

Deploy and Manage Business Process Automation Application on Amazon EKS: A Practical Guide

Contents

[ABSTRACT](#)

[INTRODUCTION](#)

[BPA DEPLOYMENT ARCHITECTURE](#)

[EKS CLUSTER SETUP](#)

[RDS DATABASE SETUP](#)

[ATLAS MONGODB SETUP](#)

[ECR AS IMAGE REGISTRY](#)

[BPA DEPLOYMENT](#)

[KEY CONCEPTS AND COMPONENTS](#)

[CONCLUSION](#)

[REFERENCES](#)

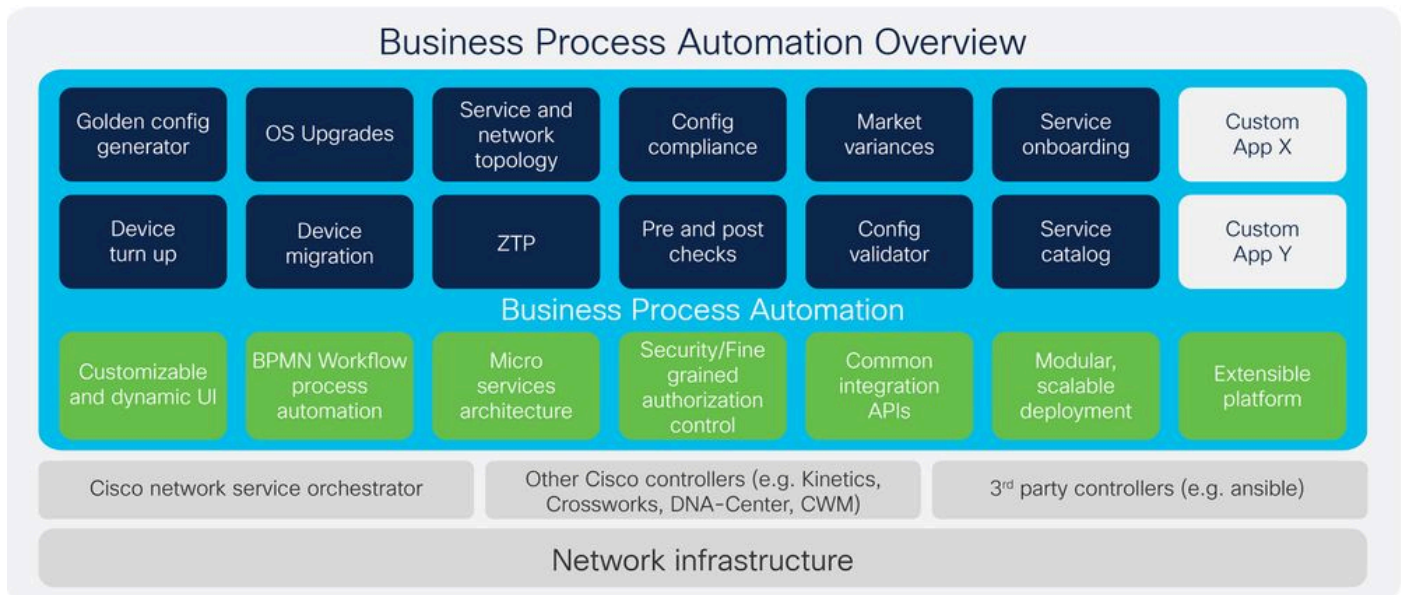
ABSTRACT

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

BUSINESS PROCESS AUTOMATION (BPA)



Cisco Business Process Automation (BPA) Services offer an end-to-end consulting and support services portfolio designed for process and workflow automation and orchestration. The BPA platform is scalable and microservices-based, featuring an embedded workflow engine, digital user interface, and common integration middleware. This platform helps automate complex network configuration changes and associated processes, making it suitable for both service provider customers and large global enterprises.

Key benefits of Cisco BPA Services include:

- Automating complicated methods and operating procedures.
- Enhancing team expertise to speed up automation initiatives.
- Accelerating the rollout of new services with an improved user interface/portal.
- Integrating legacy networks with new automation capabilities.

The BPA platform supports various business and IT/operational use cases such as OS upgrades, service provisioning, and integration with orchestration engines. Customers can access a lifecycle of services and BPA capabilities, including advisory, implementation, business-critical services, and solution support. Cisco BPA Services aim to increase operational efficiencies, reduce costly errors, improve business agility, and deliver faster returns on automation investments.

AMAZON ELASTIC KUBERNETES SERVICE (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.

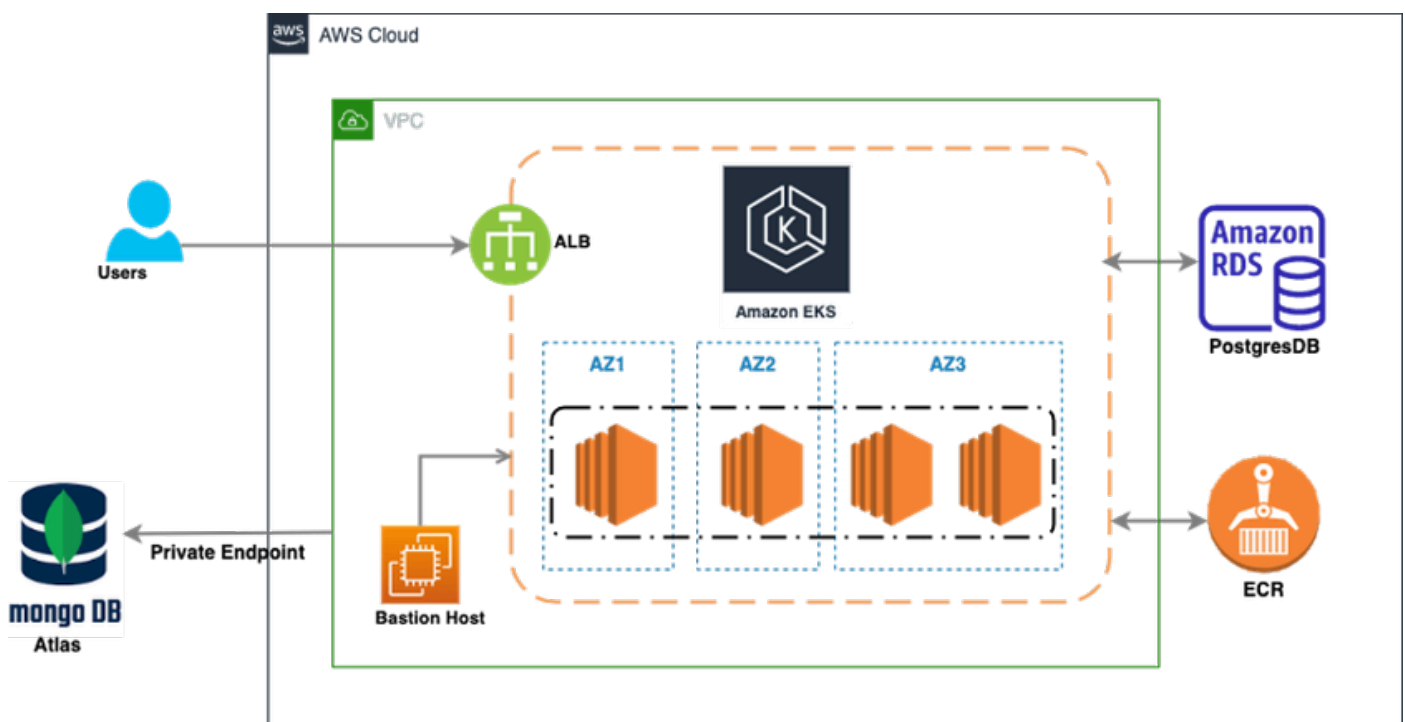
Benefits of Using Amazon EKS for Application Deployment

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.
7. 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:

1. Access the AWS Management Console and navigate to the Amazon RDS service.
2. Create a new database instance with the desired specifications.
3. Configure the security group to allow incoming connections from your Amazon EKS cluster.
 1. Using the drop-down menu, select the most recent version of PostgreSQL. In our case, it is “PostgreSQL 16.3-R1”.

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

ORACLE

☐ Microsoft SQL Server



☐ IBM Db2

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](#).

Amazon RDS

Dashboard

Databases

Query Editor

Performance Insights

Snapshots

Automated backups

Reserved instances

Proxies

Subnet groups

Parameter groups

Option groups

Events

Event subscriptions

Recommendations

Certificate update

Version

PostgreSQL 13.7-81

Templates

Choose a sample template to meet your use case.

Production

Use defaults for high availability and fast, consistent performance.

Dev/Test

This instance is intended for development use outside of a production environment.

Free tier

Use RDS Free Tier to develop new applications, test existing applications, or gain hands-on experience with Amazon RDS.

Settings

DB instance identifier

info

Type a name for your DB instance. The name must be unique across all DB instances owned by your AWS account in the current AWS Region.

tpa-postgresql

The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 63 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 instance.

long

1 to 16 alphanumeric characters. First character must be a letter.

☐ Auto generate a password

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

Master password

info

Constraints: At least 8 printable ASCII characters. Can't contain any of the following: / (slash), % (percent), ' (single quote), " (double quote) and @ (at sign).

Feedback

Looking for language selection? Find it in the new [Global Settings](#).

© 2022, Amazon Web Services, Inc. or its affiliates.

Privacy

Terms

Cookie prefer

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

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.

bpa-postgres

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.

kong

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 [Info](#) Neutral

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

Confirm master password [Info](#)

1. 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.
2. 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
3. Select appropriate values according to your use case. We have selected the default values.

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 GiB 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

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

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

Storage autoscaling

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

Services
Search
[Option+S]

Connectivity [Info](#)

☒ **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

- 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.

aws

Services

Search

[Option+S]

☰

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.

aws

Services

Search

[Option+S]

▼ 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

Services

Search

[Option+S]

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

KMS key ID

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.

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.

Cancel

Create database

Security Group Rules

1. Update the inbound security group with the pod CIDR and node CIDR block.
2. In RDS -> Databases -> DB-NAME, click configuration and refer the Parameter Group section and click the parameter group to view.

Details

Inbound rules

Outbound rules

Tags

Inbound rules (2)

🔄

Manage tags

Edit inbound rules

🔍 Search

< 1 > ⚙

<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	

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

Amazon RDS
Dashboard
Databases
Query Editor
Performance insights
Snapshots
Automated backups
Reserved instances
Proxies
Subnet groups
Parameter groups
Option groups
Events
Event subscriptions
Recommendations
Certificate update

RDS > Databases > bpa-postgresql
bpa-postgresql
Modify
Actions

Summary

DB identifier
bpa-postgresql

CPU
0.18%

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-13 In sync

Amazon Resource Name (ARN)
arn:aws:rds:us-west-1:260251831100:db:bpa-postgresql

Resource ID
db-CU6R5S7B242APGH2ZCVJ4SDAM

Created time
July 31, 2022, 15:22 (UTC+05:30)

Parameter group
bpa-postgresql-20220731094942083200000000 In sync

Deletion protection

Instance class

Availability

Storage

Performance insights

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

```
PG_ROOT_DATABASE=admin
PG_INITDB_ROOT_USERNAME=admin
PG_INITDB_ROOT_PASSWORD=xxxxxxx
AUTH_DB_NAME=kong
AUTH_DB_USER=kong
AUTH_DB_PASSWORD=xxxxxxxxxx
WFE_DB_USER=camunda
WFE_DB_PASSWORD=xxxxxxxxxx
WFE_DB_NAME=process-engine
```

- To update database passwords, modify the values in the bpa-helm-chart/bpa/env/environment.txt file. This file is used for authenticating database connections.

ATLAS MONGODB SETUP

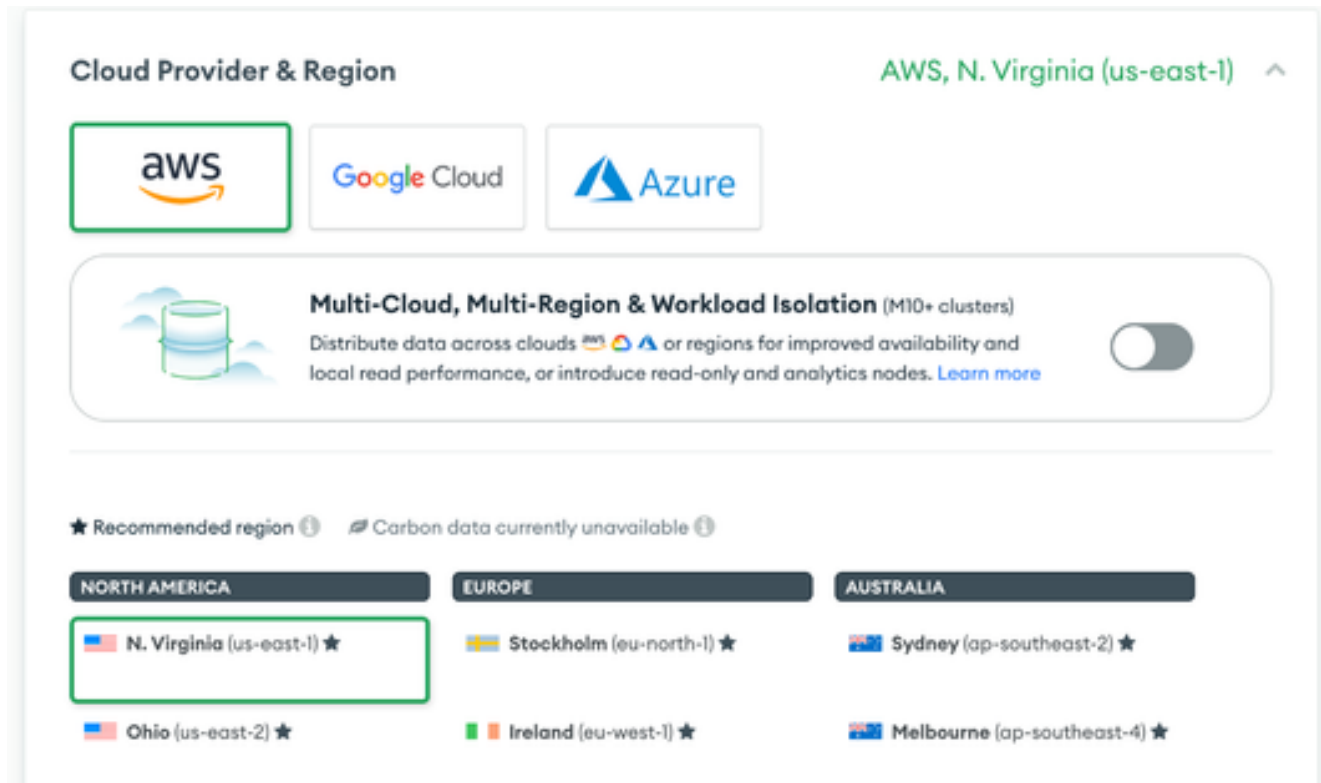
Setting up Atlas MongoDB involves:

- Logging into Atlas MongoDB.

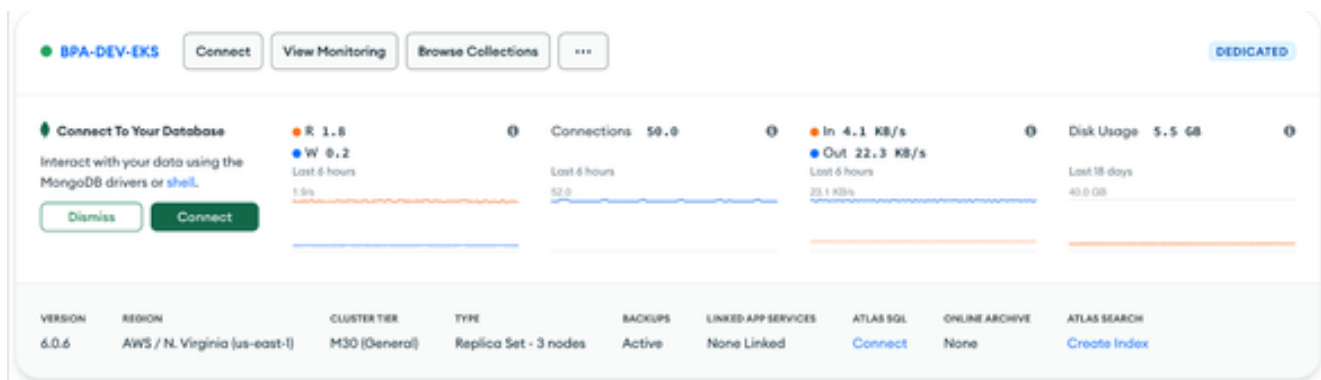
2. Selecting the organization and project.
3. Creating a dedicated cluster with the appropriate specifications and version. In our case, it is “MongoDB Atlas v5.0.29”.



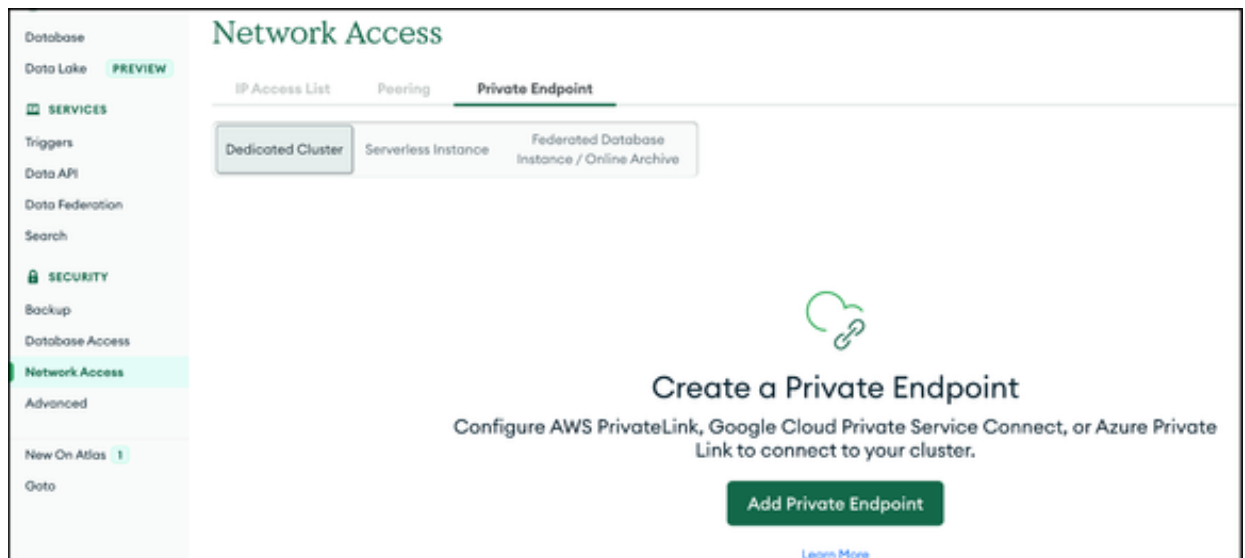
4. Select the Dedicated tier, Cloud Provider & Region.



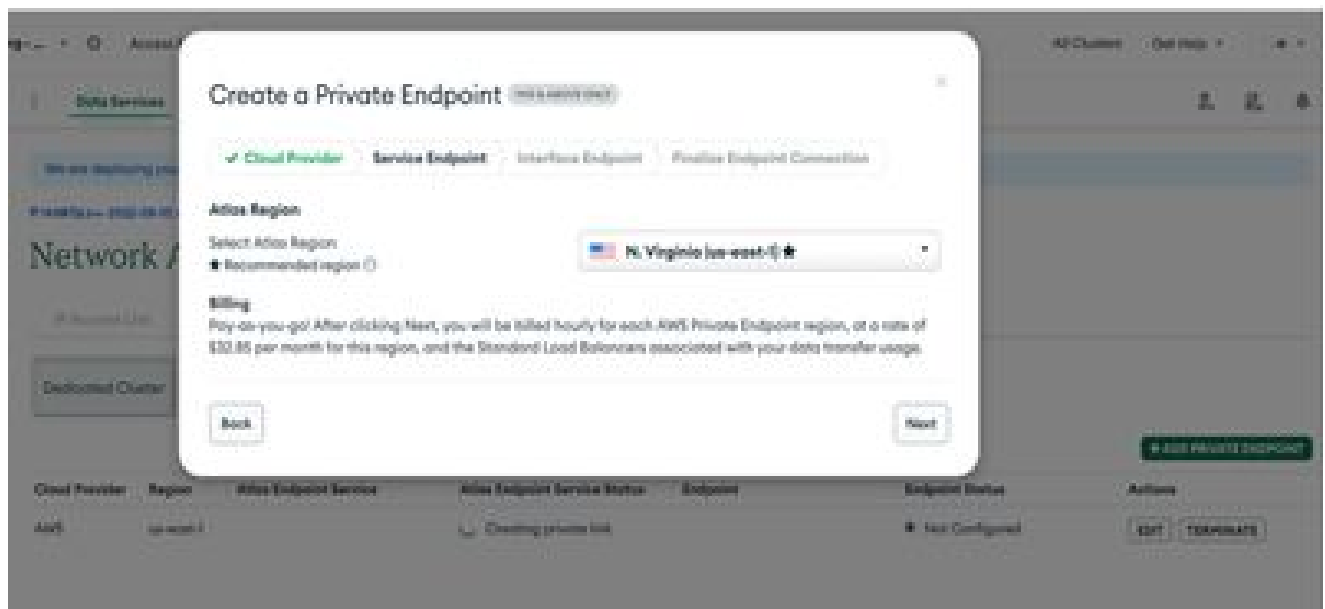
5. Select appropriate tier(we have used M30 as tier) dedicated cluster and provide appropriate cluster name and click on Create Cluster. It will initialize the Atlas monogodb cluster.



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



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



8. 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.

Access

to Services

2022-08-10

work A

ess List

ed Cluster

Region

us-east-1

us: All Good

goDB, Inc. S

Create a Private Endpoint MIG & ABOVE ONLY

✓ Cloud Provider

✓ Service Endpoint

Interface Endpoint

Finalize Endpoint Connection

✓ Atlas Endpoint Service Ready

Your Atlas Service Endpoint is ready to accept interface-endpoint connections. We will need to gather your application VPC ID and the Subnet IDs to help you create your application VPC Interface Endpoint.

Your VPC ID

You can find this in your list of VPCs in the VPC dashboard in your AWS account.

[Show instruction](#)

Your Subnet IDs

You can find this in your list of Subnets in the Subnet dashboard in your AWS account.

[Show instruction](#)

Run this command with the `aws-cli` to create your VPC Interface Endpoint

```
aws ec2 create-vpc-endpoint --vpc-id vpc-8cbb5ed86d2dc744a --region us-east-1
--service-name com.amazonaws.vpce.us-east-1.vpce-svc-02b2d38f3a8e7b201 --vpc-endpoint-type
Interface --subnet-ids subnet-034661c0a23dee34a subnet-00d7b434ba73f3e39
subnet-0bca2441a1abbb9b
```

Copy

Back

Next

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

Access

to Services

2022-08-10

work A

ess List

ed Cluster

Region

us-east-1

us: All Good

goDB, Inc. Status Terms Privacy Atlas Blog Contact Sales

Create a Private Endpoint MIG & ABOVE ONLY

✓ Cloud Provider

✓ Service Endpoint

✓ Interface Endpoint

Finalize Endpoint Connection

In order to verify and finalize your Endpoint Connection, we need to collect your VPC Endpoint ID.

Your VPC Endpoint ID

This is a 22-character alphanumeric string. Find this in the AWS Endpoints Dashboard under Endpoints / Details / Endpoint ID.

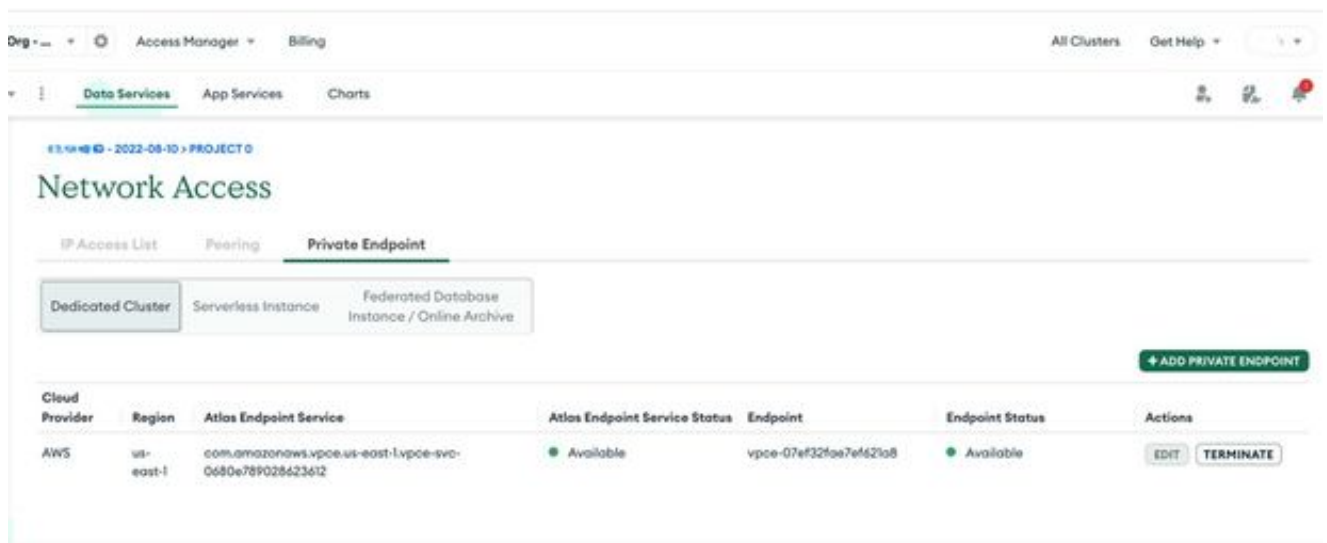
[Show instruction](#)

vpce-xxxxxxxxxx

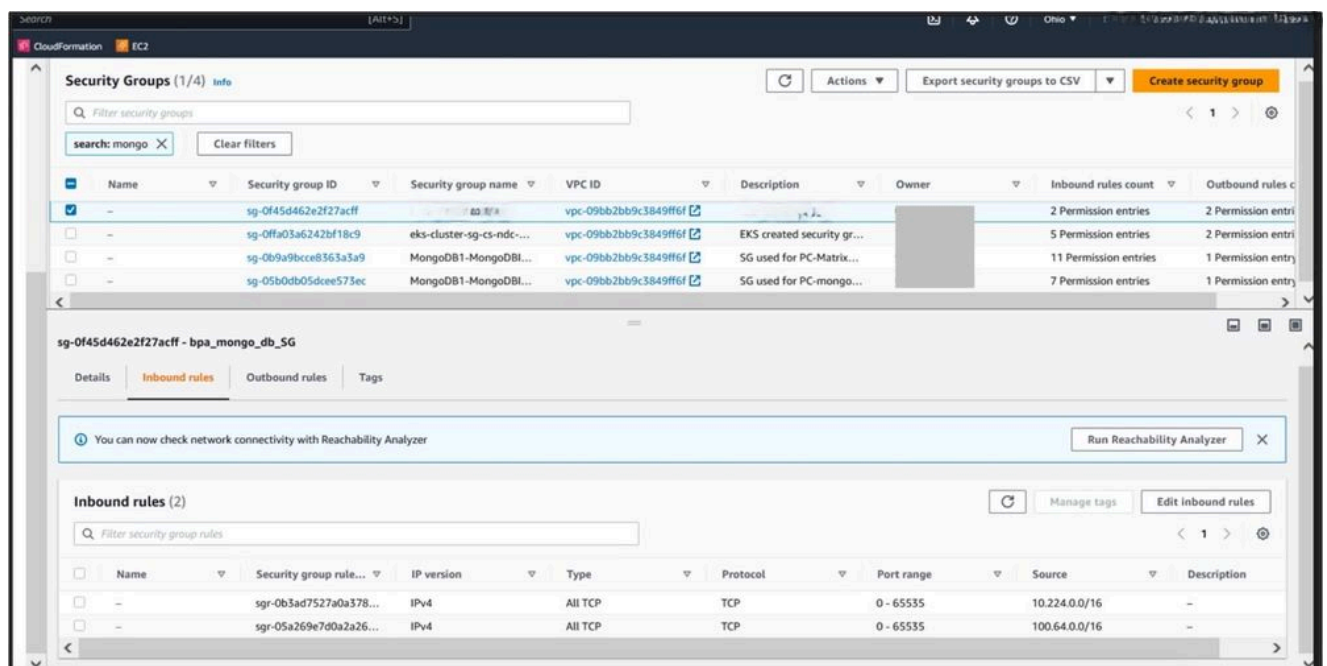
Back

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

3. Steps to tag and push the image.

- Example:

```
docker tag containers.cisco.com/bpa/sase-service:4.0.3-522
<account number>.dkr.ecr.us-west-
2.amazonaws.com/<repository_path>/sase-service:4.0.3-522

docker push <account number>.dkr.ecr.us-west-
2.amazonaws.com/<repository_path>/sase-service:4.0.3-522
```

4. Configure IAM Role for EKS Nodes

5. 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": "*"
    }
  ]
}
```

6. 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.

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

1. BPA: **4.0.3-6**
 2. RDS (Relational Database Service): **PostgreSQL 16.3-R1**
 3. MongoDB Atlas: **v5.0.29**
 4. EKS (Elastic Kubernetes Service): **v1.27**
2. These components ensure that our deployment is robust, scalable, and capable of handling the required workloads efficiently.
3. 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/
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 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/metrics/prometheus
mkdir -p /opt/bpa/data/metrics/grafana
chmod 777 /opt/bpa/data/metrics
chmod 777 /opt/bpa/data/metrics/prometheus
chmod 777 /opt/bpa/data/metrics/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:
 1. **Storage requirements:** 2TB space per nodes. Mount EBS volume to /opt and add an entry in /etc/fstab for all the nodes.
 2. **Security group inbound:** 30101, 443, 0 – 65535 TCP, 22 for ssh.
 3. **Security group outbound:** All traffic must be enabled.
 4. **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. VPC Endpoint must have security group which allows all inbound access(0 - 65535).

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>