VPC

27 May 2023 23:39

- VPC : Virtual Private Cloud

  A VPC is dedicated to an AWS account and it is isolated from other VPCs in the AWS cloud. Each VPC can be given an IP address range and we can then deploy our resources such as instances and use AWS resources within the isolated VPC. These IP addresses are in the form of a CIDR block. A VPC has a router that uses the route table. The route table controls where the traffic is directed.

1. Isolation and Security: VPC provides logical isolation of resources within a virtual network. It allows you to create a private and isolated network environment, separate from other VPCs and the public internet. You can define security groups and network access control lists (ACLs) to control inbound and outbound traffic, enhancing the security of your applications and data.

2. Network Customization: AWS VPC allows you to customize your network configuration, including IP address ranges, subnets, and route tables. This flexibility enables you to design your network architecture to match your specific requirements, such as creating public and private subnets or implementing multi-tier architectures.

3. Connectivity Options: VPC offers various connectivity options to connect your VPC to your on-premises infrastructure or other cloud resources. You can establish secure VPN (Virtual Private Network) connections, use AWS Direct Connect for dedicated private network connections, or leverage AWS Transit Gateway to connect multiple VPCs and on-premises networks.

4. Scalability and Elasticity: With VPC, you can easily scale your network resources to meet changing demands. You can add or remove subnets, modify routing tables, and dynamically allocate IP addresses as needed. This scalability allows your applications to handle increased traffic and workload without disruption.

5. High Availability: VPC provides built-in high availability features to ensure the availability of your applications. You can distribute your resources across multiple Availability Zones (AZs) within a region, which are isolated from each other in terms of power, cooling, and network connectivity. This redundancy helps to protect your applications against failures and improves fault tolerance.

6. Enhanced Network Performance: AWS offers features like Amazon VPC Traffic Mirroring and Elastic Network Interfaces (ENIs) that allow you to monitor and analyze network traffic within your VPC. Additionally, you can take advantage of Amazon VPC endpoints, which provide a direct connection to AWS services without traversing the public internet, improving performance and reducing latency.

7. Compliance and Governance: AWS VPC supports various compliance programs and provides features that help you enforce security and governance policies. You can use AWS Identity and Access Management (IAM) to control access to your VPC resources, and services like AWS CloudTrail for auditing and logging. VPC also integrates with other AWS security services, such as AWS WAF (Web Application Firewall) and AWS Shield, to protect your applications against web-based attacks.

   Let's take a look at some of the basics of a VPC

   1. **Subnets**: A subnet can be thought of as dividing a large network into smaller networks. This is done because the maintenance of smaller networks is easier and it also provides security to the network from other networks.

   2. **Route Tables**: A route table contains a set of rules called routes which determine where traffic has to be directed. You can have multiple route tables in a VPC.

   3. **Internet Gateways (IGW)**: It is a combination of hardware and software that provides your private networks with a route to the world outside. An IGW is a horizontally scaled, redundant and highly available VPC component that allows

communication between instances and the internet. Only one IGW can be attached to a VPC at a time.

4. **Network Address Translation** (NAT): Since subnet is private, the IP addresses assigned to the instances cannot be used in public. NAT maps the private IP addresses to the public address on the way out and vice versa on the way in. An Elastic IP address is a static, public IPv4 address designed for dynamic cloud computing. You can associate an **Elastic IP address** with any instance or network interface for any VPC in your account. With an Elastic IP address, you can mask the failure of an instance by rapidly remapping the address to another instance in your VPC.

5. **Security groups:** Security groups are a set of firewall rules that controls the traffic for your instance. In Amazon Firewall the only action that can be carried out is allow. You cannot create a rule to deny. The destination is always the instance on which the service security group is running. You can have a single security group associated with multiple instances.

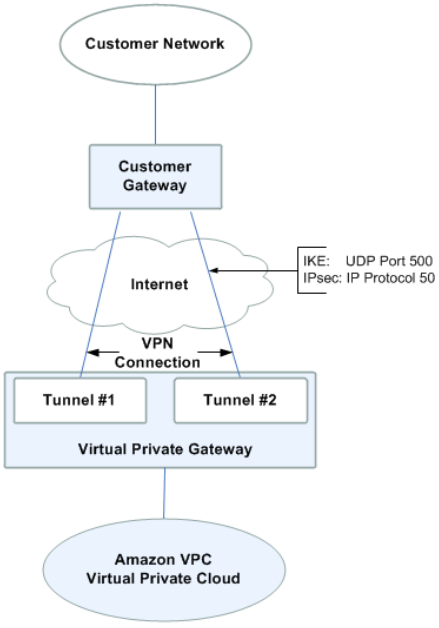
6. **Customer Gateway** — An Amazon VPC VPN connection links your data center (or network) to your Amazon VPC (virtual private cloud). A *customer gateway* is the anchor on your side of that connection. It can be a physical or software appliance.

7. **Virtual Private Gateway** — A *virtual private gateway* is the VPN concentrator on the Amazon side of the VPN connection. You create a virtual private gateway and attach it to the VPC from which you want to create the VPN connection.

8. VPN stands for 'virtual private networking', which is a popular internet security method which was originally designed for large organisations where employees needed to connect to a certain computer from different locations.

9. **VPC Peering** — A VPC peering connection allows you to route traffic between two VPC's using IPv4 or IPv6 private addresses. Instances in either VPC can communicate with each other as if they are within the same network. You can create a VPC peering connection between your own VPCs, or with a VPC in another AWS account. A VPC peering connection helps you to facilitate the transfer of data.

10. **Network Access Control Lists (NACL)**— an optional layer of security for your VPC that acts as a firewall for controlling traffic in and out of one or more subnets. You might set up network ACLs with rules similar to your security groups in order to add an additional layer of security to your VPC. The default network ACL is configured to allow all traffic to flow in and out of the subnets to which it is associated.



Source: AWS

AWS provides a number of efficient, secure connectivity options to help you get the most out of AWS when integrating your remote networks with Amazon VPC. Amazon Virtual Private Cloud (Amazon VPC) enables you to launch AWS resources into a virtual network that you've defined. This virtual network closely resembles a traditional network that you'd operate in your own data center, with the benefits of using the scalable infrastructure of AWS.

Security groups vs Network ACLs - What is the Difference?

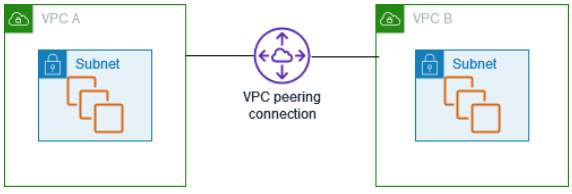
The differences between NACL and security groups have been discussed below:

NACL	Security Group
Network Access Control List that helps provide a layer of security to the amazon web services. There are two kinds of NACL- Customized and default.	A security group has to be explicitly assigned to an instance; it doesn't associate itself to a subnet.
Multiple subnets can be bound with a single NACL, but one subnet can be bound with a single NACL only, at a time	Security groups are associated with an instance of a service. It can be associated with one or more security groups which has been created by the user.
NACL can be understood as the firewall or protection for the subnet.	Security group can be understood as a firewall to protect EC2 instances.
These are stateless, meaning any change applied to an incoming rule isn't automatically applied to an outgoing rule.	These are stateful, which means any changes which are applied to an incoming rule is automatically applied to a rule which is outgoing.
Example: If a request comes through port 80, it should be explicitly indicated that its outgoing response would be the same port 80.	Example: If the incoming port of a request is 80, the outgoing response of that request is also 80 (it is opened automatically) by default.
NACL can be used to support as well as deny rules. Denial of rules can be explicitly mentioned, so that when the layer sees a specific IP address, it blocks connecting to it.	They support rules only, and the default behaviour is denial of all rules. Every VPC can belong to different security groups.
It is considered to be the second layer of defence, which helps protect AWS stack. It is an optional layer for VPC, which adds another security layer to the amazon service.	It is considered to be the first defence layer that helps protect the Amazon Web Services infrastructure.
In case of NACL, the rules are applied in the order of their priority, wherein priority is indicated by the number the rule is assigned.	In case of a security group, all the rules are applied to an instance.
This means every rule is evaluated based on the priority it has.	This means all rules are evaluated before they allow a traffic.

What is VPC peering?

A virtual private cloud (VPC) is a virtual network dedicated to your AWS account. It is logically isolated from other virtual networks in the AWS Cloud. You can launch AWS resources, such as Amazon EC2 instances, into your VPC.

A VPC peering connection is a networking connection between two VPCs that enables you to route traffic between them using private IPv4 addresses or IPv6 addresses. Instances in either VPC can communicate with each other as if they are within the same network. You can create a VPC peering connection between your own VPCs, or with a VPC in another AWS account. The VPCs can be in different Regions (also known as an inter-Region VPC peering connection).



sAWS uses the existing infrastructure of a VPC to create a VPC peering connection; it is neither a gateway nor a VPN connection, and does not rely on a separate piece of physical hardware. There is no single point of failure for communication or a bandwidth bottleneck.

A VPC peering connection helps you to facilitate the transfer of data. For example, if you have more than one AWS account, you can peer the VPCs across those accounts to create a file sharing network. You can also use a VPC peering connection to allow other VPCs to access resources you have in one of your VPCs.

When you establish peering relationships between VPCs across different AWS Regions, resources in the VPCs (for example, EC2 instances and Lambda functions) in different AWS Regions can communicate with each other using private IP addresses, without using a gateway, VPN connection, or network appliance. The traffic remains in the private IP space. All inter-Region traffic is encrypted with no single point of failure, or bandwidth bottleneck. Traffic always stays on the global AWS backbone, and never traverses the public internet, which reduces threats, such as common exploits, and DDoS attacks. Inter-Region VPC peering provides a simple and cost-effective way to share resources between regions or replicate data for geographic redundancy.

Basic Characteristics:

Private Communication:

VPC peering connections allow communication between instances in the peered VPCs using private IP addresses. The traffic traverses the AWS network without going over the internet.

No Transitive Peering:

VPC peering is not transitive. If VPC A is peered with VPC B and VPC B is peered with VPC C, there is no automatic peering relationship between VPCs A and C. Separate peering connections are needed.

Security Groups and Network ACLs:

You can control traffic between peered VPCs using security groups and network ACLs, providing a secure way to define communication rules.

Here are some common use cases for VPC peering connections:

Interconnecting VPCs in the Same Region:

When you have multiple VPCs within the same AWS region and want them to communicate with each other privately, VPC peering allows you to establish direct network connectivity between them.

Isolating Environments:

You might have separate VPCs for development, testing, and production environments. VPC peering allows you to keep these environments isolated while still enabling controlled communication between them when necessary.

Centralized Management:

VPC peering is useful for centralizing shared services, such as databases, authentication services, or other common resources, in a dedicated VPC. Other VPCs can then peer with this central VPC to access these shared services.

Cross-Account VPC Peering:

You can peer VPCs across different AWS accounts. This is beneficial when different teams or business units manage separate AWS accounts and need to share resources or data securely.

Hub-and-Spoke Architecture:

Implementing a hub-and-spoke architecture involves connecting multiple VPCs to a central VPC (hub) using peering connections. The hub VPC acts as a central point for managing shared resources, while the spoke VPCs remain isolated from each other.

Migrating Workloads:

During a migration of workloads from one VPC to another, you can use VPC peering to facilitate a phased migration strategy. This allows you to move resources gradually without disrupting the entire network.

Third-Party Access:

If you have a VPC hosting third-party services or applications that need to interact with your internal systems, you can use VPC peering to establish a secure connection without exposing your internal network to the public internet.

Disaster Recovery:

In a disaster recovery scenario, where you have redundant resources in a separate VPC or region, VPC peering can be used to enable

failover and ensure continuous operation of critical services.

Segmenting Applications:

When you have different applications with distinct security and compliance requirements, you can deploy them in separate VPCs and use peering connections to enable communication while maintaining segmentation.

Compliance and Security:

VPC peering helps organizations adhere to security and compliance requirements by allowing them to design and control network traffic between VPCs. This ensures that sensitive data remains within the intended boundaries.

When implementing VPC peering, it's essential to carefully plan and consider factors such as security, routing, and the specific requirements of your architecture. Always refer to the AWS documentation and best practices for guidance on implementing VPC peering in your specific use case.

Key Steps to Set Up VPC Peering:

1. **Create a Peering Connection:**
   In the AWS Management Console, navigate to the VPC dashboard, select "Peering Connections," and click "Create Peering Connection." Specify the IDs of the two VPCs you want to connect.
2. **Accept the Peering Connection:**
   The peering connection status will be "Pending Acceptance." In the VPC dashboard, select the pending connection, click "Actions," and choose "Accept Request" to establish the connection.
3. **Update Route Tables:**
   Update the route tables of the involved VPCs to include routes for the private IP ranges of the peer VPC. This ensures that traffic knows how to reach the peered VPC.
4. **Security Configuration:**
   Adjust security groups and network ACLs to allow the necessary traffic between instances in the peered VPCs while maintaining security.

Common Issues for VPC Peering:

CIDR Block Overlaps:

- Problem: The CIDR blocks of the peered VPCs must not overlap. Overlapping CIDR blocks can lead to routing conflicts and prevent successful peering.
- Solution: Ensure that the CIDR blocks of the VPCs do not conflict. If there is an overlap, you may need to reconfigure one of the VPCs with a non-overlapping CIDR block.

Route Table Configuration:

- Problem: Incorrect route table configuration can prevent proper routing between the peered VPCs.
- Solution: Verify that the route tables in both the source and destination VPCs are updated to include routes for the CIDR blocks of the peered VPCs, and that the routes point to the peering connection.

Pending Acceptance:

- Problem: The peering connection status may remain in "Pending Acceptance" if the peering request is not accepted.
- Solution: In the AWS Management Console, navigate to the VPC dashboard, select "Peering Connections," and accept the pending request. Ensure that both VPCs have accepted the peering connection.

Cross-Region Peering Issues:

- Problem: Peering connections across different AWS regions may have additional considerations and limitations.
- Solution: Check the AWS documentation for any specific requirements or limitations associated with cross-region VPC peering. Ensure that both regions are properly configured.

Transitive Routing:

- Problem: VPC peering is not transitive, meaning if VPC A is peered with VPC B and VPC B is peered with VPC C, VPC A is not automatically peered with VPC C.
- Solution: If you need communication between VPC A and VPC C, you'll need to set up a separate peering connection between them.

DNS Resolution:

- Problem: DNS resolution between peered VPCs may not work as expected.
- Solution: Ensure that DNS resolution is properly configured in both VPCs. If using Amazon Route 53 private hosted zones, configure DNS resolution settings accordingly.

VPC Size Limitations:

- Problem: VPC size limitations may affect peering connections, especially if the combined number of route table entries in bot h VPCs exceeds the allowed limit.
- Solution: Check and, if necessary, adjust the size of your VPCs or the number of route table entries to stay within the speci fied limits.

Performance Considerations:

- Problem: In high-performance scenarios, users may encounter latency issues.
- Solution: Consider the performance requirements of your workloads. While VPC peering generally has low latency, for very high -performance scenarios, you may need to explore alternative solutions or optimizations.

VPC (Virtual Private Cloud) Gateway Endpoints

A VPC endpoint enables customers to privately connect to supported AWS services and VPC endpoint services powered by AWS PrivateLink. Amazon VPC instances do not require public IP addresses to communicate with resources of the service. Traffic between an Amazon VPC and a service does not leave the Amazon network.
VPC endpoints are virtual devices. They are horizontally scaled, redundant, and highly available Amazon VPC components that allow communication between instances in an Amazon VPC and services without imposing availability risks or bandwidth constraints on network traffic. There are two types of VPC endpoints:

Interface endpoints
gateway endpoints

- **Interface endpoints**

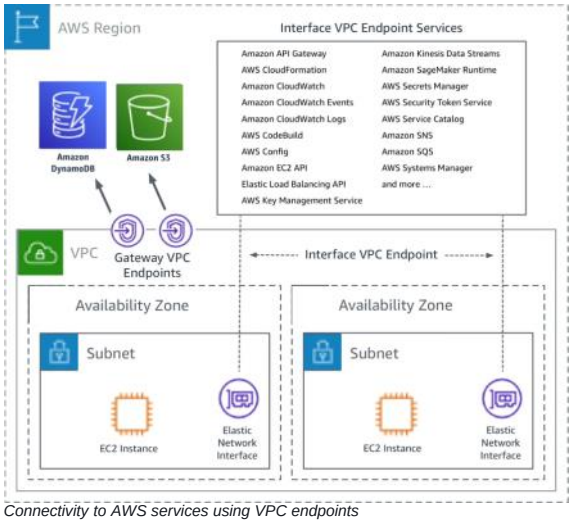
Interface endpoints enable connectivity to services over AWS PrivateLink. These services include some AWS managed services, services hosted by other AWS customers and partners in their own Amazon VPCs (referred to as endpoint services), and supported AWS Marketplace partner services. The owner of a service is a service provider. The principal creating the interface endpoint and using that service is a service consumer.

An interface endpoint is a collection of one or more elastic network interfaces with a private IP address that serves as an entry point for traffic destined to a supported service.
Interface endpoints currently support many AWS managed services.

- **Gateway endpoints**

A gateway endpoint targets specific IP routes in an Amazon VPC route table, in the form of a prefix-list, used for traffic destined to Amazon DynamoDB or Amazon Simple Storage Service (Amazon S3). Gateway endpoints do not enable AWS PrivateLink.

Instances in an Amazon VPC do not require public IP addresses to communicate with VPC endpoints, as interface endpoints use local IP addresses within the consumer Amazon VPC. Gateway endpoints are destinations that are reachable from within an Amazon VPC through prefix-lists within the Amazon VPC's route table. Refer to the following figure, which shows connectivity to AWS services using VPC endpoints.



Connectivity to AWS services using VPC endpoints

Important Considerations:

- VPC Gateway Endpoints operate at the VPC level and are associated with a specific VPC.
- When you create a VPC Gateway Endpoint, a route is added to your VPC's route tables, directing traffic to the service through the endpoint.
- The communication between your VPC and the supported AWS service through the endpoint is secure, private, and does not involve the public internet.
- VPC Gateway Endpoints do not impose additional data transfer costs. You pay only for the data transfer out of your VPC.
- Endpoint policies can be used to control access to the service through the endpoint.
  These endpoints simplify network architecture and enhance security by reducing the exposure of your VPC to the internet. They are particularly useful for scenarios where you need to access specific AWS services securely from within your VPC. Always refer to the AWS documentation for the most up-to-date and detailed information on configuring VPC Gateway Endpoints