AWS VPC: Virtual Private Cloud

Networking Fundamentals and Best Practices - Associate Level

Prepared By: Rashi Rana
AWS Corporate Trainer

What is AWS VPC

Amazon Virtual Private Cloud (VPC) is a logically isolated section of the AWS Cloud where you can launch AWS resources in a virtual network that you define. You have complete control over your virtual networking environment.

Core Purpose

- **Network Isolation:** Create isolated network environments in the cloud
- **Security Control:** Define security groups and network ACLs
- **Connectivity:** Control how resources communicate with each other and the internet
- **Customization:** Configure IP address ranges, subnets, and routing tables

Key Benefits

Complete Control

Full control over virtual networking environment including IP ranges and routing

Security

Multiple layers of security including security groups and network ACLs

Flexibility

Support for both IPv4 and IPv6, multiple connectivity options

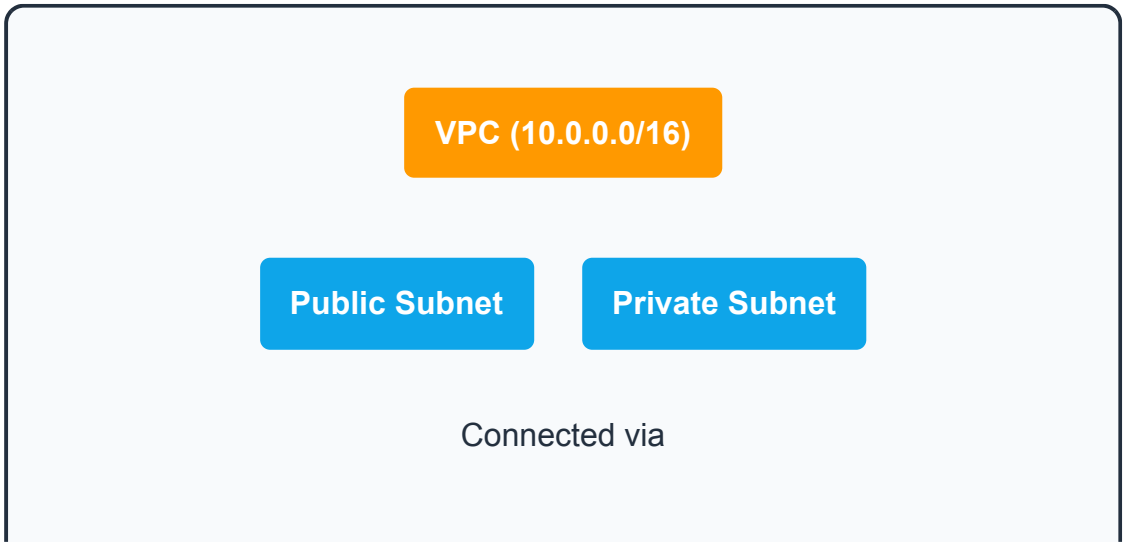
Integration

Seamless integration with other AWS services and on-premises networks

VPC vs Traditional Networking

Aspect	Traditional Data Center	AWS VPC
Infrastructure	Physical hardware and cables	Software-defined networking
Scalability	Limited by physical capacity	Virtually unlimited scalability
Setup Time	Weeks to months	Minutes to hours
Cost	High upfront capital expenditure	Pay-as-you-use operational expenditure

VPC Architecture Overview



Internet Gateway | NAT Gateway | Route Tables

Real-World Use Cases

- **Web Applications:** Host multi-tier web applications with public and private subnets
- **Hybrid Cloud:** Extend on-premises networks to the cloud
- **Disaster Recovery:** Create backup environments in different regions
- **Development/Testing:** Isolated environments for development and testing
- **Compliance:** Meet regulatory requirements with network isolation

> **Key Concept:** VPC provides the networking foundation for all your AWS resources, giving you complete control over your virtual networking environment while maintaining the scalability and flexibility of the cloud.

Prepared By: Rashi Rana
AWS Corporate Trainer

VPC Core Components

AWS VPC consists of several key components that work together to provide a complete virtual networking solution: **Subnets, Route Tables, Internet Gateways, NAT Gateways, and Security Groups**.

Essential VPC Components

Subnets

Segments of VPC IP address range where you can launch AWS resources

Route Tables

Rules that determine where network traffic is directed

Internet Gateway

Allows communication between VPC and the internet

NAT Gateway

Enables outbound internet access for private subnet resources

Security Components

Security Groups

Network ACLs

Virtual firewalls that control inbound and outbound traffic at instance level

Subnet-level firewalls that provide additional layer of security

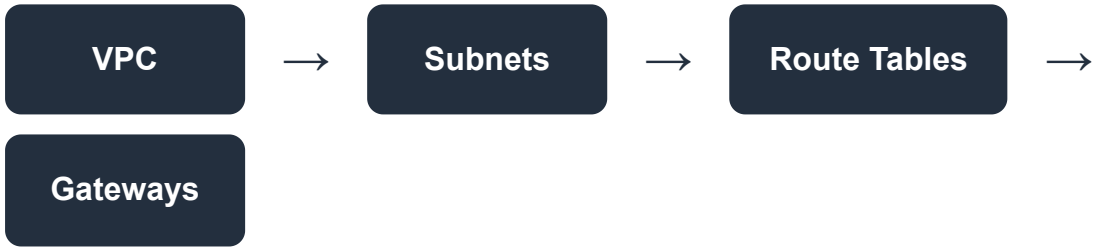
VPC Flow Logs

Capture information about IP traffic going to and from network interfaces

VPC Endpoints

Private connections to AWS services without internet gateway

Component Relationships



Component Comparison

Component	Level	Purpose	Scope
VPC	Regional	Logical isolation	Entire virtual network
Subnet	Availability Zone	Resource placement	Segment of VPC
Security Group	Instance	Traffic filtering	Individual resources

Component	Level	Purpose	Scope
Network ACL	Subnet	Additional security	All resources in subnet

VPC Limits and Considerations

- **VPCs per Region:** 5 VPCs per region (can be increased)
- **Subnets per VPC:** 200 subnets per VPC
- **Route Tables:** 200 route tables per VPC
- **Security Groups:** 2,500 security groups per VPC
- **Rules per Security Group:** 60 inbound and 60 outbound rules

> **Important:** Plan your VPC architecture carefully as some components like CIDR blocks cannot be changed after creation, though you can add additional CIDR blocks.

Prepared By: Rashi Rana
AWS Corporate Trainer

Understanding CIDR and IP Addressing

CIDR (Classless Inter-Domain Routing) is a method for allocating IP addresses and routing. Understanding CIDR is crucial for designing VPC networks and calculating available IP addresses.

CIDR Notation Basics

CIDR Format: IP Address/Prefix Length

```
Example: 10.0.0.0/16
- 10.0.0.0 = Network Address
- /16 = Subnet Mask (first 16 bits are network, remaining 16 bits are host)
- Available IPs = 2^(32-16) = 2^16 = 65,536 addresses
```

Common CIDR Blocks and Calculations

CIDR Block	Subnet Mask	Available IPs	Usable IPs*	Use Case
/16	255.255.0.0	65,536	65,531	Large VPC

CIDR Block	Subnet Mask	Available IPs	Usable IPs*	Use Case
/20	255.255.240.0	4,096	4,091	Medium subnet
/24	255.255.255.0	256	251	Small subnet
/28	255.255.255.240	16	11	Very small subnet

* AWS reserves 5 IP addresses in each subnet

CIDR Calculation Formula

Step-by-Step Calculation

```
1. Total IPs = 2^(32 - prefix length)
2. Usable IPs = Total IPs - 5 (AWS reserved)

Example: 10.0.1.0/24
- Total IPs = 2^(32-24) = 2^8 = 256
- Usable IPs = 256 - 5 = 251

AWS Reserved IPs in 10.0.1.0/24:
- 10.0.1.0: Network address
- 10.0.1.1: VPC router
- 10.0.1.2: DNS server
- 10.0.1.3: Future use
- 10.0.1.255: Broadcast address
```

Private IP Address Ranges (RFC 1918)

Class	IP Range	CIDR Notation	Total IPs
Class A	10.0.0.0 - 10.255.255.255	10.0.0.0/8	16,777,216
Class B	172.16.0.0 - 172.31.255.255	172.16.0.0/12	1,048,576
Class C	192.168.0.0 - 192.168.255.255	192.168.0.0/16	65,536

VPC CIDR Best Practices

- **Plan for Growth:** Choose larger CIDR blocks than immediately needed
- **Avoid Overlaps:** Ensure CIDR blocks don't overlap with on-premises networks
- **Use Private Ranges:** Always use RFC 1918 private IP ranges
- **Document Allocation:** Maintain clear documentation of IP allocation
- **Consider Peering:** Plan for VPC peering and avoid overlapping ranges

> **Pro Tip:** Use online CIDR calculators to verify your calculations and visualize subnet divisions. Always plan your IP addressing scheme before creating your VPC.

Prepared By: Rashi Rana
AWS Corporate Trainer

Subnets and Availability Zones

Subnets are segments of a VPC's IP address range where you can launch AWS resources. Each subnet must reside entirely within one Availability Zone and cannot span zones.

Subnet Types

Public Subnet

Has a route to an Internet Gateway, resources can have public IP addresses

Private Subnet

No direct route to Internet Gateway, resources use NAT for outbound internet access

VPN-only Subnet

Has route to Virtual Private Gateway, no route to Internet Gateway

Isolated Subnet

No routes to internet or VPN, completely isolated

Subnet Planning Example

VPC: 10.0.0.0/16 (65,536 IPs)

```
Subnet Division Strategy:

Public Subnets (Web Tier):
- 10.0.1.0/24 (AZ-1a) - 251 usable IPs
- 10.0.2.0/24 (AZ-1b) - 251 usable IPs
- 10.0.3.0/24 (AZ-1c) - 251 usable IPs

Private Subnets (App Tier):
- 10.0.11.0/24 (AZ-1a) - 251 usable IPs
- 10.0.12.0/24 (AZ-1b) - 251 usable IPs
- 10.0.13.0/24 (AZ-1c) - 251 usable IPs

Database Subnets (DB Tier):
- 10.0.21.0/24 (AZ-1a) - 251 usable IPs
- 10.0.22.0/24 (AZ-1b) - 251 usable IPs
- 10.0.23.0/24 (AZ-1c) - 251 usable IPs
```

Availability Zone Considerations

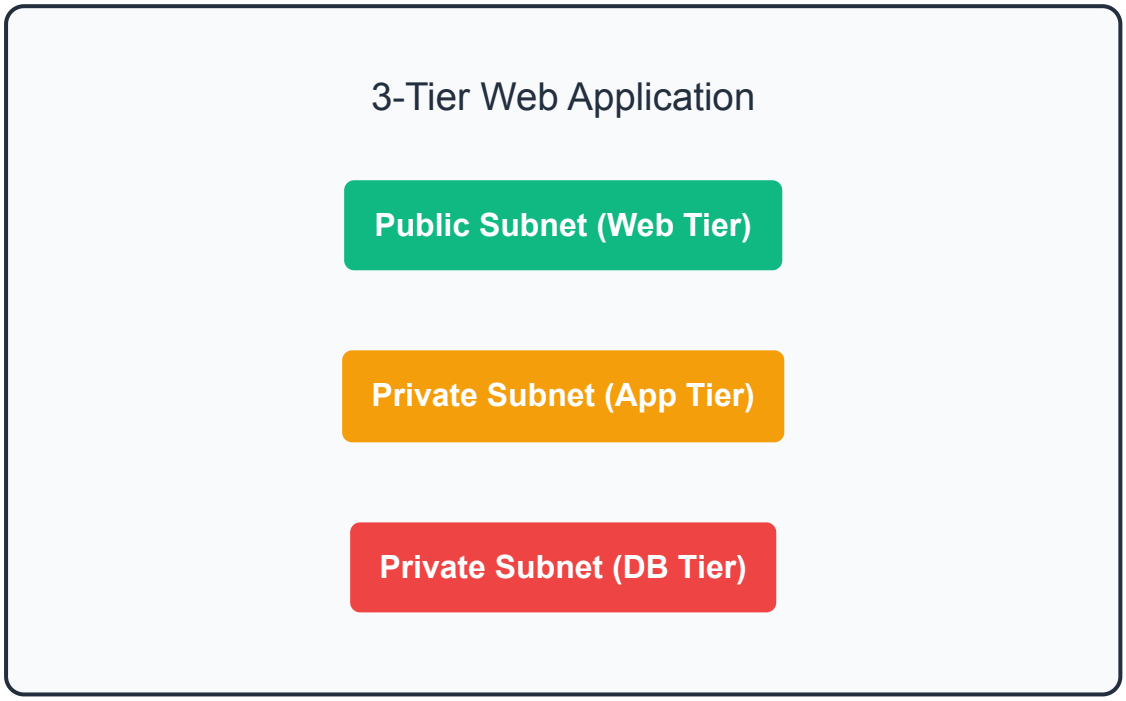
Aspect	Single AZ	Multiple AZs
High Availability	Single point of failure	Fault tolerant
Cost	Lower (single NAT Gateway)	Higher (multiple NAT Gateways)
Complexity	Simpler architecture	More complex routing
Use Case	Development/Testing	Production workloads

Subnet Sizing Guidelines

- **/24 Subnets:** Good for most use cases (251 usable IPs)
- **/20 Subnets:** For large deployments (4,091 usable IPs)

- **/28 Subnets:** For small, specific use cases (11 usable IPs)
- **Auto Scaling:** Consider future scaling needs
- **Reserved Capacity:** Leave room for growth and new services

Multi-Tier Architecture Example



Subnet Best Practices

- **Distribute Across AZs:** Use multiple AZs for high availability
- **Consistent Naming:** Use clear, consistent naming conventions
- **Right-Size Subnets:** Balance between too small and too large
- **Plan for Growth:** Leave room for additional subnets
- **Security Zones:** Group similar resources in same subnet

> **Architecture Tip:** Design your subnet strategy around your application tiers and security requirements. Each tier should have subnets in multiple AZs for high availability.

Route Tables and Routing

> **Route Tables** contain a set of rules (routes) that determine where network traffic is directed. Each subnet must be associated with a route table.

Route Table Basics

- **Main Route Table:** Default route table created with VPC
- **Custom Route Tables:** Additional route tables you create
- **Subnet Association:** Each subnet associated with exactly one route table
- **Route Priority:** Most specific route (longest prefix) takes precedence

Route Components

Component	Description	Example
Destination	CIDR block for traffic destination	0.0.0.0/0 (all traffic)
Target	Where to send the traffic	igw-12345678
Status	Route availability	Active, Blackhole
Propagated	Route source	Yes (VGW), No (manual)

Common Route Targets

Internet Gateway (IGW)

Routes traffic to/from the internet

NAT Gateway/Instance

Routes outbound internet traffic from private subnets

VPC Peering

Routes traffic to peered VPC

Virtual Private Gateway

Routes traffic to on-premises network

Route Table Examples

```
Public Subnet Route Table:
Destination Target Status
10.0.0.0/16 local Active
0.0.0.0/0 igw-12345678 Active

Private Subnet Route Table:
Destination Target Status
10.0.0.0/16 local Active
0.0.0.0/0 nat-12345678 Active

VPN Subnet Route Table:
Destination Target Status
10.0.0.0/16 local Active
192.168.0.0/16 vgw-12345678 Active
```

Routing Best Practices

- **Explicit Routes:** Create custom route tables instead of modifying main table
- **Least Privilege:** Only route traffic that needs to be routed
- **Documentation:** Name and tag route tables clearly
- **Monitoring:** Monitor route table changes and usage
- **Backup Routes:** Consider redundant paths for critical traffic

> **Important:** The local route (VPC CIDR) cannot be deleted or modified. It always has the highest priority for traffic within the VPC.

Prepared By: Rashi Rana
AWS Corporate Trainer

Internet Gateway and NAT

Internet Gateways and **NAT Gateways** provide different types of internet connectivity for your VPC resources, enabling both inbound and outbound internet access as needed.

Internet Gateway (IGW)

Purpose

Provides bidirectional internet access for public subnet resources

Scaling

Horizontally scaled, redundant, and highly available

Cost

No additional charges for using Internet Gateway

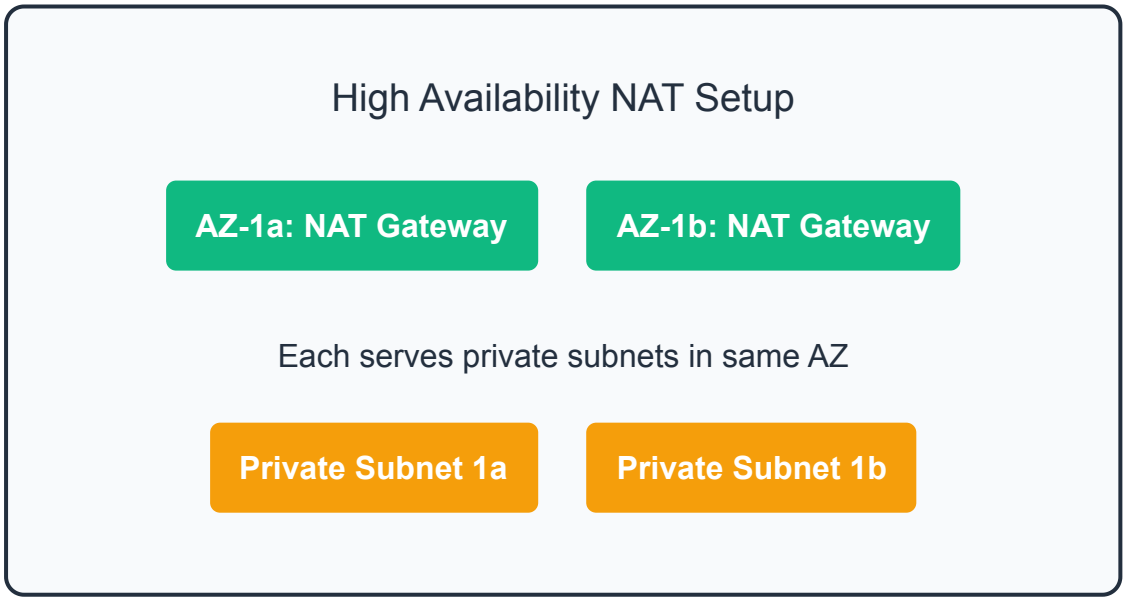
Limitations

One IGW per VPC, cannot be detached if resources depend on it

NAT Gateway vs NAT Instance

Aspect	NAT Gateway	NAT Instance
Availability	Highly available within AZ	Single point of failure
Bandwidth	Up to 45 Gbps	Depends on instance type
Maintenance	Managed by AWS	Managed by you
Cost	Higher (per hour + data)	Lower (instance cost only)
Security Groups	Cannot associate	Can associate
Bastion Host	Cannot use	Can use as bastion

NAT Gateway Deployment Patterns



Internet Connectivity Requirements

```
For Public Subnet (Bidirectional Internet Access):
1. Internet Gateway attached to VPC
```

```
2. Route to IGW (0.0.0.0/0 → igw-xxxxx)
3. Public IP address or Elastic IP
4. Security Group allows traffic
5. Network ACL allows traffic

For Private Subnet (Outbound Only):
1. NAT Gateway in public subnet
2. Route to NAT Gateway (0.0.0.0/0 → nat-xxxxx)
3. NAT Gateway has route to IGW
4. Security Group allows outbound traffic
5. Network ACL allows traffic
```

Cost Optimization Strategies

- **Single NAT Gateway:** Use one NAT Gateway for development environments
- **NAT Instance:** Consider for low-traffic, cost-sensitive workloads
- **VPC Endpoints:** Use for AWS service traffic to avoid NAT costs
- **Data Transfer:** Monitor and optimize data transfer costs
- **Regional Considerations:** NAT Gateway pricing varies by region

Troubleshooting Internet Connectivity

No Internet Access

Check route table, IGW attachment, security groups, NACLs

Intermittent Connectivity

Check NAT Gateway health, bandwidth limits, DNS resolution

High Costs

Performance Issues

Monitor data transfer, consider VPC endpoints, optimize routing

Check bandwidth limits, consider multiple NAT Gateways

Best Practice: Deploy NAT Gateways in multiple AZs for high availability, but consider the cost implications. Use VPC endpoints for AWS service traffic to reduce NAT Gateway usage.

Prepared By: Rashi Rana
AWS Corporate Trainer

Security Groups and NACLs

AWS provides two layers of security for VPC: **Security Groups** (instance-level firewalls) and **Network ACLs** (subnet-level firewalls). Understanding both is crucial for proper network security.

Security Groups vs Network ACLs

Aspect	Security Groups	Network ACLs
Level	Instance level	Subnet level
Rules	Allow rules only	Allow and Deny rules
State	Stateful	Stateless
Rule Evaluation	All rules evaluated	Rules processed in order
Default Behavior	Deny all inbound, allow all outbound	Allow all inbound and outbound
Association	Applied to ENI/instance	Applied to subnet

Security Group Rules

```
Example Security Group Rules:

Web Server Security Group (Inbound):
```

```
Type Protocol Port Source
HTTP TCP 80 0.0.0.0/0
HTTPS TCP 443 0.0.0.0/0
SSH TCP 22 10.0.0.0/16

Database Security Group (Inbound):
Type Protocol Port Source
MySQL TCP 3306 sg-web-servers
Custom TCP 5432 sg-app-servers

All Outbound: 0.0.0.0/0 (default)
```

Network ACL Rules

```
Example Network ACL Rules:

Inbound Rules:
Rule# Type Protocol Port Source Allow/Deny
100 HTTP TCP 80 0.0.0.0/0 ALLOW
110 HTTPS TCP 443 0.0.0.0/0 ALLOW
120 SSH TCP 22 10.0.0.0/16 ALLOW
130 Custom TCP 1024-65535 0.0.0.0/0 ALLOW
* ALL ALL ALL 0.0.0.0/0 DENY

Outbound Rules:
Rule# Type Protocol Port Destination Allow/Deny
100 HTTP TCP 80 0.0.0.0/0 ALLOW
110 HTTPS TCP 443 0.0.0.0/0 ALLOW
120 Custom TCP 1024-65535 0.0.0.0/0 ALLOW
* ALL ALL ALL 0.0.0.0/0 DENY
```

Security Best Practices

Least Privilege

Only allow necessary traffic, use specific ports and sources

Defense in Depth

Use both Security Groups and NACLs for layered security

Regular Audits

Regularly review and clean up unused security groups

Descriptive Names

Use clear, descriptive names and descriptions

Common Security Patterns

- **Web Tier:** Allow HTTP/HTTPS from internet, SSH from bastion
- **App Tier:** Allow traffic only from web tier security group
- **Database Tier:** Allow database ports only from app tier
- **Bastion Host:** Allow SSH from specific IP ranges only
- **Load Balancer:** Allow HTTP/HTTPS from internet, health checks

Troubleshooting Security Issues

Connection Timeout

Usually Security Group issue - check inbound rules

Connection Refused

Usually application issue - service not running

Intermittent Issues

Check NACL rules and rule order

Can't Reach Internet

Check outbound rules and routing

Security Reminder: Security Groups are stateful (return traffic automatically allowed), but NACLs are stateless (must explicitly allow return traffic). Always consider both layers when designing security.

Prepared By: Rashi Rana
AWS Corporate Trainer

VPC Connectivity Options

AWS provides multiple options for connecting VPCs to each other, to on-premises networks, and to AWS services privately. Choose the right connectivity option based on your requirements.

VPC Peering

What it is

Direct network connection between two VPCs using private IP addresses

Scope

Same region, cross-region, cross-account connections supported

Routing

Not transitive - must create direct peering for each connection

Cost

Data transfer charges apply for cross-AZ and cross-region traffic

VPN Connections

Type	Use Case	Bandwidth	Redundancy
Site-to-Site VPN	Connect on-premises to VPC	Up to 1.25 Gbps per tunnel	Two tunnels per connection
Client VPN	Remote user access to VPC	Varies by endpoint	Multiple AZ deployment
Transit Gateway	Hub for multiple connections	Up to 50 Gbps	Built-in redundancy

AWS Direct Connect

- **Dedicated Connection:** Private connection from on-premises to AWS
- **Bandwidth Options:** 1 Gbps to 100 Gbps dedicated connections
- **Virtual Interfaces:** Multiple VLANs over single physical connection
- **Consistent Performance:** Predictable bandwidth and latency
- **Cost Benefits:** Reduced data transfer costs for high volume

VPC Endpoints

Gateway Endpoints

S3 and DynamoDB access without internet gateway

Interface Endpoints

Private access to AWS services via ENI with private IP

Gateway Load Balancer Endpoints

Benefits

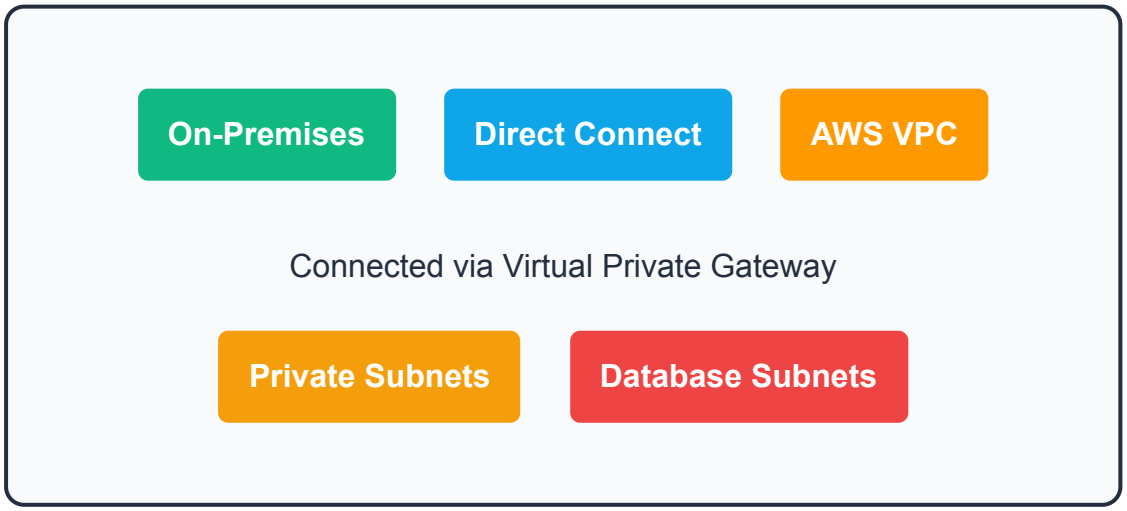
Access third-party appliances in service provider VPC

Improved security, reduced NAT costs, better performance

Connectivity Decision Matrix

Requirement	Recommended Solution	Alternative
VPC to VPC (same region)	VPC Peering	Transit Gateway
Multiple VPC connections	Transit Gateway	Multiple VPC Peering
On-premises to AWS	Direct Connect	Site-to-Site VPN
Remote user access	Client VPN	Bastion Host
AWS service access	VPC Endpoints	NAT Gateway

Hybrid Cloud Architecture Example



Connectivity Best Practice: Design your connectivity strategy based on bandwidth requirements, latency sensitivity, security needs, and cost considerations. Use VPC endpoints to reduce NAT Gateway costs for AWS service traffic.

Prepared By: Rashi Rana
AWS Corporate Trainer

VPC Best Practices

Following VPC best practices ensures optimal security, performance, cost-effectiveness, and maintainability of your AWS network infrastructure.

Network Design Principles

Plan for Scale

Design with future growth in mind, use appropriate CIDR blocks

Multi-AZ Design

Distribute resources across multiple AZs for high availability

Security Layers

Implement defense in depth with multiple security controls

Cost Optimization

Use VPC endpoints, optimize data transfer, right-size resources

IP Address Planning

- **Use RFC 1918 Ranges:** Always use private IP address ranges
- **Avoid Overlaps:** Ensure no overlap with on-premises or other VPCs

- **Plan for Growth:** Choose larger CIDR blocks than immediately needed
- **Document Allocation:** Maintain IP address management documentation
- **Reserve Space:** Leave room for future services and expansion

Security Best Practices

Layer	Best Practice	Implementation
Network	Least privilege access	Restrictive security groups and NACLs
Subnet	Proper tier separation	Public, private, and database subnets
Instance	No direct internet access	Use bastion hosts or Session Manager
Monitoring	Enable logging	VPC Flow Logs, CloudTrail

High Availability Design

Multi-AZ Architecture Pattern

```
Region: us-east-1
VPC: 10.0.0.0/16

Availability Zone 1a:
- Public Subnet: 10.0.1.0/24
- Private Subnet: 10.0.11.0/24
- Database Subnet: 10.0.21.0/24

Availability Zone 1b:
- Public Subnet: 10.0.2.0/24
```

```
- Private Subnet: 10.0.12.0/24
- Database Subnet: 10.0.22.0/24

Availability Zone 1c:
- Public Subnet: 10.0.3.0/24
- Private Subnet: 10.0.13.0/24
- Database Subnet: 10.0.23.0/24
```

Cost Optimization Strategies

- **VPC Endpoints:** Use for AWS service access to avoid NAT costs
- **NAT Gateway Optimization:** Single NAT for dev, multiple for prod
- **Data Transfer:** Minimize cross-AZ and cross-region transfers
- **Reserved Capacity:** Use Reserved Instances for predictable workloads
- **Right Sizing:** Monitor and optimize instance sizes

Monitoring and Logging

VPC Flow Logs

Monitor network traffic patterns and security

CloudWatch Metrics

Monitor NAT Gateway, VPN, and Direct Connect metrics

AWS Config

Track configuration changes and compliance

CloudTrail

Log API calls and changes to VPC resources

Naming and Tagging Strategy

```
Naming Convention Examples:

VPC: prod-vpc-us-east-1
Subnets: prod-public-1a, prod-private-1a, prod-db-1a
Security Groups: prod-web-sg, prod-app-sg, prod-db-sg
Route Tables: prod-public-rt, prod-private-rt

Required Tags:
- Environment: prod/dev/test
- Project: project-name
- Owner: team-name
- CostCenter: cost-center-code
```

Common Anti-Patterns to Avoid

Avoid These Mistakes:

- Using default VPC for production workloads
- Creating overly large or small CIDR blocks
- Not planning for multi-AZ deployment
- Overly permissive security group rules
- Not using VPC endpoints for AWS services
- Ignoring data transfer costs
- Poor naming and tagging practices

Golden Rule: Design your VPC with security, scalability, and cost-effectiveness in mind from the beginning. It's much easier to plan properly upfront than to refactor later.

Prepared By: Rashi Rana
AWS Corporate Trainer

Lab: Creating a VPC with Public and Private Subnets

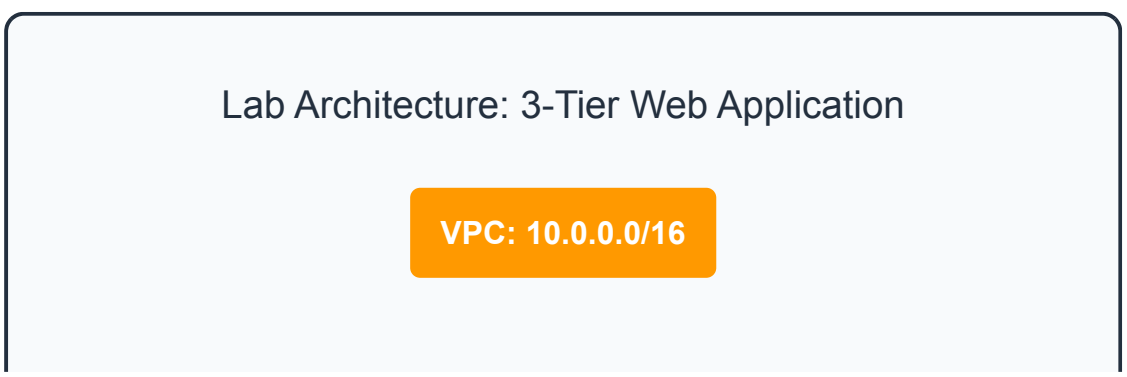
Hands-on Lab: Build a Multi-Tier VPC Architecture

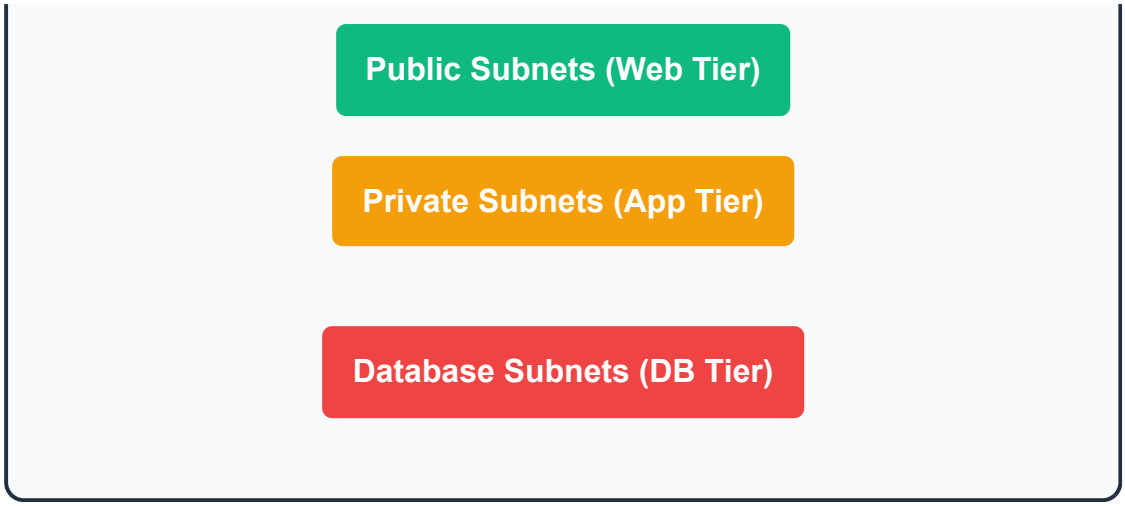
Create a production-ready VPC with public and private subnets across multiple AZs, including internet connectivity and security configuration.

Lab Objectives

- Create a VPC with appropriate CIDR block
- Set up public and private subnets in multiple AZs
- Configure Internet Gateway and NAT Gateway
- Create and configure route tables
- Set up security groups for different tiers

Architecture Overview





Step 1: Create VPC

1. **Navigate to VPC:** Go to VPC service in AWS Console
2. **Create VPC:** Click "Create VPC"
3. **VPC Settings:** Name: "MyLab-VPC", CIDR: 10.0.0.0/16
4. **Tenancy:** Default tenancy
5. **Tags:** Add Environment: Lab, Project: VPC-Training

Step 2: Create Subnets

```
Subnet Plan:

Public Subnets (Web Tier):
- MyLab-Public-1A: 10.0.1.0/24 (us-east-1a)
- MyLab-Public-1B: 10.0.2.0/24 (us-east-1b)

Private Subnets (App Tier):
- MyLab-Private-1A: 10.0.11.0/24 (us-east-1a)
- MyLab-Private-1B: 10.0.12.0/24 (us-east-1b)

Database Subnets (DB Tier):
- MyLab-DB-1A: 10.0.21.0/24 (us-east-1a)
- MyLab-DB-1B: 10.0.22.0/24 (us-east-1b)
```

Step 3: Create Internet Gateway

1. **Create IGW:** Name: "MyLab-IGW"
2. **Attach to VPC:** Select MyLab-VPC
3. **Verify:** Check attachment status is "Attached"

Step 4: Create NAT Gateway

```
# Create NAT Gateway in each public subnet for HA 1. Navigate to NAT Gateways 2. Create NAT Gateway: - Name: MyLab-NAT-1A - Subnet: MyLab-Public-1A - Connectivity: Public - Allocate Elastic IP 3. Repeat for MyLab-Public-1B (MyLab-NAT-1B)
```

Step 5: Configure Route Tables

Route Table	Associated Subnets	Routes
MyLab-Public-RT	Public subnets	10.0.0.0/16 → local 0.0.0.0/0 → IGW
MyLab-Private-RT-1A	Private subnet 1A	10.0.0.0/16 → local 0.0.0.0/0 → NAT-1A
MyLab-Private-RT-1B	Private subnet 1B	10.0.0.0/16 → local 0.0.0.0/0 → NAT-1B
MyLab-DB-RT	Database subnets	10.0.0.0/16 → local

Step 6: Create Security Groups

```
# Web Tier Security Group Name: MyLab-Web-SG Inbound Rules: - HTTP (80) from 0.0.0.0/0 - HTTPS (443) from
```

```
0.0.0.0/0 - SSH (22) from MyLab-Bastion-SG # App Tier Security Group Name: MyLab-App-SG Inbound Rules: - Custom TCP (8080) from MyLab-Web-SG - SSH (22) from MyLab-Bastion-SG # Database Tier Security Group Name: MyLab-DB-SG Inbound Rules: - MySQL (3306) from MyLab-App-SG - SSH (22) from MyLab-Bastion-SG # Bastion Host Security Group Name: MyLab-Bastion-SG Inbound Rules: - SSH (22) from YOUR-IP/32
```

> **Lab Complete:** You've successfully created a production-ready VPC architecture with proper security, routing, and high availability across multiple AZs!

Prepared By: Rashi Rana
AWS Corporate Trainer

VPC Troubleshooting

Understanding common VPC issues and their solutions is crucial for maintaining reliable network connectivity and security in your AWS environment.

Common Connectivity Issues

Cannot Access Internet

Check IGW attachment, route tables, security groups, and public IP assignment

Cannot Access from Internet

Verify public IP, security group inbound rules, and NACL settings

Private Subnet No Outbound

Check NAT Gateway status, route table configuration, and security groups

VPC Peering Issues

Verify peering connection status, route tables, and DNS resolution

Systematic Troubleshooting Approach

1. **Check Route Tables:** Verify routes exist for traffic destination

2. **Verify Security Groups:** Ensure inbound/outbound rules allow traffic
3. **Check NACLs:** Confirm subnet-level rules permit traffic
4. **Validate Gateways:** Check IGW/NAT Gateway status and configuration
5. **Test Connectivity:** Use ping, telnet, or traceroute for diagnosis
6. **Review Logs:** Check VPC Flow Logs for traffic patterns

Troubleshooting Tools

Tool	Purpose	Use Case
VPC Flow Logs	Network traffic analysis	Security analysis, troubleshooting
Reachability Analyzer	Path analysis between resources	Connectivity troubleshooting
CloudWatch Metrics	Performance monitoring	NAT Gateway, VPN monitoring
AWS Config	Configuration compliance	Security group auditing

Common Error Messages and Solutions

```
Error: "Connection timed out"
Solution: Check security group inbound rules

Error: "Connection refused"
Solution: Check if service is running on target port

Error: "Network unreachable"
Solution: Check route table configuration
```

```
Error: "No route to host"
Solution: Verify IGW/NAT Gateway and routing

Error: "DNS resolution failed"
Solution: Check DNS settings and VPC DNS options
```

Performance Troubleshooting

- **High Latency:** Check cross-AZ traffic, optimize placement groups
- **Low Throughput:** Verify instance types, enhanced networking
- **Packet Loss:** Check security group limits, instance performance
- **DNS Issues:** Verify VPC DNS resolution and hostnames settings

Security Troubleshooting

Unexpected Traffic

Review VPC Flow Logs, check security group rules

Access Denied

Verify IAM permissions, resource-based policies

Data Exfiltration

Monitor outbound traffic, implement DLP controls

Compliance Issues

Use AWS Config rules, regular security audits

Monitoring and Alerting

```
# Key metrics to monitor: - NAT Gateway bandwidth
utilization - VPN connection state - Security group rule
changes - Unusual traffic patterns - Failed connection
```

```
attempts # Set up CloudWatch alarms for: - High data transfer costs - NAT Gateway errors - VPN tunnel state changes - Unusual network activity
```

> **Troubleshooting Tip:** Always start with the most basic checks (route tables, security groups) before moving to more complex diagnostics. Document your troubleshooting steps for future reference.

Prepared By: Rashi Rana
AWS Corporate Trainer

Summary and Next Steps

Key Concepts Mastered

- Understanding of VPC fundamentals and core components
- CIDR notation and IP address calculation skills
- Subnet design and multi-AZ architecture patterns
- Route table configuration and traffic routing
- Internet connectivity with IGW and NAT Gateways
- Security implementation with Security Groups and NACLs
- VPC connectivity options and hybrid architectures
- Best practices for design, security, and cost optimization
- Hands-on experience building production-ready VPC
- Troubleshooting common VPC connectivity issues

Practical Skills Gained

Network Design

Plan and implement scalable VPC architectures

CIDR Calculations

Calculate IP ranges and subnet sizes accurately

Security Configuration

Connectivity Solutions

Implement layered security with proper access controls

Choose and implement appropriate connectivity options

Lab Accomplishments

- **VPC Creation:** Built complete VPC with proper CIDR planning
- **Multi-AZ Design:** Implemented high availability across AZs
- **Internet Connectivity:** Configured IGW and NAT Gateways
- **Security Implementation:** Created layered security with SGs and NACLs
- **Route Configuration:** Set up proper routing for all tiers

Associate-Level Exam Topics Covered

Domain	Topics Covered	Exam Weight
Design Resilient Architectures	Multi-AZ design, connectivity options	26%
Design High-Performing Architectures	Network optimization, placement strategies	24%
Design Secure Applications	Security Groups, NACLs, network isolation	30%
Design Cost-Optimized Architectures	NAT optimization, VPC endpoints	20%

VPC Design Checklist

1. **CIDR Planning:** Choose appropriate IP ranges, avoid overlaps

2. **Multi-AZ Design:** Distribute subnets across multiple AZs
3. **Security Layers:** Implement Security Groups and NACLs
4. **Internet Access:** Configure IGW and NAT Gateways appropriately
5. **Routing:** Set up proper route tables for each subnet type
6. **Monitoring:** Enable VPC Flow Logs and CloudWatch monitoring
7. **Documentation:** Maintain clear network documentation

Advanced Topics for Further Learning

- **AWS Transit Gateway:** Centralized connectivity hub for multiple VPCs
- **AWS PrivateLink:** Private connectivity to AWS services and SaaS
- **AWS Direct Connect:** Dedicated network connections to AWS
- **AWS Client VPN:** Managed client-based VPN service
- **Network Load Balancer:** High-performance load balancing
- **AWS Global Accelerator:** Improve global application performance

Real-World Implementation Tips

Start Simple

Begin with basic VPC, add complexity gradually

Plan for Scale

Design with future growth and requirements in mind

Security First

Monitor Costs

Implement security controls from the beginning

Track data transfer and NAT Gateway costs

Recommended Next Steps

1. **Practice:** Build different VPC architectures in your AWS account
2. **Certification:** Apply knowledge to AWS Solutions Architect Associate exam
3. **Advanced Learning:** Explore Transit Gateway and PrivateLink
4. **Real Projects:** Implement VPC designs for actual workloads
5. **Stay Updated:** Follow AWS networking service updates and best practices

Congratulations! You've mastered AWS VPC fundamentals and are well-prepared to design and implement secure, scalable network architectures in AWS. Continue practicing with real-world scenarios and exploring advanced networking services.

Prepared By: Rashi Rana
AWS Corporate Trainer