**Objective:**

Implement VPC peering between EKS clusters and other AWS resources.

**Documentation:**

**1. AWS VPC basics:**

A Virtual Private Cloud (VPC) is a logically isolated section of the AWS Cloud. You can control your environment by selecting your IP address range, creating subnets, and setting up route tables, network gateways, and security settings.

**2. VPC peering benefits and use cases:**

1. Securely communicate between VPCs as if they are on the same network.
2. Share resources between VPCs, like databases and file systems.
3. Enhanced performance compared to internet-based connections.
4. Reduces data transfer costs.
5. Enables decentralized application architecture.

**3. Integrating AWS resources with EKS:**

Amazon EKS is a managed Kubernetes service that lets you run Kubernetes on AWS without managing the control plane's nodes. When you integrate AWS resources with EKS, you leverage AWS's scalability and availability with Kubernetes's flexibility and community support.

**Requirements:**

1. An AWS account with required permissions to create and manage EKS clusters, EC2 instances, and EBS volumes.
2. AWS CLI installed and configured.
3. **kubectl** installed.
4. **eksctl** installed.
5. EBS CSI Driver
6. IAM Cluster Permission
7. Basic AWS VPC Knowledge

**Tasks: 1**

**1. Set up a VPC for RDS:**

1.1. Navigate to the VPC Dashboard in the AWS Management Console.

1.2. Choose “Start VPC Wizard” and select “VPC with a Single Public Subnet”.

1.3. Configure VPC details:

* IP CIDR block: **10.1.0.0/16**
* VPC name: **RDS-VPC**
* Public subnet CIDR: **10.1.1.0/24**
* Availability Zone: Select an appropriate zone.
* Enable DNS hostnames: Yes
* 1.4. Create VPC.

**Tasks: 2**

**2. Peer the VPC with an EKS cluster's VPC:**

2.1. Go to the VPC Dashboard and select “Peering Connections” then “Create Peering Connection”.

2.2. Fill in the following:

* Name tag: **EKS-RDS-Peering**
* Requester VPC (EKS VPC): Select your EKS VPC.
* Accepter VPC (RDS VPC): Select the VPC you just created **(RDS-VPC)**.

2.3. Create Peering Connection.

2.4. Navigate to Route Tables for EKS VPC. Add a route that points to the RDS VPC CIDR block **(10.1.0.0/16)** and set the target as the peering connection.

2.5. Repeat for the RDS VPC, but point the route to the EKS VPC CIDR block and set the target as the peering connection.

**Tasks: 3**

**3. Configure an application in EKS to utilize the RDS instance:**

3.1. Create an RDS instance within the RDS-VPC.

3.2. Ensure that security groups for the RDS instance allow traffic from the EKS VPC.

3.3. Deploy an application in EKS.

3.4. Configure the application to use the RDS endpoint for database operations.

**Deep Dive and Clear Examples:**

* When setting up VPC for services like RDS, you're essentially carving out a secure, isolated network segment in AWS for your database. This ensures that only resources you allow can interact with the database.
* VPC peering is like building a direct road between two different cities (VPCs) without having to pass through any other cities. It's direct and efficient. The route tables act as your GPS directions, telling traffic where to go when it needs to get from one VPC to another.
* When you configure an EKS application to use the RDS instance, you're essentially telling your application where to find the database and how to connect to it. The application will send database requests through the VPC peering connection, directly to the RDS instance.