Amazon Virtual Private Cloud (VPC)

Was ist eine Virtual Private Cloud (VPC)?

- **Definition**: Eine VPC ist eine isolierte Netzwerkumgebung in AWS, die du für maximale Kontrolle und Sicherheit konfigurieren kannst.
- **Anwendungsfall**: Ermöglicht es, Netzwerke für verschiedene Workloads sicher zu erstellen, z. B. Webanwendungen in öffentlichen Subnetzen und Datenbanken in privaten Subnetzen.
- **Vorteil**: Volle Kontrolle über Netzwerkarchitektur, Zugriff, Routing und Sicherheit.

VPC-Komponenten im Überblick

1. **Subnets**: Unterteilungen der VPC in kleinere Netzwerksegmente.
    - **Öffentliche Subnetze**: Ermöglichen Instanzen den Zugriff auf das Internet über ein Internet Gateway.
    - **Private Subnetze**: Ohne direkten Internetzugang, oft für interne Systeme und Datenbanken.
2. **Internet Gateway (IGW)**: Bietet Internetzugang für öffentliche Subnetze.
3. **NAT Gateway (Network Address Translation)**: Ermöglicht ausgehende Verbindungen von privaten Subnetzen, ohne die Instanzen öffentlich erreichbar zu machen.
4. **VPC Endpoints**: Direkte Verbindung zu AWS-Diensten wie S3 oder DynamoDB, ohne das Internet zu verwenden. Ideal für sicherheitskritische Anwendungen.
5. **Route Tables**: Definieren, wie Traffic innerhalb der VPC und zu externen Netzwerken geroutet wird.

Subnetze in einer VPC

Subnetze: Öffentlich vs. Privat

- **Öffentliche Subnetze**:
  - Verbunden mit einem Internet Gateway.
  - Für Instanzen, die auf das Internet zugreifen oder öffentlich erreichbar sein müssen (z. B. Webserver).
  - Beispiele: Webserver, Load Balancer.
- **Private Subnetze**:
  - Verwenden ein NAT Gateway oder VPC Endpoints für ausgehenden Verkehr.
  - Keine direkte öffentliche Erreichbarkeit.
  - Ideal für Datenbanken und interne Anwendungen, die keine Internetverbindung benötigen.

Internet Gateway (IGW)

- **Definition**: Komponente, die einer VPC Internetzugang ermöglicht.
- **Funktion**: Stellt eine Brücke zwischen der VPC und dem Internet dar, um Datenverkehr zwischen Instanzen und externen Netzwerken zu ermöglichen.
- **Hinweis**: Muss mit der Route Table des öffentlichen Subnetzes verknüpft sein, damit der Traffic durch das IGW geleitet wird.

NAT Gateway (Network Address Translation)

- **Definition**: Ermöglicht Instanzen in privaten Subnetzen, ausgehende Verbindungen ins Internet aufzubauen.
- **Wichtig**: Das NAT Gateway selbst befindet sich in einem öffentlichen Subnetz, wodurch ausgehender Traffic aus dem privaten Subnetz über das NAT Gateway ins Internet gelangt.
- **Anwendungsfall**: Für Updates oder Downloads von internen Ressourcen ohne direkten Internetzugang.

Sicherheitskonfigurationen: Security Groups und NACLs

Sicherheitsgruppen (Security Groups)

- **Definition**: Zustandsbehaftete Firewalls auf Instanzebene, die den ein- und ausgehenden Datenverkehr steuern.
- **Funktionsweise**:
  - Eingehende Verbindungen sind standardmäßig blockiert, ausgehende standardmäßig erlaubt.
  - Regeln sind **zustandsbehaftet**, d. h., der eingehende Traffic, der erlaubt ist, ermöglicht automatisch den zugehörigen ausgehenden Traffic.
- **Beispiel**:
  - Nur SSH-Verbindungen auf Port 22 von bestimmten IP-Adressen erlauben.
  - HTTP/HTTPS-Verbindungen auf Port 80/443 von überall erlauben.

Network Access Control Lists (NACLs)

- **Definition**: Zustandslose Firewalls auf Subnetzebene, die den Traffic steuern.
- **Eigenschaften**:
  - Regeln für eingehenden und ausgehenden Traffic müssen explizit festgelegt werden.
  - **Zustandslos**: Jede Verbindung benötigt explizite Erlaubnis für beide Richtungen.
- **Anwendungsfall**: Zusätzlicher Schutz auf Subnetzebene zur Ergänzung von Sicherheitsgruppen.

Unterschiede zwischen Security Groups und NACLs

Eigenschaft	Security Groups	NACLs
Ebene	Instanzebene	Subnetzebene
Zustandsbehaftet?	Ja	Nein
Richtungen	Nur eingehend nötig, ausgehend automatisch erlaubt	Beide Richtungen nötig
Anwendungsfall	Zugangskontrolle für einzelne Instanzen	Zugangskontrolle für Subnetze

VPC Routing und Peering

Route Tables

- **Definition**: Route Tables enthalten Regeln, die bestimmen, wie Traffic innerhalb einer VPC und zu externen Netzwerken geleitet wird.
- **Komponenten**:
  - **Destination**: Ziel-IP-Bereich.
  - **Target**: Gateway oder Ziel, wohin der Traffic geleitet wird.
- **Beispiele**:
  - Route zu einem Internet Gateway für öffentlichen Subnetz-Traffic.
  - Route zu einem NAT Gateway für privaten Subnetz-Traffic.

VPC Peering und AWS Transit Gateway

- **VPC Peering**: Direkte Verbindung zwischen zwei VPCs.
  - **Anwendungsfall**: Ermöglicht private Kommunikation zwischen Ressourcen in verschiedenen VPCs, auch regionsübergreifend.
  - Einschränkungen: Punkt-zu-Punkt-Verbindung, begrenzt auf die VPCs, die explizit verbunden sind.
- **AWS Transit Gateway**: Hub-and-Spoke-Modell zur Verbindung mehrerer VPCs und On-Premise-Netzwerke.
  - **Vorteil**: Skalierbar und zentralisierte Steuerung des Traffics für große Netzwerkinfrastrukturen.

Erweiterte Sicherheitskomponenten und Monitoring

ELB (Elastic Load Balancing)

- **Definition**: Verteilt eingehenden Traffic auf mehrere EC2-Instanzen in verschiedenen Verfügbarkeitszonen.
- **Vorteile**:
  - Hohe Verfügbarkeit durch Verteilung der Last.
  - Skalierbarkeit, da der Load Balancer den Traffic je nach Bedarf automatisch anpassen kann.

VPC Flow Logs

- **Definition**: Erfassen den IP-Traffic in der VPC und dienen zur Überwachung und Analyse des Netzwerkverkehrs.
- **Anwendungsfall**:
  - Sicherheit: Aufdecken unautorisierter Zugriffe.
  - Fehlerbehebung: Analyse des Netzwerkverhaltens und Behebung von Verbindungsproblemen.
- **Einstellungen**: Können auf VPC-, Subnetz- oder Netzwerkschnittstellenebene aktiviert werden.

Zusammenfassung: Aufbau einer sicheren und skalierbaren VPC-Umgebung

Best Practices für eine VPC-Konfiguration

1. **VPC-Isolation und Subnetzplanung**:
    - Trenne öffentliche und private Subnetze.
2. **Routing und Internetzugang**:
    - Nutze Internet Gateway und NAT Gateway für den richtigen Zugriff.
3. **Sicherheitsgruppen und NACLs**:
    - Nutze Sicherheitsgruppen auf Instanzebene und NACLs auf Subnetzebene für maximalen Schutz.
4. **Monitoring und Logging**:
    - Aktiviere VPC Flow Logs und überwache den Traffic zur Fehleranalyse und für Sicherheit.

Anleitung zur Einrichtung einer sicheren und skalierbaren AWS-VPC-Infrastruktur für ein Startup

Ziel: Eine sichere Cloud-Infrastruktur auf AWS für eine Webanwendung aufzubauen, die öffentliche und private Subnetze, Netzwerkzugriffskontrollen und grundlegende Sicherheitsvorkehrungen umfasst.

1. Schritt: Erstellen der VPC

1. **VPC anlegen**:
    - Gehe zu **AWS Management Console** > **VPC Dashboard** > **VPCs** > **Create VPC**.
    - Gib einen Namen für die VPC ein, z. B. „Startup-VPC“.
    - Wähle einen **CIDR-Block** (z. B. `10.0.0.0/16`), um den IP-Bereich der VPC zu definieren.
    - Bestätige und erstelle die VPC.

2. **Subnets erstellen**:
    - Erstelle drei Subnetze innerhalb der VPC:
        - **Öffentliches Subnetz (für Webserver)**: Wähle eine Availability Zone (AZ), z. B. `eu-central-1a`, und verwende z. B. `10.0.5.0/24` für das Subnetz.
        - **Privates Subnetz 1 (für Datenbanken)**: In derselben oder einer anderen AZ, z. B. `eu-central-1a`, mit IP-Bereich `10.0.10.0/24`. In derselben AZ, um Latenz zu minimieren.
        - erstelle gerne weitere Subnetze (public und private) für Redundanz und Skalierbarkeit. (eu-central-1b, eu-central-1c). Wir brauchen allerdings pro AZ ein public und ein private Subnetz und entsprechend ein NAT Gateway pro AZ.

2. Schritt: Hinzufügen eines Internet Gateways

1. **Internet Gateway erstellen**:
    - Gehe zu **VPC Dashboard** > **Internet Gateways** > **Create Internet Gateway**.
    - Gib einen Namen ein, z. B. „Startup-IGW“.
    - Bestätige und erstelle das Internet Gateway.

2. **Internet Gateway an die VPC anhängen**:
    - Wähle das neu erstellte Internet Gateway aus und wähle „Attach to VPC“.
    - Wähle „Startup-VPC“ und bestätige.

3. Schritt: Route Tables einrichten

1. **Öffentliche Route Table für Internetzugang**:
    - Gehe zu **Route Tables** und erstelle eine neue Route Table, z. B. „Public-Route-Table“.
    - Wähle „Startup-VPC“ als zugehörige VPC.
    - Erstelle eine Route in dieser Tabelle:
        - **Destination**: `0.0.0.0/0` (für ausgehenden Internet-Traffic).
        - **Target**: Wähle das Internet Gateway „Startup-IGW“ aus.
    - Weise die **Public-Route-Table** dem öffentlichen Subnetz zu.

2. **Private Route Table für NAT-Zugriff**:
    - Erstelle eine weitere Route Table für private Subnetze, z. B. „Private-Route-Table“.
    - Füge diese Route Table den privaten Subnetzen hinzu, damit sie über das NAT Gateway auf das Internet zugreifen können (siehe nächster Schritt).

4. Schritt: NAT Gateway für private Subnetze hinzufügen

1. **NAT Gateway erstellen**:
    - Gehe zu **VPC Dashboard** > **NAT Gateways** > **Create NAT Gateway**.
    - Wähle das **öffentliche Subnetz** (10.0.5.0/24), in dem das NAT Gateway platziert werden soll.
    - Erstelle und verknüpfe eine **Elastic IP** (diese wird dem NAT Gateway zugewiesen).
    - Bestätige die Erstellung.

2. **NAT Gateway in die Private Route Table einbinden**:
    - Gehe zur **Private-Route-Table** und erstelle eine Route:
        - **Destination**: `0.0.0.0/0`.
        - **Target**: Wähle das erstellte NAT Gateway aus.

5. Schritt: Sicherheitsgruppen und NACLs konfigurieren

1. **Sicherheitsgruppe für den Webserver (Öffentliches Subnetz)**:
    - Gehe zu **Security Groups** > **Create Security Group**.
    - Benenne die Sicherheitsgruppe „Webserver-SG“ und wähle „Startup-VPC“.
    - Füge die folgenden Regeln hinzu:
        - Eingehend:
            - **HTTP (Port 80)**: Source `0.0.0.0/0` (öffentlich zugänglich).
            - **HTTPS (Port 443)**: Source `0.0.0.0/0` (öffentlich zugänglich).
            - **SSH (Port 22)**: Source auf spezifische IP-Adressen beschränken, z. B. nur Büro-IP.
        - Ausgehend: Erlaube **alle ausgehenden Verbindungen** (Standard).

2. **Sicherheitsgruppe für die Datenbank (Privates Subnetz)**:
    - Erstelle eine weitere Sicherheitsgruppe, z. B. „Database-SG“.
    - Regeln:
        - Eingehend:
            - **Datenbankport (z. B. MySQL - Port 3306)**: Nur Verbindungen von der **Webserver-SG** zulassen.
        - Ausgehend: Erlaube **alle ausgehenden Verbindungen** (Standard).

3. **NACLs für zusätzliche Subnetz-Sicherheit**:
    - Gehe zu **Network ACLs** und erstelle eine NACL für jedes Subnetz.
    - Öffentliche NACL: Erlaube HTTP, HTTPS und SSH-Eingänge für das öffentliche Subnetz, blockiere alles andere.
    - Private NACL: Erlaube nur Verbindungen von öffentlichen IPs, die in den Sicherheitsgruppen genehmigt sind, für das private Subnetz.

6. Schritt: Instanzen starten

1. **Webserver (öffentliche Instanz)**:
    - Gehe zu **EC2 Dashboard** > **Instances** > **Launch Instance**.
    - Wähle ein Amazon Machine Image (AMI), z. B. Amazon Linux
    - Setze die Instanz in das öffentliche Subnetz (10.0.5.0/24) und verknüpfe die Sicherheitsgruppe „Webserver-SG“.
    - Starte die Instanz.

2. **Datenbank (private Instanz)**:
    - Starte eine weitere EC2-Instanz.
    - Wähle ein privates Subnetz (10.0.10.0/24) und verknüpfe die Sicherheitsgruppe „Database-SG“.
    - Starte die Instanz.
    - Installiere und konfiguriere die Datenbanksoftware (z. B. MySQL) auf der privaten Instanz mit folgenden userdata-script:

```
#!/bin/bash
yum update -y
yum install -y mariadb-server
systemctl start mariadb
systemctl enable mariadb
# configure database and user
```

7. Schritt: Optional – Einrichtung eines Bastion Hosts

Falls es erforderlich ist, administrativen Zugang zu den privaten Instanzen (z. B. Datenbank) zu haben:

1. **Bastion Host (Jump Server)**:
    - Starte eine zusätzliche EC2-Instanz im öffentlichen Subnetz.
    - Konfiguriere den Bastion Host so, dass er nur SSH-Zugriff aus einem bestimmten IP-Bereich (z. B. Büro-IP) zulässt.
    - Greife über den Bastion Host auf die privaten Instanzen zu (SSH „Hopping“ von Bastion zu Datenbank-Server).

8. Schritt: Überwachung und Logging einrichten

1. **VPC Flow Logs aktivieren**:
    - Gehe zu **VPC Dashboard** > **Your VPCs** > Wähle die „Startup-VPC“ > **Actions** > **Create Flow Log**.
    - Wähle das Logging-Ziel, z. B. **CloudWatch Logs** oder **S3**.
    - Aktiviere Flow Logs, um IP-Traffic zu überwachen und potenzielle Sicherheitsbedrohungen aufzuspüren.

2. **AWS CloudTrail aktivieren**:
    - CloudTrail protokolliert alle API-Aufrufe und ist nützlich, um Änderungen in der VPC zu verfolgen.
    - Gehe zu **CloudTrail Dashboard** und aktiviere CloudTrail für die VPC.

Zusammenfassung

Das Startup hat nun eine sichere VPC-Architektur mit den folgenden Merkmalen:

- **Öffentliches Subnetz** für den Webserver, der über das Internet zugänglich ist.
- **Private Subnetze** für die Datenbank und interne Anwendungen.
- **Internet Gateway** und **NAT Gateway** zur Steuerung des Internetzugangs.
- **Security Groups und NACLs** zum Schutz der Ressourcen und zur Kontrolle des Netzwerkzugriffs.
- **VPC Flow Logs und CloudTrail** zur Überwachung und Aufzeichnung der Aktivitäten.

Diese Konfiguration bietet eine solide Grundlage für die sichere Bereitstellung und Skalierung einer Webanwendung in der AWS-Cloud.
"""