodeBuild		•		
Region				
Asia Pacific (Mumbai)		•		
Project name Choose a build project that you ha and then return to this task.	ave already created in the AWS Code	Build console. Or create a build pr	oject in the AWS C	odeBuild consol
Q UmniDocs101Web		X	or Create	e project 🛛
Environment variables - optic Choose the key, value, and type fo CodePipeline. Learn more 🔀	onal or your CodeBuild environment varia	bles. In the value field, you can ref	ierence variables g	enerated by
Name	Value	Туре		
AWS_DEFAULT_REGION	ap-south-1	Plaintext v	Remove	
AWS_CLUSTER_NAME	Omnidocs-uat2	Plaintext v	Remove	
YAML_FILE	OmniDocs10.1Web.yml	Plaintext v	Remove	
CODE_PIPELINE_EXECU	#{codepipeline.Pipeline!	Plaintext v	Remove	
Add environment variat	ole			
Build type				
• Single build Triggers a single build.	O Batch b Triggers execution	uild multiple builds as a single n.		
	Car	ncel Previous	Skip build stag	je Nex

Figure 4.54

7. In the **Add deploy stage**, skip the deploy stage.

Deploy - optional			
Deploy provider Choose how you deploy to instances. Choose the provider, and then provide the provider of the	ne configuration de	etails for that provider.	
Cancel	Previous	Skip deploy stage	Next



8. In the Review, click **Create pipeline**.

Step 4: Add deploy stage	
Deploy action provider	
Deployment stage No deploy	
	Cancel Previous Create pipeline
	Figure 4.56

NOTE:

As soon as you create the pipeline, it starts its first pipeline execution. This execution failed as expected when you have not yet integrated the AWS ECR into the pipeline. You need to do the same.

- 9. Specify the following to integrate the AWS ECR into the AWS CodePipeline:
 - i. Open the created pipeline **OmniDocs101Web-ProdStage** in **Edit** mode.
 - ii. Select the Edit source stage.
 - iii. Select + Add action.

Developer Tools > CodePipeline > Pipe	lines > OmniDocs101Web-ProdSt	age > Edit OmniDocs101Web-ProdStage	
Editing: OmniDocs101	Editing: OmniDocs101Web-ProdStage		Delete Cancel Save
Edit: Source			Cancel Done
+ Add action group			
Source ③ AWS CodeCommit	+ Add action		
+ Add action group		- -	



- iv. In the Edit action panel, specify the unique Action name, that is, AWS-ECR-Registry
- v. In the Action provider, select the Amazon ECR.
- vi. In the **Repository name**, select the **omnidocs10.1web** Docker image that needs to deploy to the Prod stage.
- vii. In the **Image tag optional**, select the image tag that you want to use to set up the continuous deployment trigger.
- viii. In the **Variable namespace optional**, specify the unique namespace, that is, **AWS-ECR**. This is required to use its output variable in the following sections.
- ix. In the **Output artifacts**, specify the unique variable name, that is, **SourceArtifact1**. SourceArtifact is already used by AWS CodeCommit action.
- x. Click Done.

Edit action
Action name Choose a name for your action
AWS-ECR-Registry
No more than 100 characters
Action provider
Amazon ECR
Repository name Choose an Amazon ECR repository as the source location.
Q omnidocs10.1web X C
Image tag - optional Choose the Image tag that triggers your pipeline when a change occurs in the image repository.
Q patch3hf19-alpine-openidk-prod X
If an image tag is not selected, defaults to latest
Variable namespace - optional Choose a namespace for the output variables from this action. You must choose a namespace if you want to use the variables this action produces in your configuration. Learn more [
AWS-ECR
Output artifacts Choose a name for the output of this action.
SourceArtifact1
No more than 100 characters


- xi. Click **Done** on the Edit source stage.
- xii. Select the Edit build stage.
- xiii. Click the **Edit** icon for AWS CodeBuild action.

Edit: Build			Cancel	Delete Do
+ Add action group	р			
Build AWS CodeBuild	(j) (j)	+ Add action		
+ Add action group	p			



10. Add the three new environment variables as given in the table below:

Name	Value	Туре
IMAGE_REGISTRY_ID	#{AWS-ECR.RegistryId}	Plaintext
IMAGE_REPOSITORY_NAME	#{AWS-ECR.RepositoryName}	Plaintext
IMAGE_TAG	#{AWS-ECR.ImageTag}	Plaintext

Where, **AWS-ECR** is the name of the variable namespace, created in Amazon ACR action.

ame	Value	Туре	
AWS_DEFAULT_REGION	ap-south-1	Plaintext	▼ Remove
AWS_CLUSTER_NAME	Omnidocs-uat2	Plaintext	▼ Remove
YAML_FILE	OmniDocs10.1Web.yml	Plaintext	▼ Remove
CODE_PIPELINE_EXECUTION_ID	#{codepipeline.PipelineExecutionId}	Plaintext	▼ Remove
IMAGE_REGISTRY_ID	#{AWS-ECR.RegistryId}	Plaintext	▼ Remove
IMAGE_REPOSITORY_NAME	#{AWS-ECR.RepositoryName}	Plaintext	▼ Remove
IMAGE_TAG	#{AWS-ECR.ImageTag}	Plaintext	▼ Remove

Figure 4.60

- 11. On the Edit Action Panel, click **Done**.
- 12. On the Edit build stage, click Done. Since this is the PROD stage and it must be an **approvalbased** pipeline then select the **+ Add stage**.

aws Services ▼	Q. Search for services, features, marketplace products, and docs [Alt+S]	∑ 🔷 omnidocs⊛ngdev 🔻 Mumbai 🔻 Support 🔻
Developer Tools X CodePipeline	Developer Tools > CodePipeline > Pipelines > OmniDocs101Web-ProdStage > Edit OmniDocs101Web-ProdStage Editing: OmniDocs101Web-ProdStage	Delete Cancel Save
Artifacts • CodeArtifact	Edit: Source	Edit stage
 Build • CodeBuild Deploy • CodeDeploy Pipeline • CodePipeline 	Source O AWS-ECR-Registry O AWS CodeCommit Amazon ECR	
Getting started Pipelines Pipeline	+ Add stage	
History Settings	Edit: Build	Edit stage
Settings Q Go to resource	Build O NWS CodeBuild	
🛱 Feedback	+ Add stage	
Feedback English (US) 🔻	© 2008 - 2021, Amazon Internet Services Priv	ate Lid, or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

Figure 4.61

- 13. Specify the Stage name such as Approval.
 - i. Click +Add action group under the Approval stage.

Edit: Approval	Cancel	Delete Done
+ Add action group		



- ii. Specify the unique action name such as **Approval-for-PROD** in the **Action name**.
- iii. Select the Manual approval in the Action provider.
- iv. Select the ARN of SNSTopic1 created in Configuration of Notification in the **SNS topic ARN – optional**.
- v. Specify a comment to display for the reviewer in the email notifications or the console in Comments-optional. For Example, provide your approval for **PROD** deployment.
- vi. Keep the other settings as default and click **Done**.

Edit action			×
Action name Choose a name for your action			
Approval-for-PROD			
No more than 100 characters			
Action provider			
Manual approval	•		
Configure the approval request.			
SNS topic ARN - optional			
Q arn:aws:sns:ap-south-1:678035612169:SNSTopic1	\times		
URL for review - optional Type the URL you want to provide to the reviewer as part of the approval request. The URL must begin with 'http://` or 'https://`.			
Comments - optional Comments you type here display for the reviewer in email notifications or the console.			
Please provide your approval for PROD deployment.			
Variable namespace - optional Choose a namespace for the output variables from this action. You must choose a namespace if you want to use the variables this action produces in your configuration. Learn more 🗹			
		Cancel	Done

Figure 4.63

- 14. The pipeline has a multi-level approval system, you need to add multiple Approval actions with different SNS topics.
- 15. Click + Add action under the Approval stage and specify the following:

Edit: Approval		Cancel	Delete Done
+ Add action group			
Approval-for-PROD (3) Manual approval	+ Add action		
☐ × → Add action group			

Figure 4.64

- i. Specify the unique action name such as **Approval-for-PROD-2** in the Action name.
- ii. Select Manual approval in the Action provider.
- iii. In the SNS topic ARN optional, select the ARN of SNSTopic2 created in the configuration of notification.
- iv. For **Comments optional**, specify the comment to display for the reviewer in email notifications or the console, for example, provide your approval for PROD deployment.
- v. Keep the other settings as default and click **Done**.

Edit action	×
Action name Choose a name for your action	
Approval-for-PROD-2	
No more than 100 characters	
Action provider	
Manual approval	
Configure the approval request.	
SNS topic ARN - optional	
Q am:aws:sns:ap-south-1:678035612169:SNSTopic2	
URL for review - optional Type the URL you want to provide to the reviewer as part of the approval request. The URL must begin with 'http:// or 'https://'.	
Comments - optional Comments you type here display for the reviewer in email notifications or the console.	
Please provide your approval for PROD deployment.	
Variable namespace - optional Choose a namespace for the output variables from this action. You must choose a namespace if you want to use the variables this action produces in your configuration. Learn more 🛛]
	Cancel Done

Figure 4.65

- vi. Click Done.
- vii. Click **Save** to save the pipeline.

	50510	Delete Cancel Sa		
dit: Source				Edit stage
Source AWS CodeCommit	٤	AWS-ECR-Registry Amazon ECR	١	
				+ Add stage
				Edit stage
dit: Approval				

Figure 4.66

NOTE:

As soon as you integrate the AWS ECR into the pipeline, it adds to the existing CloudWatch event rule that acts as a deployment trigger. Now, whenever you push the new Docker image with the same image tag name that is defined in the Amazon ECR action in the source stage, it triggers the pipeline. But as per the PROD deployment specification, PROD deployments are approval based and are available on-demand. So, you need to disable the continuous deployment for the PROD pipeline.

Perform the below steps to Remove the target from the existing CloudWatch event rule created against AWS ECR action:

- 1. Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.
- 2. Go to the Rules given under Events in the navigation pane.
- 3. Search the rule created against the Amazon ECR with the same image tag name that is defined in the source stage.

CloudWatch × Rules > C	odepipeline-omnido-pa	tchh-208522-rule		Actions -
Dashboards Summary				
▼ Alarms ARN 0	arn:aws:events:ap-south-1:678035612169:rule/co	depipeline-omnido-patchh-208522-rule		
In alarm O Event pattern 🖲	4			
Insufficient data	"source": [
ок 🔘	"aws.ecr"			
Billing	"detail": {			
▼ Logs	"action-type": ["PUSH"			
Log groups	b			
Insights	"image-tag": ["patch3bf19-alpine-openidk-prod"			
 Metrics 	b			
Explorer	"repository-name": [
Streams New],			
▼ Events	"result": [
Rules]		*	
Event Buses Status	Enabled			
Service Map Description	Amazon CloudWatch Events rule to automatically	start your pipeline when a change occur	s in the Amazon ECR image tag. Deleting this may prevent cha	inges from being detected in that pipeline. Read more:
Traces	http://docs.aws.amazon.com/codepipeline/latest/u	userguide/pipelines-about-starting.html		
Container Insights New	Show metrics for the rule			
Resources Targets				
Performance				
monitoring Filter:				« < Viewing 1 to 1 of 1 Targets > »
Lambda Insights New Type	Name	Input	Role	Additional parameters
monitoring CodePipeline	OmniDocs101Web-ProdSta	ge Matched event	cwe-role-ap-south-1-OmniDocs101Web-ProdStage	
Synthetics				
Canaries				
Contributor Insights				
Settinas 👻				
Feedback English (US) ▼			© 2008 - 2021, Amazon Internet Services Private Ltd. or its affiliates. A	All rights reserved. Privacy Policy Terms of Use Cookie preferences

Figure 4.67

4. Select the rule, go to the **Actions** menu, and select **Disable** or **Delete**.

Rules	
Rules route e	events from your AWS resources for processing by selected targets. You can create, edit, and delete rule
Create rul	Ie Actions - Edit
	Statt Disable
0	codepipeline-Genesi-main-476342-rule
\bigcirc	codepipeline-omnido-patchh-134251-rule
\bigcirc	codepipeline-omnido-patchh-186837-rule
۲	codepipeline-omnido-patchh-208522-rule
0	pipeline-execution-status
0	startec2instance
0	stopec2instance



Now, it does not trigger the pipeline whenever you push the Docker image to the AWS ECR.

NOTE:

As discussed in the **Add source stage**, the AWS CodeCommit action creates a new CloudWatch event rule, and it triggers the AWS CodePipeline whenever there is a change in the CodeCommit repository. You can disable the CloudWatch event rule once the pipeline is created.

Perform the below steps to disable the CloudWatch event rule created against AWS CodeCommit action:

- 1. Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.
- 2. Go to the Rules under Events in the navigation pane.
- Search the rule created against the AWS CodeCommit repository Genesis-CodeCommit-Repository.

aws Servie	es 🔻		Q Search for services, fea	tures, marketplace products, an	d docs [Alt+S]			🔻 Mumbai 🔻 Support 🔻
CloudWatch	×	Rules > c	odepipeline-Genesi-m	nain-476342-rule	9			Actions -
Dashboards		Summary						
▼ Alarms		ARN ()	arn:aws:events:ap-south-1:678035612169:rule	codepipeline-Genesi-main-47634	2-rule			
In alarm	0	Event pattern 6						
Insufficient da	ta 🧿		"source": [Î		
ОК	Ō		"aws.codecommit"					
Billing	-		"detail-type": [
▼ Logs			"CodeCommit Repository State Change"					
Log groups			"resources": [
Insights			"arn:aws:codecommit:ap-south-1:678035612	2169:Genesis-CodeCommit-Repositor	r -			
 Metrics], "detail": {					
Explorer			"event": [
Streams Ne			"referenceUpdated",					
Events			Ъ					
Rules			"referenceType": ["branch"			*		
Event Buses		Status	Enabled					
▼ ServiceLens		Description	Amazon CloudMatch Events rule to automatics	ally start your ninalina when a char	the accurs in the AWS CodeCommit source r	appeiton, and branch. Deleting this	may prevent changes from bein	a datacted in that nineline. Read
Service Map		Description	more: http://docs.aws.amazon.com/codepipelir	ne/latest/userguide/pipelines-about	-starting.html	spository and branch. Deleting this	may prevent changes nom bei	g detected in that pipeline. Read
Traces		Monitoring	Show metrics for the rule					
Container insigni	s New	Targets						
Performance		· · · · · · · · · · · · · · · · · · ·						
monitoring		Filter:					« < View	ing 1 to 3 of 3 Targets > >>
 Lambda Insights 	New	Туре	Name	Input	Role	Add	ditional parameters	
Performance monitoring		CodePipeline	OmniDocs101Web-DevS	tage Matched event	cwe-role-ap-south-1-OmniDo	cs101Web-DevStage		
 Synthetics 		CodePipeline	OmniDocs101Web-ProdS	Stage Matched event	cwe-role-ap-south-1-OmniDo	cs101Web-ProdStage		
Canaries		CodePipeline	OmniDocs101Web-UATS	tage Matched event	cwe-role-ap-south-1-OmniDo	cs101Web-UATStage		
Contributor Insig	nts			•		a-		
Settinas Feedback Epolish) IS) -				@ 2000 - 2021 Amazon Internet Cen-	or Drivate Ltd. or its affiliator. All violate	manual Drivery Dolicy	Torms of Lisa Caakia assistances
reeuback English	()) •				© 2008 - 2021, Amazon Internet Servio	tes Private Lto, or its amilates. All rights	reserved. Privacy Policy	terms of ose Cookie preferences



4. Select the rule, go to the Actions menu, and select Disable or Delete.

Rules	
Rules route eve	nts from your AWS resources for processing by selected targets. You can create, edit, and delete rules
Create rule	Actions -
	Edit
Status All	Delete
St	tatu Disable
\bigcirc	CW_SQL-POC_START
\bigcirc	CW_SQL-POC_STOP
\bigcirc	ECRTrigger
	codepipeline-Genesi-main-476342-rule
\bigcirc	codepipeline-omnido-patchh-134251-rule
\bigcirc	startec2instance
\bigcirc	stopec2instance

Figure 4.70

Now, it does not trigger the pipeline whenever you make any change in the AWS CodeCommit repository.

NOTE:

As per the production deployment specification, even after taking approvals from all stakeholders, the deployment to the production environment is not triggered automatically. A manual intervention mail is sent to the engineer who is supposed to deploy to production with a checklist.

Configuration of Manual Intervention Mail:

To configure the manual intervention mail, follow the below steps:

- Create an SES identity
- Create a Lambda function
- Add a stage to the PROD pipeline

Create an SES identity:

In AWS SES, an identity is an email address or domain that you use to send an email. Before sending/receiving an email using AWS SES, you must verify each identity that you are going to use as a sender or recipient. In other words, you can say that a verified identity is an email address or domain that you have proven that you own.

To create a verified SES identity, follow the below steps:

- 1. Sign in to the Amazon SES console <u>https://console.aws.amazon.com/ses/home</u>
- 2. In the console, use the region selector to select the **AWS Region** where want to verify the email address.
- 3. Select the Verified identities under Configuration in the navigation pane.
- 4. Select the Create identity.
- 5. On the Create identity page, select Identity type as Email Address.
- 6. Specify the email address that you want to use as sender or recipient.
- 7. Click **Create identity.**

reate identity	
verified identity is a domain, subdomain, or email address you the domain level extends to all email addresses under one ve	use to send email through Amazon SES. Identity verification rified domain identity.
Identity details Info	
Identity type	
 Domain To verify ownership of a domain, you must have access to its DNS settings to add the necessary records. 	• Email address To verify ownership of an email address, you must have access to its inbox to open the verification email.
Email address	
vivek_kumar@newgen.co.in	
Assign a default configuration set Enabling this option ensures that the assigned configuration set is configuration set isn't specified at the time of sending.	applied to messages sent from this identity by default whenever a
Tags - <i>optional</i> Info You can add one or more tags to help manage and organize your resou	irces, including identities.
No tags associated with the resource.	
Add new tag	
You can add 50 more tags.	
	Cancel Create identity
	concer electre lacinety

Figure 4.71

- Select the inbox for the email address for verification and receives a message with the following subject line: Amazon Web Services - Email Address Verification Request in region RegionName, where RegionName is the name of the AWS Region.
- 9. Click the **link** in the message. A confirmation message appears **You have successfully verified an email address**.

Create a Lambda function

- Open the function page on the lambda console: <u>https://console.aws.amazon.com/lambda/home</u>
- 2. Select Create function.
- 3. In the Function name, specify the unique function name such as lambda-fuction-ses.
- 4. In the **Runtime**, select **python 3.8** or the latest version. Keep the other settings as default.
- 5. Select Create function.

aws	Services V	Q Search for services, features, marketplace products, and docs [Alt+S]	∑ 🗘 omnidocs @ ngdev ▼ Mum	bai 🔻 Support 🔻
=	Create function Info Choose one of the following options to create your func	ion.		^ (3)
	Author from scratch • Start with a simple Hello World example.	Use a blueprint O Build a Lambda application from sample code and configuration presets for common use cases. Select a container image to deploy for your function.	Browse serverless app repository Deploy a sample Lambda application from the Serverless Application Repository.	AWS
	Basic information			
	Function name Enter a name that describes the purpose of your function.			
	lambda-fuction-ses			
	Use only letters, numbers, hyphens, or underscores with no spaces. Runtime Info Choose the language to use to write your function. Note that the o	unsole code editor supports only Node.js, Python, and Ruby.		
	Python 3.8	▼		
	Permissions Info By default, Lambda will create an execution role with permissions I	o upload logs to Amazon CloudWatch Logs. You can customize this default role later when adding triggers.		
	Change default execution role			
	► Advanced settings			
			Cancel Create fur	ction
Feedback	English (US) 🔻	© 2008 - 2021, Amazon Internet Services Private Ltd. or its affilia	iates. All rights reserved. Privacy Policy Terms of U:	e Cookie preferences

Figure 4.72

- 6. Under the Code tab, select lambda_function.py.
- 7. Replace the default code snippet using the below code snippet, and select **Deploy**:

```
import boto3
from botocore.exceptions import ClientError
# Create a new Codepipeline event to set the Job status
code pipeline = boto3.client('codepipeline')
def lambda handler(event, context):
    #Getting JobID of the pipeline
    JobId=event['CodePipeline.job']['id']
execution id=event['CodePipeline.job']['data']['actionConfiguration']['configu
ration']['UserParameters']
    # Update vivek kumar@abc.co.in with your "From" address. This address must
be verified with Amazon SES.
    SENDER = "DevOps Admin <vivek kumar@abc.co.in>"
    #Update ToRecipients, CcRecipients, and BccRecipients addresses. If your
account is still in the sandbox, this address must be verified.
    ToRecipients = ["vivek kumar@abc.co.in","vivekkumarpandey185@gmail.com"]
    #CcRecipients = ["vivek kumar@abc.co.in"]
    #BccRecipients = ["vivek kumar@abc.co.in"]
    # If necessary, replace ap-south-1 with the AWS Region you're using for
Amazon SES.
   AWS REGION = "ap-south-1"
    # The subject line for the email.
```

```
SUBJECT = "Manual Intervention Mail for the "+execution_id
    # The HTML body of the email.
    BODY HTML = """<html>
<head></head>
<body>
color:#1F4E79">Dear Recipient,
Before deploying to the production, make sure that the below checklist points
are completed:
   • All the Major and Catastrophic bugs must be fixed.
   • The latest images must be thoroughly tested on the Dev and UAT stages.
   • Approval has been taken from all stakeholders.
   • Deployment downtime has been taken from the client.
Regards:</BR>DevOps Admin
</body>
</html>"""
    # The character encoding for the email.
   CHARSET = "UTF-8"
    # Create a new SES resource and specify a region.
    client = boto3.client('ses', region name=AWS REGION)
    # Try to send the email.
    try:
        #Provide the contents of the email.
        response = client.send email(
            Destination={'ToAddresses':ToRecipients},
#Destination={'ToAddresses':ToRecipients,'CcAddresses':CcRecipients},
#Destination={'ToAddresses':ToRecipients,'CcAddresses':CcRecipients,'BccAddres
ses':BccRecipients},
       Message={
            'Body': {
                'Html': {
                    'Charset': CHARSET,
                    'Data': BODY HTML,
                },
            },
            'Subject': {
                'Charset': CHARSET,
                'Data': SUBJECT,
            },
        },
        Source=SENDER,
    )
    # Display an error if something goes wrong.
    except ClientError as e:
       print("Email is not sent!"),
       print(e.response['Error']['Message']),
       put job failure (JobId, 'Unable to send mail')
    else:
       print("Email sent! with below Message ID:"),
```

print(response['MessageId']),

```
put_job_success(JobId, "Mail sent successfully")
def put_job_success(job, message):
    print('Putting job success')
    print(message)
    code_pipeline.put_job_success_result(jobId=job)
def put_job_failure(job, message):
    print('Putting job failure')
    print(message)
    code_pipeline.put_job_failure_result(jobId=job, failureDetails={'message':
    message, 'type': 'JobFailed'})
```

- 8. In the above lambda function, you can update the following:
 - **SENDER:** You can update your email ID in the From address. This address must be verified with Amazon SES.
 - **ToRecipients:** Specify the multiple ToAddresses separated by a comma (,). These email addresses must be verified.
 - **CcRecipients** and **BccRecipients**: By-default, CcRecipients and BccRecipients are commented. If you want to use these then you just uncomment at two places.
 - **AWS_REGION:** If required, replace ap-south-1 with the AWS Region you are using for Amazon SES.
 - **SUBJECT**: Update the email subject line as per your requirement.
 - **BODY_HTML**: Update the mail body in HTML form.



Figure 4.73

9. Go to the **Permissions** under the **Configuration** tab.

10. Select the created IAM role for this lambda function. The IAM role Summary screen appears.

Code Test Monitor	Configuration Aliases Versions
General configuration Triggers	Execution role
Permissions	Role name lambda-fuction-ses-role-lked3e2w 🛂
Destinations	

Figure 4.74

- 11. Select Add inline policy.
- 12. Select SES in the **Service**.
- 13. Select SendEmail and SendRawEmail in the Actions.
- 14. Select **All resources** in the **Resources**.
- 15. Click Review policy.

aws	Services	▼ Q	Search for services, features, marketplace products, and docs	[Alt+S]	🗢 omnidocs @ ngdev 🔻	Global 🔻	Support 🔻
		✓ SES (2 actions)			Clone Remove		*
		Service	SES				
		➤ Actions	Write SendEmail SendRawEmail				- 1
Documentation		✓ Resources close	Specific All resources As a best practice, define permissions for only specific resourcing condition keys. Learn more	rrces in specific accounts. Alternatively, you ca	n grant least privilege		
		Request conditions	Specify request conditions (optional)				
				0 A	dd additional permissions	Ţ	
		Character count: 140 of 10,240. The current character count includes character	for all inline policies in the role: lambda-fuction-ses-role-lked3e2w.		Cancel Review policy		
Feedback	k English (US	;) ▼	© 2008 - 2021, Amaz	zon Internet Services Private Ltd. or its affiliates. All rights re	served. Privacy Policy Term	s of Use Co	ookie preferences

Figure 4.75

16. Specify the policy name such as **ses-lambda-policy**.

17. Select the **Create policy**. Add another inline policy by selecting **Add inline policy**.

- 18. Select the CodePipeline in the Service.
- 19. Select the PutJobSuccessResult and PutJobFailureResult in the Actions.
 - By default, the above actions support all resources so there is no need to select.
- 20. Select Review policy.

aws	Services	▼ Q.S	earch for services, features, marketplace prod	lucts, and docs [Alt+S]	\$° omnidocs@ngdev ▼	Global 🔻	Support 🔻
		← CodePipeline (2 actions)			Clone Remove		•
		▶ Service	CodePipeline				
		✓ Actions	Specify the actions allowed in CodePipelin	ne 🕐	Switch to deny permissions		
		0050	Q Put				- 1
			PutActionRevision ③ PutApprovalResult ③	 PutJobFallureResult ② PutJobSuccessResult ③ 	PutThirdPartyJobSuccessResult (2)		
ation			PutWebhook ③	PutThirdPartyJobFailureResult ③			
Document		Resources	The actions you chose support all resources.				
		Request conditions	Specify request conditions (optional)				
					• Add additional permissions		
		Champeler count: 215 of 10 240			Consultant Providence Consultant		
		Character count: 315 of 10,240. The current character count includes character f	or all Inline policies in the role: lambda-fuction-	ses-rale-lked3e2w.	Cancel Review policy		
Feedback	English (US) 🔻			t. or its affiliates. All rights reserved. Privacy Policy Term		ookie preferences

Figure 4.76

21. Specify the policy name such as codepipelie-lambda-policy and select Create policy.

Add a stage to the PROD pipeline:

NOTE:

As per the production deployment specification, even after taking approvals from all stakeholders, the deployment to the production environment is not triggered automatically. A manual intervention mail is sent to the engineer who is supposed to deploy to production with a checklist. If all the checklist points are covered or not, then the deployment to the production gets rejected. To handle this use case, you need to add a stage just after the Approval stage and add 2 actions: Firstly, execute the Lambda function, and secondly, Manual approval action without the SNS topic.

To add a stage to the PROD pipeline, follow the below steps:

- 1. Open the created pipeline **OmniDocs101Web-ProdStage** in **Edit** mode.
- 2. Select + Add stage and specify the stage name such as Manual-Intervention.

aws Services ▼	Q Search for services, features, marketplace products, and docs [Alt+S]	∑ 🙏 omnidocs@ngdev ▼ Mumbai ▼ Support ▼
Developer Tools X CodePipeline	Editing: OmniDocs101Web-ProdStage	Delete Cancel Save
▶ Source + CodeCommit	Edit: Source	Edit stage
 Artifacts • CodeArtifact Build • CodeBuild Deploy • CodeDeploy 	Source AWS-ECR-Registry AWS-CodeCommit Amazon ECR IS	
▼ Pipeline * CodePipeline Getting started	+ Add stage	
Pipelines Pipeline History	Edit: Approval	Edit stage
Settings	Approval-for-PROD Approval-for-PROD-2 Manual approval	
Q Go to resource ☐ Feedback	+ Add stage	
	Edit: Build	Edit stage
	Build (C) AWS CodeBuild	
Feedback English (US) 🔻	© 2008 - 2021, Amazon Internet Services Private Ltd. or its affilia	ates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

Figure 4.77

- 3. Specify the following in the Manual-Intervention stage:
 - i. Click +Add action group under the Manual-Intervention stage.

Edit: Manual-Intervention	Cancel	Delete Done
+ Add action group		



- ii. Specify the action name such as **Execute-Lambda** in the **Action** name.
- iii. Select AWS Lambda in the Action provider.
- iv. Select the AWS region where the Lambda function is created in the Region.
- v. Select lambda function lambda-function-ses in the Function name.
- vi. In the User parameters optional, specify the parameters as given below:
 - Pipeline "<Name of the Pipeline>" with Execution Id

"#{codepipeline.PipelineExecutionId}"

For example,

Pipeline "OmniDocs101Web-ProdStage" with Execution Id

"#{codepipeline.PipelineExecutionId}"

- vii. Keep the other settings as default.
- viii. Click Done.

aws	Edit action	×	Support 💌 📩
Develop CodeF	Action name Choose a name for your action		î (
	Execute-Lambda		
▶ Sour	No more than 100 characters		
	Action provider		
▶ Artif	AWS Lambda v		
▶ Build	Region		
▶ Depl	Asia Pacific (Mumbai)		
♥ Pipe Getti	Input artifacts Chose an input artifact for this action. Learn more 🕜		
Pipel	¥		
P H	Add No more than 100 characters		
S	Function name Choose a function that you have already created in the AWS Lambda console. Or create a function in the AWS Lambda console and then return to this task.		
▶ Setti	Q lambda-fuction-ses X C		
	Function name contains only letters, numbers, hyphens, or underscores with no spaces. This does not include the function alias or function ARN.		
Q Go t	User parameters - optional This string will be used in the event data parameter passed to the handler in AWS Lambda.		
La reed	Pipeline "OmniDocs101Web-ProdStage" with Execution Id "#(codepipeline.PipelineExecutionId)"		
	Variable namespace - optional Choose a namespace for the output variables from this action. You must choose a namespace if you want to use the variables this action produces in your configuration. Learn more 🗹		
	Output artifacts Choose a name for the output of this action.		- 1
	Add		
	No more tran i Lux characters		
Feedback	Cancel De	one	okie preferences

Figure 4.79

4. To add another action for the Manual approval, click **+Add action** under the **Manual**-**Intervention** stage and specify the following:

Edit: Manual-Intervention	Cancel	Delete Done
+ Add action group		
Execute-Lambda (i) AWS Lambda (i) E X		
+ Add action group		



- i. Specify the Action name that is, as Manual-Approval.
- ii. Select Manual approval in the Action provider.
- iii. For **Comments optional**, specify the comment to display for the reviewer in email notifications or the console.

Ensure that all the checklist points shared over the mail are completed for the Pipeline " OmniDocs101Web-ProdStage " with Execution Id "#{codepipeline.PipelineExecutionId}". Where **OmniDocs101Web-ProdStage** is the name of the pipeline.

iv. Keep the other settings as default and click **Done.**

Edit action	×
Action name Choose a name for your action	
Manual-Approval	
No more than 100 characters	
Action provider	
Manual approval	
Configure the approval request.	
SNS topic ARN - optional	
٩	
URL for review - optional Type the URL you want to provide to the reviewer as part of the approval request. The URL must begin with "http://" or "https://".	
Comments - optional Comments you type here display for the reviewer in email notifications or the console.	
Please make sure all the checklist points shared over the mail are completed for the Pipeline "OmniDocs101Web-ProdStage" with Execution Id "#(codepipeline.PipelineE)	
Variable namespace - optional Choose a namespace for the output variables from this action. You must choose a namespace if you want to use the variables this action produces in your configuration. Learn more 🗹	
	Cancel Done

Figure 4.81

- v. Click Done on the Manual-Intervention stage.
- vi. Click **Save** in the upper-right to save the pipeline.

aws Services ▼	Q Search for services, features, marketplace products, and docs [Alt+5]	∑	lev 🔻 Mumbai 🔻	Support 🔻
Developer Tools X CodePipeline	Editing: OmniDocs101Web-ProdStage	Delete	ancel Save	•
 Source • CodeCommit Artifacts = CodeArtifact 	Edit: Source		Edit stage	
 Build • CodeBuild Deploy • CodeDeploy 	Source O AWS-ECR-Registry O AWS CodeCommit Amazon ECR &			
 Pipeline • CodePipeline Getting started Pipelines 	+ Add stage			
Pipeline History	Edit: Approval		Edit stage	
Settings	Approval-for-PROD () Approval-for-PROD-2 () Manual approval Manual approval			
Q Go to resource	+ Add stage			-
	Edit: Manual-Intervention		Edit stage	
	Execute-Lambda () Manual-Approval () AWS Lambda Manual approval			-
eedback English (US) 🔻	© 2008 - 2021, Amazon Internet Services Private Ltd. or its affiliat	es. All rights reserved. Privacy Policy	Terms of Use	Cookie preferenc

Figure 4.82

NOTE:

To add an entry in the Lambda function lambda_function1 created in Creation of AWS CodeCommit Repository for each newly created pipeline and its associated SNS topic that you can use. This is required to notify the recipient(s) about the pipeline execution status whether it is succeeded or failed.

5. Add an entry of the pipeline 'OmniDocs101Web-ProdStage' in the lambda function lambda_function1.

Code source Info							
-	▲ File Edit Find View Go Tools Window Test 💌 Deploy Changes deployed						
Q	Go to Anything (Ctrl-P)	■ lambda_function× ⊕					
Environment	Iambda-fuction1 - / Iambda_function.py	<pre>import json import boto3 sns = boto3.client('sns') pipeline_sns_map = { "OmniDocs101Web-DevStage": "arn:aws:sns:ap-south-1:678035612169:SNSTopic1", "OmniDocs101Web-UATStage": "arn:aws:sns:ap-south-1:678035612169:SNSTopic1", "moniDocs101Web-VarStage": "arn:aws:sns:ap-south-1:678035612169:SNSTopic1", "pipeline3": "sns_arn_2" } def lambda_handler(event, context): detai1 = event['detai1'] pipeline = detai1['pipeline'] execution_id = detai1['tate1] state = detai1['tate1] sns1 = pipeline = map[ipeline] subject = "pipeline = " + pipeline + " has " + state message = "Pipeline name: " + pipeline + " has " + state + " with execution id : " + execution_id # print(message) response = sns.publish(TopicArn = sns1, Message=message, Subject=subject } return(response)</pre>					

Figure 4.83

6. Trigger the pipeline manually by clicking on **Release change**.

aws Services ▼	Q Search for services, features, marketplace products, and docs	[Alt+S]	ג לי omnidocs @ ngdev א Mumbai א Mumbai א	Support 🔻
Developer Tools X CodePipeline	Developer Tools > CodePipeline > Pipelines > OmniDocs101Web-ProdStage OmniDocs101Web-ProdStage	▲ Notify ▼ Edit Stop execution	Clone pipeline Release change	()
 Source • CodeCommit Artifacts • CodeArtifact Build • CodeBuild 	Source Succeeded Pipeline execution ID: bd39119c-4a2b-4d79-99a6-52c774675c13			
Deploy • CodeDeploy Pipeline • CodePipeline Getting started Pipelines	Source ③ AWS-ECR-Registry ④ AWS CodeCommit Amazon ECR 년 ④ Succeeded - 19 hours ago ④ Didn't Run AND executions yet			0
Pipeline History Settings ▶ Settings	7tb6f314 Source: ok Disable transition			0
Q Go to resource ⊟ Feedback	Approval Didn't Run Approval-for-PROD Approval-for-PROD-2 Manual approval Didn't Run Moneutations yet No executions yet			
Feedback English (US) 🔻	Disable transition	© 2008 - 2021, Amazon Internet Services Private Ltd, or its affiliates. All rig	phis reserved. Privacy Policy Terms of Use	Cookie preferences

Figure 4.84

Appendix

This guide contains third-party product information about configuring Amazon Web Services (AWS) CodePipeline for Container Deployment on EKS and AWS Kubernetes Cluster. Newgen Software Technologies Ltd does not claim any ownership on such third-party content. This information is shared in this guide only for convenience of our users and could be an excerpt from the AWS documentation. For latest information on configuring the AWS Kubernetes Cluster and AWS CodePipeline refer to the AWS documentation.