**What Is 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.

**Amazon VPC Concepts**

As you get started with Amazon VPC, you should understand the key concepts of this virtual network, and how it is similar to or different from your own networks. This section provides a brief description of the key concepts for Amazon VPC.

### VPCs and Subnets

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 your AWS resources, such as Amazon EC2 instances, into your VPC. You can specify an IP address range for the VPC, add subnets, associate security groups, and configure route tables.

A subnet is a range of IP addresses in your VPC. You can launch AWS resources into a specified subnet. Use a public subnet for resources that must be connected to the internet, and a private subnet for resources that won't be connected to the internet. For more information about public and private subnets

To protect the AWS resources in each subnet, you can use multiple layers of security, including security groups and network access control lists (ACL). For more information, see

By launching your instances into a VPC instead of EC2-Classic, you gain the ability to:

* Assign static private IPv4 addresses to your instances that persist across starts and stops
* Optionally associate an IPv6 CIDR block to your VPC and assign IPv6 addresses to your instances
* Assign multiple IP addresses to your instances
* Define network interfaces, and attach one or more network interfaces to your instances
* Change security group membership for your instances while they're running
* Control the outbound traffic from your instances (egress filtering) in addition to controlling the inbound traffic to them (ingress filtering)
* Add an additional layer of access control to your instances in the form of network access control lists (ACL)
* Run your instances on single-tenant hardware

## Pricing for Amazon VPC

There's no additional charge for using Amazon VPC. You pay the standard rates for the instances and other Amazon EC2 features that you use. There are charges for using an Site-to-Site VPN connection and using a NAT gateway.

Amazon Virtual Private Cloud (Amazon VPC) lets you provision a logically isolated section of the Amazon Web Services (AWS) cloud where you can launch AWS resources in a virtual network that you define. 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. You can use both IPv4 and IPv6 in your VPC for secure and easy access to resources and applications.

You can easily customize the network configuration for your Amazon Virtual Private Cloud. For example, you can create a public-facing subnet for your webservers that has access to the Internet, and place your backend systems such as databases or application servers in a private-facing subnet with no Internet access. You can leverage multiple layers of security, including security groups and network access control lists, to help control access to Amazon EC2 instances in each subnet.

The link between VPC and Azure virtual network will use an IPsec tunnel created with the help of Strongswan Linux package on AWS side

and the virtual network gateway on Azure side.

traffic between the AWS VPC private subnet will go through the VM from the AWS VPC public subnet.

There are two subnets in VPC, one public and one private: AWS

Azure Portal provides basic monitoring for Azure Virtual Network Gateway. Users that require advanced monitoring,

auto-scaling or self-healing features

Azure Virtual Network Gateway stable, CloudMonix also provides powerful dashboards, historical reporting

The EC2 instance acting as an IPsec peer for the Azure virtual network gateway runs strongswan to establish the IPsec tunnels.

Subnet

1.

An AWS VPC spans all the Availability Zones (AZs) in that region, hence, subnets in AWS VPC are mapped to Availability Zones (AZs).

Azure VNet subnets are defined by the IP Address block assigned to it.

2.AWS VPC subnets can either be private or public.

A subnet is public if it has an internet gateway (IGW) attached.

AWS allows only one IGW per VPC and the public subnet allow resources deployed in them access to the internet.

Azure VNet does not provide a default VNet and does not have private or public subnet as in AWS VPC

Resources connected to a VNet have access to the Internet, by default.

3.IP Addresses –

Azure VNet assigns resources connected and deployed to the VNet a private IP address from the CIDR block specified.

In Azure VNet, the smallest subnet supported is /29 and the largest is a /8.

AWS also allows IP addresses from the same RFC 1918 or publicly routable IP blocks.

Currently, AWS does not support direct access to the internet from publicly routable IP blocks,

hence they are not reachable from the internet even through the Internet gateway (IGW).

They are only reachable via the Virtual Private Gateway. Because of this, Windows instances cannot boot correctly if launched into a VPC with ranges from 224.0.0.0 to 255.255.255.255 (Class D and Class E IP address ranges).

For the subnet, AWS recommends a minimum address block of /28 and maximum of /16.

Microsoft Azure VNet support for IPv6 is limited at the time of writing this blog,

however, AWS VPC supports IPv6 for all regions AWS VPC supports IPv6 for all regions except China, as of January 2017.

In IPv6, every address is internet-routable and can talk to the Internet by default. AWS VPC provides Egress-Only Internet Gateway (EGW)

Routing Table :

============

Routing Table – AWS uses the route table to specify the allowed routes for outbound traffic from the subnet. All subnets created in a VPC is automatically associated with the main routing table, hence, all subnets in a VPC can allow traffic from other subnets unless explicitly denied by security rules. In Azure VNet, all resources in the VNet allow the flow of traffic by using the system route. You don’t have to configure and manage routes because by default, Azure VNet provides routing between subnets, VNets, and on-premises networks. The use of system routes facilitates traffic automatically but there are cases in which you want to control the routing of packets through a virtual appliance. Azure VNet uses the system route table to ensure that resources connected to any subnet in any VNet communicate with each other by default. However, there are scenarios when you might want to override the default routes. For such scenarios, you can implement the user-defined routes (UDR) — control where traffic is routed for each subnet — or/and BGP routes (your VNet to your on-premises network using an Azure VPN Gateway or ExpressRoute connection). The UDR applies to only traffic leaving the subnet and can provide a layer of security for Azure VNet deployment, if the goal of UDR is to send traffic to some kind of inspection NVA or the like. With UDR, packets sent to one subnet from another can be forced to go through a network virtual appliance on a set of routes. In a hybrid setup, Azure VNet may use any of the three route tables – UDR, BGP (if ExpressRoute is used) and System routing tables. In Azure VNet, the subnet relies on the system routes for its traffic until a route table is explicitly associated with a subnet. Once an association is established, i.e. a UDR and/or a BGP route exists, routing is done based on Longest Prefix Match (LPM). In cases where there is more than one route with the same prefix length, a route is selected based on its origin in the following order: User defined route, BGP route (when ExpressRoute is used) and System route. Whereas, in AWS VPC the routing tables may be more than one, but of the same type.

**Security:**

======

– AWS VPC provides two levels of security for resources deployed to the network. The first is called Security Groups (SG). The Security Group is a stateful object that is applied at the EC2 instance level – technically, the rule is applied at the Elastic Network Interface (ENI) level. The response traffic is automatically allowed once a traffic is allowed. The second security mechanism is called Network Access Controls (NACLs). NACLs are stateless filtering rules that are applied at the subnet level and applies to every resource deployed to the subnet. It’s stateless because if an ingress traffic is allowed, the response is not automatically allowed unless explicitly allowed in the rule for the subnet. NACLs operates at the subnet level by examining the traffic entering and exiting the subnet. NACLs can be used to set both Allow and Deny rules. You can associate a NACL with multiple subnets; however, a subnet can be associated with only one NACL at a time. The NACL rules are numbered and evaluated in order, starting with the lowest numbered rule, to determine whether traffic is allowed in or out of any subnet associated with the network ACL. The highest number that you can use for a rule is 32766. The last rule numbered is always an asterisk, and denies traffic to the subnet. Note, you reach this rule only if no rules in the NACL list matches the traffic. Azure VNet provides Network Security Groups (NSGs) and it combines the functions of the AWS SGs and NACLs. NSGs are stateful and can be applied at the subnet or NIC level. Only one NSG can be applied to a NIC, but in AWS you can apply more than one Security Group (SG) to an Elastic Network Interface (ENI)

Gateways

=========

Both VNet and VPC offer different gateways for different connectivity purposes. AWS VPC uses mostly three gateways, four, if you add the NAT gateway. AWS allows one Internet Gateway (IGW) to provide connectivity to the internet via IPv4 and Egress-only Internet Gateway for internet connectivity to resources with IPv6. In AWS, any subnet without the IGW is regarded as private subnet and have no internet connectivity without NAT gateway or NAT instance (AWS recommends NAT Gateway for high availability and scalability). Another AWS gateway, Virtual Private Gateway (VPG) allows AWS to provide connectivity from AWS to other networks via VPN or Direct Connect. On the non-AWS network, AWS requires Customer Gateway (CGW) on the customer side to connect to AWS VPC. Azure VNet provides two types of gateway namely VPN Gateway and ExpressRoute Gateway. The VPN Gateway allows encrypted traffic for VNet to VNet or VNet to on-premises location across a public connection or across Microsoft’s backbone in the case of VNet to VNet VPN. However, the ExpressRoute and VPN Gateway also require a gateway subnet. The gateway subnet contains the IP addresses that the virtual network gateway services use. Azure VNET to VNET can connect natively via VPN but in AWS, such VPC to VPC requires a 3rd party NVA if the VPCs are in different regions.

Hybrid Connectivity

====================

Both AWS VPC and Azure VNet allow hybrid connections using VPN and/or Direct Connect and ExpressRoute respectively. With Direct Connect or ExpressRoute, connections of up to 10Gbps are available. An AWS DC connection consists of a single dedicated connection between ports on your router and an Amazon router. With one DC connection, you can create virtual interfaces directly to public AWS services (for example, to Amazon S3) or to Amazon VPC. Before using AWS DC, you must create a virtual interface. AWS allows 50 virtual interfaces per AWS Direct Connect connection, and this can be increased by contacting AWS. AWS DC connection is not redundant and a second connection is needed, if redundancy is required. AWS VPN creates two tunnels between AWS VPC and the on-premises network. To provide fault tolerance for Direct Connect, AWS recommends using one of the tunnels to connect to the on-premises data network via VPN and BGP. Azure ExpressRoute also provides two links and an SLA for connectivity — Azure guarantees a minimum of 99.95% ExpressRoute Dedicated Circuit availability — and hence predictable network performance.