Fundamental Concepts

1. **What is AWS VPC (Virtual Private Cloud)?**
    - **Answer:** AWS VPC is a service that lets you provision a logically isolated section of the AWS Cloud where you can launch AWS resources (like EC2 instances) in a virtual network that you define.[1][2][3][4][5] It's like having your own private data center within AWS. You have complete control over your virtual networking environment, including selection of your own IP address range, creation of subnets, and configuration of route tables and network gateways.[2]
2. **Why would you use a VPC? What are its main benefits?**
    - **Answer:** The main benefits of using a VPC are:
        - **Isolation and Security:** It provides a private, isolated environment for your resources, separating them from other networks and AWS customers.[2] You control who can access your resources using security groups and network ACLs.[6][7]
        - **Network Control:** You define the network topology, including IP address ranges (CIDR blocks), subnets, route tables, and gateways (Internet Gateway, NAT Gateway).[5]
        - **Hybrid Cloud Capabilities:** VPCs can be securely connected to your on-premises data center using VPN connections or AWS Direct Connect.[6]
        - **Scalability:** Like other AWS services, VPC infrastructure is scalable.
3. **What is a CIDR block in the context of a VPC?**
    - **Answer:** CIDR (Classless Inter-Domain Routing) notation is how you define the IP address range for your VPC (e.g., 10.0.0.0/16). The number after the slash (/16) indicates how many bits are fixed for the network portion, which determines the total number of available IP addresses within the VPC. Choosing the right CIDR block is important for network design and preventing IP address overlap if you plan to connect multiple VPCs or connect to an on-premises network.

Core VPC Components

1. **What is a Subnet in VPC?**
    - **Answer:** A subnet is a range of IP addresses within your VPC.[1][2][3][8][9] You launch AWS resources, like EC2 instances, into subnets.[1][2][3][4][5] Each subnet must reside entirely within one Availability Zone (AZ) and cannot span AZs. Subnets allow you to segment your VPC network, often based on security requirements or application tiers.
2. **What's the difference between a Public Subnet and a Private Subnet?**
    - **Answer:**
        - **Public Subnet:** A subnet is considered "public" if its associated route table has a route directly to an Internet Gateway (IGW).[4][5][9] Instances in a public subnet can communicate directly with the internet if they have a public IP address or an Elastic IP address.[9] They are typically used for resources that need to be publicly accessible, like web servers or bastion hosts.

        - **Private Subnet:** A subnet is "private" if its route table does *not* have a direct route to the Internet Gateway.[9] Instances in a private subnet cannot be directly reached from the internet.[1][9] They are used for backend resources like databases or application servers that shouldn't be exposed publicly. They can optionally access the internet *outbound* via a NAT Gateway or NAT Instance located in a public subnet.

3. **What is an Internet Gateway (IGW)?**
    - **Answer:** An Internet Gateway is a horizontally scaled, redundant, and highly available VPC component that allows communication between instances in your VPC and the internet.[4][9] It serves two main purposes: it provides a target in your VPC route tables for internet-routable traffic, and it performs network address translation (NAT) for instances that have been assigned public IPv4 addresses. You attach one IGW to your VPC to enable internet access.[2][9]
4. **What is a NAT Gateway? Why would you use it?**
    - **Answer:** A NAT (Network Address Translation) Gateway is an AWS managed service that enables instances in a private subnet to initiate outbound traffic to the internet or other AWS services, but prevents unsolicited inbound connections from the internet from reaching those instances.[4][10][11][12] You deploy a NAT Gateway in a public subnet and update the route table associated with the private subnet(s) to direct internet-bound traffic (0.0.0.0/0) to the NAT Gateway. This is commonly used for tasks like downloading software updates or patches on instances in private subnets without exposing them directly.[9][13]
5. **What is the difference between a NAT Gateway and a NAT Instance?**
    - **Answer:**
        - **NAT Gateway:** An AWS managed service.[12][13] It's highly available within an Availability Zone, provides better bandwidth, and requires less administrative effort (no patching, scaling is managed by AWS). It is generally the preferred option.
        - **NAT Instance:** An EC2 instance that you configure and manage yourself to perform NAT. You are responsible for patching, scaling, and ensuring high availability (e.g., using scripts or multiple instances). It offers more control but comes with more overhead.
6. **What are Route Tables?**
    - **Answer:** A route table contains a set of rules, called routes, that determine where network traffic from your subnet or gateway is directed.[2][6][14] Each VPC has a main route table by default, and you can create custom route tables. You associate subnets with specific route tables.[15][16] For example, a public subnet's route table typically has a route sending 0.0.0.0/0 traffic to the Internet Gateway, while a private subnet needing outbound internet access would route 0.0.0.0/0 traffic to a NAT Gateway.[5][9]

Security in VPC

1. **What are Security Groups?**

- **Answer:** Security Groups act as a virtual firewall for your EC2 instances (and other resources like RDS instances or ELBs) to control *inbound* and *outbound* traffic at the *instance level*.[17] They are stateful, meaning if you allow inbound traffic, the corresponding outbound return traffic is automatically allowed, regardless of outbound rules. You specify allow rules (by protocol, port range, source/destination IP/Security Group); all traffic not explicitly allowed is denied.

2. **What are Network Access Control Lists (NACLs)?**
   - **Answer:** NACLs act as a firewall for controlling traffic *in* and *out* of one or more *subnets*.[18][19] They operate at the *subnet level*.[9][17][20] NACLs are stateless, meaning return traffic must be explicitly allowed by a corresponding rule.[17] They have numbered rules that are evaluated in order, from lowest to highest, to determine whether to allow or deny traffic. NACLs support both allow and deny rules.[18][19]
3. **What are the key differences between Security Groups and NACLs?**
   - **Answer:**
     - **Level:** Security Groups operate at the instance level; NACLs operate at the subnet level.[7][8]
     - **Statefulness:** Security Groups are stateful; NACLs are stateless.[17]
     - **Rules:** Security Groups only support allow rules (deny by default); NACLs support both allow and deny rules.
     - **Evaluation:** Security Groups evaluate all rules before deciding; NACLs process rules in numbered order, stopping at the first match.
     - **Association:** An instance can be associated with multiple Security Groups; a subnet can only be associated with one NACL at a time.[21]

Connectivity Options

1. **What is VPC Peering?**
   - **Answer:** VPC Peering allows you to connect two VPCs privately using AWS's backbone network, enabling them to communicate with each other as if they are within the same network.[2] Traffic does not traverse the public internet. Peering connections are non-transitive (if VPC A is peered with B, and B is peered with C, A cannot talk to C directly via B).[4] The CIDR blocks of peered VPCs cannot overlap.[4]
2. **What are VPC Endpoints?**
   - **Answer:** VPC Endpoints enable you to privately connect your VPC to supported AWS services (like S3, DynamoDB, EC2 API) and VPC endpoint services powered by AWS PrivateLink without requiring an Internet Gateway, NAT device, VPN connection, or AWS Direct Connect connection.[2] Traffic between your VPC and the other service does not leave the Amazon network.[16] There are two main types: Interface Endpoints (using an Elastic Network Interface with a private IP) and Gateway Endpoints (used for S3 and DynamoDB, acting as a target in your route table).

Scenario/Design Questions

1. **Describe how you would set up a basic VPC for a two-tier web application (web servers in a public subnet, database servers in a private subnet).**
    - **Answer:**
        1. **Create VPC:** Define a CIDR block (e.g., 10.0.0.0/16).[5]
        2. **Create Subnets:**
            - Create at least one public subnet (e.g., 10.0.1.0/24) in an Availability Zone.[5]
            - Create at least one private subnet (e.g., 10.0.2.0/24) in the same or different Availability Zone (for HA).
        3. **Create Internet Gateway (IGW):** Create an IGW and attach it to the VPC.[15]
        4. **Create Route Tables:**
            - **Public Route Table:** Create a route table, associate it with the public subnet(s). Add a route 0.0.0.0/0 pointing to the IGW.[5][9][10]
            - **Private Route Table:** Create a route table, associate it with the private subnet(s). *Do not* add a route to the IGW directly. (Optionally, if the DB needs internet access for updates, create a NAT Gateway in the public subnet and add a 0.0.0.0/0 route pointing to the NAT Gateway in the private route table).
        5. **Configure Security Groups:**
            - **Web Server SG:** Allow inbound traffic on ports 80/443 from 0.0.0.0/0 (or a specific source like a Load Balancer). Allow outbound traffic as needed. Allow inbound traffic from the Database SG on the required port(s) if the web server initiates communication.
            - **Database Server SG:** Allow inbound traffic only from the Web Server Security Group on the database port (e.g., 3306 for MySQL). Deny all other inbound traffic. Allow outbound traffic as needed (e.g., back to the web server SG, or to the NAT gateway if required).
        6. **Launch Instances:** Launch web servers into the public subnet with the Web Server SG. Launch database servers into the private subnet with the Database Server SG.
2. **An instance in a private subnet needs to download software patches from the internet. How would you enable this securely?**
    - **Answer:** The standard and most secure way is to use a NAT Gateway.
        1. Create a NAT Gateway in a public subnet within the same VPC. This requires allocating an Elastic IP address.[9][15]
        2. Modify the route table associated with the private subnet where the instance resides.
        3. Add a route for internet-bound traffic (0.0.0.0/0) that targets the NAT Gateway created in step 1.

4. Ensure the instance's Security Group allows the necessary outbound traffic (e.g., port 80/443).
5. Ensure the NACL associated with both the private and public subnets allows the necessary traffic (inbound and outbound, since NACLs are stateless).

Tips for Answering:

- **Be Clear and Concise:** Explain concepts simply.
- **Use Keywords:** Use the correct AWS terminology (Subnet, Route Table, IGW, NAT Gateway, Security Group, NACL, CIDR, AZ, etc.).
- **Understand the "Why":** Don't just define components, explain *why* they are used and the benefits they provide (e.g., security, isolation, connectivity).
- **Draw Diagrams:** If it's an in-person or virtual whiteboard interview, offer to draw a simple diagram to illustrate your points, especially for scenario questions.
- **Be Honest:** If you don't know an answer, it's better to admit it than to guess incorrectly. You can mention what you *do* know that's related.
- **Focus on Fundamentals:** For an internship, a strong grasp of the core concepts is more important than knowing every obscure feature.

Good luck with your interview preparation!