(Senior Academy - IT training center)

The Place You Can Be A Senior

www.seniorsteps.net
https://www.facebook.com/seniorsteps.it
contact us: 0224153419 - 01090873748

فرع مدينة نصر 1 : عمارة 4 – شارع محمد توفيق دياب – عباس العقاد - مدينة نصر – الدورال 1

DevOps Engineer Diploma

DevOps Engineer Diploma

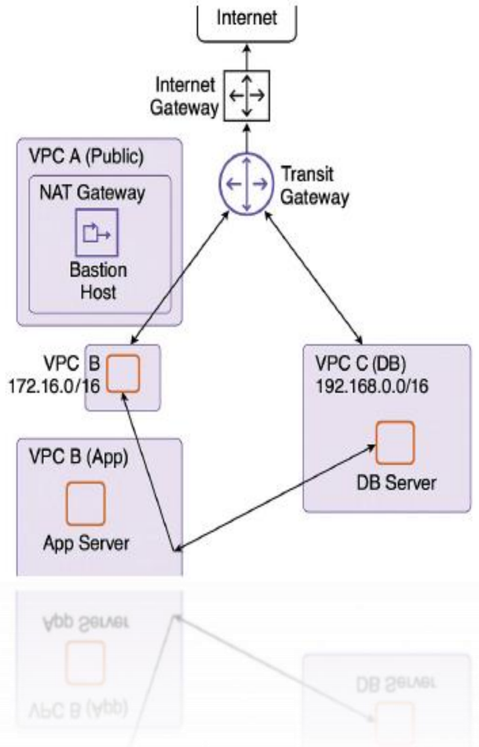
AWS Labs
Lab 17

Design and Configure a Multi-VPC AWS Network Architecture Using a Transit Gateway

### Lab Objectives

- **Set up three VPCs** (Public, App, and DB) with the CIDR ranges shown in the diagram.
- **Implement proper routing** between VPCs using an AWS Transit Gateway.
- **Deploy required components** such as a NAT Gateway, Bastion Host, App Server, and DB Server.
- **Enable secure connectivity** between all tiers following the architecture design.
- **Ensure controlled and secure internet access** via NAT and Internet Gateway configurations.
- **Apply best-practice segmentation** for public, application, and database layers.

Task



Design, deploy, and document a multi-VPC network architecture on AWS using a Transit Gateway to interconnect three isolated tiers—Public (Bastion + NAT), Application (App Server), and Database (DB Server)—while enforcing best practices for routing, segmentation, controlled internet access, and secure east-west communication.

Architecture Components

1) VPC A (Public Tier)

- **Role:** Acts as the public entry point and outbound internet provider.
- **Components:**
  • NAT Gateway for outbound internet access from private VPCs.
  • Bastion Host for secure SSH/administrative access.
- **Connectivity:**
  • Attached to the Internet Gateway.
  • Connected to the Transit Gateway for internal traffic.
- **Exposure:**
  • Public subnet for Bastion and NAT.

• Acts as the only internet-facing VPC.

---

2) VPC B (Application Tier)

- **Role:** Hosts the application server providing business logic or API functionality.
- **Connectivity:**
  • Communicates with VPC C (DB) strictly via Transit Gateway.
  • Routes all outbound internet traffic through VPC A's NAT Gateway.
- **Security:**
  • No direct internet connectivity.
  • Access limited to Bastion Host and DB tier as needed.

---

3) VPC C (Database Tier)

- **Role:** Contains the database layer used only by the application server.
- **Exposure:**
  • Internal use only—no public access.
- **Connectivity:**
  • App Server in VPC B is the only allowed consumer.
  • Connected to the Transit Gateway for controlled private routing.
- **Security:**
  • Highly restricted security groups; no inbound internet routes.
  • Optional network ACL hardening.

Transit Gateway (Core Interconnect Layer)

- **Purpose:** Central routing hub allowing scalable VPC-to-VPC communication.
- **Configuration:**
  • Attachments for VPC A, VPC B, and VPC C.
  • Route tables designed so that App → DB communication is permitted while maintaining strict segmentation.
  • Ensures all east-west traffic is isolated from the internet.

Deployment on AWS (Conceptual Flow)

1. VPC Creation

- Provision the three VPCs with their CIDR blocks:
  • VPC A (Public): *as shown in diagram*
  • VPC B (App): 172.16.0.0/16
  • VPC C (DB): 192.168.0.0/16

- Create subnets, route tables, and appropriate associations.

---

2. IGW, NAT, and Public Access

- Attach an Internet Gateway to VPC A.
- Deploy the NAT Gateway inside a public subnet.
- Configure routing so only VPC B and VPC C use this NAT via Transit Gateway when accessing the internet.

---

3. Transit Gateway Integration

- Attach all VPCs to the Transit Gateway.
- Configure propagation and association for TGW route tables.
- Ensure:
  • App Tier can reach DB Tier.
  • Bastion Host can reach both private VPCs.
  • No unintended cross-tier lateral exposure.

---

4. Database Tier

- Deploy the DB Server in a private subnet in VPC C.
- Restrict inbound access solely to App Server in VPC B.
- Ensure no direct NAT/IGW access.

---

5. Application Tier

- Deploy App Server in VPC B private subnet.
- Configure routing for:
  • Outbound → NAT (via VPC A).
  • East-west → DB Server (via Transit Gateway).
- Enforce least-privilege security rules.

---

6. Bastion Host & Administrative Access

- Deploy Bastion Host in VPC A public subnet.
- Allow controlled SSH access to App and DB servers (via security groups and TGW).

---

7. Networking & Routing

- Validate all VPC route tables:
  • Correct TGW attachments.
  • NAT routing for private tiers.
  • No direct public exposure for App/DB.
- Ensure DNS resolution is enabled in each VPC.

8. Security & Governance

- Apply restrictive security group rules per tier.
- Implement IAM least-privilege for administrative roles.
- Enforce segmentation by blocking unused ports and routes.

9. Scalability & Resilience

- Allow independent scaling of each server tier.
- Optionally replicate App or DB components as needed.
- Design routing and TGW attachments to support multi-AZ deployments.

10. Observability & Operations

- Collect VPC Flow Logs for monitoring internal/external traffic patterns.
- Track the health of Bastion, App, and DB servers.
- Monitor NAT Gateway usage, TGW metrics, and subnet routing behavior.

Verification Checklist (No Commands)

- Three VPCs created with correct CIDR ranges and attachments.
- Transit Gateway configured with proper associations and propagations.
- VPC A serves as the only internet-facing VPC through IGW/NAT.
- App Server can reach DB Server; DB cannot reach internet.
- Bastion Host can reach App and DB servers securely.
- No tier has unintended public access.
- All routing tables reflect correct TGW paths.
- Security groups enforce least privilege across all tiers.

Deliverables (Suggested Evidence)

- Short summary describing the network, components, and traffic flow.

- Screenshots of:
  • VPCs, subnets, and route tables.
  • Transit Gateway attachments and route tables.
  • Bastion, App, and DB instances running.
  • Security groups showing allowed paths.
  • NAT Gateway and IGW setup.

Expected Outcome

You will have a production-style, secure, multi-VPC AWS architecture featuring:

- Clear separation between public, application, and database tiers.
- Controlled internal communication through Transit Gateway.
- Centralized outbound internet access via NAT.
- Strong security boundaries and least-privilege controls.
- Operational observability and routing transparency.

Cleanup (Conceptual)

• Detach and remove TGW attachments.
• Delete NAT Gateway, IGW, and VPCs.
• Remove EC2 instances (Bastion/App/DB).
• Clean up route tables, subnets, and security groups.
• Optionally archive screenshots before teardown.

You are Welcome

Senior Steps - IT training center
The place You can be A Senior