PRJ-NET-037: Multi-Account VPC Sharing

Certification: AWS Certified Advanced Networking – Specialty
Domain: Network Management and Automation

---

1. Project Overview

This project demonstrates how to use **VPC Sharing**, a powerful feature that allows multiple AWS accounts within the same **AWS Organization** to launch their application resources into a shared, centrally managed Virtual Private Cloud (VPC). This model provides significant benefits by separating the roles and responsibilities of network administration from application development.

A central networking team can own and manage the VPC, subnets, route tables, and network gateways, while application teams (in their own separate accounts) can deploy resources like EC2 instances, RDS databases, and Lambda functions into subnets that are shared with them. This approach reduces the number of VPCs to manage, simplifies network topology, lowers costs by reusing NAT Gateways and VPC endpoints, and enforces a clear separation of duties.

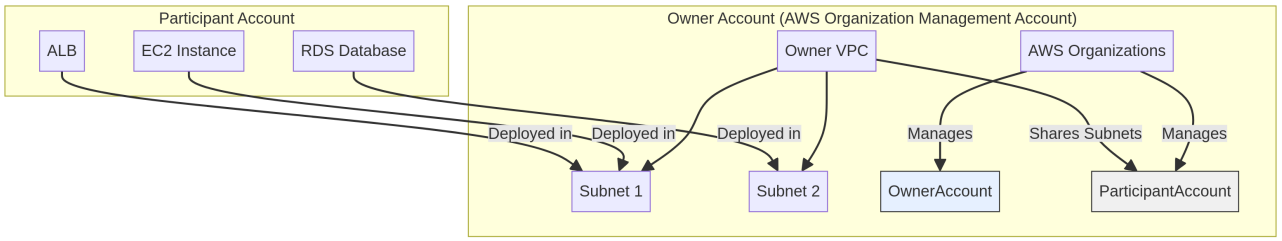
Key Objectives

- Understand the owner/participant model of VPC sharing.
- Enable resource sharing within an AWS Organization.
- Use **AWS Resource Access Manager (RAM)** to share subnets from an owner account to participant accounts.
- Launch resources from a participant account into a shared subnet.
- Verify that resources in different accounts within the same VPC can communicate with each other.
- Understand the security responsibilities and boundaries in a shared VPC environment.

---

2. Architecture

The architecture consists of a single “Owner” account that creates and manages the VPC, and one or more “Participant” accounts that use the shared network resources.



Architectural Components:

1. **AWS Organizations:** VPC sharing is only possible between accounts that are part of the same AWS Organization. The management account of the organization must enable resource sharing.
2. **Owner Account:**
    - This account owns the VPC, including its CIDR block, subnets, route tables, Internet Gateways, NAT Gateways, and VPC endpoints.
    - The network administrator in the owner account uses **AWS RAM** to create a “Resource Share”.
    - They select specific subnets from their VPC to be included in this share.
    - They then share it with other accounts or, more commonly, with an entire Organizational Unit (OU).
3. **Participant Account:**
    - This account is a member of the same AWS Organization.
    - The application owner in the participant account accepts the resource share invitation in the AWS RAM console.
    - Once accepted, the shared subnets appear in the participant’s EC2 and VPC consoles as if they were native to that account.
    - The participant can now launch resources (EC2, RDS, ALB, etc.) directly into these shared subnets.
4. **Resource Communication:**

- Resources launched by different participant accounts into the same shared VPC can communicate with each other using their private IP addresses, subject to security group rules.
- All resources share the same route tables, NAT Gateways, and Internet Gateway configured by the owner account.

---

3. Prerequisites

- An active **AWS Organization** with at least two member accounts (one to act as the owner, one as the participant).
- Administrative access to both the organization's management account and the participant accounts.

---

4. Step-by-Step Implementation Guide

Step 4.1: Enable Resource Sharing in AWS Organizations

1. Log in to the **management account** of your AWS Organization.
2. Go to the **AWS Resource Access Manager (RAM) console**.
3. In **Settings**, click **Enable sharing with AWS Organizations**.

Step 4.2: Create and Share the VPC Subnets (Owner Account)

1. Log in to the account you have designated as the **Owner Account**.
2. Create a VPC with at least two subnets (e.g., a public and a private subnet).
3. Go to the **AWS RAM console -> Resource shares -> Create resource share**.
4. **Name:** `Shared-VPC-Subnets`
5. **Select resource type:** Subnets.
6. **Select the subnets** you want to share.
7. **Principals:**

- Select **Allow sharing with anyone** (if you want to share with specific accounts) or **Allow sharing only within my organization**.
- Add the **AWS Account ID** of your participant account or the ID of the **Organizational Unit (OU)** that contains your participant accounts.

8. **Create resource share**.

Step 4.3: Accept the Share and Launch Resources (Participant Account)

1. Log in to the **Participant Account**.
2. Go to the **AWS RAM console**.
3. In **Shared with me -> Resource shares**, you will see the invitation for `Shared-VPC-Subnets`.
4. Select the share and click **Accept resource share**.
5. **Verify Access:**
   - Go to the **VPC Console -> Subnets**. You should now see the shared subnets listed. The “Owner account” column will show the ID of the owner account.
6. **Launch an EC2 Instance:**
   - Go to the **EC2 Console -> Launch instances**.
   - When you get to the **Network settings** step, you will be able to select the shared VPC.
   - Choose one of the **shared subnets** to launch your instance into.
   - Create a new security group for this instance.
   - Launch the instance.

Step 4.4: Verify Connectivity

1. In the **Owner Account**, launch an EC2 instance into the same subnet that you shared.
2. Note the private IP addresses of both the instance in the owner account and the instance in the participant account.
3. SSH into one of the instances.
4. Try to ping the private IP address of the other instance.

5. The ping should succeed (provided the security groups you created allow ICMP traffic between them), proving that resources in different accounts are communicating within the same VPC network space.

---

5. Security and Governance Model

Understanding the separation of responsibilities is crucial in a shared VPC environment:

Resource	Managed by Owner Account (Network Admin)	Managed by Participant Account (App Admin)
VPC, Subnets, Route Tables	✅ Yes	❌ No
Internet Gateway, NAT Gateway	✅ Yes	❌ No
Network ACLs (NACLs)	✅ Yes	❌ No
Security Groups	❌ No	✅ Yes
EC2, RDS, ALB, Lambda	❌ No	✅ Yes

- **Owner's Responsibility:** The owner controls the overall network topology, routing, and broad network access controls (NACLs). They are responsible for the flow of traffic in and out of the VPC.
- **Participant's Responsibility:** The participant is responsible for the security of their own applications. They control the instance-level firewall rules by managing their own **security groups**. A participant cannot see or modify the security groups of another participant, even if their instances are in the same subnet.

---

6. Cleanup

1. **Terminate all resources** launched by participant accounts in the shared subnets.

2. In the **Owner Account**, go to the **AWS RAM console**.
3. Select the `Shared-VPC-Subnets` resource share and **delete** it. This will revoke access for all participant accounts.
4. You can now safely delete the VPC and other networking resources in the owner account.

Business Context

The Problem

Cloud networks are complex and vulnerable to misconfigurations that expose resources to the internet. Organizations lack visibility into network traffic and struggle to implement proper network segmentation. Manual network management leads to security gaps and compliance violations.

The Solution

Secure network architecture with automated traffic inspection, network segmentation, and centralized management. Implements VPC design best practices, transit gateway for multi-VPC connectivity, and network firewall for deep packet inspection. Provides complete network visibility and control.

Business Value

- **Zero Trust Architecture:** Micro-segmentation prevents lateral movement of threats
- **Network Visibility:** Complete traffic logging and analysis for security and troubleshooting
- **Centralized Management:** Single pane of glass for multi-account network control
- **Compliance Ready:** Network logs and flow records for audit requirements

Risk Mitigation

Prevents unauthorized network access, blocks malicious traffic, contains breaches through segmentation, and ensures compliance with network security requirements.

GRC Mapping

Compliance Frameworks

- **NIST CSF:** PR.AC-5 (Network segregation), PR.PT-4 (Network protection), DE.CM-1 (Network monitoring)
- **ISO 27001:** A.13.1 (Network security management), A.13.2 (Information transfer)
- **CIS Controls:** Control 12 (Network Infrastructure Management), Control 13 (Network Monitoring)
- **Zero Trust Architecture:** NIST SP 800-207

Security Controls Implemented

- Network segmentation and micro-segmentation
- Stateful firewall and deep packet inspection
- VPC flow logs and traffic analysis
- Network access control lists (NACLs)
- Private connectivity (VPN/Direct Connect)

Audit Evidence

- VPC flow logs and network traffic records
- Firewall rules and policy configurations
- Network topology diagrams
- Security group and NACL configurations

Regulatory Alignment

- **PCI DSS:** Requirement 1 (Firewall configuration), Requirement 2 (Network segmentation)
- **HIPAA:** § 164.312(e) (Transmission security)
- **GDPR:** Article 32 (Network security measures)
- **SOC 2:** CC6.6 (Logical access - network security)