Distribuire e gestire le applicazioni di automazione dei processi aziendali su Amazon EKS: a Practical Guide

Sommario

Riassunto

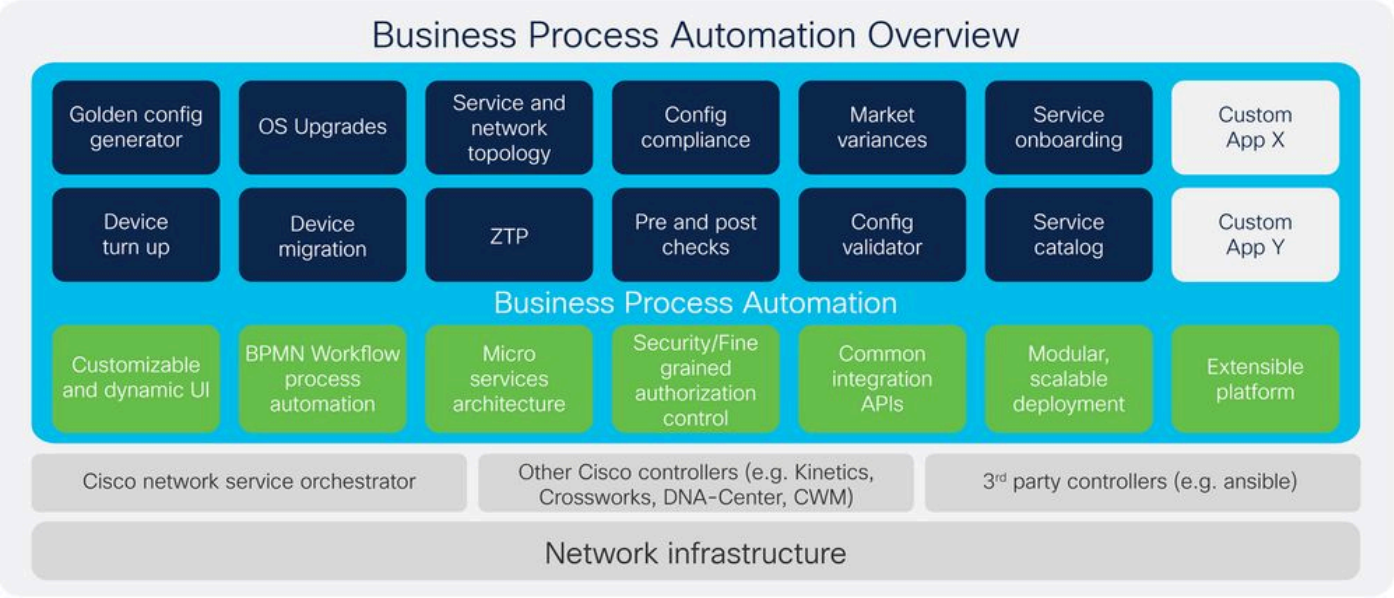
Questo documento presenta una guida completa sull'installazione e la gestione delle applicazioni BPA (Business Process Automation) utilizzando il servizio EK (Amazon Elastic Kubernetes Service). Illustra i prerequisiti, evidenzia i vantaggi dell'utilizzo di EKS e fornisce istruzioni dettagliate per la configurazione di un cluster EKS, di un database RDS Amazon e di un atlante MongoDB. Inoltre, il documento analizza l'architettura di distribuzione e specifica i requisiti dell'ambiente, offrendo una risorsa completa per le organizzazioni che intendono utilizzare EK per le proprie applicazioni BPA containerizzate.

Parole chiave

Amazon EK, Kubernetes, AWS, RDS, MongoDB Atlas, DevOps, cloud computing, Business Process Automation.

Introduzione

BPA



Nell'odierna era digitale, le aziende cercano di semplificare e automatizzare i processi aziendali complessi in una vasta gamma di ambienti IT. L'automazione dei processi aziendali (BPA, Business Process Automation) è emersa come una tecnologia fondamentale che consente alle organizzazioni di migliorare l'efficienza operativa, ridurre gli errori e migliorare l'erogazione dei servizi. BPA introduce diverse innovazioni e miglioramenti chiave volti a migliorare l'automazione dei flussi di lavoro, il provisioning dei servizi e le applicazioni di automazione preconfigurate.

La piattaforma BPA ospita casi di utilizzo e applicazioni aziendali e IT/operative, quali aggiornamenti del sistema operativo, provisioning dei servizi e integrazione nei motori di orchestrazione. I clienti hanno accesso a un ciclo di vita di servizi e funzionalità BPA, tra cui consulenza, implementazione, servizi business critical e supporto alle soluzioni, forniti dagli esperti Cisco, best practice e metodologie comprovate che consentono di automatizzare i processi aziendali e di eliminare i rischi associati ai sistemi.

Queste funzionalità del ciclo di vita possono essere basate su sottoscrizione o personalizzate in base alle singole esigenze. I servizi di implementazione consentono di definire, integrare e implementare strumenti e processi per accelerare l'automazione. Gli esperti Cisco conducono un processo formale per la raccolta di requisiti, progettano e sviluppano storie di utenti basate su processi flessibili e strumenti CID (Continuous Integration and Continuous Delivery) e implementano servizi flessibili con test automatizzati di flussi di lavoro, dispositivi e servizi nuovi o esistenti. Grazie al supporto per le soluzioni, i clienti possono accedere a un supporto centralizzato 24 ore su 24, 7 giorni su 7, con particolare attenzione ai problemi relativi al software, abbinato al supporto multifornitore e open source offerto dal modello software su più livelli di Cisco. Gli esperti di supporto delle soluzioni Cisco aiutano a gestire il caso dalla prima chiamata alla risoluzione finale e fungono da principale punto di contatto per lavorare con più fornitori contemporaneamente. Grazie alla collaborazione con esperti a livello di soluzione, è possibile ridurre fino al 44% i problemi, garantendo la business continuity e un ritorno più rapido sull'investimento BPA.

Caratteristiche tecniche chiave, come il supporto per FMC e i dispositivi Ansible Managed, le esecuzioni parallele con Advanced Queuing Framework (AQF) e la conformità di configurazione estesa per i dispositivi NDFC e FMC, posizionano BPA come una soluzione completa per l'automazione aziendale su larga scala. Grazie alle nuove funzionalità di gestione SD-WAN, all'onboarding dei dispositivi e alla

gestione delle policy firewall, questa versione affronta gli aspetti critici della sicurezza e dell'automazione della rete, soddisfacendo le esigenze degli ambienti multivendor su larga scala.

EK

Amazon Elastic Kubernetes Service (EKS) è un servizio Kubernetes completamente gestito fornito da Amazon Web Services (AWS). Lanciato nel 2018, EKS semplifica il processo di distribuzione, gestione e scalabilità delle applicazioni containerizzate utilizzando Kubernetes, una piattaforma di orchestrazione di contenitori open-source. EKS riassume le complessità della gestione dei cluster Kubernetes, consentendo agli sviluppatori di concentrarsi sulla creazione e l'esecuzione di applicazioni senza la necessità di gestire l'infrastruttura sottostante.

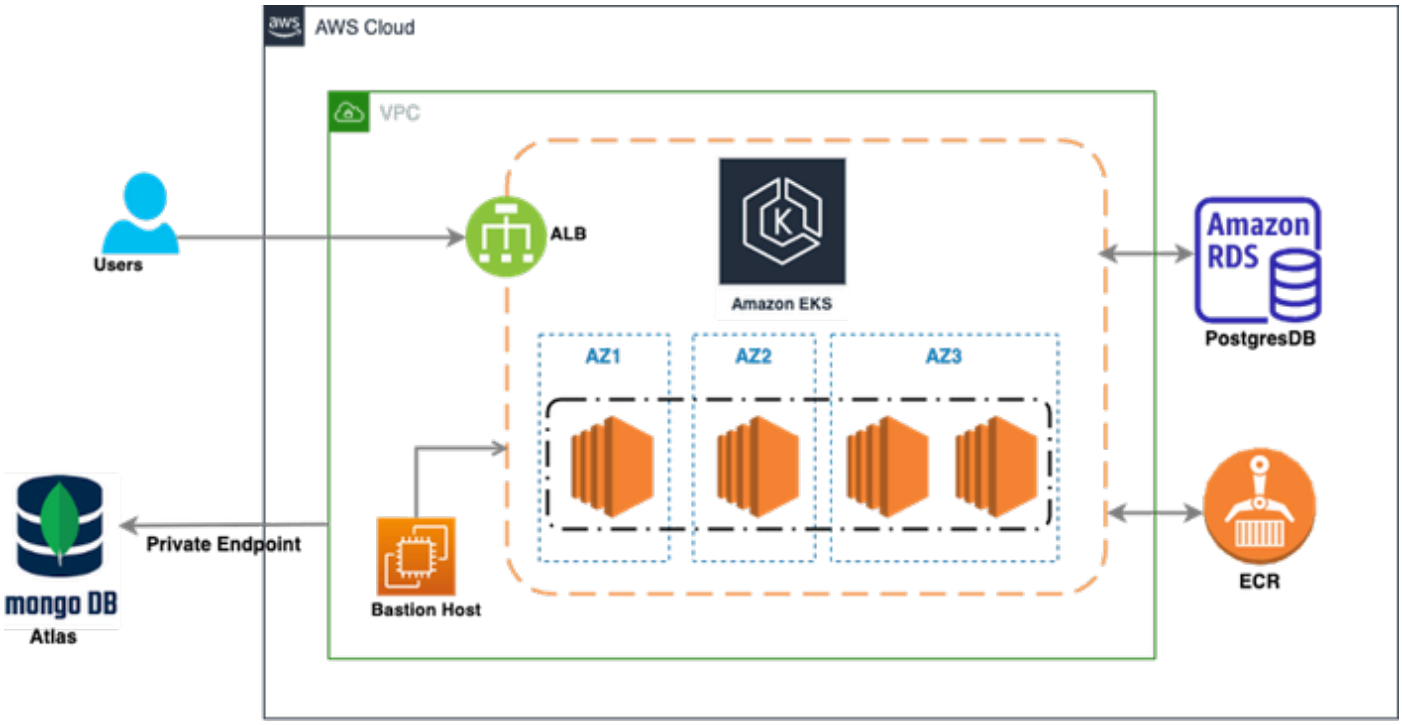
Vantaggi dell'utilizzo di Amazon EK per la distribuzione delle applicazioni

Amazon EK offre diversi vantaggi per l'installazione delle applicazioni, rendendola una scelta diffusa per le organizzazioni che utilizzano applicazioni e microservizi containerizzati.

I vantaggi principali includono:

- **Control Plane Kubernetes gestito**: EK gestisce l'installazione, la scalabilità e la manutenzione del control plane Kubernetes, riducendo il carico operativo.
- **Gestione semplificata dei cluster**: EK astrae le complessità della configurazione e della gestione dei cluster Kubernetes.
- **Scalabilità**: il protocollo EK consente di scalare facilmente i cluster per gestire carichi di lavoro in aumento.
- **Alta disponibilità**: EK supporta le installazioni di più zone di disponibilità, migliorando la disponibilità e la tolleranza di errore.
- **Integrazione con AWS Services**: EKS si integra perfettamente con vari servizi AWS.
- **DevOps Automation**: EKS supporta l'integrazione continua e l'installazione continua (CI/CD) per applicazioni containerizzate.

Architettura di distribuzione BPA



Questa immagine rappresenta un'architettura di alto livello di un'infrastruttura basata su cloud installata in **AWS**, che utilizza diversi componenti chiave. Ecco una scomposizione del diagramma:

1. **Amazon EKS (Elastic Kubernetes Service)**: al centro del diagramma, Amazon EKS è distribuito in tre zone di disponibilità (AZ1, AZ2, AZ3), con nodi di lavoro Kubernetes all'interno di ciascuna zona. Ciò indica un'impostazione a elevata disponibilità e tolleranza di errore, in quanto i carichi di lavoro sono distribuiti su più zone di disponibilità.

2. **ALB (Application Load Balancer)**: si trova nella parte anteriore, riceve il traffico dagli utenti e lo distribuisce attraverso il cluster EKS per la gestione dei carichi di lavoro delle applicazioni. Il bilanciamento del carico garantisce che le richieste siano distribuite in modo uniforme e in grado di gestire la scalabilità in base alla domanda di traffico.

3. **Amazon RDS (Relational Database Service)** - PostgreSQL: sul lato destro del diagramma è presente un'istanza Amazon RDS che esegue PostgreSQL. È possibile accedere a questo database da applicazioni in esecuzione all'interno del cluster EKS.

4. **ECR (Elastic Container Registry)**: è il luogo in cui vengono archiviate e gestite le immagini dei contenitori Docker, che vengono quindi distribuite in Amazon EK per l'esecuzione dei carichi di lavoro.

5. **MongoDB Atlas**: sul lato sinistro, MongoDB Atlas è integrato nell'architettura tramite un endpoint privato. MongoDB Atlas è un servizio di database NoSQL ospitato nel cloud, utilizzato qui per gestire i requisiti di database basati su documenti. L'endpoint privato garantisce una comunicazione protetta e privata tra l'istanza di MongoDB Atlas e altri componenti AWS.

6. **Bastion Host**: posizionato all'interno del VPC (Virtual Private Cloud), Bastion Host fornisce un punto di ingresso sicuro per gli amministratori per accedere alle risorse all'interno del VPC senza esporle direttamente a Internet.

Nel complesso, questa architettura fornisce una soluzione altamente disponibile, scalabile e sicura per l'installazione e la gestione di applicazioni containerizzate mediante Amazon EK, con supporto per database relazionali (PostgreSQL) e NoSQL (MongoDB).

- **Configurazione cluster EKS**
  Per creare un cluster di Amazon EKS utilizzando AWS CLI, è possibile utilizzare l'utilità da riga di comando `eksctl`. Questo è un comando di esempio:

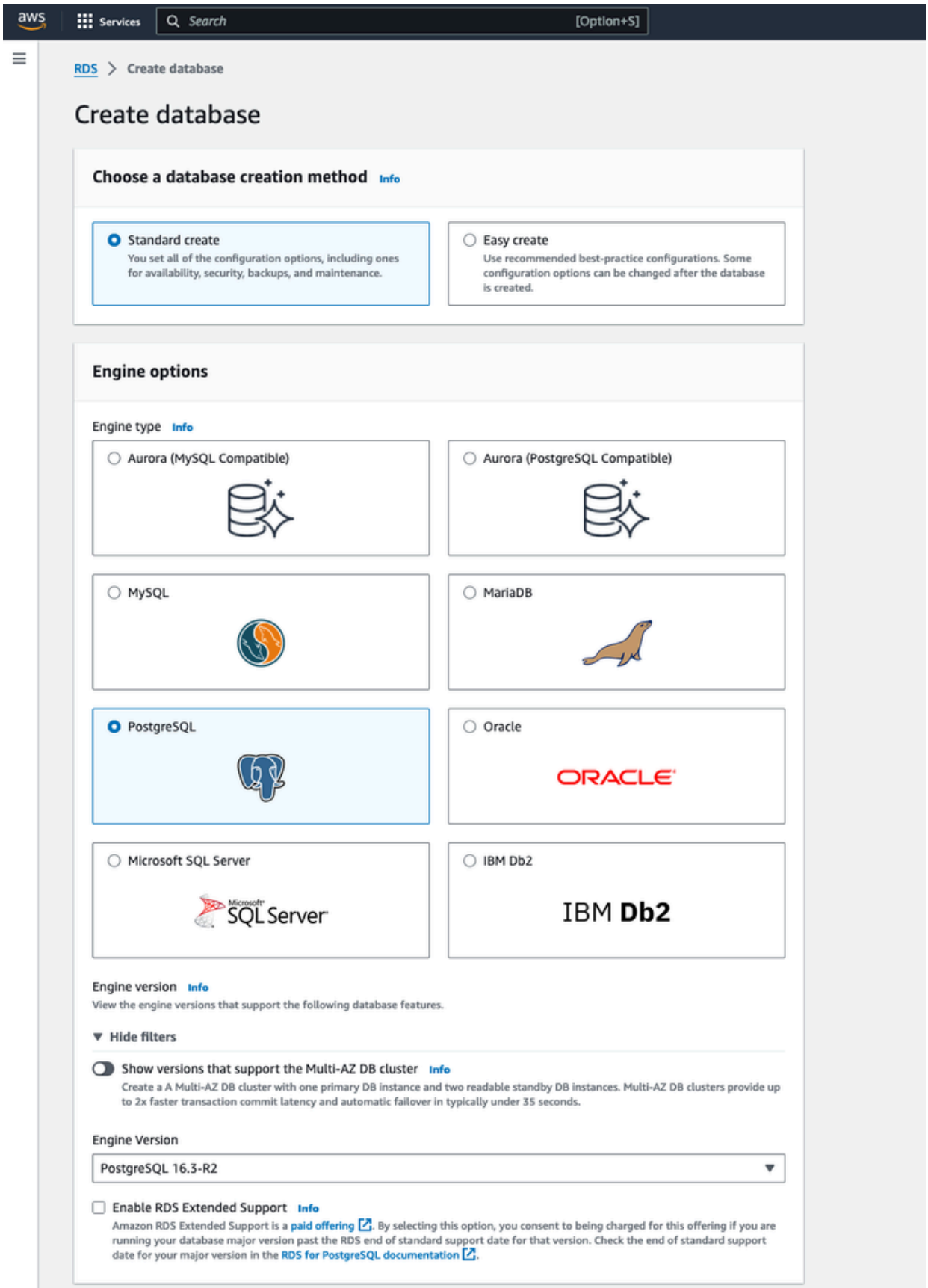
```
eksctl create cluster \
  --name
```

```
\ --region us-west-2 \ --nodegroup-name standard-workers \ --node-type t3.medium \ --node
```

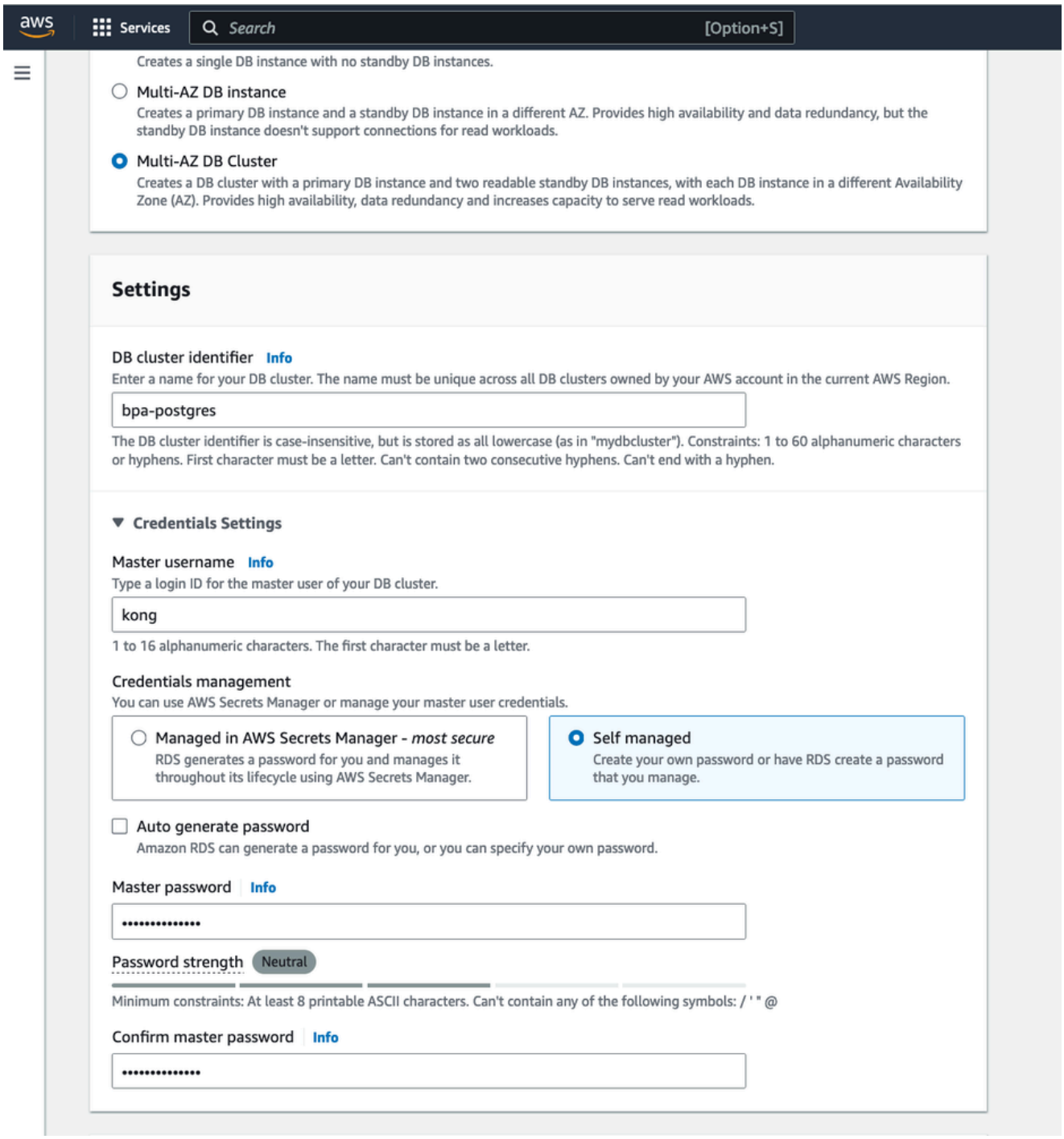
- **Impostazione database RDS**

  La distribuzione di un database relazionale su Amazon RDS prevede i seguenti passaggi:

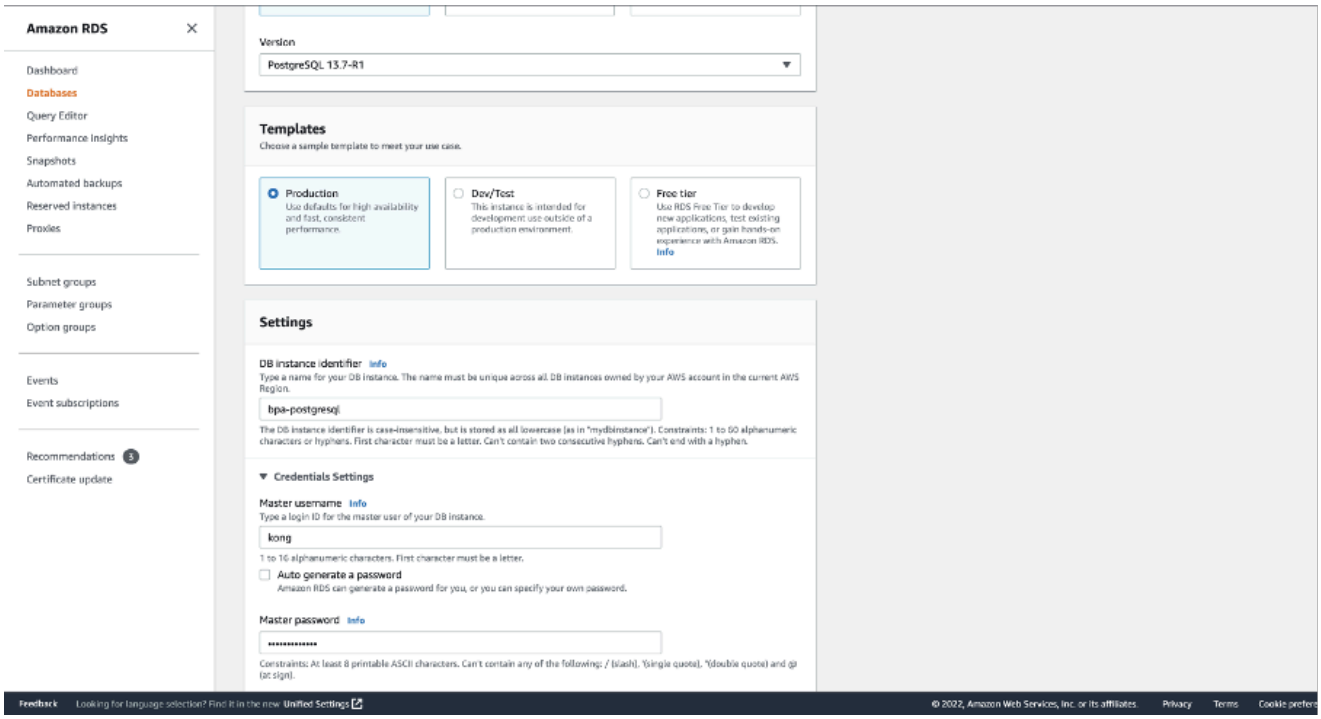
  - Accedere a AWS Management Console e selezionare il servizio Amazon RDS.
  - Creare una nuova istanza di database con le specifiche desiderate.
  - Configurare il gruppo di sicurezza in modo da consentire le connessioni in ingresso dal cluster EK Amazon.



Dal menu a discesa, selezionare la versione più recente di PostgreSQL. Nel nostro caso, è "PostgreSQL 16.3-R1".



Assegnare un nome all'istanza di database e creare un nome utente e una password.

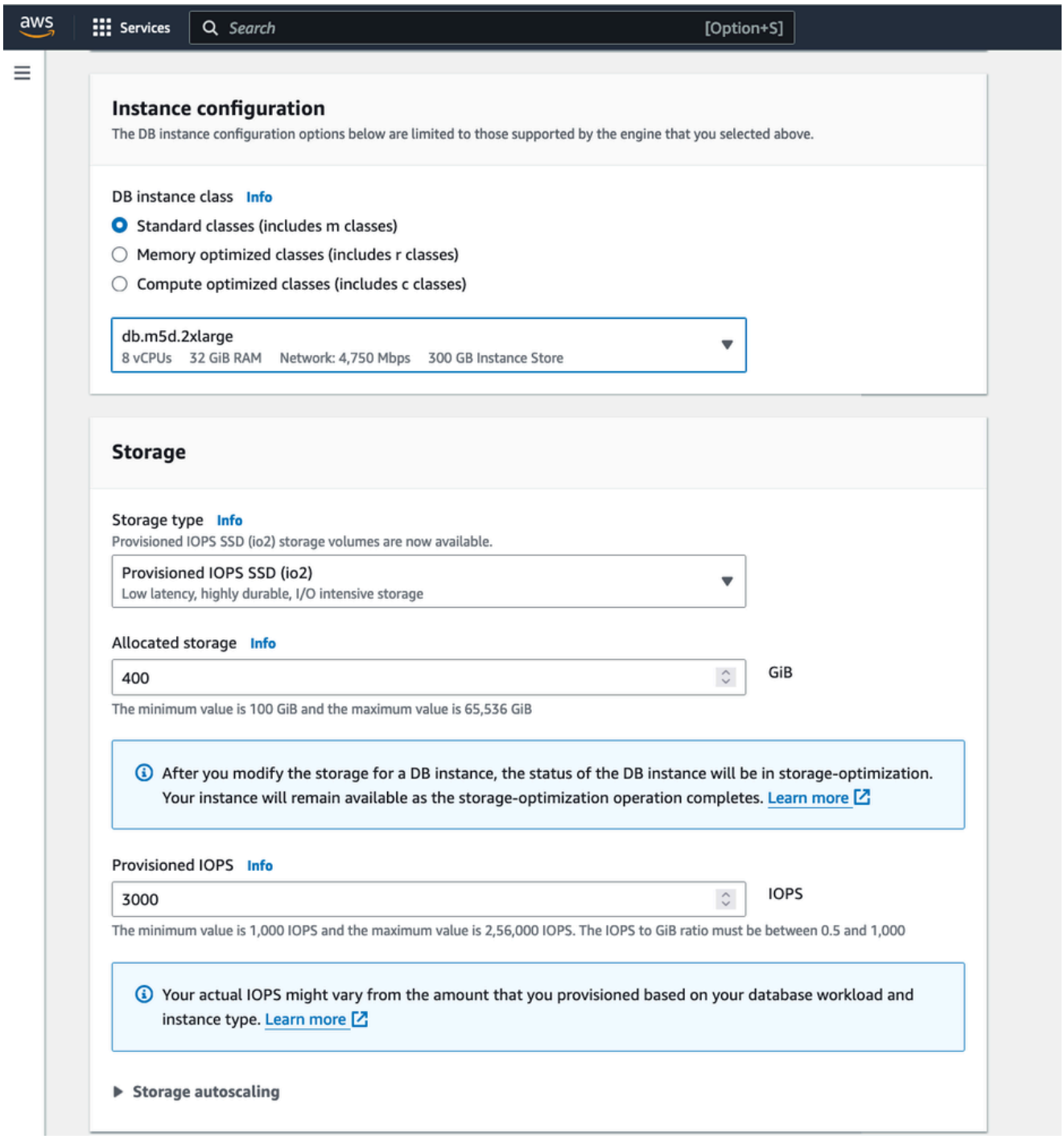


Accertarsi che siano selezionate le impostazioni predefinite per "Dimensione istanza database" e "Archiviazione".

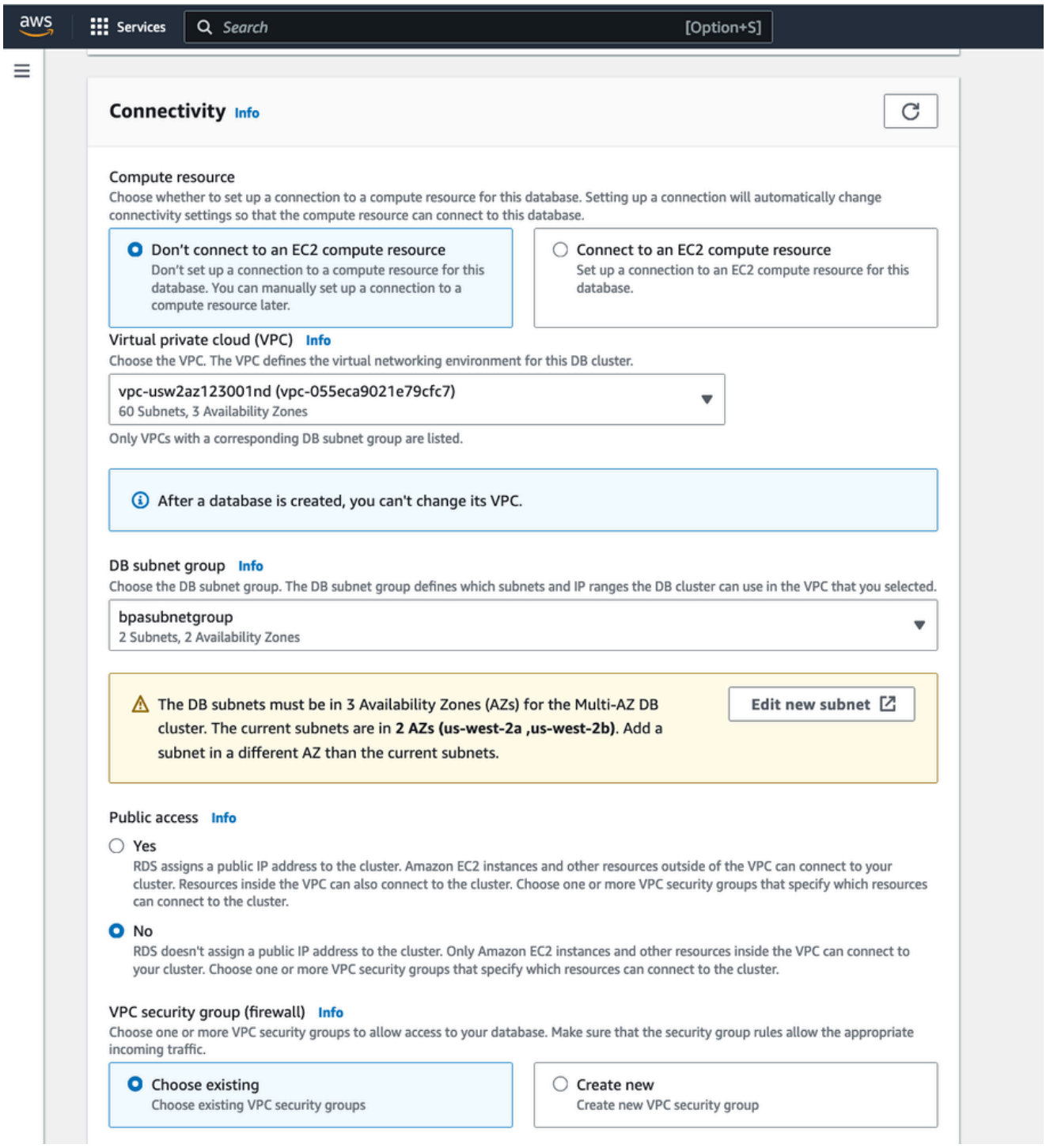
In base alle dimensioni del cluster e ai requisiti dei dati, selezionare le dimensioni e il tipo di archiviazione appropriati per l'istanza di database.

In base al nostro caso di utilizzo, abbiamo scelto la seguente configurazione:

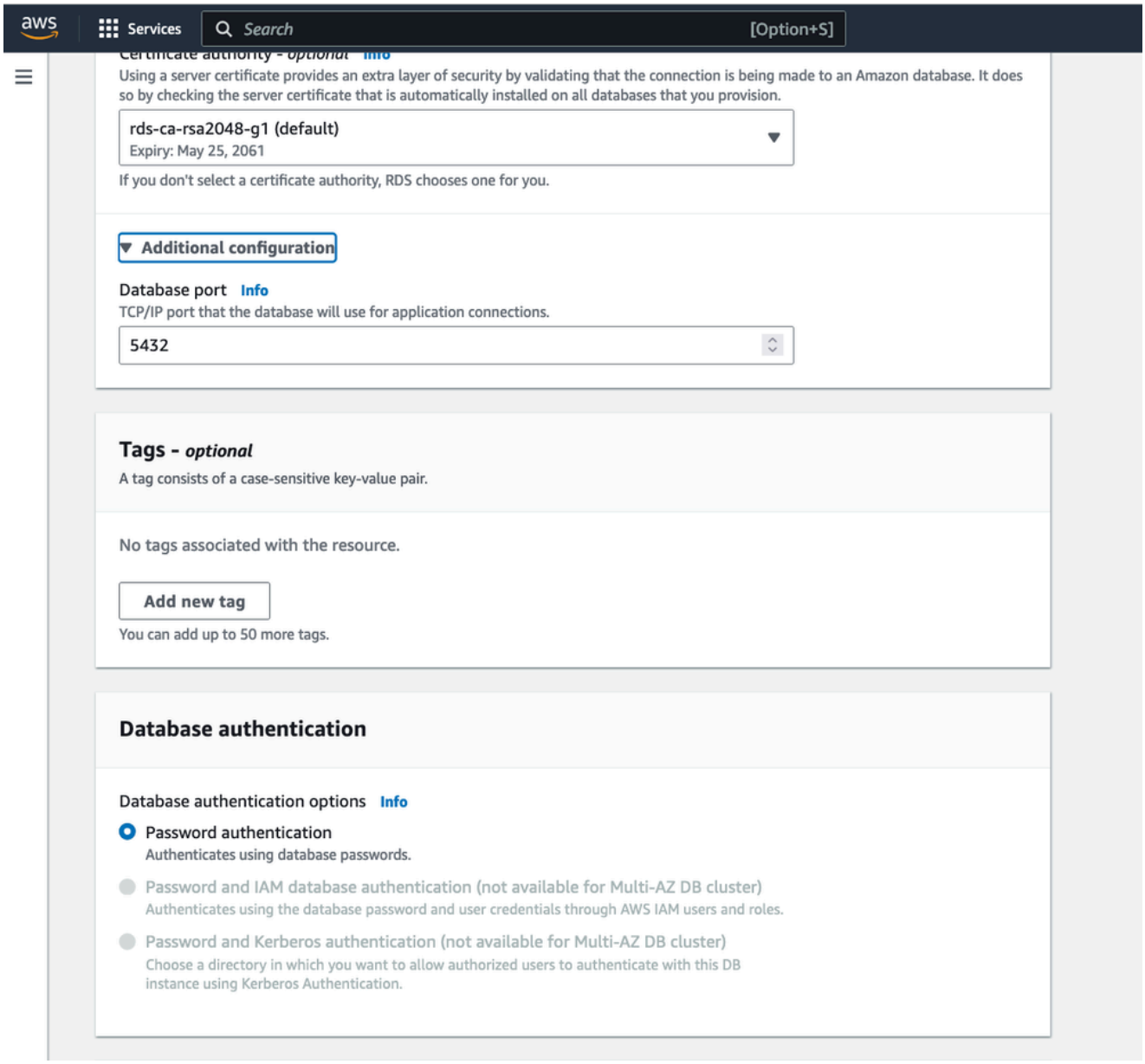
- **Dimensioni istanza database**: db.m5d.2xlarge
  - 8 vCPU
  - 32 GB di RAM
  - Rete: 4.750 Mbps
  - Archivio istanza da 300 GB

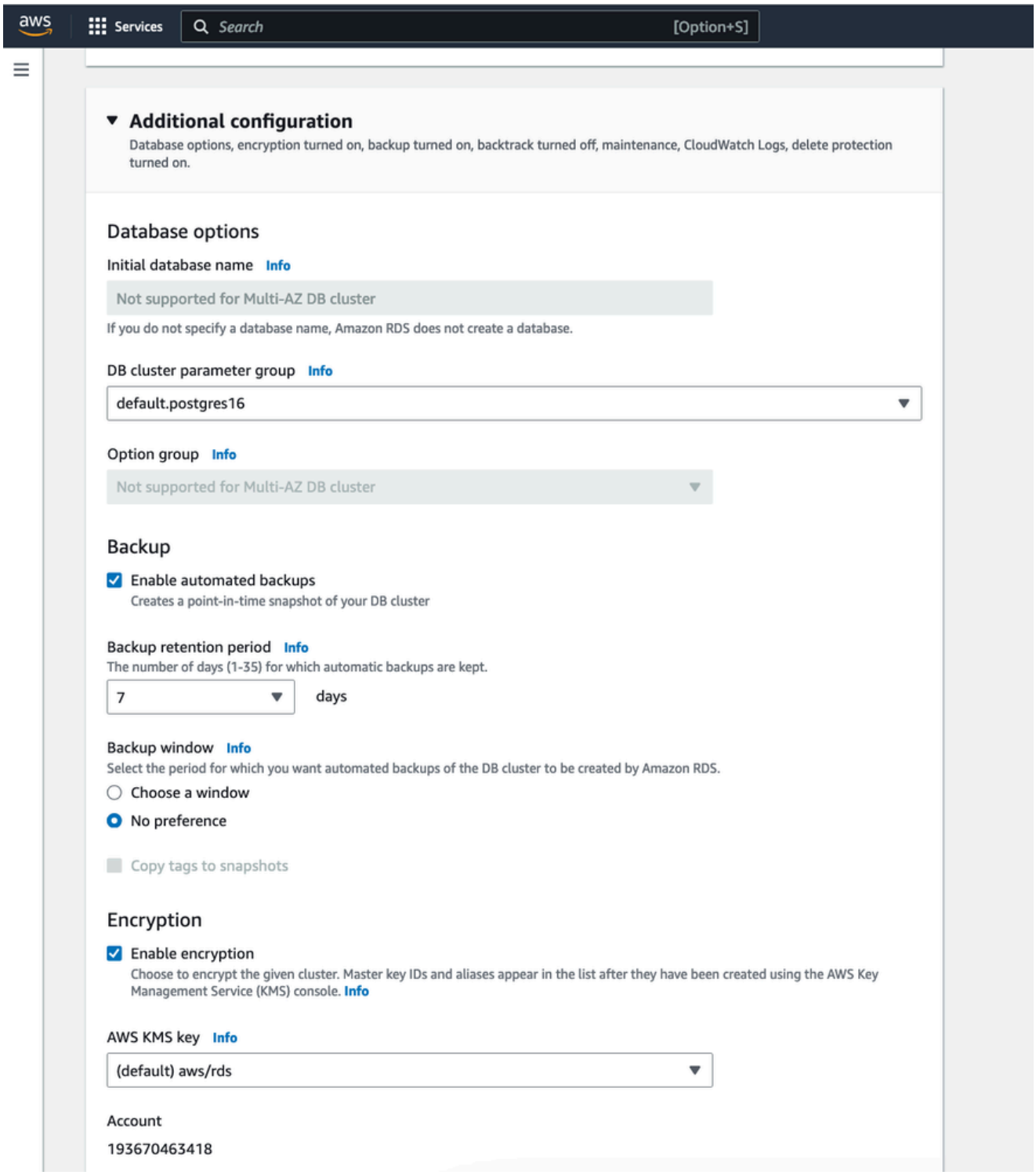


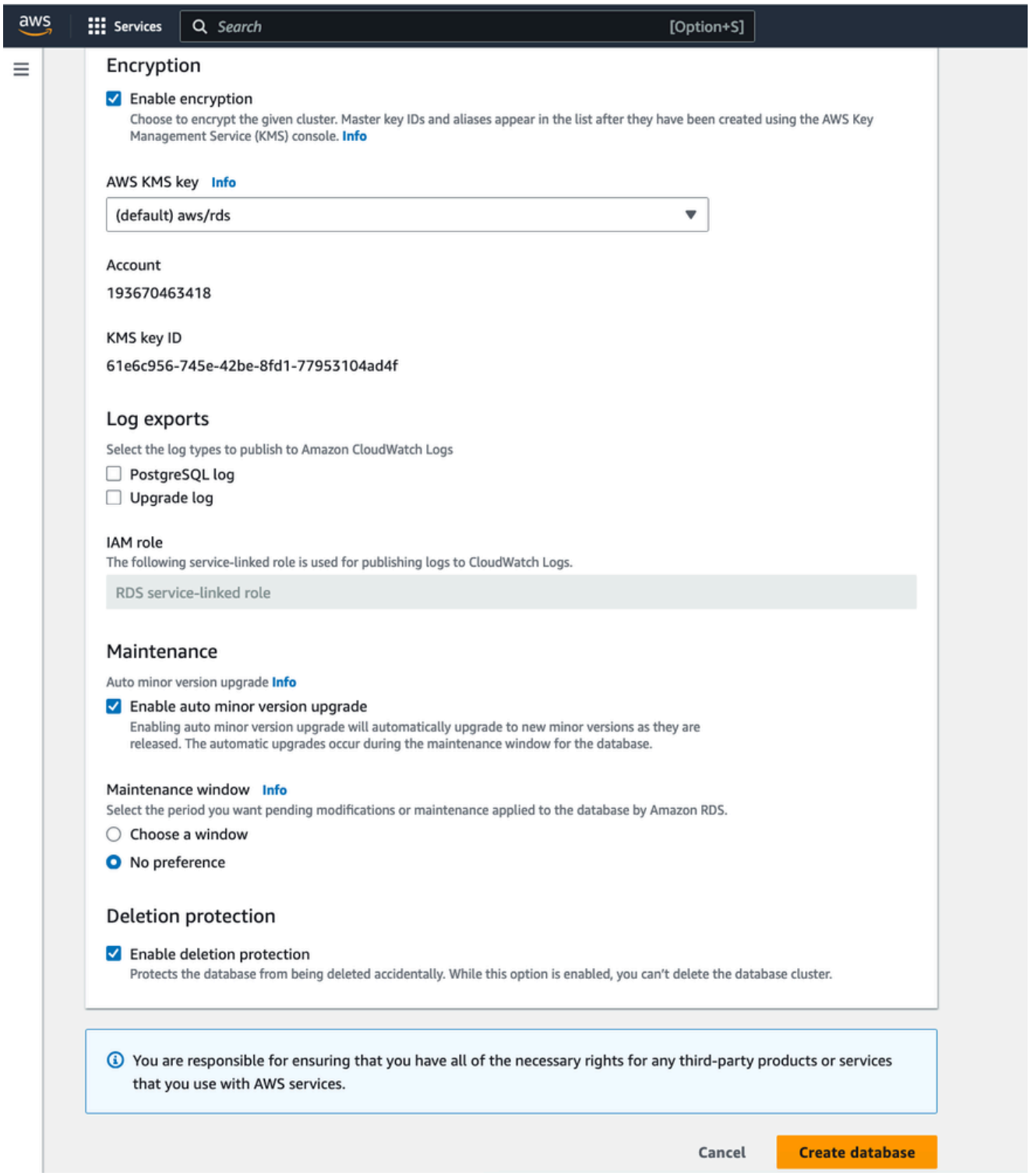
Selezionare i valori appropriati in base allo Use Case. Sono stati selezionati i valori predefiniti.



Assicurarsi che in "Autenticazione database" sia stata selezionata l'opzione Autenticazione password. Esegue l'autenticazione utilizzando le password del database.


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.

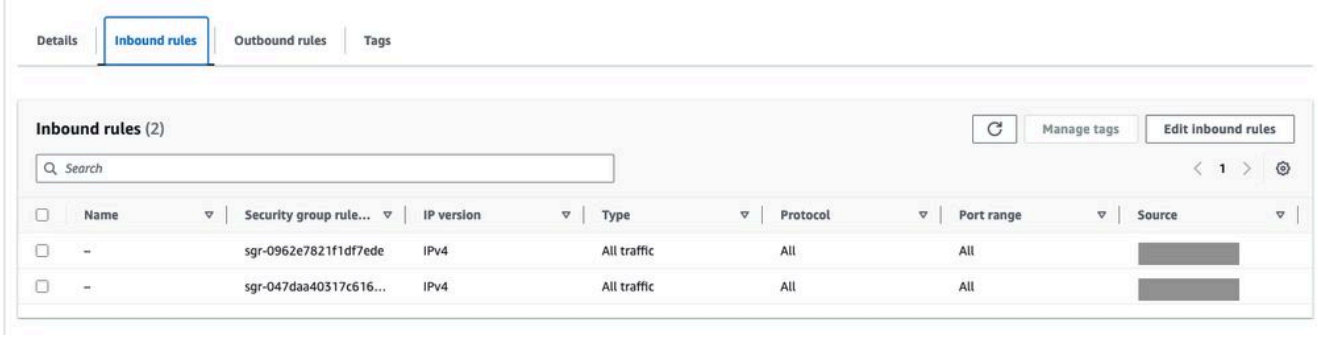

aws
Services
Search
[Option+S]
Additional configuration
Database options, encryption turned on, backup turned on, backtrack turned off, maintenance, CloudWatch Logs, delete protection turned on.
Database options
Initial database name Info
Not supported for Multi-AZ DB cluster
If you do not specify a 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
193670463418



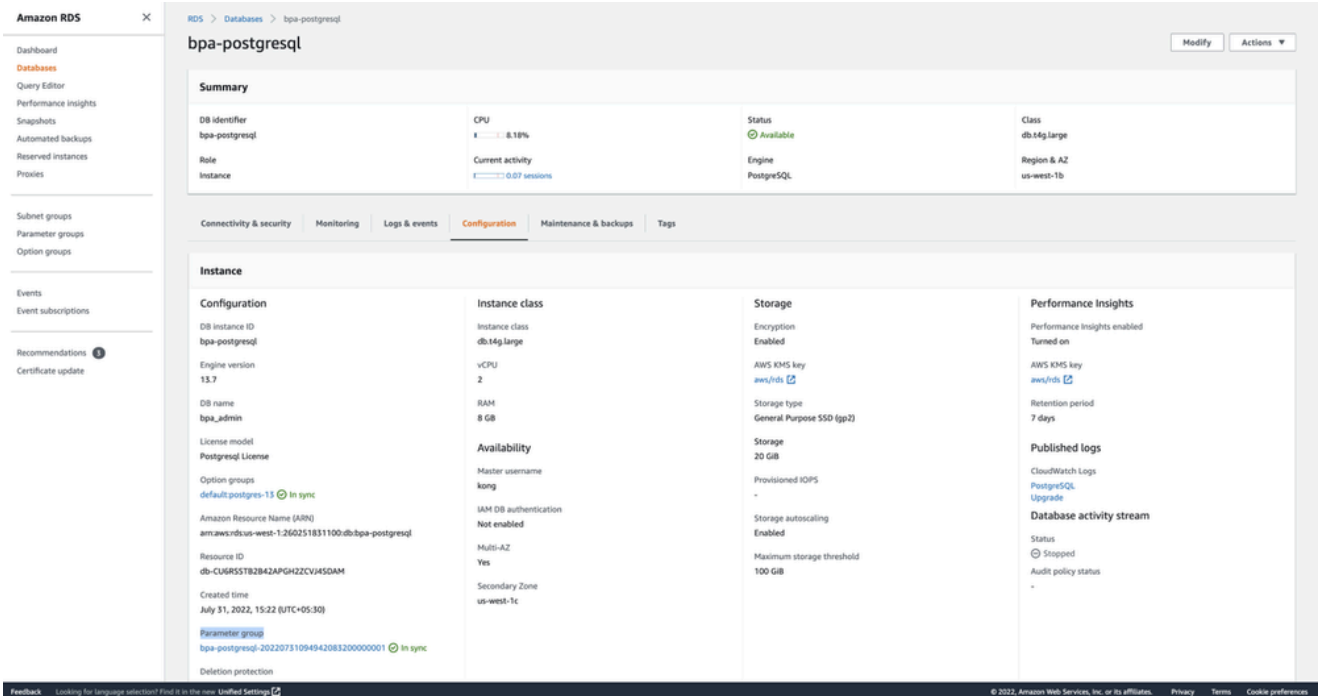
Una volta verificato, è possibile creare il database. Tornate al dashboard Amazon RDS. Confermate che l'istanza sia disponibile per l'uso.

Regole gruppo di sicurezza

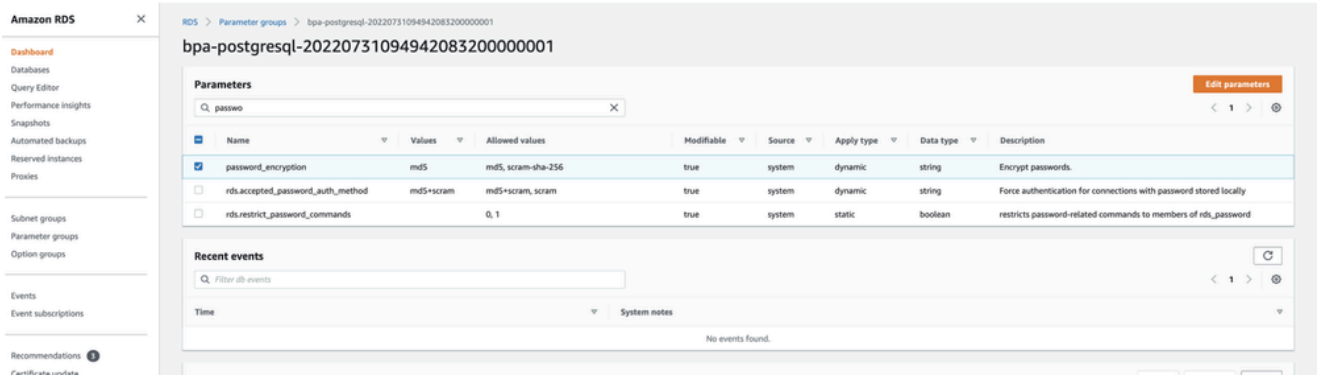
Aggiornare il gruppo di sicurezza in ingresso con il blocco CIDR del pod e il blocco CIDR del nodo.



In RDS -> Database -> DB-NAME, fare clic su configuration e fare riferimento alla sezione Gruppo di parametri, quindi fare clic sul gruppo di parametri da visualizzare.



Cercare "password_encryption" e modificare il valore in md5 da blank / other value. Questa operazione è necessaria per il funzionamento delle configurazioni della camunda.



Creare questi database insieme agli utenti connettendosi a RDS.

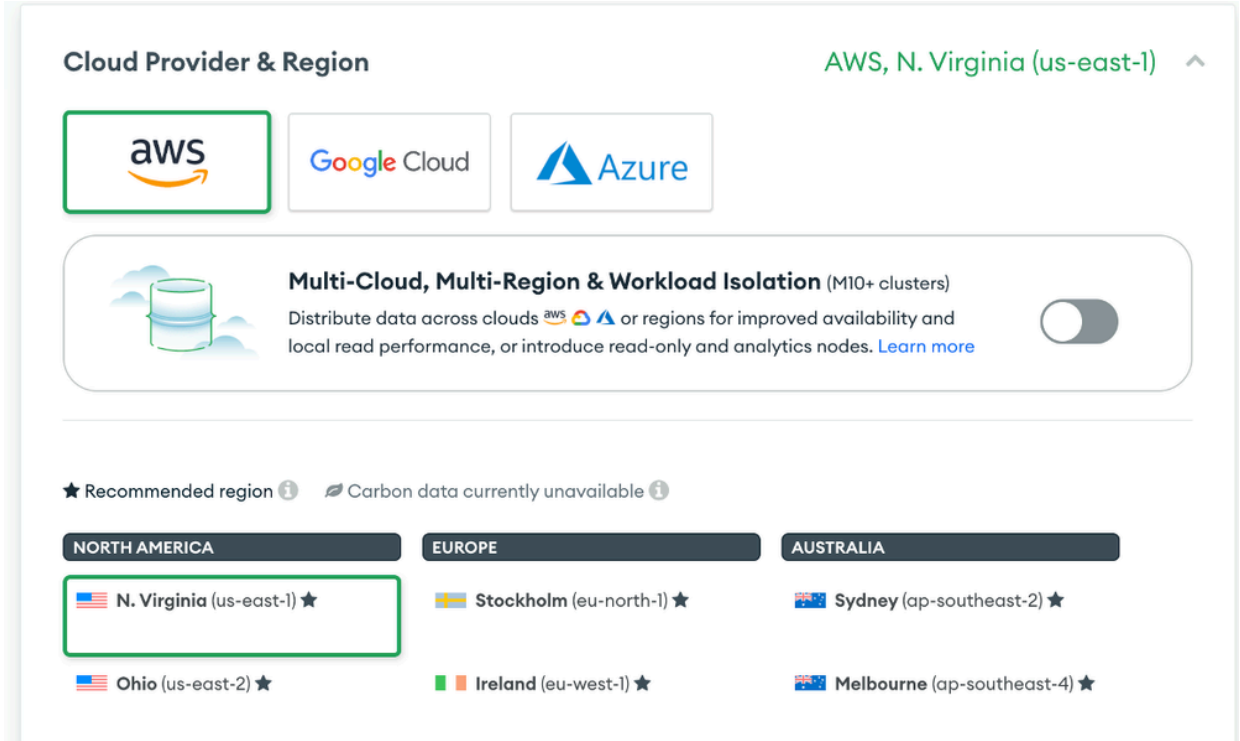
```
PG_ROOT_DATABASE=admin
PG_INITDB_ROOT_USERNAME=admin
PG_INITDB_ROOT_PASSWORD=Bp@Chang3d!
AUTH_DB_NAME=kong
AUTH_DB_USER=kong
AUTH_DB_PASSWORD=K@ngPwdCha*g3
WFE_DB_USER=camunda
WFE_DB_PASSWORD=WOrkFlo#ChangeNow
WFE_DB_NAME=process-engine
```

- Autenticazione password

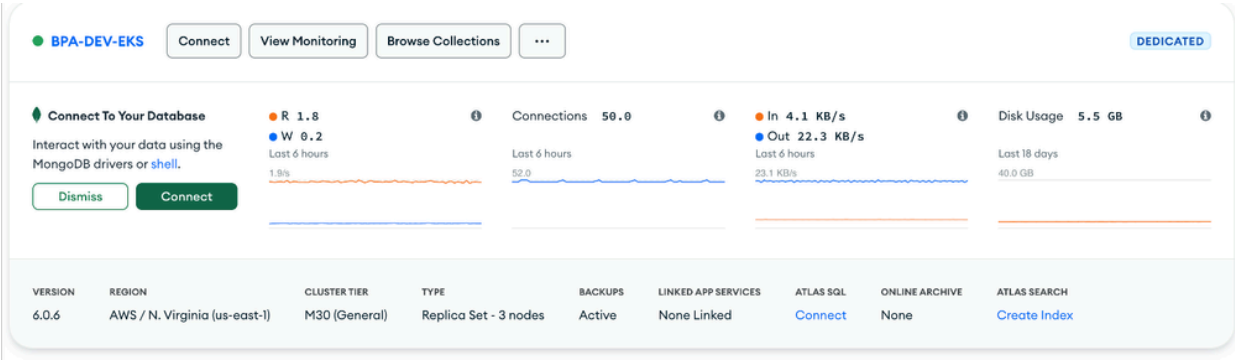
  Esegue l'autenticazione utilizzando le password del database.

- **Installazione di Atlas MongoDB**
  La creazione di Atlas MongoDB implica:
  - **Accesso ad Atlas MongoDB.**
  - **Selezione dell'organizzazione e del progetto.**
  - **Creazione di un cluster dedicato con le specifiche appropriate.**

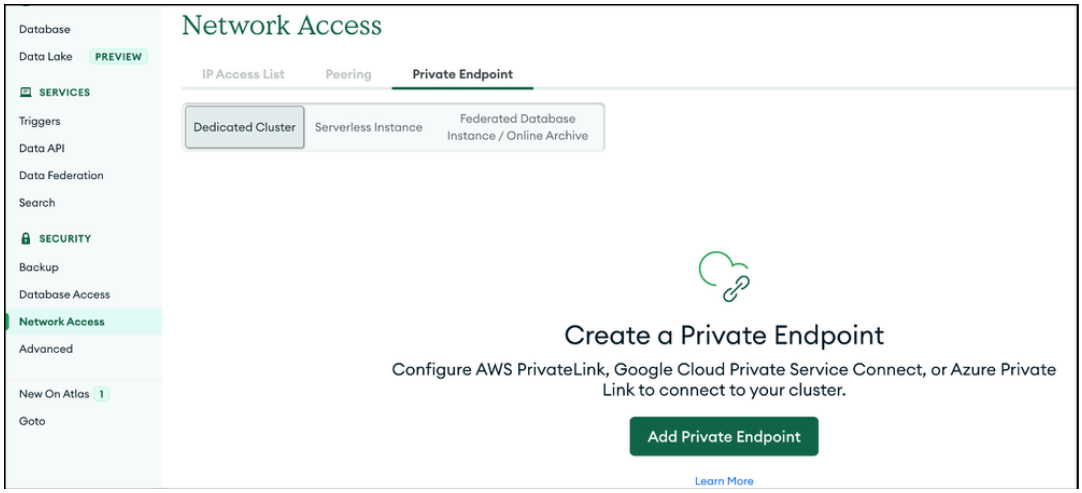
  - **Selezionare il livello dedicato, Cloud Provider & Region.**



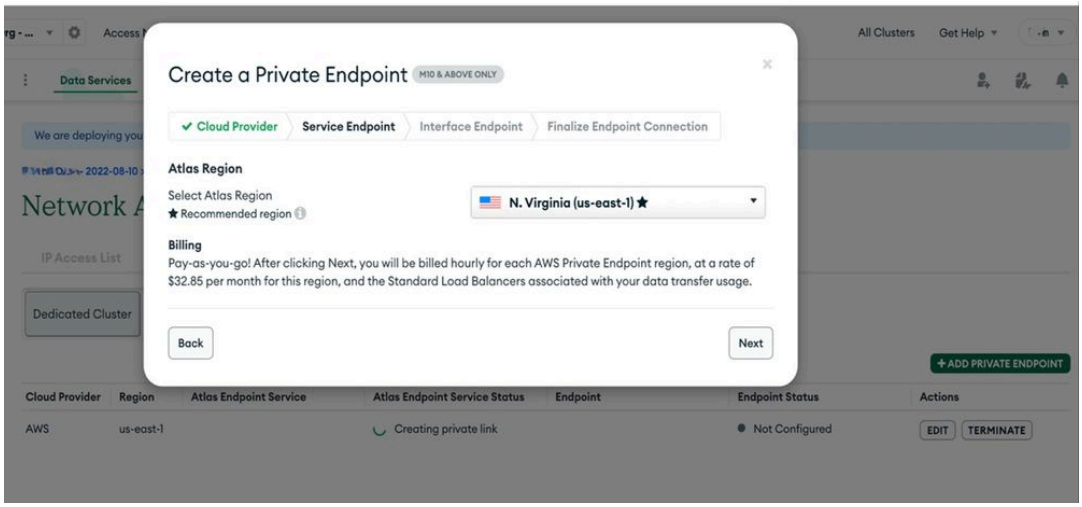
- **Selezionare il cluster dedicato di livello appropriato (M30 è stato utilizzato come livello), fornire il nome del cluster appropriato e fare clic su Crea cluster. Verrà inizializzato il cluster monogodb Atlas.**



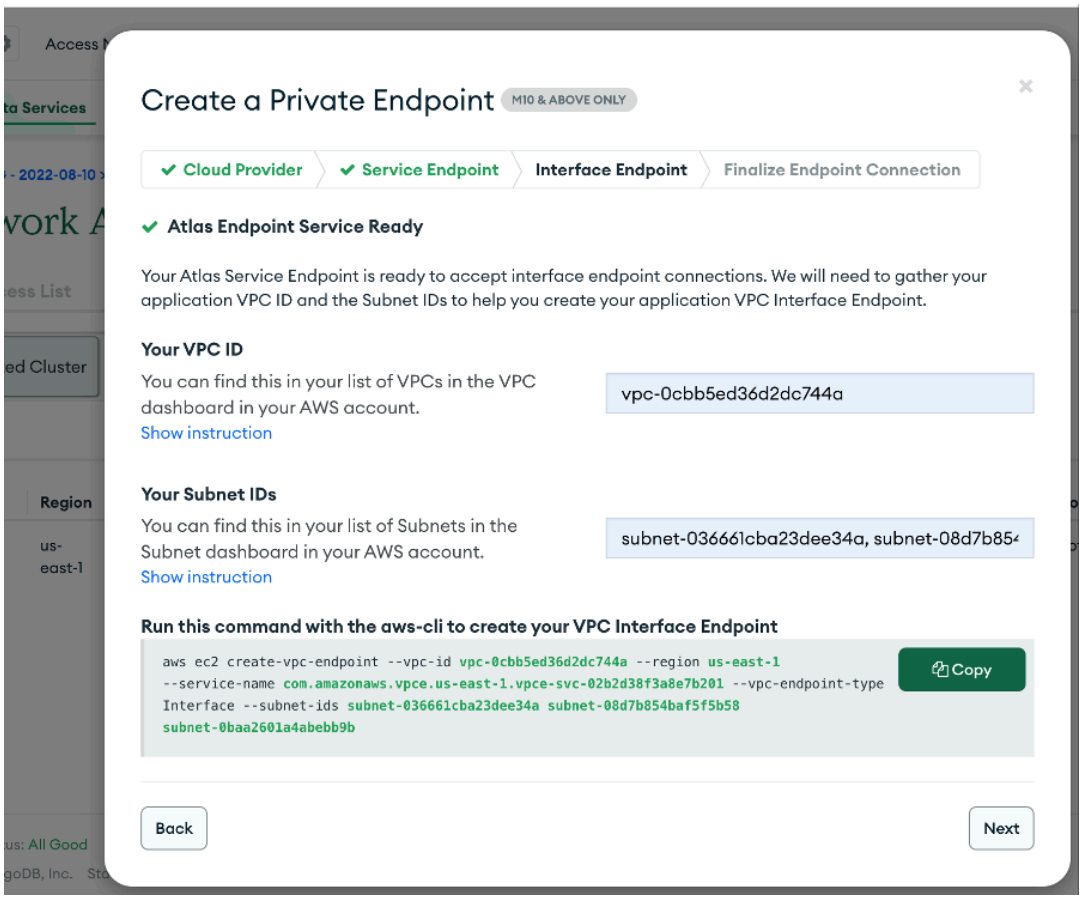
- **Configurazione dell'endpoint privato VPC per il cluster Atlas e K8S.**
  - **Fare clic su Accesso alla rete, selezionare Endpoint privato, quindi fare clic su Aggiungi endpoint privato.**



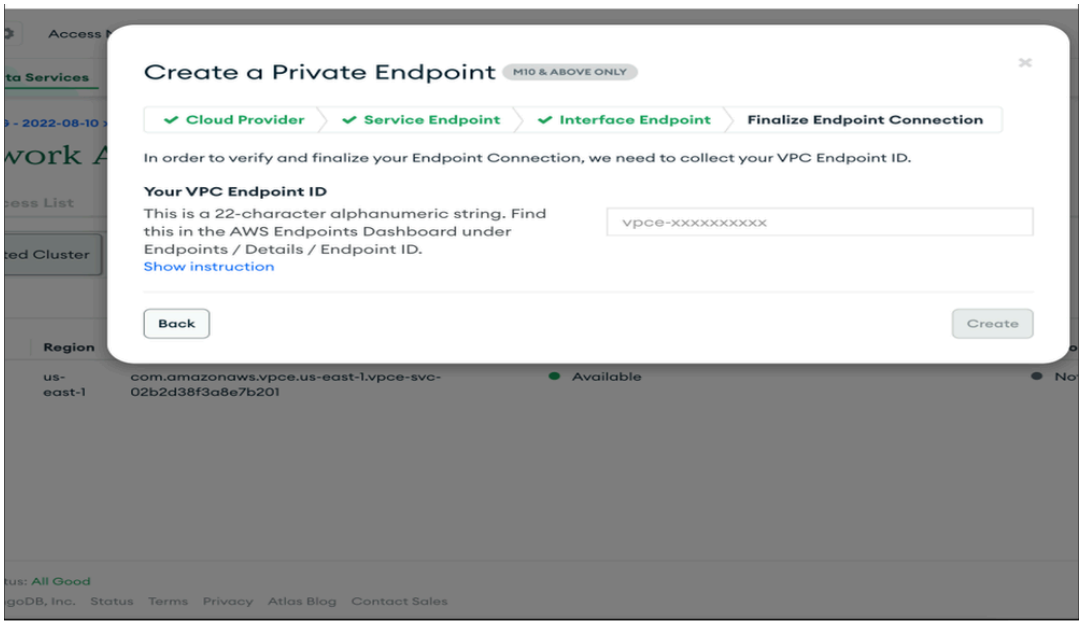
  - **Selezionare Cloud Provider come AWS, selezionare la rispettiva area e fare clic su Avanti.**



- **Fornire gli ID PVC e subnet corrispondenti. Una volta immessi i dettagli, copiare il comando vpc end point creation ed eseguirlo nella console di aws. L'ID dell'endpoint vpc verrà restituito come output.**

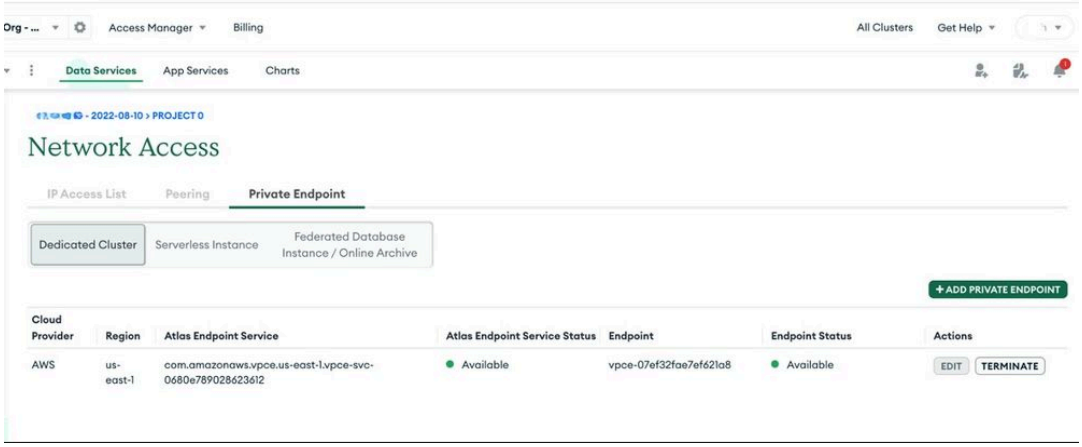


- **Fare clic su Next per incollare l'ID dell'endpoint VPC e fare clic su Create.**

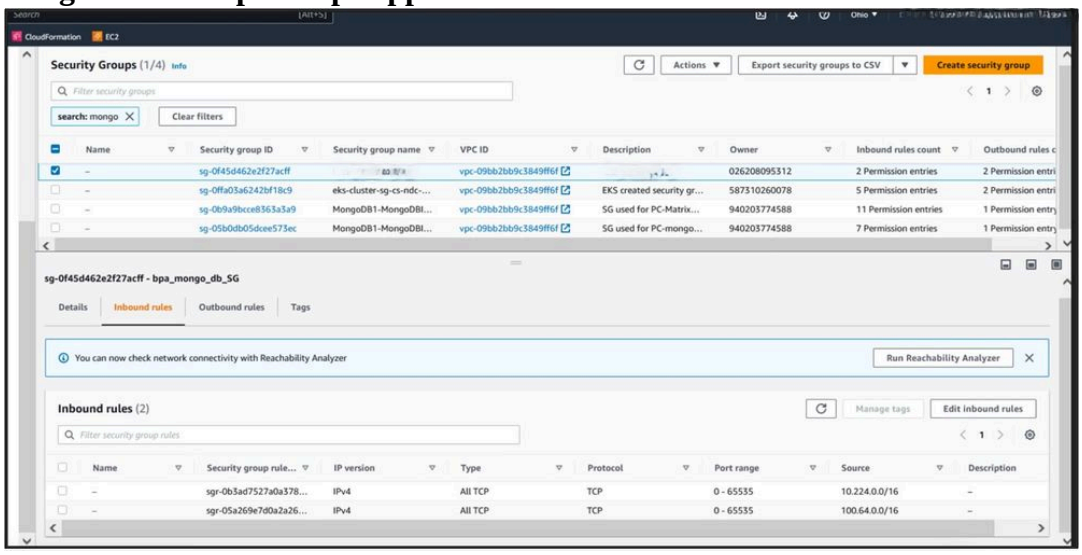


- **Una volta creato, lo stato dell'endpoint sarà Disponibile, come mostrato**

nell'immagine seguente. È necessario creare l'endpoint VPC per il cidr pod. Nel nostro caso abbiamo usato "100.64.0.0/16" .



- **Aggiungere le regole in entrata all'endpoint vpc appena creato. L'endpoint vpc verrà incluso nell'account padre e un gruppo di sicurezza deve essere assegnato all'endpoint vpc appena creato.**



ECR come registro immagini

La creazione di repository di Amazon ECR e il trasferimento di immagini Docker in tali repository richiede diversi passaggi. Di seguito vengono illustrati i passaggi per creare un repository ECR, assegnare tag a un'immagine Docker e spostarla nel repository utilizzando la CLI di AWS.

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

Sostituisci:

- **nome-immagine-**con il nome desiderato per il repository ECR.

- **la tua** regione con la tua regione AWS

Configura ruolo IAM per nodi EK

Verificare che ai nodi di lavoro EK (istanze EC2) sia associato il ruolo IAM necessario con le autorizzazioni per eseguire il pull di immagini da ECR. I criteri IAM richiesti sono:

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

Allegare questo criterio al ruolo IAM associato ai nodi di lavoro EKS.

Distribuzione BPA
L'implementazione di BPA comporta diversi passaggi, tra cui l'etichettatura dei nodi di lavoro EK, la preparazione delle directory sui nodi, la copia dei pacchetti BPA e la distribuzione di BPA utilizzando Helm.

Per l'installazione presso i clienti, sono state utilizzate le seguenti versioni di software e servizi cloud:

- **BPA:** 4,0,3-6
- **RDS (Relational Database Service)**: 16.3-R2
- **Atlante MongoDB**: v5.0.29
- **EK (Elastic Kubernetes Service)**: v1.27

Questi componenti garantiscono che l'implementazione sia solida, scalabile e in grado di gestire in modo efficiente i carichi di lavoro richiesti.

- **Assegnazione di etichette ai nodi di lavoro EK**

```
kubectl label node
```

```
name=node-1 kubectl label node
```

```
name=node-2 kubectl label node
```

```
name=node-3 kubectl label node
```

```
name=node-4
```

- **Preparazione delle directory sui nodi**
  Nodo 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
```

Nodo 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
```

Nodo 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
```

Nodo 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
```

- Copia di pacchetti BPA

```
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/
```

- Distribuzione di BPA mediante Helm

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

Configurazione in ingresso

- **Abilitazione in ingresso**
  Aggiorna **`valori.`**yamlper abilitare l'ingresso:

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

- **Creazione di un segreto mediante un certificato BPA**
  Passare alla directory dei certificati e creare un segreto:

```
cd /opt/bpa/
```

```
/bpa/conf/common/certs/ kubectl create secret tls bpa-certificate-ingress --cert=bap-cert
```

- **Aggiornamento del controller in ingresso**
  Aggiungere il nuovo segreto creato nel **`controller in ingresso.yaml`** file:

```
cd /opt/bpa/
```

```
/templates/ vi ingress-controller.yaml "- --default-ssl-certificate=$(POD_NAMESPACE)/bpa-
```

- **Aggiornamento del certificato in ingresso**
  Eseguire l'eliminazione e l'installazione del timone per aggiornare il certificato in entrata.

Specifiche dell'ambiente
Le specifiche dell'ambiente includono i requisiti per le istanze EC2, i load balancer, gli endpoint VPC e le istanze RDS. Le specifiche principali sono:

Requisiti EC2:

Requisiti di storage: 2 TB di spazio per nodi. Montare il volume EBS su /opt e aggiungere una voce in /etc/fstab per tutti i nodi.

Gruppo di sicurezza in entrata: 30101, 443, 0 - 65535 TCP, 22 per ssh.

Gruppo di sicurezza in uscita: tutto il traffico deve essere abilitato.

Sistema di risoluzione DNS: EC2 deve disporre di resolver locali in /etc/resolve.conf.

Requisiti del servizio di bilanciamento del carico:

- Le porte dei listener devono essere 443, 30101.
- Requisiti degli endpoint VPC (Atlas MongoDB).
- Gli endpoint VPC creati per la connettività Atlas sono disponibili nell'account padre (aws-5g-ndc-prod). L'endpoint VPC deve avere un gruppo di sicurezza che consenta tutti gli accessi in entrata (0 - 65535).

Requisiti RDS:

Tipo RDS: db.r5b.2xlarge

Versione motore Postgres: 13.7

Gruppo di sicurezza: Inbound deve consentire il traffico dall'origine CIDR POD.

Concetti e componenti principali
Comprendere i fondamenti di Kubernetes è essenziale per implementare e gestire in modo efficace le applicazioni che utilizzano Amazon EK.

Conclusioni
Questo documento fornisce una guida dettagliata per l'installazione e la gestione delle applicazioni BPA (Business Process Automation) utilizzando Amazon EK. Seguendo le fasi descritte e comprendendo i concetti chiave, le organizzazioni possono sfruttare i vantaggi di EKS per le loro applicazioni BPA containerizzate.

Riferimenti

- **Servizi Web Amazon, "Documentazione di Amazon EKS" [Online]. Disponibile:https://docs.aws.amazon.com/eks/**
- **Kubernetes, "Documentazione di Kubernetes" [Online]. Disponibile:https://kubernetes.io/docs/home/**
- **Cisco BPA in breve https://www.cisco.com/c/en/us/solutions/collateral/service-provider/at-a-glance-c45-742579.html**
- **Guida operativa BPA https://www.cisco.com/c/dam/en/us/support/docs/bpa/v403/cisco-bpa-operations-guide-v403.pdf**
- **Guida per gli sviluppatori BPA https://www.cisco.com/c/dam/en/us/support/docs/bpa/v403/cisco-bpa-developer-guide-v403.pdf**

Informazioni su questa traduzione

Cisco ha tradotto questo documento utilizzando una combinazione di tecnologie automatiche e umane per offrire ai nostri utenti in tutto il mondo contenuti di supporto nella propria lingua. Si noti che anche la migliore traduzione automatica non sarà mai accurata come quella fornita da un traduttore professionista. Cisco Systems, Inc. non si assume alcuna responsabilità per l’accuratezza di queste traduzioni e consiglia di consultare sempre il documento originale in inglese (disponibile al link fornito).