PRJ-NET-035: Centralized Egress and Inspection VPC

Certification: AWS Certified Advanced Networking – Specialty
Domain: Network Security and Inspection

1. Project Overview

This project demonstrates how to build a centralized architecture for inspecting all outbound (egress) traffic from multiple VPCs. In a large AWS environment, giving each VPC its own Internet Gateway and NAT Gateway can be costly and creates a security blind spot, as there is no single point to monitor or filter traffic. This project solves that by routing all internet-bound traffic from multiple “spoke” VPCs through a single, dedicated “Inspection VPC”.

Inside the Inspection VPC, we will deploy a fleet of third-party firewall appliances (which we will simulate with a simple Linux instance) behind a **Gateway Load Balancer (GWLB)**. This architecture allows for highly scalable, resilient, and centralized traffic inspection, ensuring that all outbound connections adhere to corporate security policies before reaching the internet.

Key Objectives

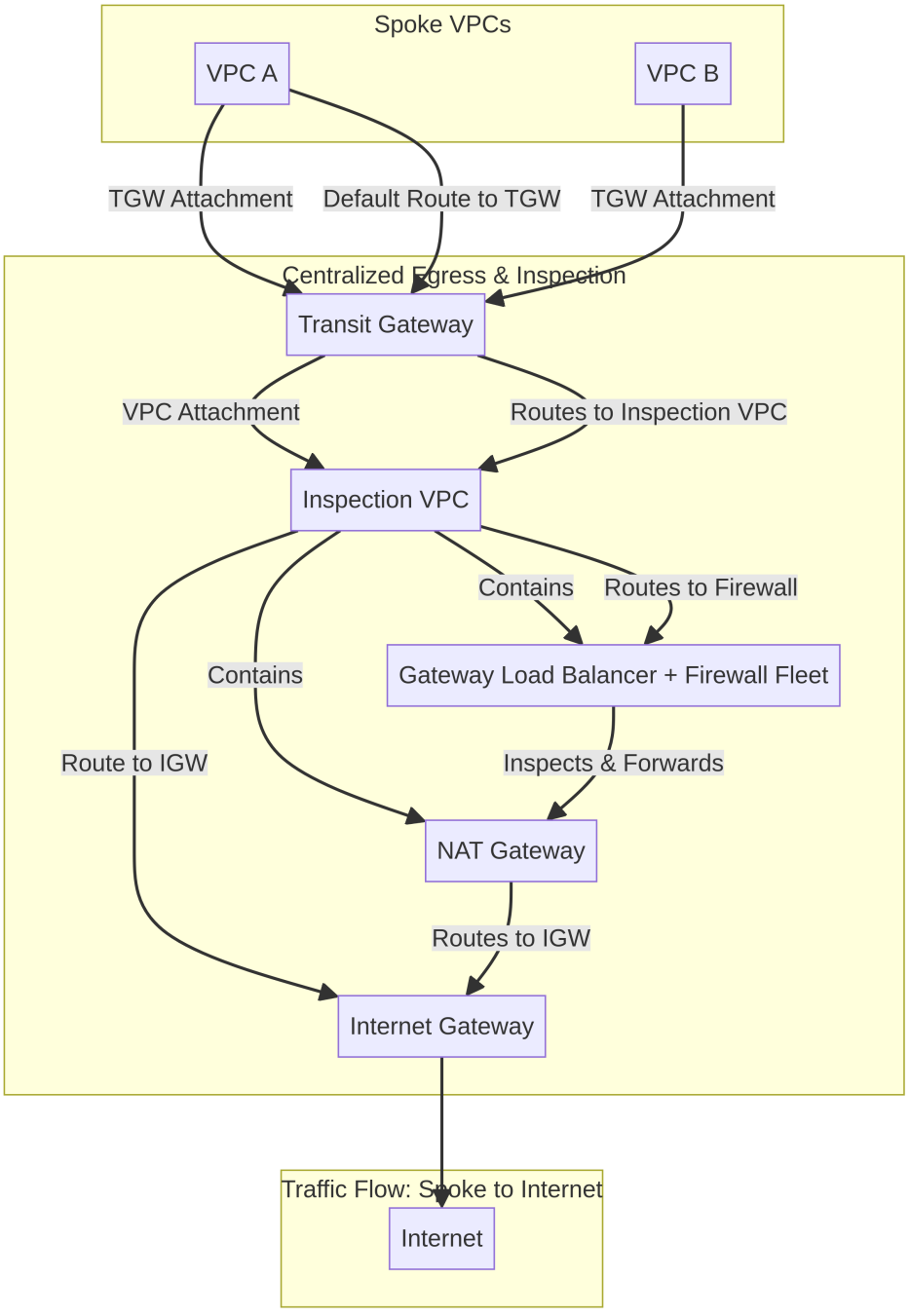
- Design a hub-and-spoke network topology with Transit Gateway for centralized routing.
- Create a dedicated Inspection VPC to house security appliances.
- Deploy a Gateway Load Balancer (GWLB) to distribute traffic to a fleet of virtual firewalls.
- Configure complex routing to force all internet-bound traffic from spoke VPCs through the inspection appliances.

- Understand the concept of “bump-in-the-wire” insertion of security appliances using GWLB Endpoints.
- Achieve a scalable and highly available security inspection solution.

---

2. Architecture

The architecture uses Transit Gateway to centralize routing and the Gateway Load Balancer to transparently insert a fleet of security appliances into the network path.



Traffic Flow (Spoke VPC to Internet):

1. **Initial Hop:** An EC2 instance in a private subnet of a **Spoke VPC** (e.g., VPC A) needs to access the internet. Its route table has a default route (`0.0.0.0/0`) that points to the **Transit Gateway**.
2. **Central Routing:** The Transit Gateway receives the packet. It has a route table that directs all default traffic to the attachment for the **Inspection VPC**.
3. **To the Firewall:** The packet arrives in the TGW attachment subnet of the Inspection VPC. This subnet has a route table that directs all traffic to the **Gateway Load Balancer Endpoint (GWLBE)**.
4. **Inspection:** The GWLBE acts as a transparent "bump-in-the-wire". It forwards the packet to the **Gateway Load Balancer**, which in turn selects one of the registered firewall instances. The firewall inspects the packet, and if the traffic is allowed, it sends the packet back to the GWLB.
5. **To the Internet:** The GWLB sends the approved packet back to the GWLBE. The GWLBE's subnet has a route table that now directs the traffic to a **NAT Gateway**.
6. **Egress:** The NAT Gateway performs network address translation and sends the packet out to the internet via the **Internet Gateway (IGW)** attached to the Inspection VPC.

This entire process is symmetrical. Return traffic from the internet follows the reverse path, ensuring that it is also inspected by the same firewall instance that handled the outbound request.

---

3. Prerequisites

- An AWS account with administrative permissions.
- At least one "spoke" VPC with a private subnet.
- A Transit Gateway (can be created as part of the guide).
- An AMI for a firewall appliance (we will use a standard Amazon Linux 2 AMI for simulation).

---

4. Step-by-Step Implementation Guide

Step 4.1: Set Up the Network Foundation

1. **Create VPCs:**
    - **Inspection VPC:** Create a VPC with a CIDR of `10.10.0.0/16`. Create three subnets:
        - `TGW-Attachment-Subnet` (`10.10.1.0/24`)
        - `GWLBE-Subnet` (`10.10.2.0/24`)
        - `Firewall-Subnet` (`10.10.3.0/24`)
    - **Spoke VPC:** Use an existing VPC or create a new one (e.g., `10.1.0.0/16`) with a private subnet.
2. **Create Transit Gateway:** Deploy a Transit Gateway in your region if you don't already have one.
3. **Attach VPCs to TGW:**
    - Create a TGW attachment for the **Spoke VPC**.
    - Create a TGW attachment for the **Inspection VPC**, placing the attachment in the `TGW-Attachment-Subnet`.

Step 4.2: Deploy the Inspection Fleet

1. **Create Gateway Load Balancer (GWLB):**
    - Go to the **EC2 Console -> Load Balancers -> Create Load Balancer**.
    - Choose **Gateway Load Balancer**.
    - **Name:** `inspection-gwlb`
    - **VPC:** Select the **Inspection VPC**.
    - **Target Group:** Create a new target group.
        - **Target type:** Instances
        - **Protocol:** GENEVE
        - **Port:** 6081

        - **VPC:** Inspection VPC

2. **Launch Firewall Instances:**
    - Launch one or more EC2 instances from your chosen firewall AMI into the `Firewall-Subnet`.
    - **Important:** In the instance's network configuration, ensure that **Source/Destination Check** is **disabled**. This allows the instance to handle traffic for other IP addresses.
    - Register these instances with the target group you created for the GWLB.

Step 4.3: Configure the GWLB Endpoints

1. **Create GWLB Endpoint Service:**
    - Go to the **VPC Console -> Endpoint Services -> Create endpoint service**.
    - **Name:** `inspection-service`
    - **Load balancer:** Select the `inspection-gwlb` you created.
    - Require acceptance for the endpoint.

2. **Create GWLB Endpoint:**
    - Go to **Endpoints -> Create endpoint**.
    - **Service category:** Find services by name.
    - **Service name:** Enter the service name of the endpoint service you just created.
    - **VPC:** Select the **Inspection VPC**.
    - **Subnet:** Place the endpoint in the `GWLBE-Subnet`.
    - Accept the endpoint connection request in the Endpoint Services console.

Step 4.4: Configure Routing (The Critical Part)

This is where we stitch everything together.

1. **Spoke VPC Route Table:**

- In the route table for the private subnet in your **Spoke VPC**, add a default route (`0.0.0.0/0`) with the target set to your **Transit Gateway**.

2. **Transit Gateway Route Tables:**
   - Create a new TGW route table, `Spoke-TGW-RT` .
   - Associate your **Spoke VPC attachment** with this route table.
   - In `Spoke-TGW-RT` , add a default route (`0.0.0.0/0`) that points to the **Inspection VPC attachment**.
   - In the default TGW route table (associated with the Inspection VPC), add a route for the Spoke VPC's CIDR (`10.1.0.0/16`) pointing to the Spoke VPC attachment. This handles the return traffic.
3. **Inspection VPC Route Tables:**
   - **TGW Attachment Subnet (`10.10.1.0/24`):**
     - Create a route table for this subnet.
     - Add a default route (`0.0.0.0/0`) with the target set to the **GWLB Endpoint**.
   - **GWLBE Subnet (`10.10.2.0/24`):**
     - Create a route table for this subnet.
     - Add a default route (`0.0.0.0/0`) with the target set to the **NAT Gateway** (which you will create in the next step).
   - **Firewall Subnet (`10.10.3.0/24`):**
     - Create a route table for this subnet.
     - Add a default route (`0.0.0.0/0`) with the target set to the **NAT Gateway**.

Step 4.5: Finalize Egress Path

1. **Create NAT Gateway:** Create a NAT Gateway and place it in a public subnet within your Inspection VPC (you may need to create a new public subnet for this).
2. **Create Internet Gateway:** Create and attach an IGW to your Inspection VPC.
3. **Update Public Subnet Route Table:** Ensure the public subnet's route table has a default route pointing to the IGW.

5. How to Test

1. SSH into an EC2 instance in the private subnet of your **Spoke VPC** (you may need a bastion host for this).
2. From that instance, try to ping an external IP address (e.g., `ping 8.8.8.8`).
3. If your routing is correct, the traffic will flow through the entire architecture and succeed.
4. You can use **VPC Flow Logs** and **Transit Gateway Flow Logs** to trace the path of the traffic and verify it is passing through the correct components.
5. On your firewall instance, you can use `tcpdump` to see the GENEVE-encapsulated traffic arriving from the GWLB.

---

6. Cleanup

1. **Delete the GWLB Endpoint and Endpoint Service.**
2. **Delete the Gateway Load Balancer.**
3. **Terminate the firewall instances.**
4. **Delete the NAT Gateway and Internet Gateway.**
5. **Delete the Transit Gateway attachments and the Transit Gateway itself.**
6. **Remove the custom route table entries** from all VPCs.

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)