

VPC

Networking 101

IP Addresses

Every host or device on a network must be addressable, meaning that they should have something that can be referenced by in order to reach it as a destination under a defined system of addresses. That thing is called IP Addresses and this is how we can address host in a network.

If one computer want to communicate with another computer, then the address can be used the address to reach each other and send information.

The IP address must be unique on its own network.

IPv4 Addresses

IP addresses are made up of two parts. The network address which is to identify the network that the address is part of. Then the part after is used to identify the host within the network.

IPv4 in the old days were divided into five different classes, A through E.

1. Class A: The first bit is 0 so it includes network address range from `0.0.0.0` to `127.0.0.0` can be used as network ID. 24 bits for host
2. Class B: The first bit is 1 and second bit is 0 so it includes network address range from `128.0.0.0` to `191.255.0.0`. 16 bits for host.
3. Class C: The two bit is 1 so it includes network address ranges from `192.0.0.0` to `223.255.255.0`. 8 bits for host.

So if a company wants a IPv4 address they would go through of first picking a class of address that they want, which dictate how many host they can have in their network. But as you can see it is limited and the size isn't best fit, since the differences between two classes is huge.

Now after you receive a network address from either class, you can further divide the network into smaller network sections and is called subnetting. And by default each network has only one subnet without subnetting, because it contains all the host addresses defined within. To do subnetting you would basically provide a subnet mask to mask out the subnets that you are looking into.

Netmask is used for identifying the network that the destination IP address falls under, think of the classes, it is for finding out "which network you belong to". Then you would use Subnet mask if you

have subnets to find out which subnet you belong to, then finally use the remaining bits to find the actual host.

Example:

You're issued the IP address `10.10.0.0/16`, here the Netmask is `255.255.0.0`. But if you are going to divide it into say 4 different subnets then you would need a Subnet mask

```
10.10.0.0/18
10.10.64.0/18
10.10.128.0/18
10.10.192.0/18
```

The remaining 14 bits will be used to identify each host within each of the subnets.

$2^{14} * 4 = 2^{16}$. It is still covering all the host that the original big subnet has, but just divided further for ease of management

CIDR

Stands for classes inter-domain routing.

CIDR let you have a variable length of subnet masking, which is much more efficient compared to those classes we say in the previous section.

CIDR consist of two numbers, the network address then the second part is the Netmask indicating how many bits from left to right to mask the network address.

Let's take an example:

```
11000000.10101000.01111011.10000100 -- Dest IP address (192.168.123.132)
11111111.11111111.11111111.00000000 -- Subnet mask (255.255.255.0)
192.168.123.132 / 24
```

As you can see the part highlighted in green is the network address, while the last 8 bits gives the host address. When the package arrives it will arrive in the 192.168.123.0 subnet and then be processed at the destination address of 192.168.123.132.

To know for AWS: It just basically help define the IP ranges that's all.

IP is made up of segments or octets. /32 says that no octet can change for the host, so it only one host address.

/24 means only last octet can change for the host addresses, and so on.

For example: 192.168.0.0 /16 means that the IP ranges from 192.168.0.0 - 192.168.255.255, you can two octets that can change.

Public vs Private IP

Public IP is assigned by IANA organization, and they also establish standard for private IP uses.

Private IP are as follows:

1. 10.0.0.0 - 10.255.255.255 (10.0.0.0 / 8) This is for big networks
2. 172.16.0.0 - 172.31.255.255 (172.16.0.0 / 12) This is for AWS default VPC private network range
3. 192.168.0.0 - 192.168.255.255 (192.168.0.0 / 16) This is for home networks

Then rest of the IP addresses are public.

VPC

Virtual private cloud.

Max of 5 per region but you can increase it. Max CIDR per VPC is 5, so you are allowed to add more CIDR into your VPC and it isn't limited to just one CIDR block (defines the range of IPs that you can have for your EC2 instances).

Only private IPv4 range are allowed because it is private cloud.

Your VPC CIDR should not overlap with your other networks, in case you wanna connect two VPCs together. So it means that you can but you shouldn't

Default VPC

All new AWS account have a default VPC. New EC2 instances are launched into the default VPC if no subnet is not specified.

Default VPC have internet connectivity and all EC2 instances inside it have public and private IPv4 addresses. And you also get a public and private IPv4 DNS name.

Subnets

You divide the CIDR block that you are given into further subnets for better management. This is done via increasing the subnet mask bits.

Subnets are associated with an availability zone, so you would put subnets into different availability zone to get high availability.

For each subnet that you created it will reserve 5 IP the first 4 and last 1 in each subnet which means they cannot be used for assignment.

Ex: 10.0.0.0/24, then:

1. 10.0.0.0: Is for Network address
2. 10.0.0.1: Reserved by AWS for VPC router
3. 10.0.0.2: Reserved by AWS for mapping Amazon provided DNS
4. 10.0.0.3: Reserved for future use
5. 10.0.0.255: For network broadcast, but isn't supported in VPC so it is reserved

So when you are creating your subnets keep in mind of the 5 reserved IPs. If you need 29 IP address for a subnet, then you can't use subnet size of /27. $2^5 = 32 - 5 = 27 < 29$.

Subnets can be public or private. For public subnet choose a CIDR block that's small since they are just for public front-facing resources like ALB.

Internet gateway

Internet gateway allows resources like EC2 instances in a VPC to connect to the internet. It scales horizontally and is highly available and redundant.

One VPC can only be attached to one internet gateway and vice versa.

Internet gateway on their own don't allow internet access you have to also configure the route tables. You have to edit the route table to connect the EC2 instances with the router then connect to the internet gateway to finally be able to access the internet.

Route table

Route table tells how the network traffic from your subnet is directed, where does it go.

Each route table have subnet associations. They work together with internet gateway to actually provide internet access for the instances launched in the subnet.

If it is location address, then route it locally. Then any other IP address that's connecting to then route it to the internet gateway. It is just a configuration for the route table, that's all to enable internet access.

Bastion Hosts

Users want to access EC2 instances in a private subnet. How do we do it? We use bastion host, it is an EC2 instance named Bastion Host. It is EC2 instance in public subnet and the Bastion Host have access to the private subnet, **so to access the private subnet EC2 instance the user first connect to Bastion Host then connect from Bastion Host to the EC2 instance in private subnet.**

Bastion Host security group must allow access from public internet on port 22 for SSH and from restricted CIDR from your company for example, or your own IP address. This is for security you don't want everyone from the internet able to SSH into your Bastion Host.

The security group of the private EC2 instance must also allow security group of Bastion Host or the private IP of the bastion host to SSH into it.

NAT instances

NAT = Network address translation. Outdated but still in the exam

So NAT instances are still EC2 instance that acts as the NAT in normal world basically. They are not highly available and resilient right out of the box thus it is outdated.

The private EC2 instance doesn't have public internet access because it doesn't have routing table set up. But how do we give it internet access? NAT instances, gives those EC2 instances in private subnet to be connected to the internet.

NAT instances must be launch in public subnet and must have an elastic IP attach to it. Must disable source / destination IP check otherwise it won't work.

Then you would configure your routing table to route all the traffic from the private subnet to the NAT instances. NAT instance will be responsible for proxying the traffic, meaning the IP address of the private EC2 instance is hidden, instead the packet will be sent with the IP address of the NAT instance. When the response comes back NAT is smart enough to remember which EC2 instance requested this packet and forwards it back.

NAT gateway

Compared to NAT instance it has higher bandwidth, higher availability, and you don't need administration. Up to 45 Gbps and is all managed by AWS.

NAT gateway will be created for a specific availability zone, and uses an elastic IP.

Can't be used by EC2 instance if they are in the same subnet, only used for another subnet. And they must be launched in the public subnet.

No need to manage any security groups it is not needed!

Then all you need to do is to configure the route table to send traffic from the subnet to the NAT gateway.

Availability

NAT gateway is only resilient in a single AZ. You have to create multiple NAT in multi AZ for fault tolerance.

Security groups and NACLs

Network traffics goes to the NACL before going into the subnet. It has inbound / outbound rules.

NACL are stateless. If it is allowed in, then the response after passing security group will be checked again for outbound rules because it doesn't remember that it allowed the same request in.

After it reaches into the subnet it will go through security group. It has inbound rule and outbound rules. Security group are stateful. If it is accepted in, then automatically the response from the EC2 instance are automatically accepted out.

If this time the request is outgoing from the EC2 instances. The outbound rules are first evaluated in the security group, then it will reach the NACL and be evaluated. Then the response will come back as incoming request then NACL will evaluate again, but for security group because it is stateful it won't be evaluated and automatically accepted in.

NACL

One NACL per subnet. New subnet are assigned default NACL. By default it allows everything in and out with the subnet it is associated with.

You define rules in NACL, and the higher precedence with a lower number and first rule match will drive the decision.

Last rule is * and denies all request if no rule match.

It also supports deny rule. It is stateless so every traffic needs to be evaluated. Security group return traffic are automatically accepted.

Ephemeral ports with NACL

You need to configure the appropriate ephemeral ports for the NACL.

VPC peering

Privately connect two VPC using AWS's network. This is achieved by making them behave as if they were in the same network, and this is why their CIDR should not overlap otherwise it will not work!

Peering connection are not transitive. If A can connect to B, and B can connect to C. But A cannot connect to C. It needs an explicit peering connection.

You must update route tables in each VPC's subnet to ensure EC2 instances can communicate with each other after you created the VPC peering connection. Basically you want to send the traffic going to the other VPC via the peering connection in both of the routing tables.

Peering connection can happen cross account and cross region.

VPC endpoints

If you want to access AWS resources using AWS's private network and not have to go through the public internet then you can use VPC Endpoints. It is associated with a VPC.

There is two kinds of endpoint:

1. Interface endpoint (powered by PrivateLink): It will provision ENI + security group. ENI will give you a private IP which will be the entry point to access the other services. This kind will allow you to communicate with lots of AWS services
2. Gateway endpoint: Used as a target in a route table, so no IP address you need to worry about. This only works for S3 and DynamoDB, but this is free and scales automatically.

In the exam Gateway endpoint is usually preferred because it is free and scales automatically. Even though you can use interface endpoint.

Interface endpoint is only perferrable if you want access from your on-premise data center to the AWS center using private network.

VPC Flows logs

Capture information about IP traffic going into your interfaces.

Help monitor and troubleshoot connectivity issues. The logs can be sent to S3 or CloudWatch.

You can use Athena for S3 and CloudWatch insights.

AWS Site-toSite VPN

To enable connection from on-premise to AWS VPC you need two things. Virtual Private Gateway: this is the gateway created and attached to the VPC on AWS side.

Then you also need customer gateway which is set up on the on-premise data center that you want to establish the connection to. It is either a software or physical device to install.

VGW <-> CGW connection. Or if your CGW is private then you need to use NAT. CGW <-> NAT <-> VGW.

You need to enable route propagation to make site-to-site connection work.

CloudHub

Let you set up secure communication between multiple on-premise sites if you have multiple VPN connection.

It is low cost and hub-spoke model.

Direct Connect

You get a dedicated private connection from a remote network to your VPC.

It is used to get a more consistent network experience, and increase bandwidth throughput. You can access to both the public and private resources on the same connection.

However, setting up direct connect require you to find a physical location to set up and connect to it. You will get a private connectivity, none of the traffics are routed via public internet.

Connect to one or more VPC

If you want to connect the on-premise data center to one or more VPC using Direct Conect then you need to use Direct Connect Gateway. Allowing you to connect to multiple VPC and multiple region as well.

Connection type

Dedicated connection: 1 Gbps, 10 Gbps, and 100 Gbps.

Hosted connection: 50 Mbps, 500 Mbps, and 10 Gbps. cpaacity can be added or removed

However, it will take more than one month to set up this connection.

Encryption

Data is not encrypted but it doesn't matter because the connection is private. But if you want the connection to be encrypted then you can use a VPN on top of the connection.

Good for extra level of security.

Resiliency

You need to set up Direct Connection with multiple on-premise center.

To get maximum resiliency for critical workloads, then each data center need two Direct Connect locations that are independent with each other for max.

Site to Site VPN and Direct Connect

Use direct connect as primary connection, then use Site to Site VPN as the backup in case DX failed.

The traffic for site-to-site VPN goes through the public internet however, since it is a VPN the traffic is encrypted. Direct connect on the other hand doesn't encrypt the data but you can set up, this is because the connection is private it doesn't go through the public internet.

Direct connect requires more effort and physical labor to set up while site to site VPN is wayyy easier to set up.

Transit gateway

This allows you to have transitive peering between thousands of VPCs and on-premises connections. VPC Peering doesn't have transitive peering, but transit gateway allows you that.

It is regional resources and can work cross-region. To limit the connection of which VPC can talk to which is done by the route tables.

Is the only service that supports IP Multicast.

Site-to-site VPN ECMP

Equal-cost multi-path routing. Allow you to forward packet over multiple best path. This is used to create multiple site-to-site VPN connection to increase the bandwidth of your connection to AWS.

Share direct connect between multiple account

Transit gateway allows you to share a Direct Connect between multiple account.

VPC Traffic mirroring

Allow you to capture and inspect network traffic in your VPC.

You have to define the source ENI, and to ENI or NLB. Traffic mirroring can basically mirror the traffic that's sent to the source ENI will also be sent to the mirror ENI or NLB then you can analyze the traffic separately without disturbing the functionality.

Source and target can be same or different VPC via VPC peering.

Content inspection, threat monitoring, and troubleshooting.

IPv6 for VPC

IPv4 soon will be exhausted soon. So IPv6 is come up with to not have be exhausted in our near time. All IPv6 addresses are public.

x.x.x.x.x.x.x where x is hexadecimal.

You can enable IPv6 support for your VPC.

Egress-only internet gateway

Only used for IPv6 only. Similar to NAT gateway but for IPv6.

Allow instances in your VPC outbound connection over IPv6 while preventing internet to initiate an IPv6 connection to your instance. You have to update the route table to make it work.

Networking cost in AWS

Incoming traffic to EC2 instances are free. Communication between EC2 instances within availability zone using their private IP are also free.

Cross AZ using public IP / Elastic IP, i.e. the traffic leaves AWS, then has to go back into AWS will cost \$0.02.

However, if you use private IP cross AZ then it is \$0.01. This is the preferred method. Use private IP for cheaper and faster option.

Cross region is \$0.02 per GB.

- Use private IP instead of public IP for good saving and better network performance
- Use same AZ for maximum savings. However, if you lose high availability.

Minimizing egress traffic network cost

Egress traffic = outbound, from AWS to outside

Ingress traffic = inbound traffic, from outside to AWS. This is typically free.

Try to keep as much internet within AWS to minimize the cost.

S3 data transfer pricing

Ingress traffic is free. However, egress traffic are not free, \$0.09 per GB.

S3 transfer acceleration is additional cost per GB to get faster data transfer.

S3 to CloudFront is free.

Cross region replication is \$0.02 per GB.

NAT Gateway vs Gateway VPC endpoint

Use public internet via NAT gateway to say access S3 bucket. \$0.045 NAT gateway / hour + \$0.045 data process / GB. So it isn't cheap.

However, if you use VPC endpoint gateway the cost is free. You pay \$0.01 data transfer in/out for the same region. Significant lower cost.

AWS Network Firewall

Used to protect your entire Amazon VPC. From layer 3 to layer 7.

Will monitor any traffic, from internet and out from VPC.

Internally it uses AWS Gateway load balancer, but instead we don't have to set up our 3rd party appliances, but AWS manages it for us.

You have fine grain control to support thousands of rules. Allow, drop, alert traffic that matches rules.

Logs can also be analyze.

Revision #3

Created 27 February 2023 01:58:25 by Tamarine

Updated 27 February 2023 21:49:43 by Tamarine