AWS VPC Peering: Configuration Guide and Best Practices

Introduction

Amazon VPC Peering enables private communication between two VPCs using AWS's internal network. Peering supports secure, low-latency traffic flow without requiring internet gateways, VPN connections, or transit gateways.

Overview

VPC Peering establishes a one-to-one, bidirectional connection between two VPCs. These VPCs can belong to the same AWS account or different accounts, and they can reside in the same or different AWS Regions (intra- or inter-Region).

Pre-requisites

- Two VPCs with non-overlapping IPv4/IPv6 CIDR blocks
- IAM permissions for `ec2:CreateVpcPeeringConnection`, `ec2:AcceptVpcPeeringConnection`, and route table modification
- Understanding of route tables, DNS settings, and NACLs

Use Cases / Benefits

- Private connectivity between microservices in different VPCs
- Connecting dev/staging and production environments
- Multi-account network design using AWS Organizations
- Cross-Region low-latency communication

How-To: Create a VPC Peering Connection

1. Initiate Peering

- In the **VPC console**, select **Peering Connections → Create Peering Connection**
- Select Requester and Accepter VPC IDs
- For cross-account peering, enter the Accepter's AWS Account ID

2. Accept the Peering Request

- On the Accepter account (if cross-account), go to **Peering Connections**, select the pending request, and **Accept**

3. Update Route Tables

- Add routes in both VPCs to direct traffic to the peered VPC's CIDR block via the Peering Connection

4. (Optional) Enable DNS Resolution

- In **Peering Connection settings**, enable DNS resolution if required
- Ensure `enableDnsHostnames` and `enableDnsSupport` are enabled on both VPCs

Considerations

- **Transitive Peering Not Supported**: A → B and B → C does NOT imply A → C
- **Overlapping CIDRs**: Peering will be blocked
- **Security Groups & NACLs**: Still apply; you must allow traffic explicitly
- **Cost**: Data transfer between peered VPCs is charged at standard intra-/inter-AZ/Region data rates
- **Scalability**: For many VPCs, consider **Transit Gateway** as a hub-and-spoke alternative

Advanced Topics

- **Cross-Region Peering**: Supported; note higher latency and different pricing
- **Terraform Integration**: Use `aws_vpc_peering_connection`, `aws_route`, and optionally `aws_vpc_peering_connection_accepter`
- **Peering with Shared VPCs**: Ensure proper RAM (Resource Access Manager) permissions are in place

Documentation

- AWS Docs: https://docs.aws.amazon.com/vpc/latest/peering/what-is-vpc-peering.html
- AWS Peering Limits: https://docs.aws.amazon.com/vpc/latest/peering/vpc-peering-limits.html