😊 VPC Sharing (Subnets)

In AWS, VPC (Virtual Private Cloud) sharing is a feature that allows you to share your VPC resources with other AWS accounts within the same AWS organization. This feature simplifies collaboration between different teams or departments within an organization by allowing them to use shared resources while maintaining the isolation of other resources.

With VPC sharing, you can share the following VPC resources:

1. Subnets: You can share one or more subnets within your VPC with other AWS accounts.
2. Route tables: Shared subnets can use route tables from the owner account or from a shared account.
3. Security groups: You can share security groups with other AWS accounts, allowing them to apply the same security policies to their resources.
4. Network access control lists (ACLs): Similar to security groups, you can share network ACLs with other accounts for controlling inbound and outbound traffic.

VPC sharing simplifies network management and resource sharing within complex AWS environments while maintaining security and isolation between different accounts. It's particularly useful in scenarios where multiple teams or departments need to collaborate on AWS resources while having separate AWS accounts.

😊 Resource Access Manager

AWS Resource Access Manager (RAM) is a service provided by Amazon Web Services (AWS) that allows you to securely share your AWS resources with other AWS accounts. This can be particularly useful in scenarios where different teams, departments, or even different organizations need to collaborate and use the same resources without duplicating them.

Key Features of AWS RAM:

1. **Resource Sharing**: AWS RAM lets you share AWS resources across AWS accounts. Resources can be shared with specific AWS accounts, organizational units, or even entire AWS Organizations.

2. **Supported Resources**: You can share various types of resources, including but not limited to Amazon VPC subnets, AWS Transit Gateways, AWS License Manager configurations, and Amazon Route 53 Resolver rules.

3. **Granular Permissions**: You can control which specific resources are shared and with whom, providing fine-grained access control to your AWS resources.

4. **Consistency and Cost Efficiency**: By sharing resources, you can maintain consistency across environments and avoid duplicating resources, potentially reducing costs.

5. **Integration with AWS Organizations**: AWS RAM integrates with AWS Organizations, making it easier to manage and share resources across multiple accounts within an organization.

Use Cases

- **Multi-Account Architectures**: Organizations often use multiple AWS accounts for different departments or teams. AWS RAM allows these accounts to share resources like VPCs, eliminating the need to replicate resources in each account.
- **Centralized Management**: Centralized teams can manage resources like licensing, directory services, or network infrastructure and share these resources with other teams, ensuring compliance and uniformity.
- **Cost Optimization**: By sharing resources rather than duplicating them, organizations can optimize costs, reducing the need for additional resources in separate accounts.

How It Works

1. **Create a Resource Share**: Define a resource share by selecting the resources you want to share and the accounts with which you want to share them.
2. **Add Principals**: Specify the principals (AWS accounts, organizational units, or entire organizations) that will have access to the shared resources.
3. **Accept Invitations**: The recipient accounts must accept the invitation to access the shared resources. Once accepted, they can use the resources according to the permissions granted.

AWS RAM simplifies resource management and sharing in multi-account environments, helping organizations streamline operations, maintain security, and optimize costs.

🤔 What are we doing in this Lab?

In this exercise, you're setting up and sharing a Virtual Private Cloud (VPC) between two AWS accounts using AWS Resource Access Manager (RAM). The main goal is to create a VPC in one account (the "main" account) and share its subnet with another account (the "member" account) within the same AWS Organization. By doing this, you can demonstrate how to use shared resources in AWS for cross-account collaboration.

Key Steps:

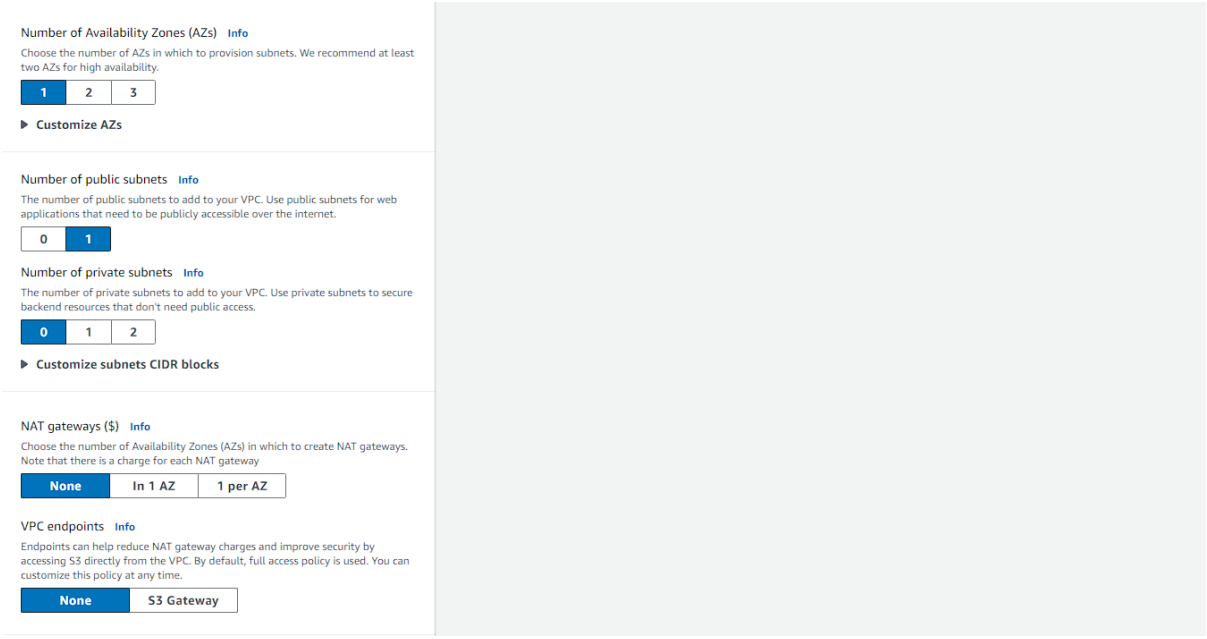
1. **Create and Share a VPC Subnet**: You create a VPC and its subnet in the main account, then share the subnet with the member account via AWS RAM.
2. **Launch EC2 Instances**: You launch an EC2 instance in each account using the shared VPC.
3. **Test Connectivity**: You establish network connectivity between the instances by configuring security group rules, allowing them to communicate.

End Goal:

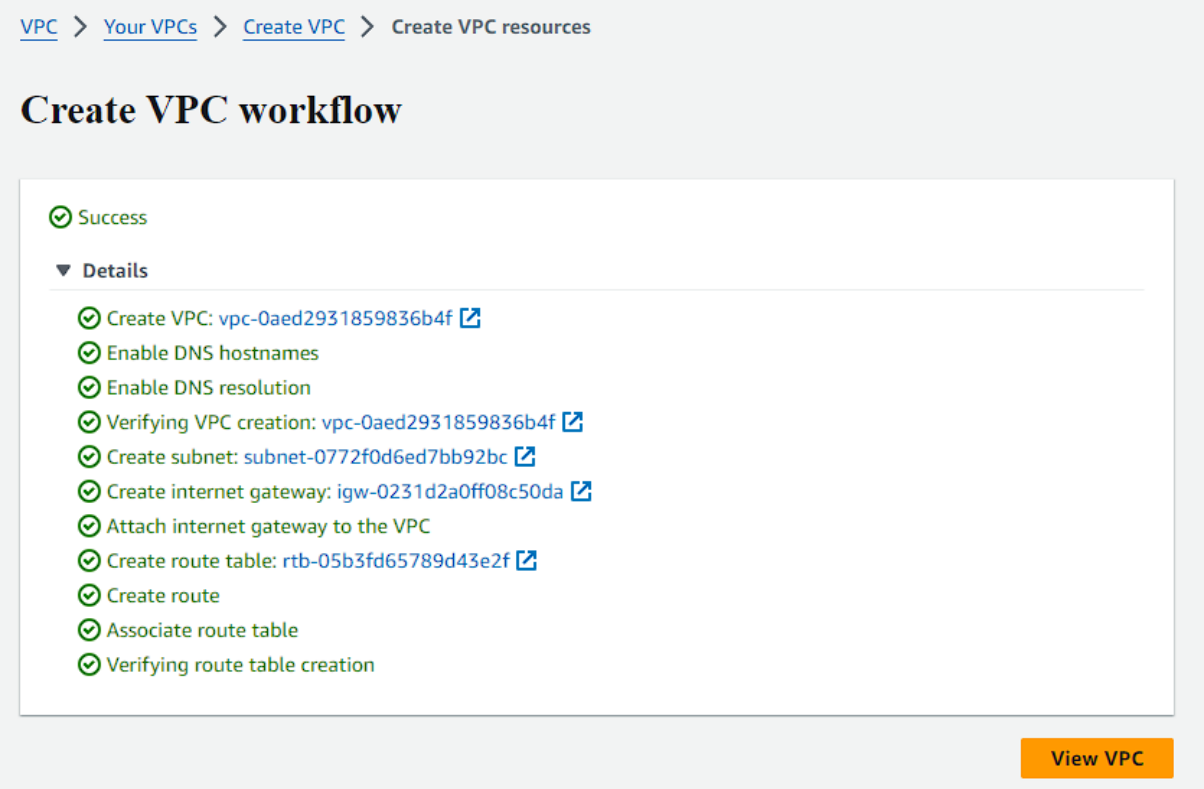
The purpose is to understand how to share AWS resources securely across multiple accounts and how to set up network configurations to allow communication between resources in different accounts. This setup is useful for scenarios where different teams or departments need to access shared infrastructure while maintaining separate accounts.

😊 To begin with the Lab:

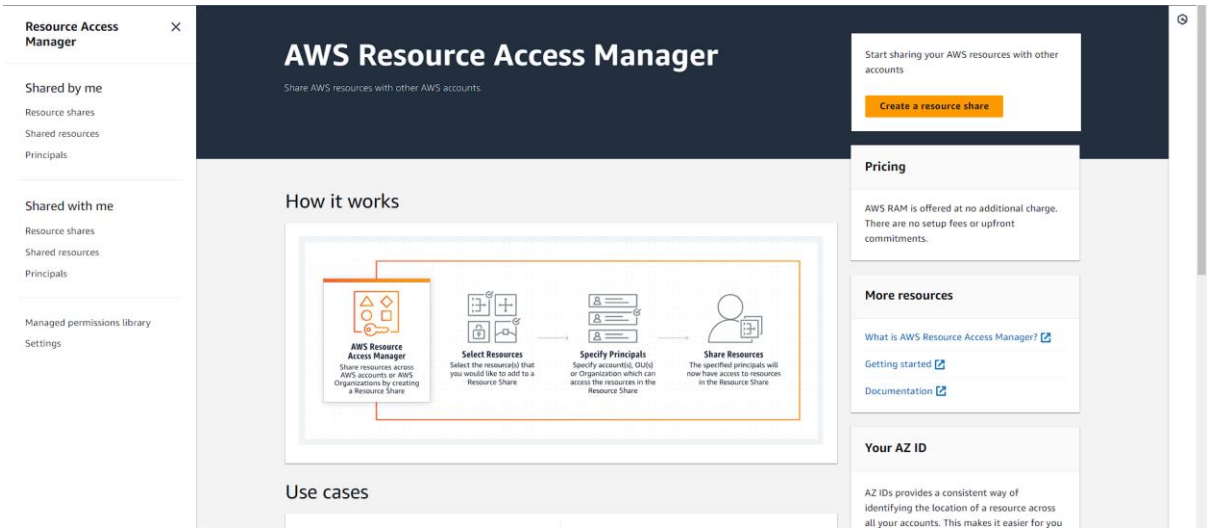
1. There are some prerequisites for this lab. You should have two AWS Accounts.
2. Login to both of them in the AWS Console.
3. They should be connected via AWS Organizations.
4. Now you have to navigate to VPC in your main account which ever it may be.
5. Then you are going to create a new VPC.
6. While creating the VPC use the VPC and more option. Then choose the options shown below and create your VPC.



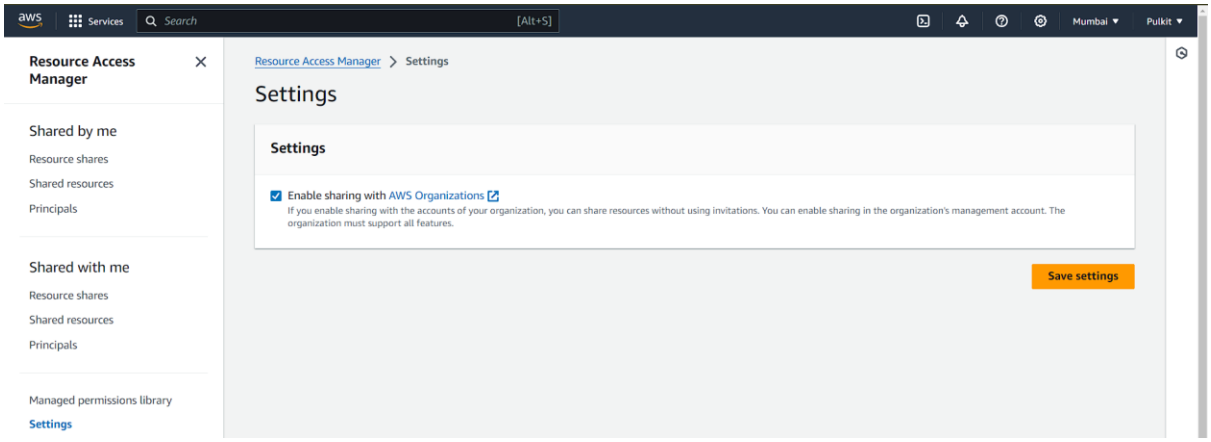
7. You will see that your VPC is created in no time.



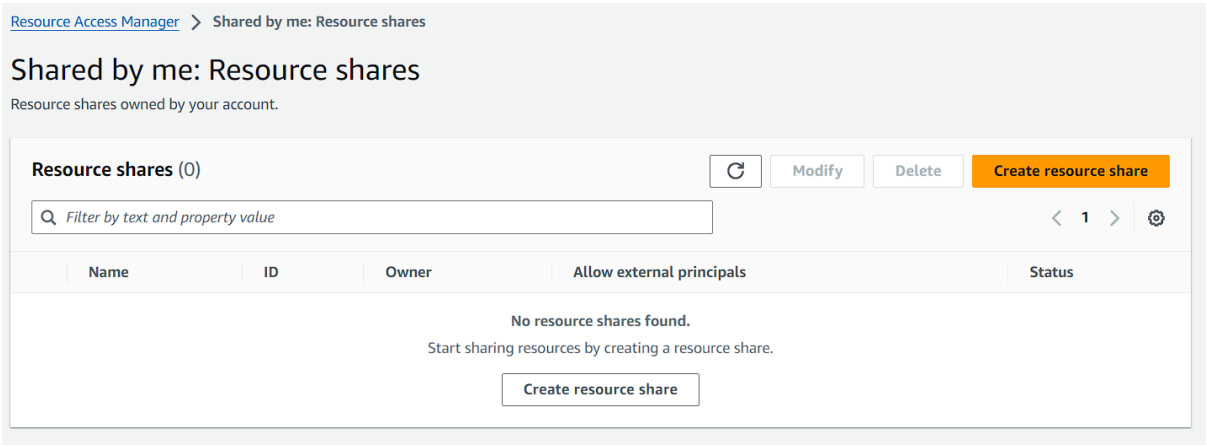
8. Now open Resource Access Manager in both of the accounts.



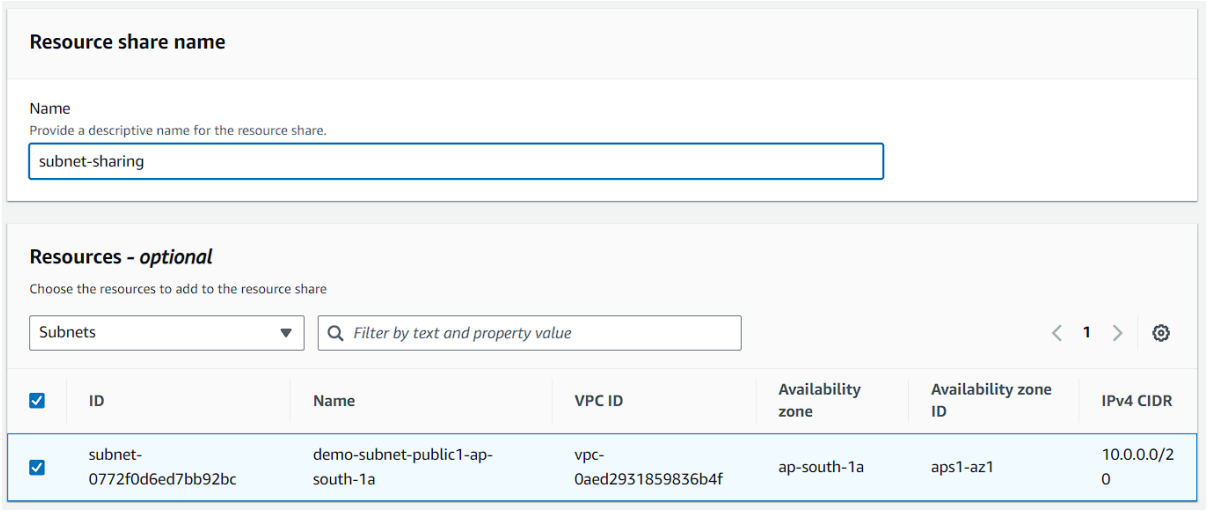
9. Now in here click on settings and enable the sharing with AWS Organizations.



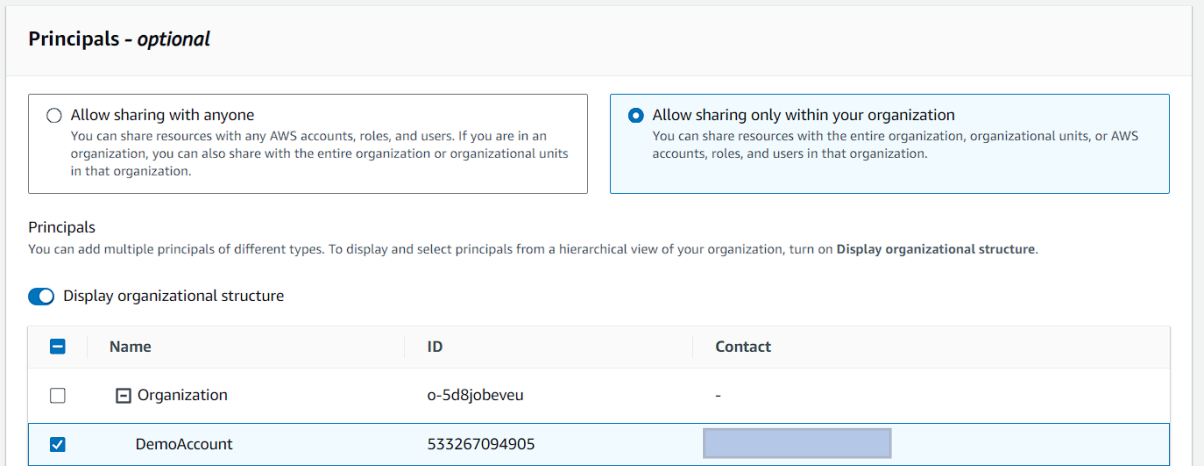
10. Once this is done go to Resource share and click on create resource share.



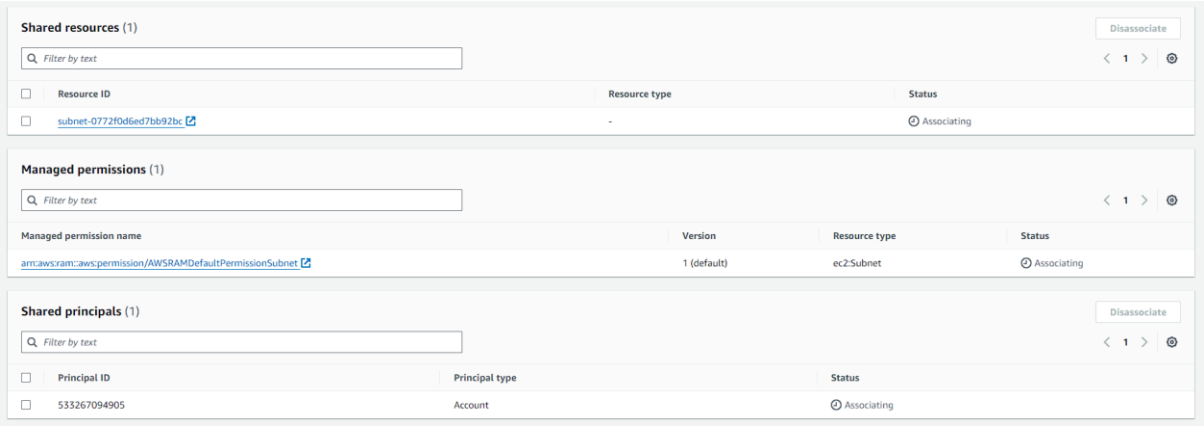
11. In there you have to give it a name then in the resources search for subnets and select your newly created subnet. Then move to next page.



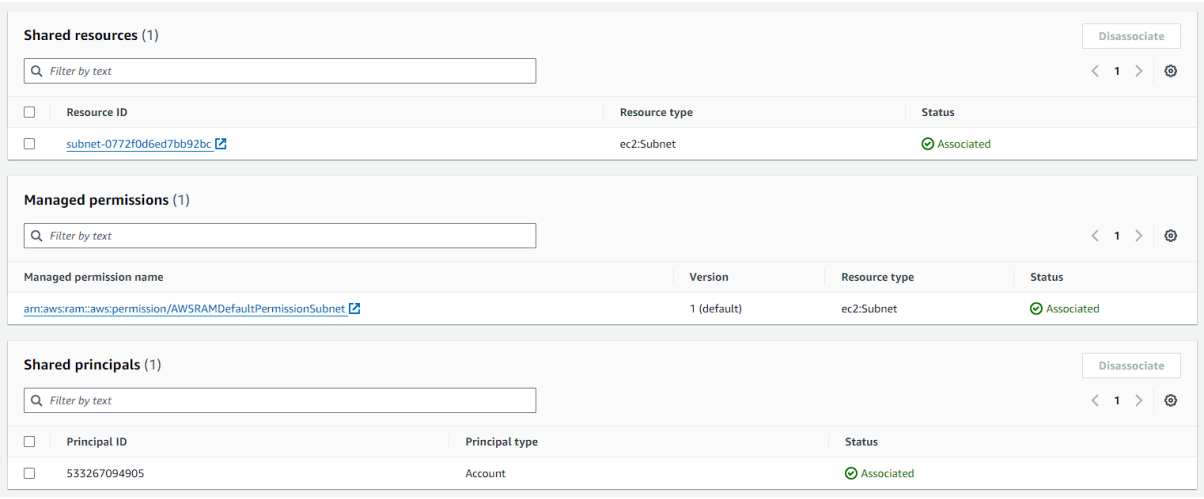
12. One the page or say step 2 leave as it is and move to next.
13. Now here you need to click on display organizational structure. Then select your member account on which you are going to share your subnet.



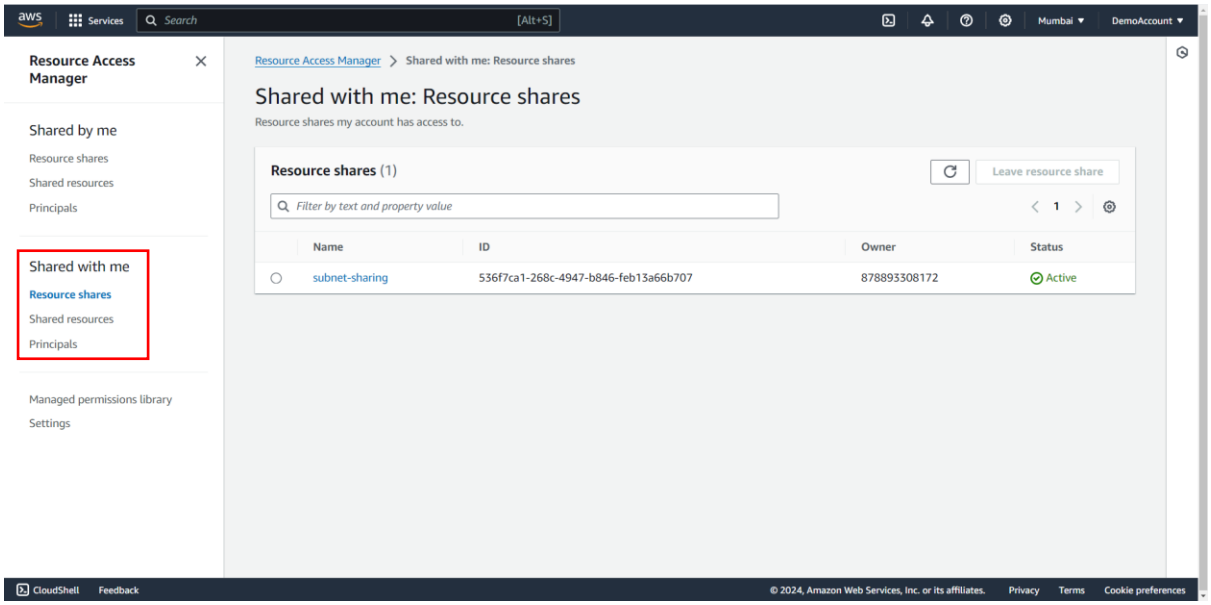
14. Then just move to review page and create your subnet.
15. Below you can see that the status is associating. So, now you have to wait until it gets to associated.



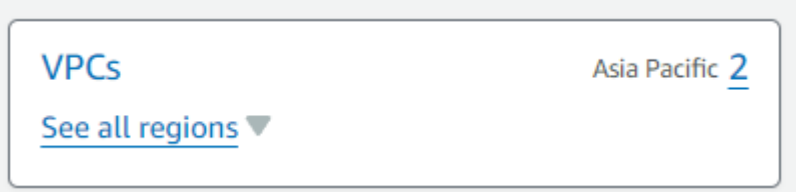
16. After sometime just refresh the page and you will see that the status is green now. Always check that the status should be green or say Associated.



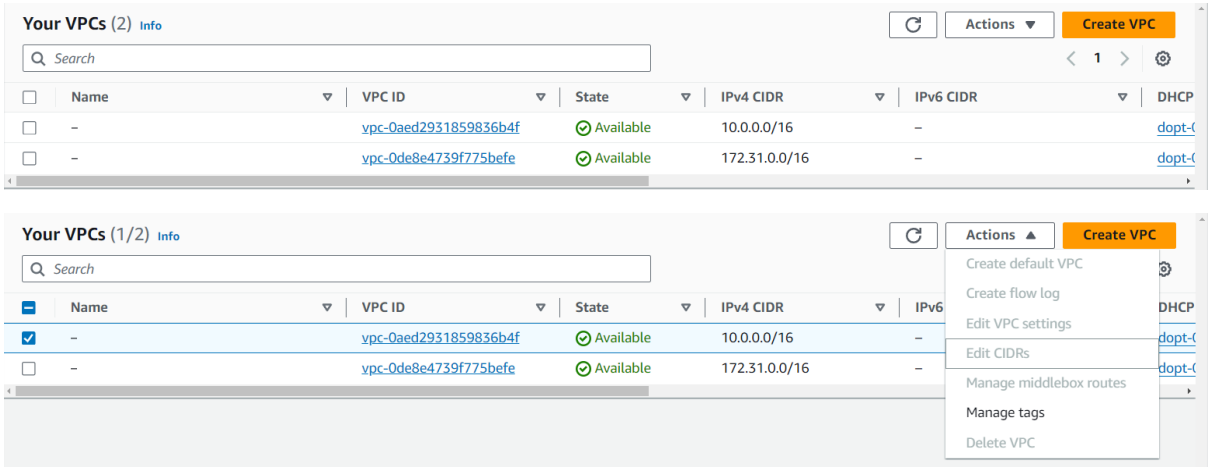
17. Now move to the other account and navigate to Resource Access Manager. You will see those shared resources in this account.



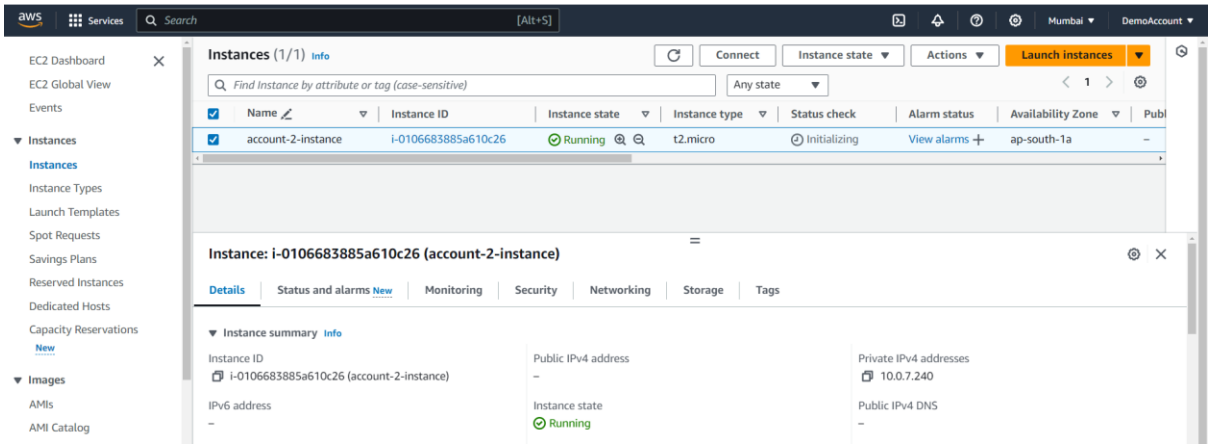
18. Now for a quick review go to VPC in this account and check for the VPC. Now in the VPC you will see that there are two VPCs.



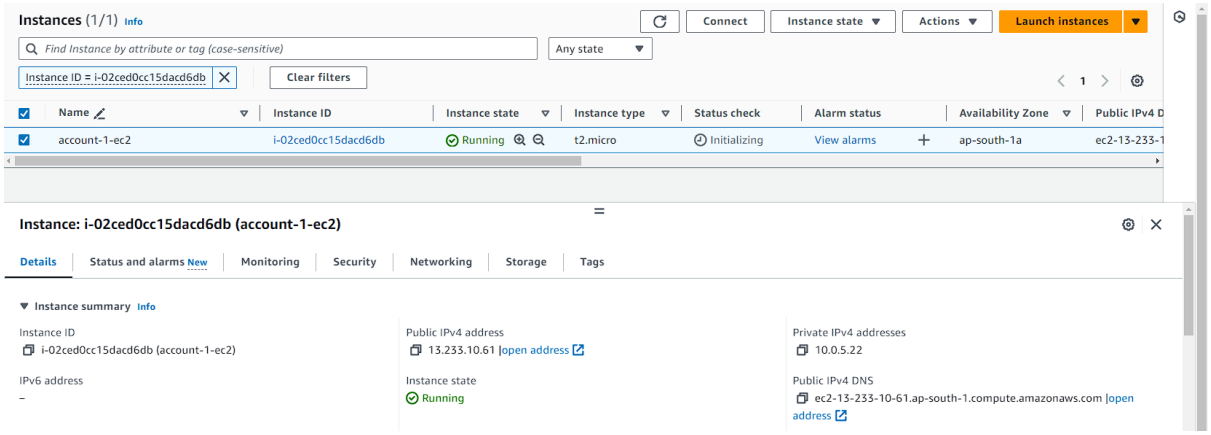
19. There are two ways two ways to recognize your shared VPC. One is to see the CIDR and other one is to select it and click on action you won't be able to perform any action on this VPC.



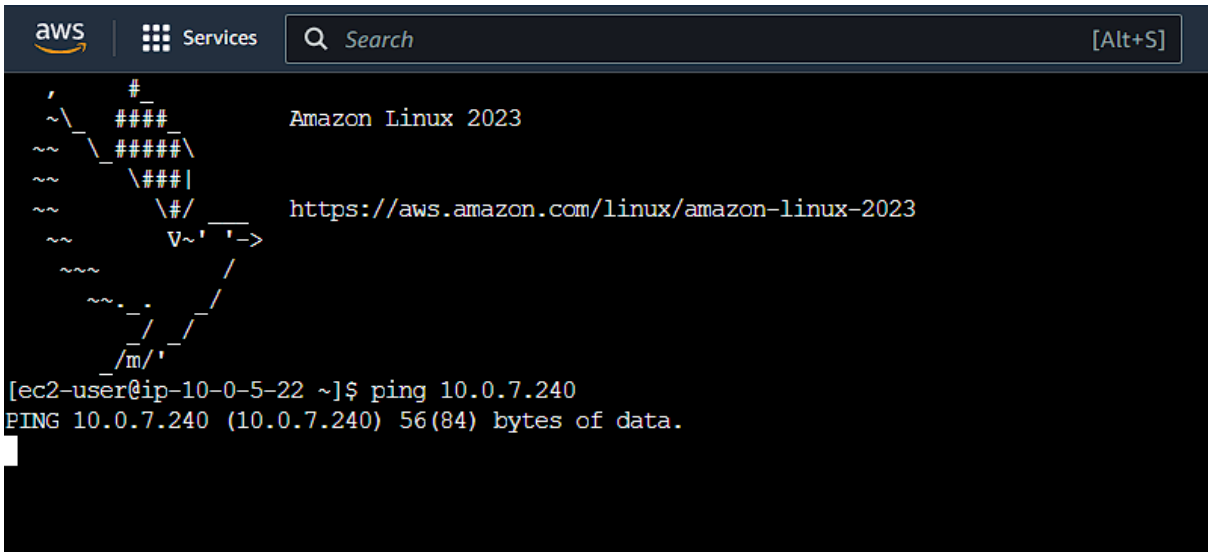
20. Now in your member account or say account 2 navigate to EC2 and launch an instance using the shared VPC.



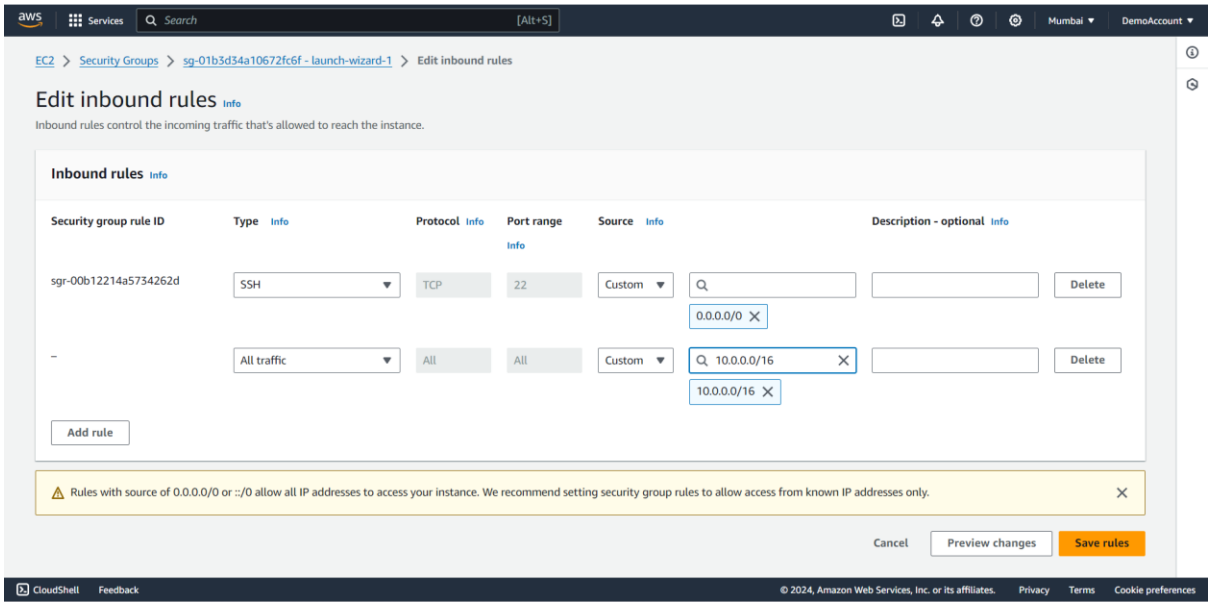
21. Now you need to do the same thing in your main account, create an EC2 instance with the same VPC and remember to enable public IP address.



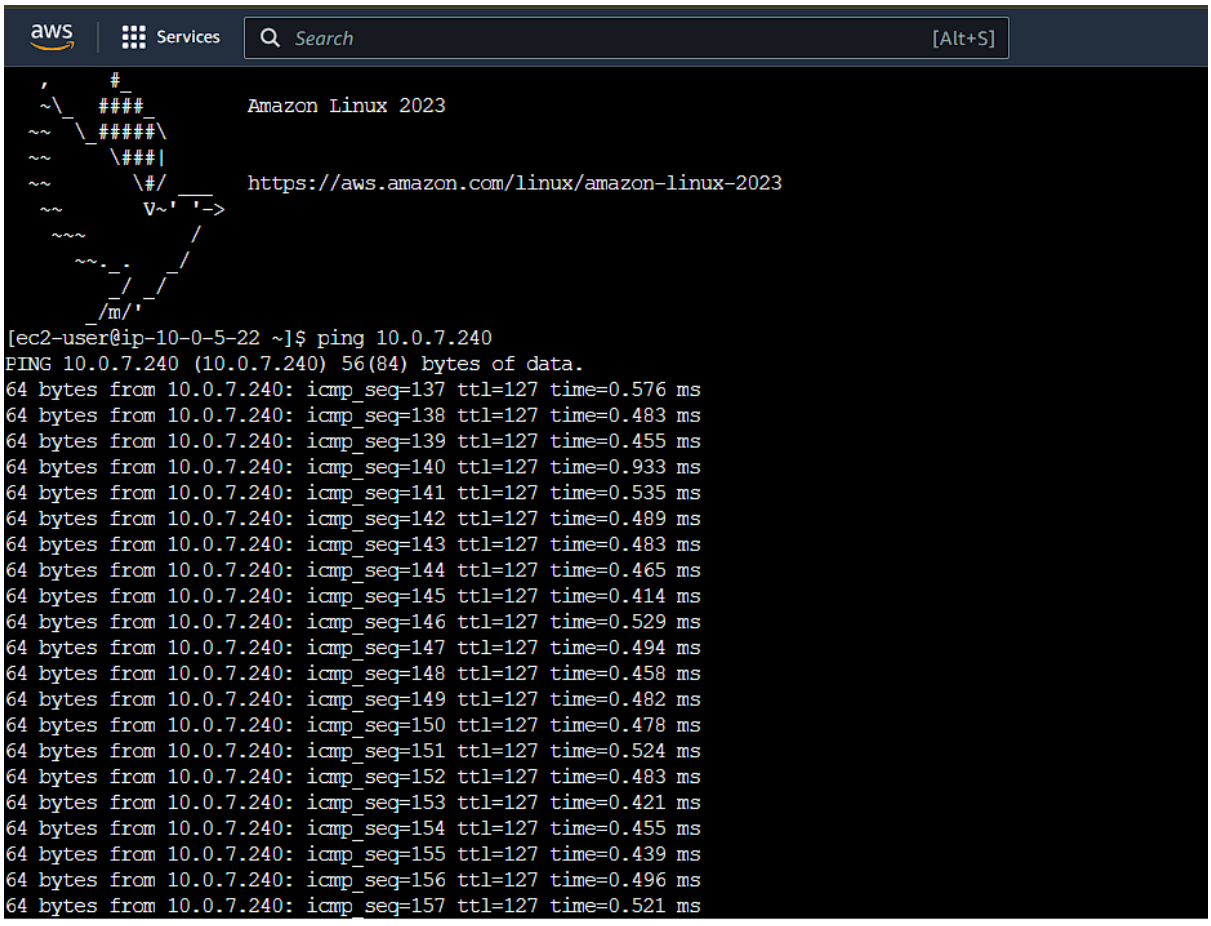
22. Now SSH into your main account instance and try to ping the other account instance.
23. Currently, you will see that is not able to connect to the other instance. For that you are going to add rule in the security group of the account 2.



24. Now from EC2 quickly open security click on edit inbound rules. And then add all traffic from VPC CIDR block. Then save the rules.



25. Now if you will go back, you'll be able to see the connection working.



26. Do not forget to terminate the instance in both of your accounts.