

Deploy and Manage Business Process Automation Applications on Amazon EKS: a Practical Guide

Contents

Abstract

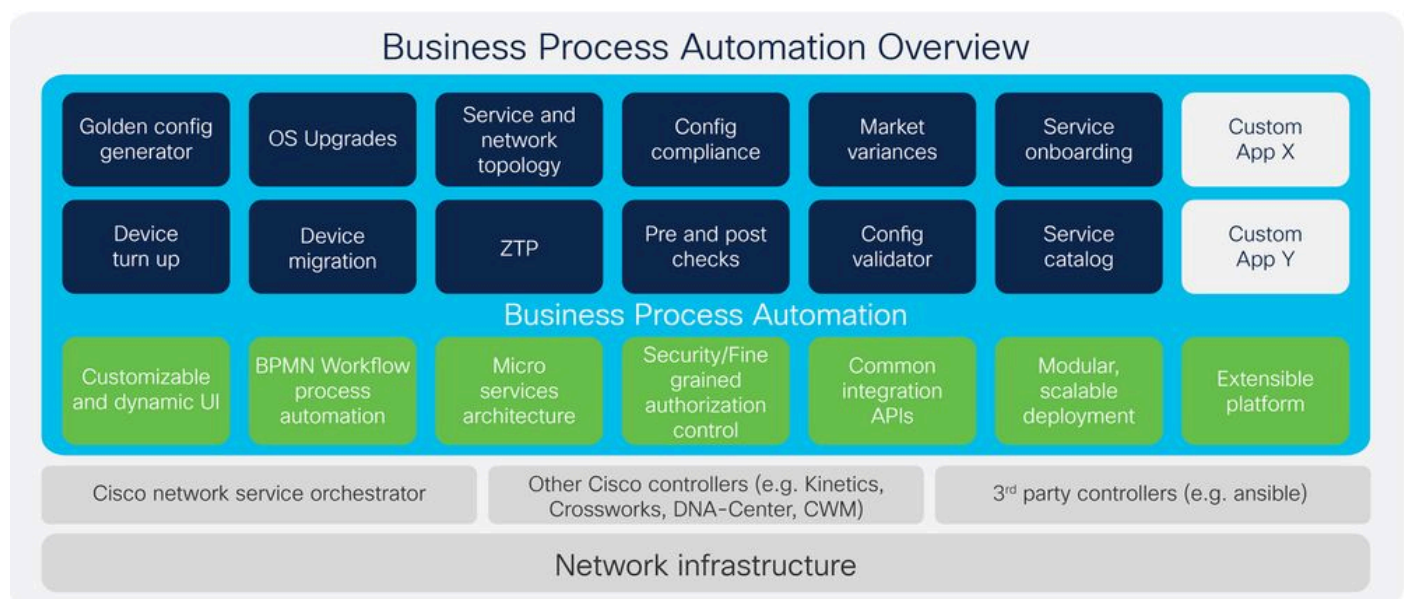
This paper presents a comprehensive guide on deploying and managing Business Process Automation (BPA) applications using Amazon Elastic Kubernetes Service (EKS). It outlines the prerequisites, highlights the benefits of utilizing EKS, and provides step-by-step instructions for setting up an EKS cluster, Amazon RDS database, and MongoDB Atlas. Additionally, the paper delves into the deployment architecture and specifies the environment requirements, offering a thorough resource for organizations aiming to leverage EKS for their containerized BPA applications.

Keywords

Amazon EKS, Kubernetes, AWS, RDS, MongoDB Atlas, DevOps, Cloud Computing, Business Process Automation.

Introduction

BPA



In today's digital era, enterprises seek to streamline and automate complex business processes across a diverse range of IT environments. Business Process Automation (BPA) has emerged as a pivotal technology, enabling organizations to enhance operational efficiency, reduce errors, and improve service delivery. BPA introduces several key innovations and enhancements aimed at advancing workflow

automation, service provisioning, and off-the-shelf automation applications.

The BPA platform hosts business and IT/operational use cases and applications, such as OS upgrades, service provisioning, and integration to orchestration engines. Customers have access to a lifecycle of services and BPA capabilities including advisory, implementation, business critical services, and solution support delivered through Cisco experts, best practices, and proven techniques and methodologies that help automate their business processes and de-risk their systems.

These lifecycle capabilities can be subscription-based or customized to individual needs. Implementation services help define, integrate, and deploy tools and processes to accelerate automation. Cisco experts conduct a formal process for gathering requirements, designs and develops user stories based on agile processes and Continuous Integration and Continuous Delivery (CICD) tools, and implements flexible services with automated testing of new or existing workflows, devices, and services. With Solution Support, customers get access to 24/7, centralized support with a focus on software-centric issues coupled with multivendor and open-source support offered through Cisco's tiered software model. Cisco solution support experts help manage your case from first call to final resolution and act as the main point of contact working with multiple vendors simultaneously. You could experience up to 44 percent fewer issues working with solution-level experts, helping you maintain business continuity and get faster return on your BPA investment.

Key technical features, such as support for FMC and Ansible-managed devices, parallel executions using the Advanced Queuing Framework (AQF), and expanded configuration compliance for NDFC and FMC devices, position BPA as a comprehensive solution for large-scale enterprise automation. With added capabilities in SD-WAN management, device onboarding, and firewall policy governance, the release addresses critical aspects of network security and automation, catering to the demands of large-scale, multi-vendor environments.

EKS

Amazon Elastic Kubernetes Service (EKS) is a fully managed Kubernetes service provided by Amazon Web Services (AWS). Launched in 2018, EKS simplifies the process of deploying, managing, and scaling containerized applications using Kubernetes, an open-source container orchestration platform. EKS abstracts the complexities of Kubernetes cluster management, allowing developers to focus on building and running applications without the need to handle the underlying infrastructure.

Benefits of Using Amazon EKS for Application Deployment

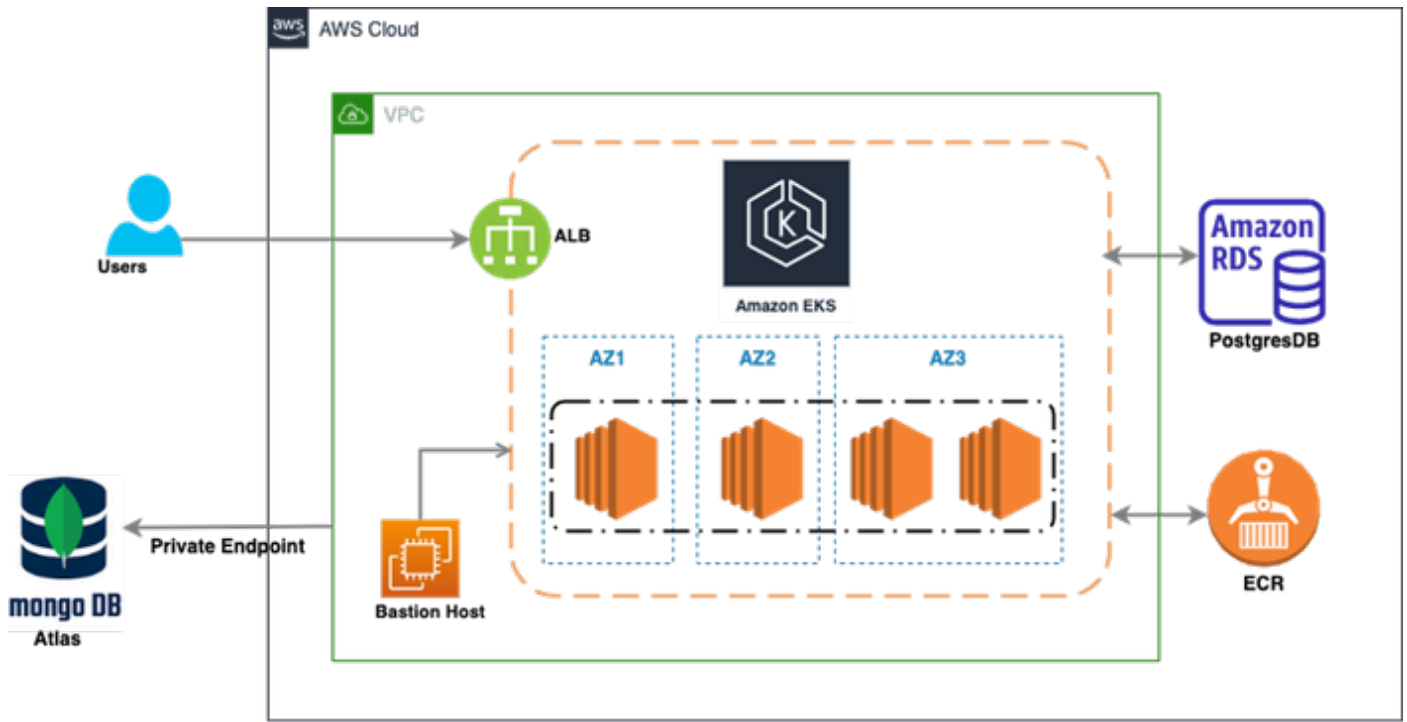
Amazon EKS offers several benefits for application deployment, making it a popular choice for organizations leveraging containerized applications and microservices.

Key advantages include:

- **Managed Kubernetes Control Plane:** EKS handles the deployment, scaling, and maintenance of the Kubernetes control plane, reducing operational burden.
- **Simplified Cluster Management:** EKS abstracts the complexities of setting up and managing Kubernetes clusters.
- **Scalability:** EKS allows for easy scaling of clusters to accommodate growing workloads.
- **High Availability:** EKS supports multi-Availability Zone deployments, enhancing availability and fault tolerance.

- **Integration with AWS Services:** EKS integrates seamlessly with various AWS services.
- **DevOps Automation:** EKS supports continuous integration and continuous deployment (CI/CD) for containerized applications.

BPA Deployment Architecture



This image represents a high-level architecture of a cloud-based infrastructure deployed on **AWS**, using several key components. Here's a breakdown of the diagram:

1. **Amazon EKS (Elastic Kubernetes Service):** At the core of the diagram, Amazon EKS is deployed across three availability zones (AZ1, AZ2, AZ3), with Kubernetes worker nodes inside each zone. This indicates a highly available and fault-tolerant setup, as the workloads are spread across multiple availability zones.
2. **ALB (Application Load Balancer):** This is positioned at the front, receiving traffic from users and distributing it across the EKS cluster for handling application workloads. The load balancer ensures that the requests are evenly distributed and can handle scaling based on traffic demand.
3. **Amazon RDS (Relational Database Service) - PostgreSQL:** On the right side of the diagram, an Amazon RDS instance running PostgreSQL is present. This database can be accessed by applications running within the EKS cluster.
4. **ECR (Elastic Container Registry):** This is where Docker container images are stored and managed, which are then deployed to Amazon EKS for running the workloads.
5. **MongoDB Atlas:** On the left side, MongoDB Atlas is integrated into the architecture through a private endpoint. MongoDB Atlas is a cloud-hosted NoSQL database service, used here to handle document-based database requirements. The private endpoint ensures secure, private communication between the MongoDB Atlas instance and other AWS components.
6. **Bastion Host:** Positioned within the VPC (Virtual Private Cloud), a Bastion Host provides a secure

entry point for administrators to access resources inside the VPC without directly exposing them to the internet.

Overall, this architecture provides a highly available, scalable, and secure solution for deploying and managing containerized applications using Amazon EKS, with support for both relational (PostgreSQL) and NoSQL (MongoDB) databases.

- **EKS Cluster Setup**

To create an Amazon EKS cluster using the AWS CLI, the `eksctl` command-line utility can be used. This is an example command:

```
eksctl create cluster \  
  --name <my-eks-cluster> \  
  --region us-west-2 \  
  --nodegroup-name standard-workers \  
  --node-type t3.medium \  
  --nodes 4 \  
  --nodes-min 4 \  
  --nodes-max 6
```

- **RDS Database Setup**

Deploying a relational database on Amazon RDS involves these steps:

- Access the AWS Management Console and navigate to the Amazon RDS service.
- Create a new database instance with the desired specifications.
- Configure the security group to allow incoming connections from your Amazon EKS cluster.

aws Services Search [Option+S]

RDS > Create database

Create database


Choose a database creation method [Info](#)


Standard create
You set all of the configuration options, including ones for availability, security, backups, and maintenance.


Easy create
Use recommended best-practice configurations. Some configuration options can be changed after the database is created.


Engine options


Engine type [Info](#)


Aurora (MySQL Compatible) 


Aurora (PostgreSQL Compatible) 


MySQL 

MariaDB 

PostgreSQL 

Oracle 

Microsoft SQL Server 

IBM Db2 

Engine version [Info](#)
View the engine versions that support the following database features.

▼ Hide filters

Show versions that support the Multi-AZ DB cluster [Info](#)
Create a Multi-AZ DB cluster with one primary DB instance and two readable standby DB instances. Multi-AZ DB clusters provide up to 2x faster transaction commit latency and automatic failover in typically under 35 seconds.

Engine Version
PostgreSQL 16.3-R2 ▼

Enable RDS Extended Support [Info](#)
Amazon RDS Extended Support is a [paid offering](#). By selecting this option, you consent to being charged for this offering if you are running your database major version past the RDS end of standard support date for that version. Check the end of standard support date for your major version in the [RDS for PostgreSQL documentation](#).

Using the drop-down menu, select the most recent version of PostgreSQL. In our case, it is “PostgreSQL 16.3-R1.”

aws Services Search [Option+S]

Creates a single DB instance with no standby DB instances.

- Multi-AZ DB instance
Creates a primary DB instance and a standby DB instance in a different AZ. Provides high availability and data redundancy, but the standby DB instance doesn't support connections for read workloads.
- Multi-AZ DB Cluster
Creates a DB cluster with a primary DB instance and two readable standby DB instances, with each DB instance in a different Availability Zone (AZ). Provides high availability, data redundancy and increases capacity to serve read workloads.

Settings

DB cluster identifier [Info](#)
Enter a name for your DB cluster. The name must be unique across all DB clusters owned by your AWS account in the current AWS Region.

The DB cluster identifier is case-insensitive, but is stored as all lowercase (as in "mydbcluster"). Constraints: 1 to 60 alphanumeric characters or hyphens. First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.

▼ **Credentials Settings**

Master username [Info](#)
Type a login ID for the master user of your DB cluster.

1 to 16 alphanumeric characters. The first character must be a letter.

Credentials management
You can use AWS Secrets Manager or manage your master user credentials.

- Managed in AWS Secrets Manager - most secure**
RDS generates a password for you and manages it throughout its lifecycle using AWS Secrets Manager.
- Self managed**
Create your own password or have RDS create a password that you manage.

Auto generate password
Amazon RDS can generate a password for you, or you can specify your own password.

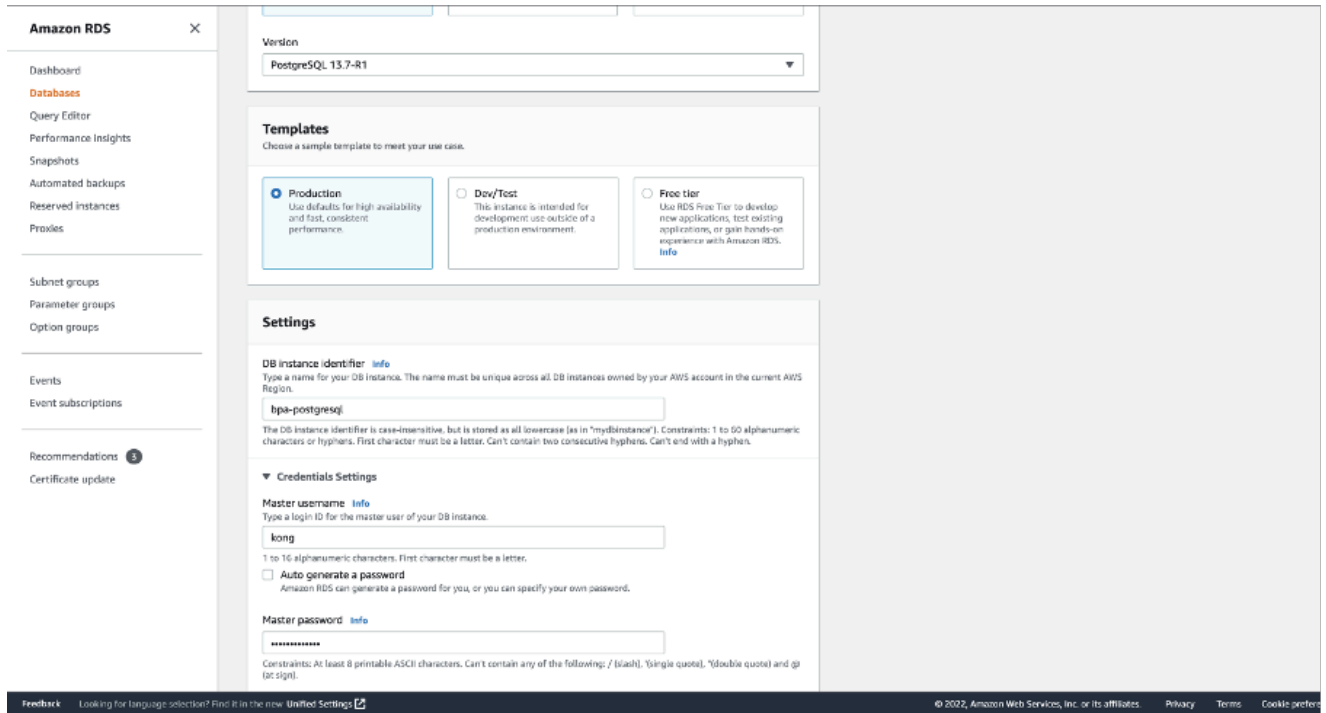
Master password [Info](#)

Password strength Neutral

Minimum constraints: At least 8 printable ASCII characters. Can't contain any of the following symbols: / ' " @

Confirm master password [Info](#)

For this give the database instance a name and create a username and password.



Ensure that the default settings for “DB instance size” and “Storage” are selected.

Depending on the cluster size and data requirements, select the appropriate DB instance size and storage type.

Based on our use case, we have chosen the following configuration:

- **DB Instance Size:** db.m5d.2xlarge
 - 8 vCPUs
 - 32 GiB RAM
 - Network: 4,750 Mbps
 - 300 GB Instance Store

aws Services Search [Option+S]

Instance configuration

The DB instance configuration options below are limited to those supported by the engine that you selected above.

DB instance class [Info](#)

- Standard classes (includes m classes)
- Memory optimized classes (includes r classes)
- Compute optimized classes (includes c classes)

db.m5d.2xlarge
8 vCPUs 32 GiB RAM Network: 4,750 Mbps 300 GB Instance Store

Storage

Storage type [Info](#)
Provisioned IOPS SSD (io2) storage volumes are now available.

Provisioned IOPS SSD (io2)
Low latency, highly durable, I/O intensive storage

Allocated storage [Info](#)

400 GiB
The minimum value is 100 GiB and the maximum value is 65,536 GiB

i After you modify the storage for a DB instance, the status of the DB instance will be in storage-optimization. Your instance will remain available as the storage-optimization operation completes. [Learn more](#)

Provisioned IOPS [Info](#)

3000 IOPS
The minimum value is 1,000 IOPS and the maximum value is 2,56,000 IOPS. The IOPS to GiB ratio must be between 0.5 and 1,000

i Your actual IOPS might vary from the amount that you provisioned based on your database workload and instance type. [Learn more](#)

► Storage autoscaling

Select appropriate values according to your use case. We have selected the default values.

aws Services Search [Option+S]

Connectivity [Info](#)

Compute resource
Choose whether to set up a connection to a compute resource for this database. Setting up a connection will automatically change connectivity settings so that the compute resource can connect to this database.

Don't connect to an EC2 compute resource
Don't set up a connection to a compute resource for this database. You can manually set up a connection to a compute resource later.

Connect to an EC2 compute resource
Set up a connection to an EC2 compute resource for this database.

Virtual private cloud (VPC) [Info](#)
Choose the VPC. The VPC defines the virtual networking environment for this DB cluster.

vpc-usw2az123001nd (vpc-055eca9021e79cfc7)
60 Subnets, 3 Availability Zones

Only VPCs with a corresponding DB subnet group are listed.

ⓘ After a database is created, you can't change its VPC.

DB subnet group [Info](#)
Choose the DB subnet group. The DB subnet group defines which subnets and IP ranges the DB cluster can use in the VPC that you selected.

bpasubnetgroup
2 Subnets, 2 Availability Zones

⚠ The DB subnets must be in 3 Availability Zones (AZs) for the Multi-AZ DB cluster. The current subnets are in 2 AZs (us-west-2a ,us-west-2b). Add a subnet in a different AZ than the current subnets. [Edit new subnet ↗](#)

Public access [Info](#)

Yes
RDS assigns a public IP address to the cluster. Amazon EC2 instances and other resources outside of the VPC can connect to your cluster. Resources inside the VPC can also connect to the cluster. Choose one or more VPC security groups that specify which resources can connect to the cluster.

No
RDS doesn't assign a public IP address to the cluster. Only Amazon EC2 instances and other resources inside the VPC can connect to your cluster. Choose one or more VPC security groups that specify which resources can connect to the cluster.

VPC security group (firewall) [Info](#)
Choose one or more VPC security groups to allow access to your database. Make sure that the security group rules allow the appropriate incoming traffic.

Choose existing
Choose existing VPC security groups

Create new
Create new VPC security group

Make sure in “Database authentication” we have selected Password authentication. Authenticates using database passwords.

**Certificate authority - optional** [Info](#)

Using a server certificate provides an extra layer of security by validating that the connection is being made to an Amazon database. It does so by checking the server certificate that is automatically installed on all databases that you provision.

rds-ca-rsa2048-g1 (default) ▼

Expiry: May 25, 2061

If you don't select a certificate authority, RDS chooses one for you.

Additional configuration**Database port** [Info](#)

TCP/IP port that the database will use for application connections.

5432

Tags - optional

A tag consists of a case-sensitive key-value pair.

No tags associated with the resource.

Add new tag

You can add up to 50 more tags.

Database authentication**Database authentication options** [Info](#)

- Password authentication
Authenticates using database passwords.
- Password and IAM database authentication (not available for Multi-AZ DB cluster)
Authenticates using the database password and user credentials through AWS IAM users and roles.
- Password and Kerberos authentication (not available for Multi-AZ DB cluster)
Choose a directory in which you want to allow authorized users to authenticate with this DB instance using Kerberos Authentication.



▼ Additional configuration

Database options, encryption turned on, backup turned on, backtrack turned off, maintenance, CloudWatch Logs, delete protection turned on.

Database options

Initial database name [Info](#)

Not supported for Multi-AZ DB cluster

If you do not specify a database name, Amazon RDS does not create a database.

DB cluster parameter group [Info](#)

default.postgres16

Option group [Info](#)

Not supported for Multi-AZ DB cluster

Backup

Enable automated backups

Creates a point-in-time snapshot of your DB cluster

Backup retention period [Info](#)

The number of days (1-35) for which automatic backups are kept.

7 days

Backup window [Info](#)

Select the period for which you want automated backups of the DB cluster to be created by Amazon RDS.

Choose a window

No preference

Copy tags to snapshots

Encryption

Enable encryption

Choose to encrypt the given cluster. Master key IDs and aliases appear in the list after they have been created using the AWS Key Management Service (KMS) console. [Info](#)

AWS KMS key [Info](#)

(default) aws/rds

Account

193670463418

Encryption

Enable encryption
Choose to encrypt the given cluster. Master key IDs and aliases appear in the list after they have been created using the AWS Key Management Service (KMS) console. [Info](#)

AWS KMS key [Info](#)
(default) aws/rds

Account
193670463418

KMS key ID
61e6c956-745e-42be-8fd1-77953104ad4f

Log exports
Select the log types to publish to Amazon CloudWatch Logs

PostgreSQL log
 Upgrade log

IAM role
The following service-linked role is used for publishing logs to CloudWatch Logs.
RDS service-linked role

Maintenance
Auto minor version upgrade [Info](#)

Enable auto minor version upgrade
Enabling auto minor version upgrade will automatically upgrade to new minor versions as they are released. The automatic upgrades occur during the maintenance window for the database.

Maintenance window [Info](#)
Select the period you want pending modifications or maintenance applied to the database by Amazon RDS.

Choose a window
 No preference

Deletion protection

Enable deletion protection
Protects the database from being deleted accidentally. While this option is enabled, you can't delete the database cluster.

Information: You are responsible for ensuring that you have all of the necessary rights for any third-party products or services that you use with AWS services.

Buttons: Cancel, Create database

Once that is verified, we are ready to create the database. Return to the Amazon RDS dashboard. Confirm that the instance is available for use.

Security Group Rules

Update the inbound security group with the pod CIDR and node CIDR block.

Details **Inbound rules** Outbound rules Tags

Inbound rules (2)

Search

<input type="checkbox"/>	Name	Security group rule...	IP version	Type	Protocol	Port range	Source
<input type="checkbox"/>	-	sgr-0962e7821f1df7ede	IPv4	All traffic	All	All	
<input type="checkbox"/>	-	sgr-047daa40317c616...	IPv4	All traffic	All	All	

In RDS -> Databases -> DB-NAME, click configuration and refer the Parameter Group section and click the parameter group to view.

Amazon RDS

RDS > Databases > bpa-postgresql

bpa-postgresql Modify Actions

Summary

DB identifier bpa-postgresql	CPU 6.18%	Status Available	Class db.t4g.large
Role Instance	Current activity 0.07 sessions	Engine PostgreSQL	Region & AZ us-west-1b

Connectivity & security | Monitoring | Logs & events | **Configuration** | Maintenance & backups | Tags

Instance

Configuration	Instance class	Storage	Performance Insights
DB instance ID bpa-postgresql Engine version 13.7 DB name bpa_admin License model Postgresql License Option groups default:postgres-11 In sync Amazon Resource Name (ARN) arn:aws:rds-us-west-1:260251831100:db:bpa-postgresql Resource ID db-CUGR55TB2B4ZAPGH2ZCVJ4SDAM Created time July 31, 2022, 15:22 (UTC-05:30) Parameter group bpa-postgresql-20220731094942083200000001 In sync Deletion protection	Instance class db.t4g.large vCPU 2 RAM 8 GB Availability Master username kong IAM DB authentication Not enabled Multi-AZ Yes Secondary Zone us-west-1c	Encryption Enabled AWS KMS key aws/rds Storage type General Purpose SSD (gp2) Storage 20 GiB Provisioned IOPS - Storage autoscaling Enabled Maximum storage threshold 100 GiB	Performance Insights enabled Turned on AWS KMS key aws/rds Retention period 7 days Published logs CloudWatch Logs PostgreSQL Upgrade Database activity stream Status Stopped Audit policy status -

Search for “password_encryption” and change the value to md5 from blank / other value. This is needed for camunda configurations to work.

Amazon RDS

RDS > Parameter groups > bpa-postgresql-20220731094942083200000001

bpa-postgresql-20220731094942083200000001 Edit parameters

Parameters

password

<input type="checkbox"/>	Name	Values	Allowed values	Modifiable	Source	Apply type	Data type	Description
<input checked="" type="checkbox"/>	password_encryption	md5	md5, scram-sha-256	true	system	dynamic	string	Encrypt passwords.
<input type="checkbox"/>	rds.accepted_password_auth_method	md5+scram	md5+scram, scram	true	system	dynamic	string	Force authentication for connections with password stored locally
<input type="checkbox"/>	rds.restrict_password_commands	0, 1		true	system	static	boolean	restricts password-related commands to members of rds_password

Recent events

Filter db events

Time System notes

No events found.

Create these Databases along with users by connecting to the RDS.

PG_ROOT_DATABASE=admin

```
PG_INITDB_ROOT_USERNAME=admin
PG_INITDB_ROOT_PASSWORD=Bp@Chang3d!
AUTH_DB_NAME=kong
AUTH_DB_USER=kong
AUTH_DB_PASSWORD=K@ngPwdCha*g3
WFE_DB_USER=camunda
WFE_DB_PASSWORD=W0rkFlo#ChangeNow
WFE_DB_NAME=process-engine
```

- Password authentication

Authenticates using database passwords.

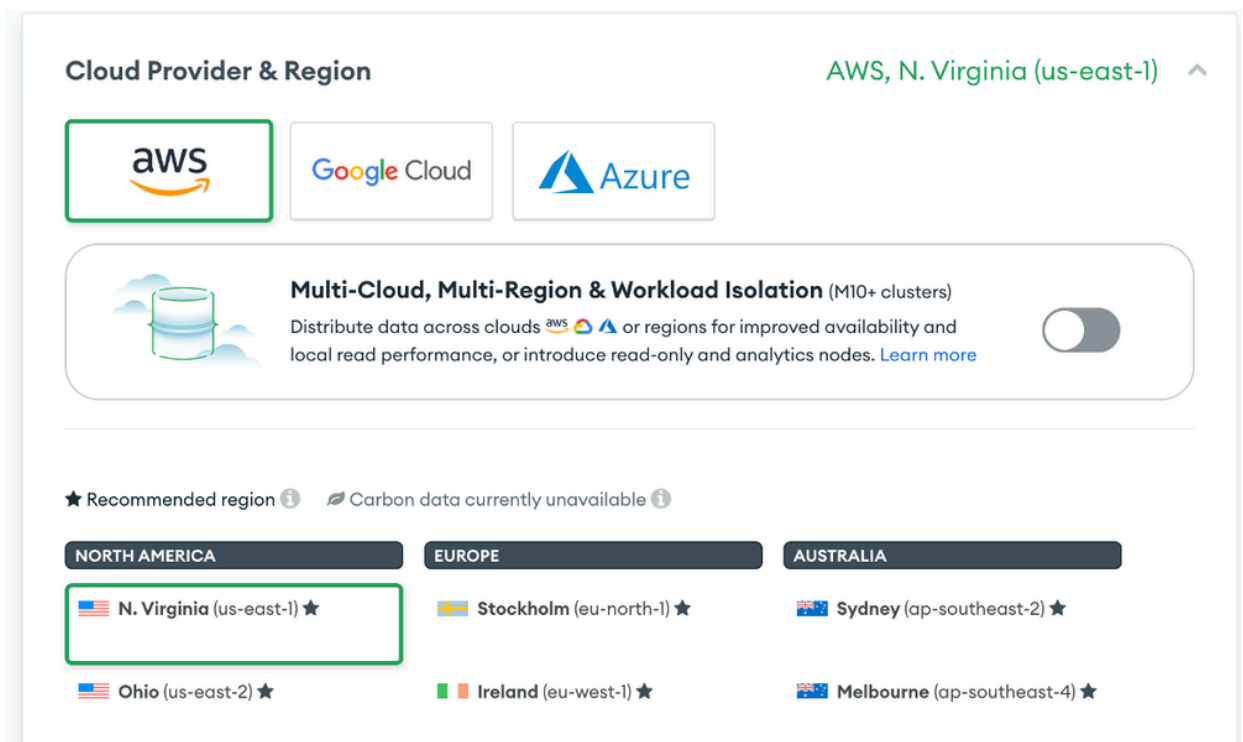
- **Atlas MongoDB Setup**

Setting up Atlas MongoDB involves:

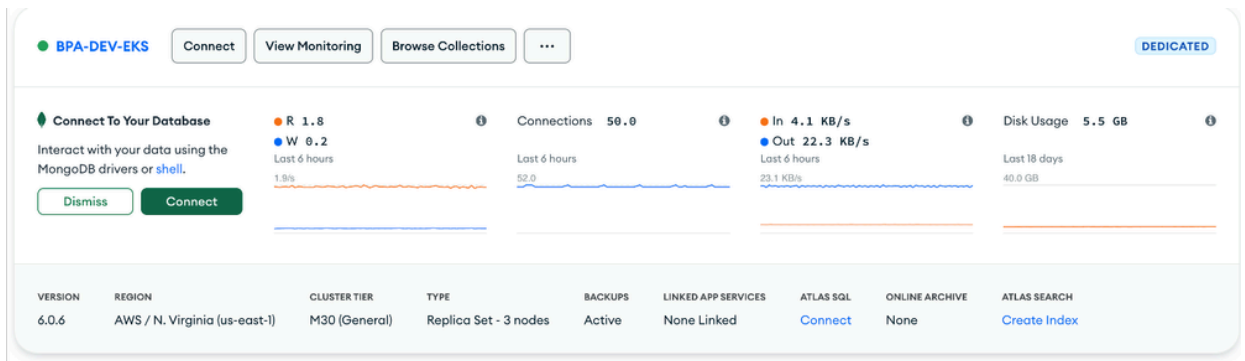
- **Logging into Atlas MongoDB.**
- **Selecting the organization and project.**
- **Creating a dedicated cluster with the appropriate specifications.**



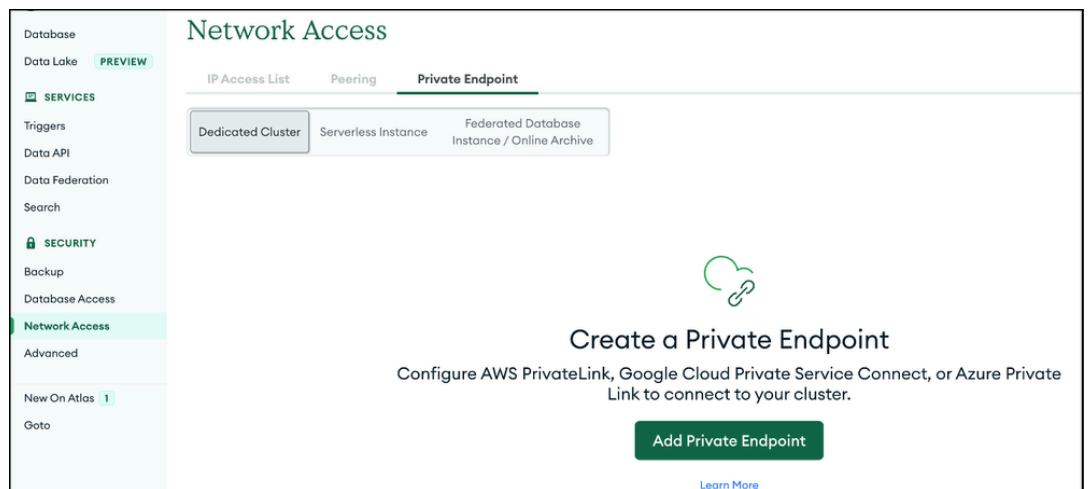
- **Select the Dedicated tier, Cloud Provider & Region.**



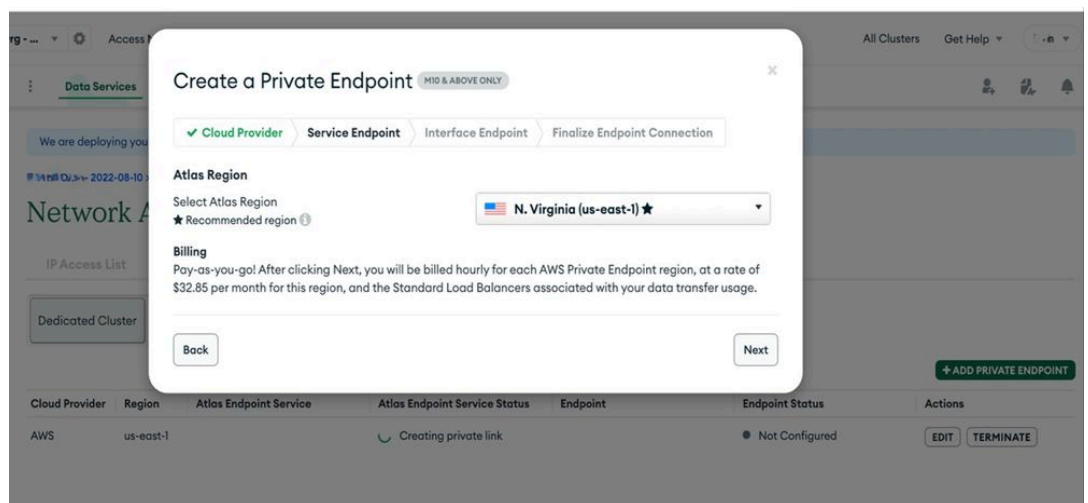
- **Select appropriate tier(we have used M30 as tier) dedicated cluster and provide appropriate cluster name and click on Create Cluster. It will initialize the Atlas monogoddb cluster.**



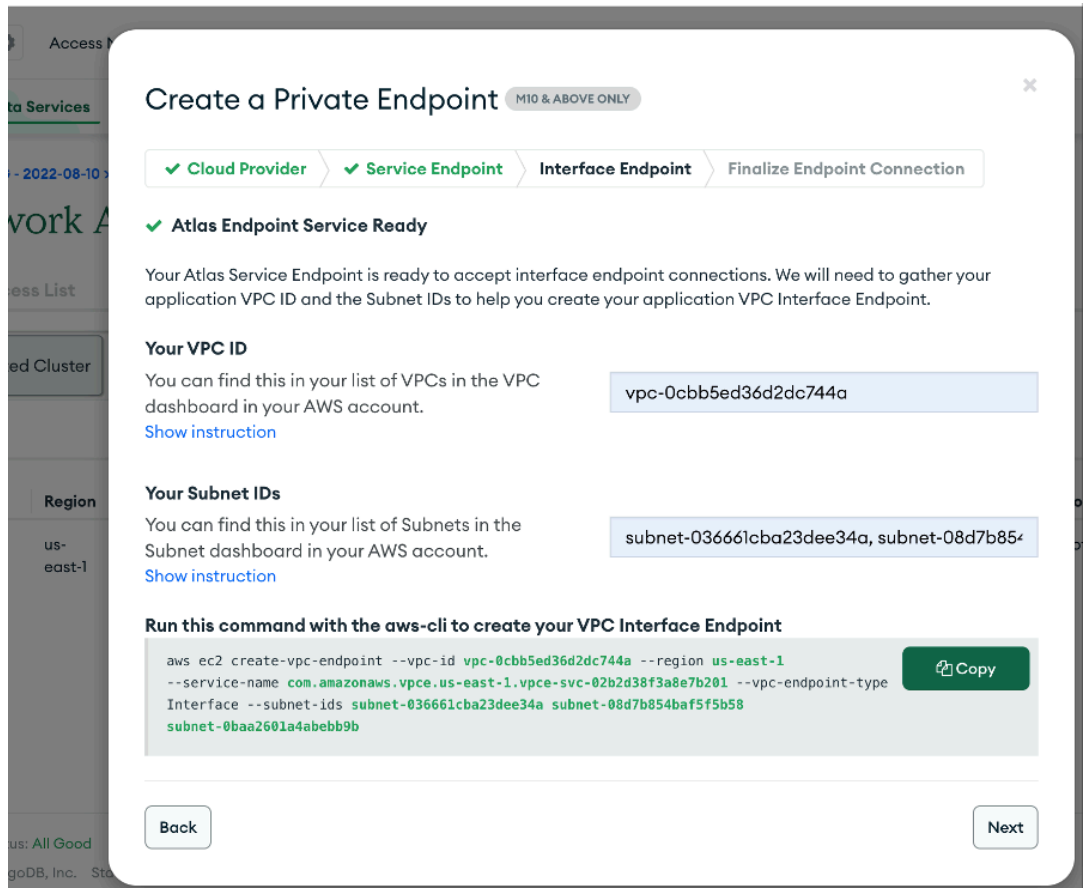
- **Setting up VPC private endpoint for the Atlas and K8S cluster.**
 - **Click on the Network Access Select Private Endpoint à Click on Add Private Endpoint.**



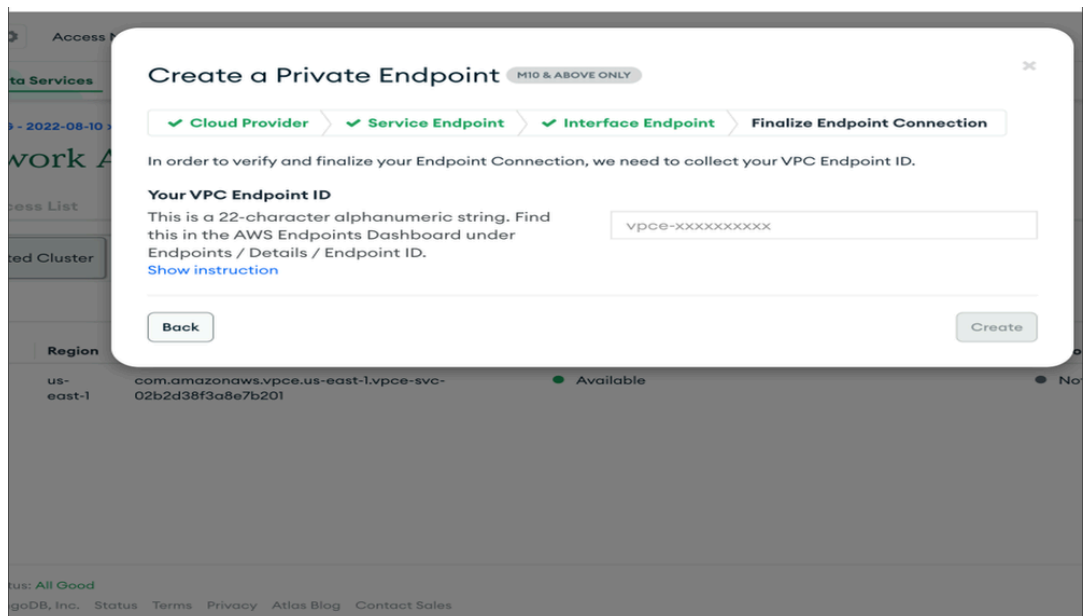
- **Select Cloud Provider as AWS, select respective Region and click on Next.**



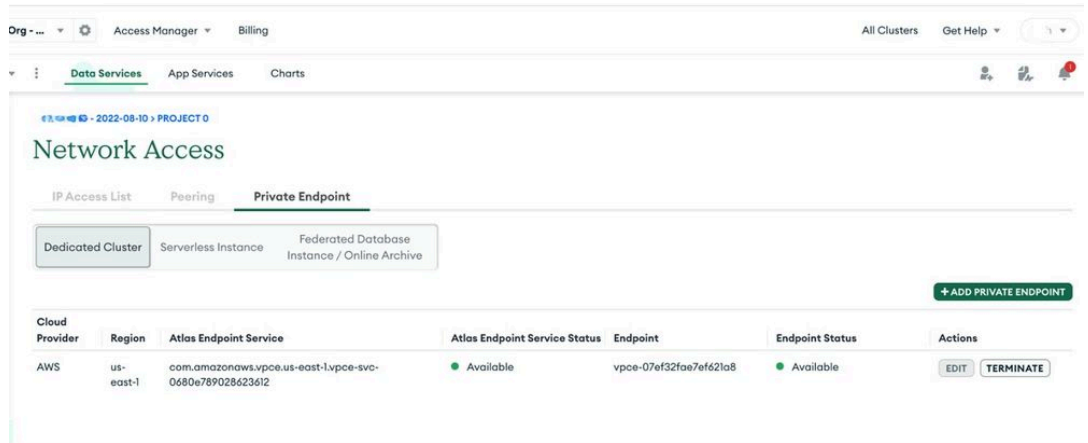
- **Provide Respective PVC id and subnet ids. Once you enter the details, Copy the vpc end point creation command and execute it in aws console. You will get the vpc endpoint id as output.**



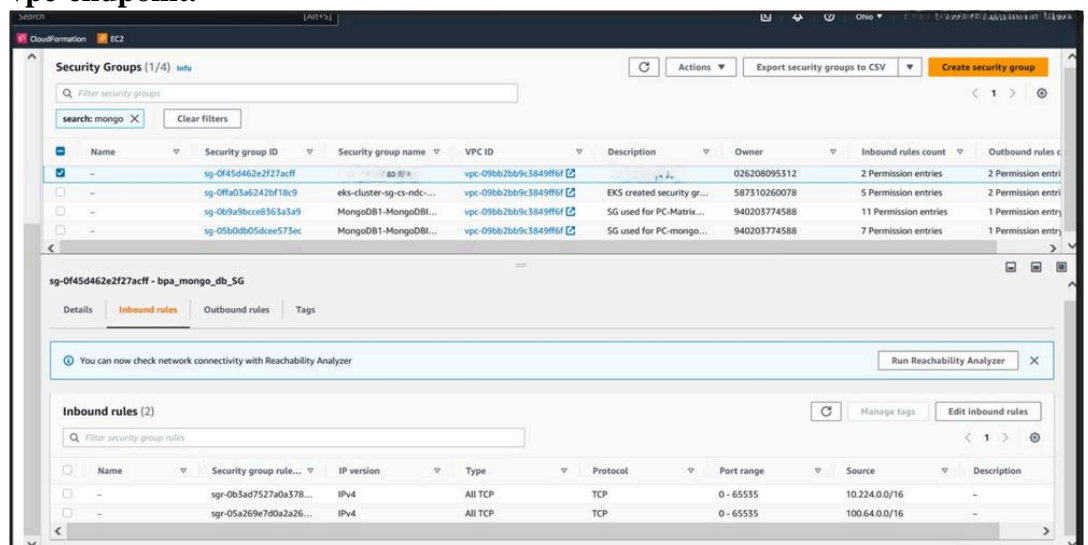
- **Click on Next to paste the VPC endpoint ID and click on Create.**



- **Once it is successfully created, Endpoint status will be Available as shown in the next picture. VPC end-point must be created for pod cidr. In our case we have used "100.64.0.0/16" .**



- **Add inbound rules to newly created vpc-endpoint. The vpc-endpoint will be in the parent account and a security group must be assigned to the newly created vpc-endpoint.**



ECR as image registry

Creating Amazon ECR repositories and pushing Docker images into them involves several steps. These are the steps to create an ECR repository, tag a Docker image, and push it to the repository using the AWS CLI.

```
aws ecr create-repository --repository-name your-image-name --region your-region
```

Replace:

- **your-image-name** with the desired name for your ECR repository.
- **your-region** with your AWS region

Configure IAM Role for EKS Nodes

Ensure that the EKS worker nodes (EC2 instances) have the necessary IAM role attached with permissions to pull images from ECR. The IAM policy required is:

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ecr:GetDownloadUrlForLayer",
        "ecr:BatchGetImage",
        "ecr:BatchCheckLayerAvailability"
      ],
      "Resource": "*"
    }
  ]
}

```

Attach this policy to the IAM role associated with your EKS worker nodes.

BPA Deployment

The deployment of BPA involves several steps, including labeling EKS worker nodes, preparing directories on nodes, copying BPA packages, and deploying BPA using Helm.

For our customer deployment, we have utilized the following versions of software and cloud services:

- **BPA:** 4.0.3-6
- **RDS (Relational Database Service):** 16.3-R2
- **MongoDB Atlas:** v5.0.29
- **EKS (Elastic Kubernetes Service):** v1.27

These components ensure that our deployment is robust, scalable, and capable of handling the required workloads efficiently.

- **Labeling EKS Worker Nodes**

```

kubectl label node <worker_node_1> name=node-1
kubectl label node <worker_node_2> name=node-2
kubectl label node <worker_node_3> name=node-3
kubectl label node <worker_node_4> name=node-4

```

- **Preparing Directories on Nodes**
Node 1:

```

rm -rf /opt/bpa/data/
mkdir -p /opt/bpa/data/zookeeper1
mkdir -p /opt/bpa/data/zookeeper4
mkdir -p /opt/bpa/data/zookeeper5
chmod 777 /opt/bpa/data/zookeeper1
chmod 777 /opt/bpa/data/zookeeper4
chmod 777 /opt/bpa/data/zookeeper5
mkdir -p /opt/bpa/data/kafka1
chmod 777 /opt/bpa/data/kafka1
sysctl -w vm.max_map_count=262144

```

Node 2:

```
rm -rf /opt/bpa/data
sysctl -w vm.max_map_count=262144
mkdir -p /opt/bpa/data/kafka2
mkdir -p /opt/bpa/data/zookeeper2
mkdir -p /opt/bpa/data/zookeeper4
mkdir -p /opt/bpa/data/zookeeper5
chmod 777 /opt/bpa/data/kafka2
chmod 777 /opt/bpa/data/zookeeper2
chmod 777 /opt/bpa/data/zookeeper4
chmod 777 /opt/bpa/data/zookeeper5
```

Node 3:

```
rm -rf /opt/bpa/data
sysctl -w vm.max_map_count=262144
mkdir -p /opt/bpa/data/kafka3
mkdir -p /opt/bpa/data/zookeeper3
mkdir -p /opt/bpa/data/zookeeper4
mkdir -p /opt/bpa/data/zookeeper5
chmod 777 /opt/bpa/data/kafka3
chmod 777 /opt/bpa/data/zookeeper3
chmod 777 /opt/bpa/data/zookeeper4
chmod 777 /opt/bpa/data/zookeeper5
```

Node 4:

```
mkdir -p /opt/bpa/data/elk
mkdir -p /opt/bpa/data/metrices/prometheus
mkdir -p /opt/bpa/data/metrices/grafana
chmod 777 /opt/bpa/data/metrices
chmod 777 /opt/bpa/data/metrices/prometheus
chmod 777 /opt/bpa/data/metrices/grafana
sysctl -w vm.max_map_count=262144
```

- Copying BPA Packages

```
scp -r packages to node1:/opt/bpa/
scp -r packages to node2:/opt/bpa/
scp -r packages to node3:/opt/bpa/
scp -r packages to node4:/opt/bpa/
```

- Deploying BPA Using Helm

```
helm install bpa-rel --create-namespace --namespace bpa-ns /opt/EKS/bpa-helm-chart
```

Ingress Setup

- **Enabling Ingress**

Update `values.yaml` to enable ingress:

```
ingress_controller: {create: true}
```

- **Creating a Secret Using BPA Certificate**

Navigate to the certificate directory and create a secret:

```
cd /opt/bpa/<BPA helm chart location>/bpa/conf/common/certs/  
kubectl create secret tls bpa-certificate-ingress --cert=bap-cert.pem --key=bap-key.pem -n bpa-ns
```

- **Updating Ingress Controller**

Add the newly created secret in the `ingress-controller.yaml` file:

```
cd /opt/bpa/<BPA helm chart location>/templates/  
vi ingress-controller.yaml  
"- --default-ssl-certificate=$(POD_NAMESPACE)/bpa-certificate-ingress"
```

- **Updating Ingress Certificate**

Perform Helm delete and install to update the ingress certificate.

Environment Specifications

The environment specifications include requirements for EC2 instances, load balancers, VPC endpoints, and RDS instances. Key specifications are:

EC2 Requirements:

Storage requirements: 2TB space per nodes. Mount EBS volume to `/opt` and add an entry in `/etc/fstab` for all the nodes.

Security group inbound: 30101, 443, 0 – 65535 TCP, 22 for ssh.

Security group outbound: All traffic must be enabled.

DNS Resolver: EC2 must have on-prem resolvers in `/etc/resolve.conf`.

Load balancer requirements:

- Listeners ports must be 443, 30101.
- VPC End point Requirements (Atlas MongoDB).
- VPC end points created for Atlas connectivity is available in the parent account(aws-5g-ndc-prod). VPC Endpoint must have security group which allows all inbound access(0 - 65535).

RDS Requirements:

RDS Type: db.r5b.2xlarge

Postgres Engine version: 13.7

Security group: Inbound must allow traffic from the POD CIDR source.

Key Concepts and Components

Understanding Kubernetes fundamentals is essential for effectively deploying and managing applications using Amazon EKS.

Conclusion

This paper provides a detailed guide for deploying and managing Business Process Automation (BPA) applications using Amazon EKS. By following the outlined steps and understanding the key concepts, organizations can leverage the benefits of EKS for their containerized BPA applications.

References

- Amazon Web Services, "Amazon EKS Documentation," [Online]. Available:<https://docs.aws.amazon.com/eks/>
- Kubernetes, "Kubernetes Documentation," [Online]. Available:<https://kubernetes.io/docs/home/>
- Cisco BPA at a Glance <https://www.cisco.com/c/en/us/solutions/collateral/service-provider/at-a-glance-c45-742579.html>
- BPA Operations Guide <https://www.cisco.com/c/dam/en/us/support/docs/bpa/v403/cisco-bpa-operations-guide-v403.pdf>
- BPA Developer Guide <https://www.cisco.com/c/dam/en/us/support/docs/bpa/v403/cisco-bpa-developer-guide-v403.pdf>