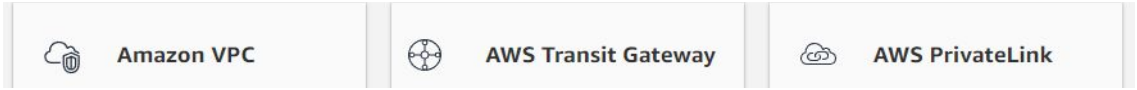
Redes en AWS

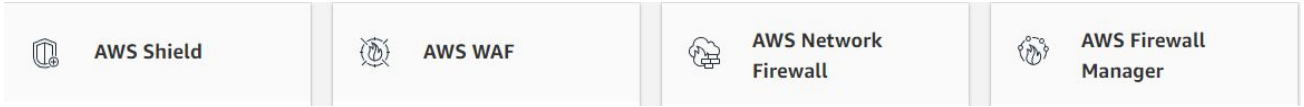
- Fundamentos de Redes



- Conectividad Híbrida

- Redes de borde

- Seguridad de la red



Analogía: Red Eléctrica

- Red masiva que brinda servicio a un menor costo (y mayor eficiencia) que generar nuestra propia electricidad
- Pago por lo que uso

Cloud Computing

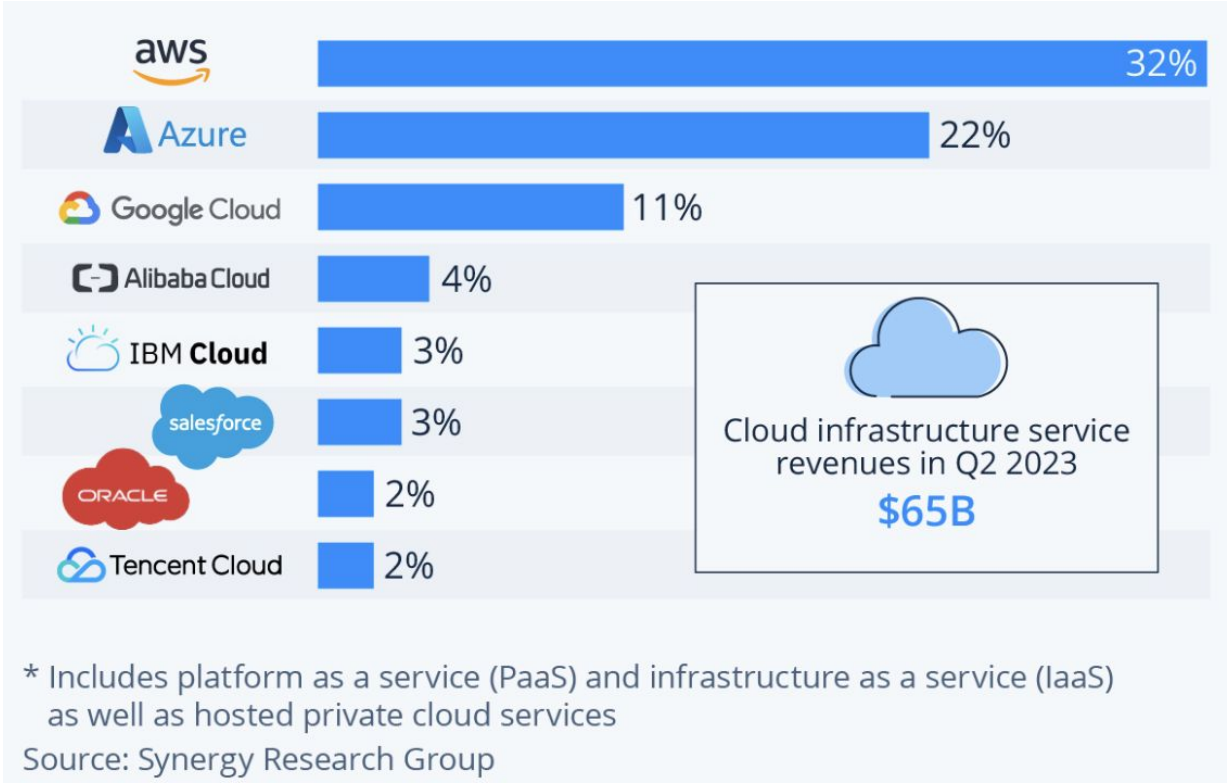
- Acceso a recursos de tecnología manejados por expertos
- Disponibles a demanda
- Mayor eficiencia que hacerlo nosotros mismos

- Acceso a través de internet*
- Sin inversión inicial
- Pago por lo que uso

Cloud Computing

Cloud Market Share (2023)

Esto es como se veia el año pasado. AWS fue el primero en crearse y hoy en dia es el mas importante. Distintos proveedores tienen sus diferencias, pero generalmente todos se acoplan a AWS.



Virtual Data Center – AWS

Punto de presencia: ubicaciones donde no hay un datacenter completo, sino que hay pocos servicios.
Por ejemplo, nuestra region mas cercana de AWS esta en brasil, pero hay un punto de presencia con algunos servicios que esta en Argentina.

31 regiones lanzadas
cada una con varias zonas de disponibilidad (AZ)

99 zonas de disponibilidad

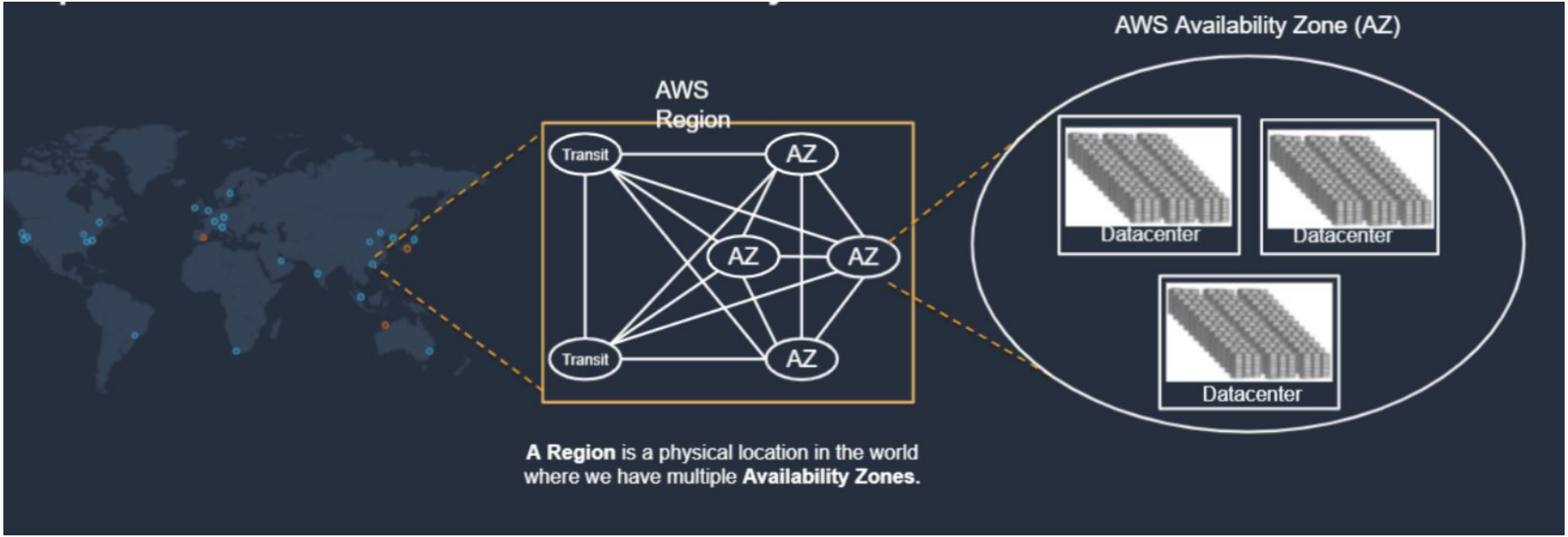
Más de 410 puntos de presencia
Más de 400 ubicaciones periféricas y 13 cachés periféricas regionales

Virtual Data Center – AWS

Virtual Data Center – Regiones

- Región AWS: ubicación física en el mundo
  - ej: us-east-1, us-west-1, eu-central-1
- Compuestas por múltiples Availability Zones (AZ)

En cada region, tenemos muchas zonas de disponibilidad. La mayoria de las regiones tiene al menos 3 AZ, aunque algunas de las nuevas tienen solo una. Podemos pensar las AZ como distintos datacenters en el mismo area.



Virtual Data Center – Availability Zones

- Zonas de fallo independientes entre sí
- Separadas máximo 100 km
- Infraestructura en 1 o más datacenters
- Ejemplos
  - us-east-1a
  - us-east-1b
  - etc.

Podemos pensar a las AZ como datacenters distintos dentro de una region (aunque cada AZ puede tener multiples datacenters).
Las AZ estan separadas un maximo de 100km por temas de latencia. Generalmente se replica los servicios en las distintas AZ (por ejemplo mi DB primaria esta en us-east-1a y la secundaria en us-east-1b).

La convencion de nombres es {CODIGO_AREA}-{ORIENTACION}-{NUMERO}-{LETRA_AZ}

Virtual Data Center – Availability Zones

- Millones de servidores a escala
- Alimentación independiente
- Datacenters conectados a través de fibra dedicada y redundante
- Amplio ancho de banda
- Baja latencia

Entre regiones se nota la diferencia de latencia segun en que lugar del mundo este. La idea de las AZ es que, como estan cerca y conectadas por cable, el usuario ni se de cuenta a que AZ se esta conectando.
En cada AZ puede haber muchos datacenters (en la misma cuadra por ejemplo), y el usuario nunca sabe en cual de ellos estan.

Virtual Data Center – Instancias EC2

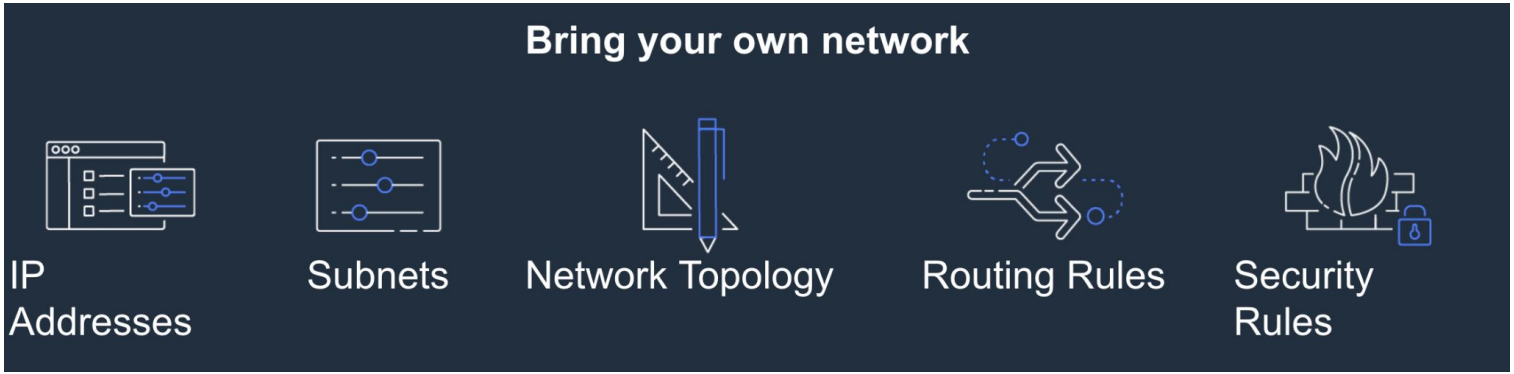
- Máquinas virtuales provistas por AWS
- Basada en una imagen AMI de de un sistema operativo (ej: Windows 11, Red Hat 7/8, Ubuntu 22.04, etc.)
- Puedo attacharle discos y placas de red virtuales
- Se pueden detener y reiniciar a demanda

Si yo freno mi EC2, AWS no me cobra por running time, pero me cobra por guardar esto en disco.

Virtual Private Cloud - VPC

- Parte de la nube AWS lógicamente segregada, donde podemos lanzar recursos en una red virtual
- Las VPC se definen dentro de una región AWS determinada

Una VPC es una porcion de la nube de AWS que esta separada logicamente, y esta reservada para nosotros.
Posible pregunta de parcial: las VPC se definen en una region determinada. NO se puede definir una VPC cross-region.



Virtual Private Cloud - VPC

Siempre que yo tenga que crear algun servicio en un cloud provider, tengo que crear una VPC para attachear el servicio.

- Una VPC se define con una lista de bloques CIDR
  - Ejemplo: 172.16.0.0/12
- Solo bloques de IP privadas
- Mínimo /28 (16 IPs)
- Máximo /16 (65536 IPs)

Este es un bloque CIDR PRIVADO.
Si tengo una sola VPC, no me importa mucho esto.
Pero si tengo muchas VPC y quiero que se conecten entre si, tal vez me conviene hacer que no haya colisiones de red.
Una vez que hago la VPC, NO puedo cambiar el CIDR.

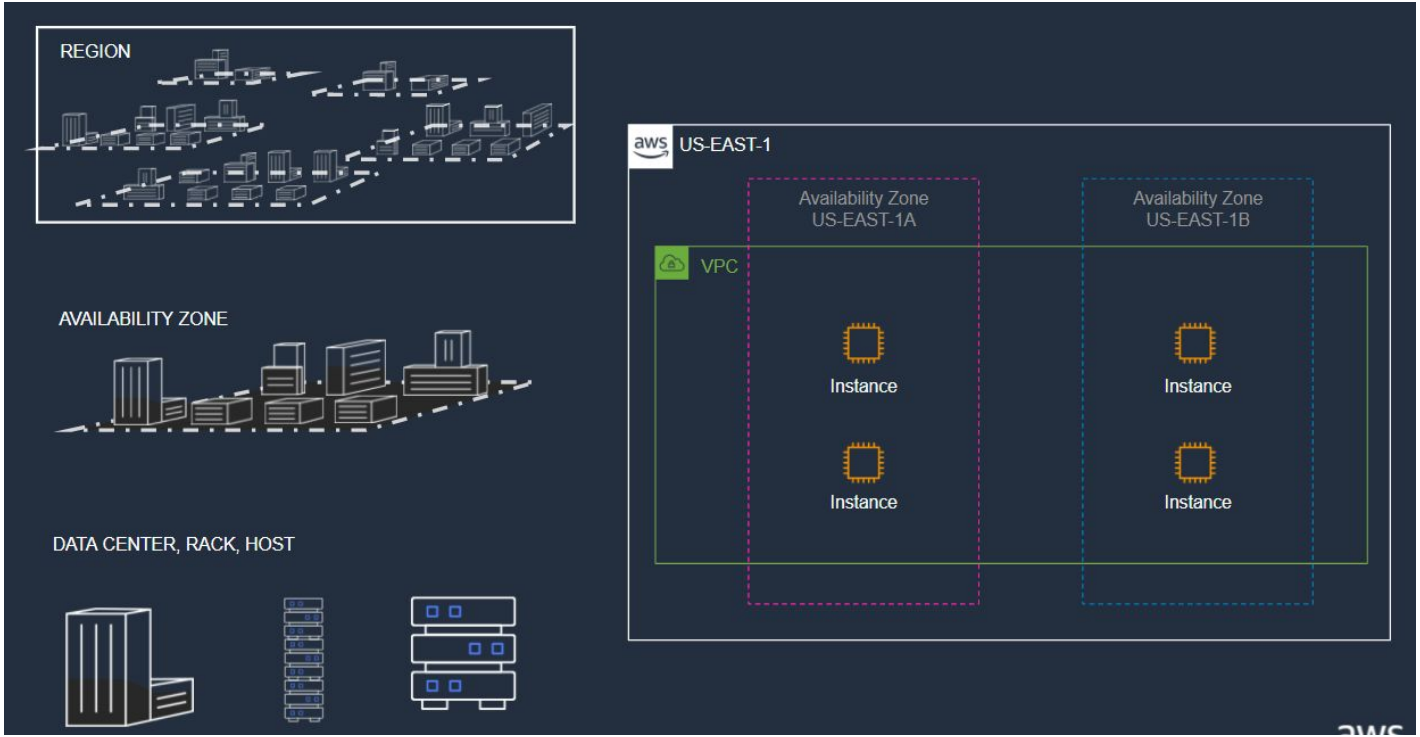
Virtual Private Cloud - VPC

- Las instancias EC2 creadas dentro de la VPC reciben una IP privada
- No se puede cambiar el CIDR original

  Para cambiar el CIDR original tendria que borrar la VPC junto a todos sus servicios asociados, y crear nuevos.
- Se puede agregar nuevos CIDR para expandir la VPC
- Evitar usar los CIDR de otras redes con las cual tenemos pensado conectarnos (si tengo muchas VPC me conviene que no haya colisiones para que sea mas facil conectarlas)

Virtual Data Center – Ejemplo

En este ejemplo vemos un VPC es US-EAST-1. Tenemos dos AZ, 1a y 1b. Cada AZ tiene multiples instancias de recursos.



VPC - Subnets

- Cada subnet se define con un bloque CIDR dentro del CIDR de la VPC
- Debe ser un subset del CIDR de la VPC
  - Ejemplo:
    - VPC: 172.16.0.0/16
    - Subnet 1: 172.16.1.0/24
    - Subnet 2: 172.16.2.0/24
    - Subnet 3: 172.16.3.0/24
- Se define en una AZ específica
  - Ejemplo:
    - Subnet 1: us-east-1a
    - Subnet 2: us-east-1a
    - Subnet 3: us-east-1c

Yo puedo decir que las subnets 1 y 2 vayan a la misma AZ, pero que la subnet 3 vaya a una AZ distinta (y usarla en caso de que haya un problema en la primer AZ).
Las subnets si o si tienen que estar asociada a UNA AZ en especifico.

VPC - Route Tables

- Controlan el tráfico de red de la VPC
- La VPC tiene una tabla de ruteo default que aplica a toda la VPC
- Puedo agregar tablas de ruteo por subnet para hacer override específicos

  Se define una tabla de ruteo por cada subnet.
- Las reglas más específicas tienen precedencia
  - Ejemplo: 172.16.0.0/16 le gana a 0.0.0.0/0

Elastic Network Interface (ENI)

- También conocida como “network interface”
- Es una placa de red virtual
- Se crean en una AZ determinada
- Se pueden attachar a las instancias EC2 de la misma AZ
- Pueden tener asociadas IPs pública y/o privadas

Elastic IP

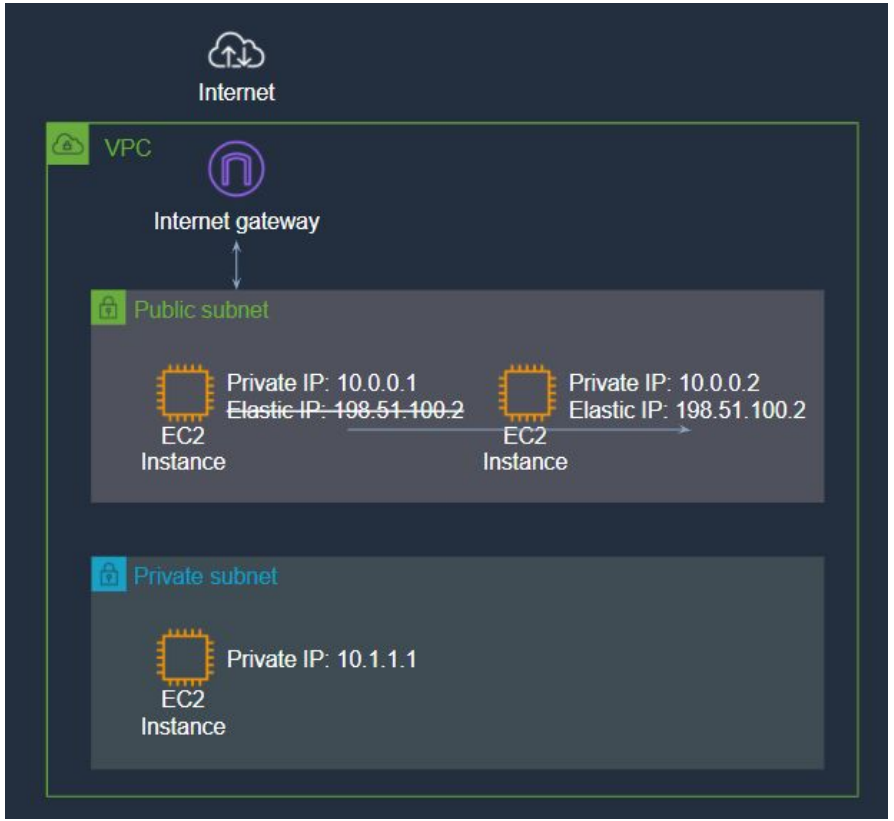
- Dirección IPv4 pública estática
  - Amazon EIP Pool
  - Bring Your Own IP (BYOIP) Pool
- Vinculada a la cuenta de AWS y a una región específica
- Se pueden asociar a una instancia EC2 o ENI específica

Cuando lanzamos una instancia viene con IP privada.
Para hacerlo publico, asociamos un EIP, en la cual AWS nos da un IPV4 de su pool propio. Tambien podemos elegir usar una IP propia.
AWS se encarga de hacer BGP para que todos puedan llegar a nuestro servicio.

Failover con Elastic IP

- Pueden ser remapeadas hacia otra instancia
- Útil para redundancia básica
- Hay mejores opciones

En este ejemplo, tenemos una EC2 instance primaria y una secundaria. Si se cae la primaria, puedo entrar y ponerle la IP de la primaria a la secundaria.
Sin embargo, este proceso hay que hacerlo manualmente.

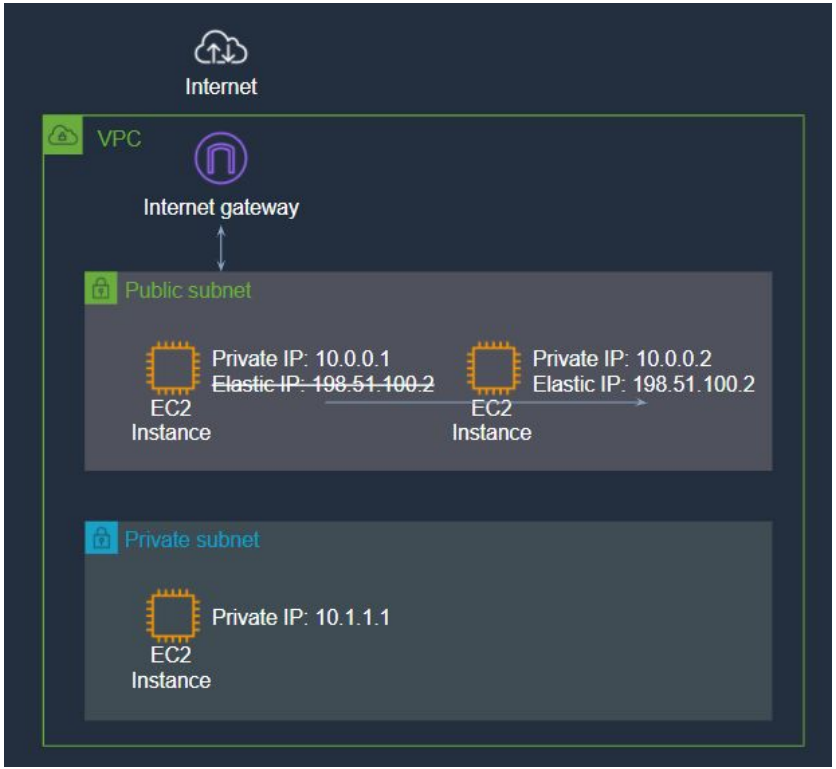


VPC - Subnets

- De acuerdo a cómo configuremos el ruteo, se clasifican en:
  - Public Subnet
    - Subred con conexión a Internet
    - Las instancias EC2 obtienen una IP pública y una IP privada
    - Permite asignar Elastic IPs
    - Usos típicos: DMZ para web servers, load balancers, etc.
  - Private Subnet
    - Subred sin acceso directo a Internet
    - Las instancias EC2 obtienen solo IP privada
    - Si algún recurso requiere salir a internet debe usar NAT
    - Usos típicos: application servers, bases de datos

Internet Gateway (IGW)

- Conecta subnets a Internet
- Debe estar atachado a una VPC
- Crea NAT stateless (llamada NAT 1:1) entre IP Pública y Privada
- Si se necesita stateful se debe usar NAT Gateway (siguiente slide)
- Es el default Gateway de la VPC



Yo me conecto directamente a la IP publica de mi instancia, mi IGW no tiene IP propia.
Es solo para public subnets, porque los servicios tienen que tener IP propia.

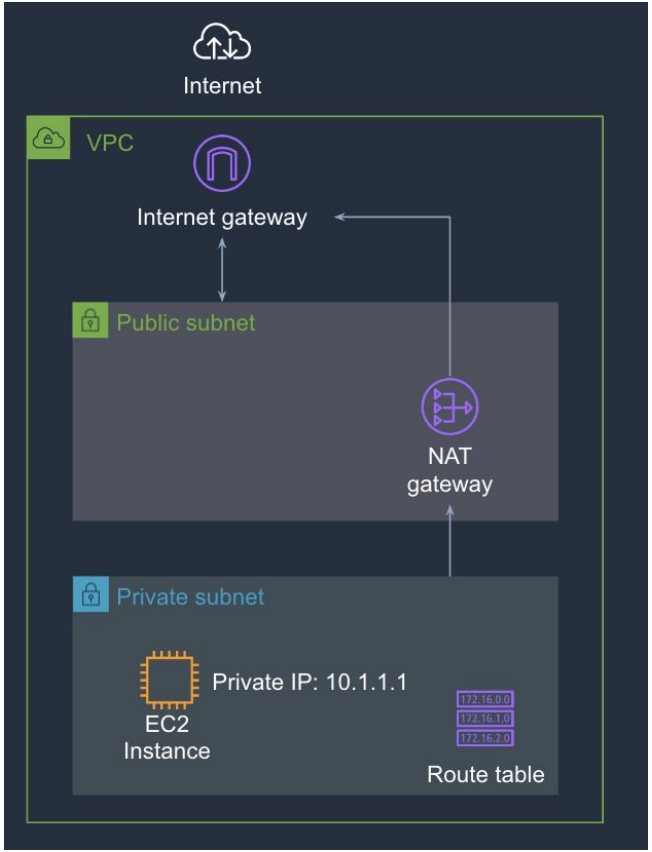
IGW - Ruteo

Everything that isn't destined for the VPC:
Send to the Internet

Destination	Target	Status	Propagated
172.31.0.0/16	local	Active	No
2600:1f16:14d:6300::/56	local	Active	No
0.0.0.0/0	igw-09ef761d872bd7540	Active	No
::/0	igw-09ef761d872bd7540	Active	No

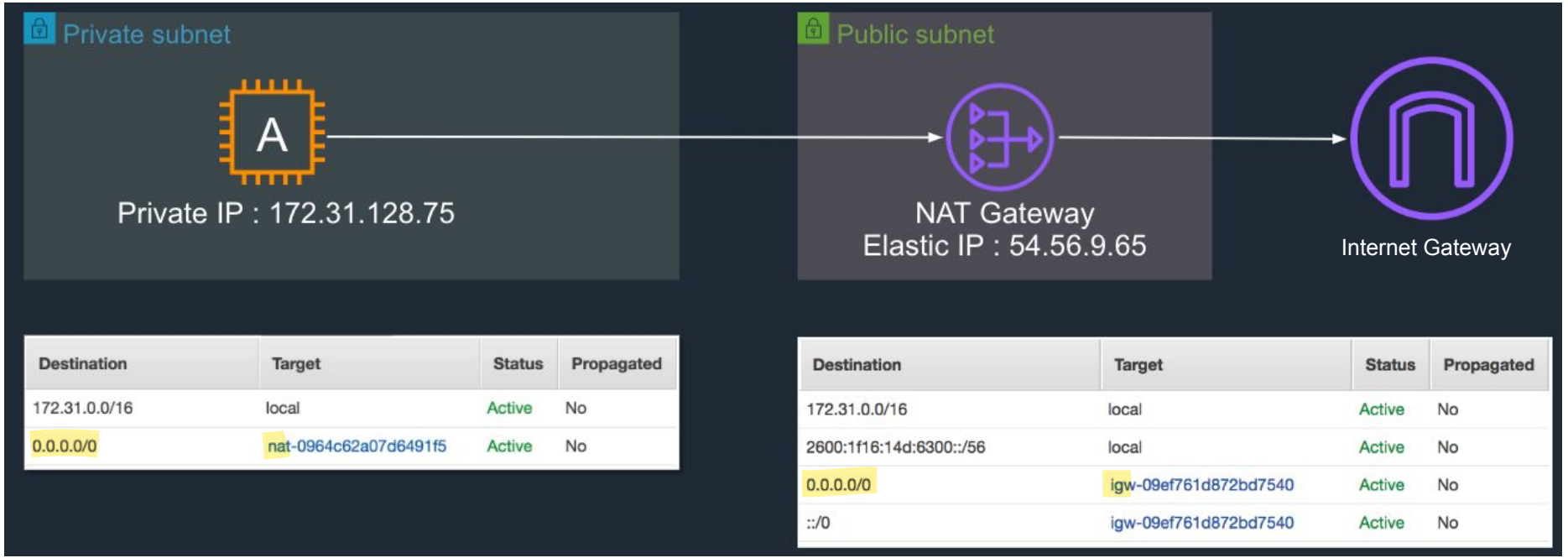
NAT Gateway

- Conexiones de salida hacia otras redes (ej: Internet)
- No permite entrada de datos
- Alta disponibilidad
- Requiere asignarle una Elastic IP



La diferencia con el IGW es que aca no tengo las IP mapeadas uno a uno. Aca todas las instancias salen con la IP del NAT GW. IMPORTANTE: Sigo necesitando el IGW para que el NAT GW salga a internet.

NAT Gateway - Ruteo



Destination	Target	Status	Propagated
172.31.0.0/16	local	Active	No
0.0.0.0/0	nat-0964c62a07d6491f5	Active	No

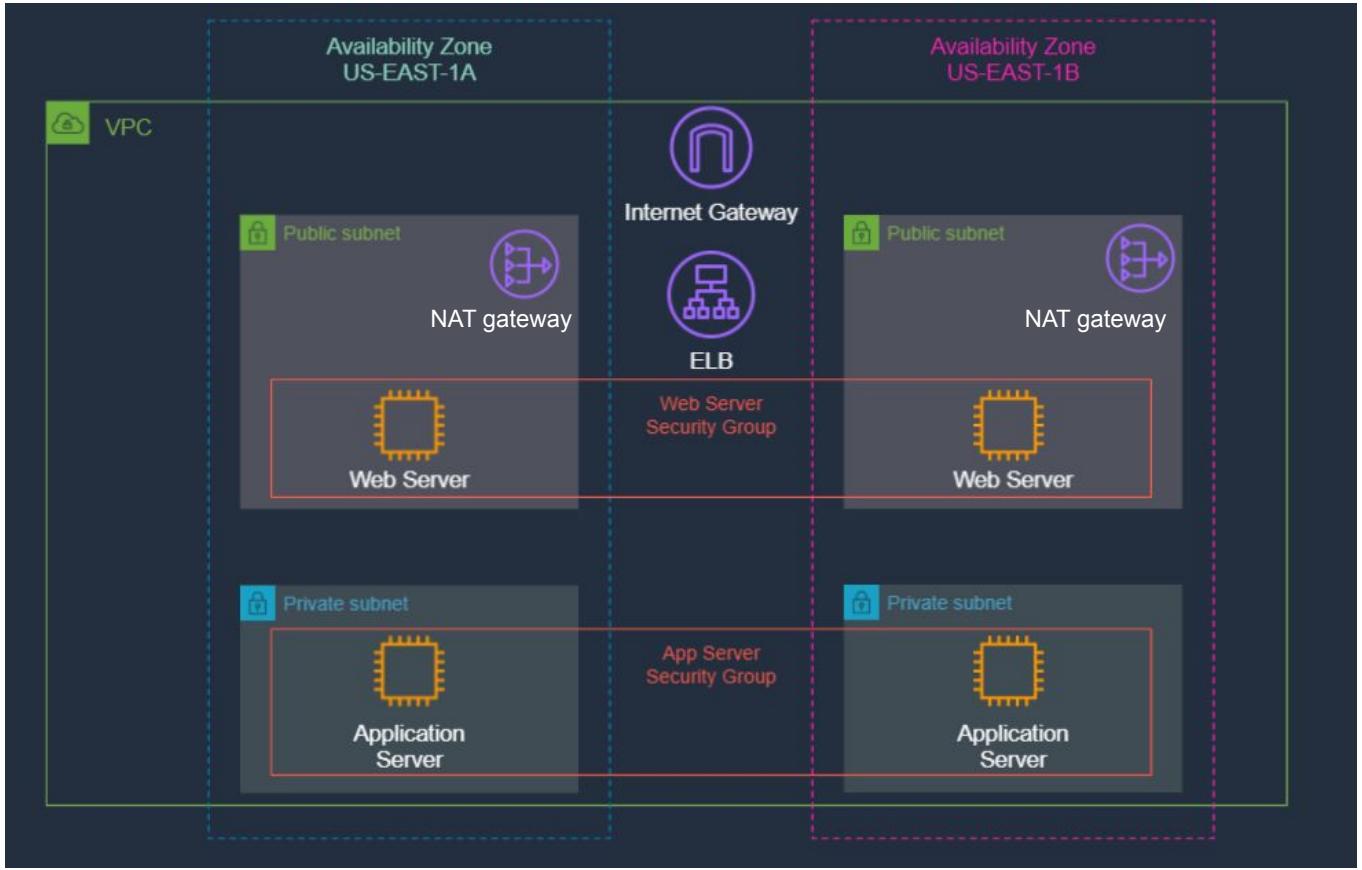
Destination	Target	Status	Propagated
172.31.0.0/16	local	Active	No
2600:1f16:14d:6300::/56	local	Active	No
0.0.0.0/0	igw-09ef761d872bd7540	Active	No
::/0	igw-09ef761d872bd7540	Active	No

Subredes Públicas y Privadas

NAT Gateway vs IGW

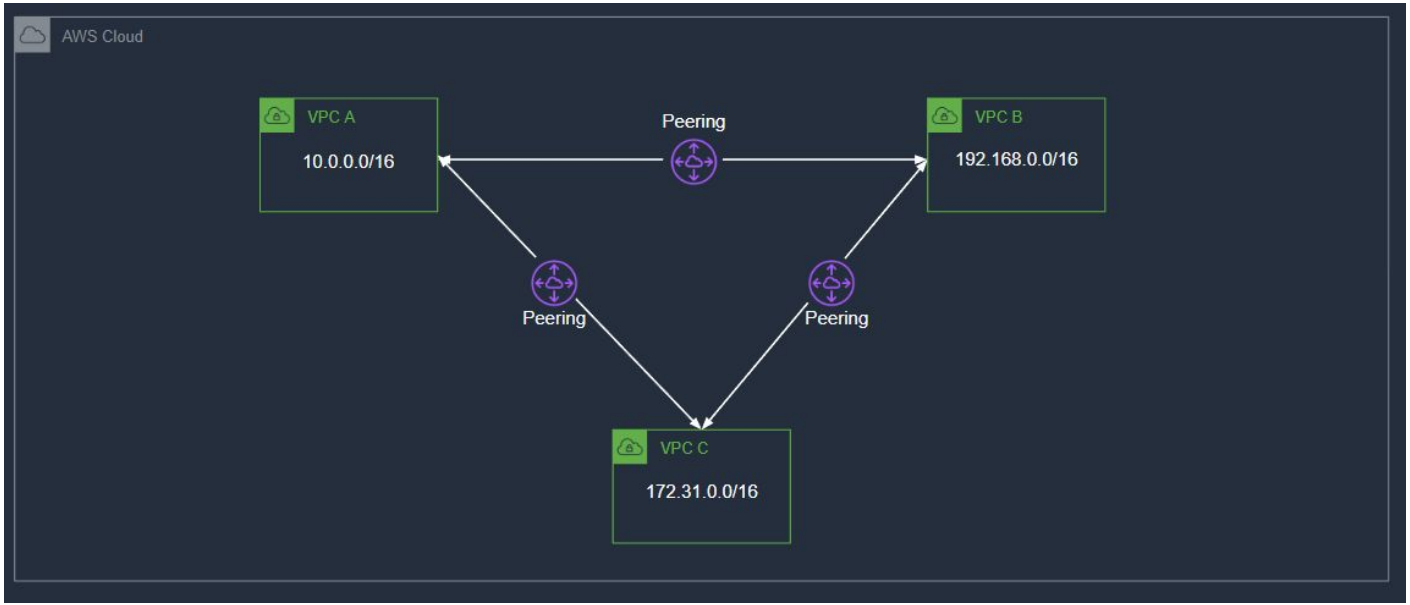
	NAT Gateway	Internet Gateway
Dirección	solo salida a Internet (u otra red)	entrada/salida desde/hacia internet
Ubicación	Public subnet	VPC-level
NAT	Stateful Todos salen a internet con la IP del NAT	Stateless (porque no tiene una IP asociada)
Usos típico	Descargar updates de Internet, conectarse a redes externas usando NAT	Publicar una instancia/servicio en Internet

Aplicación Web Ejemplo



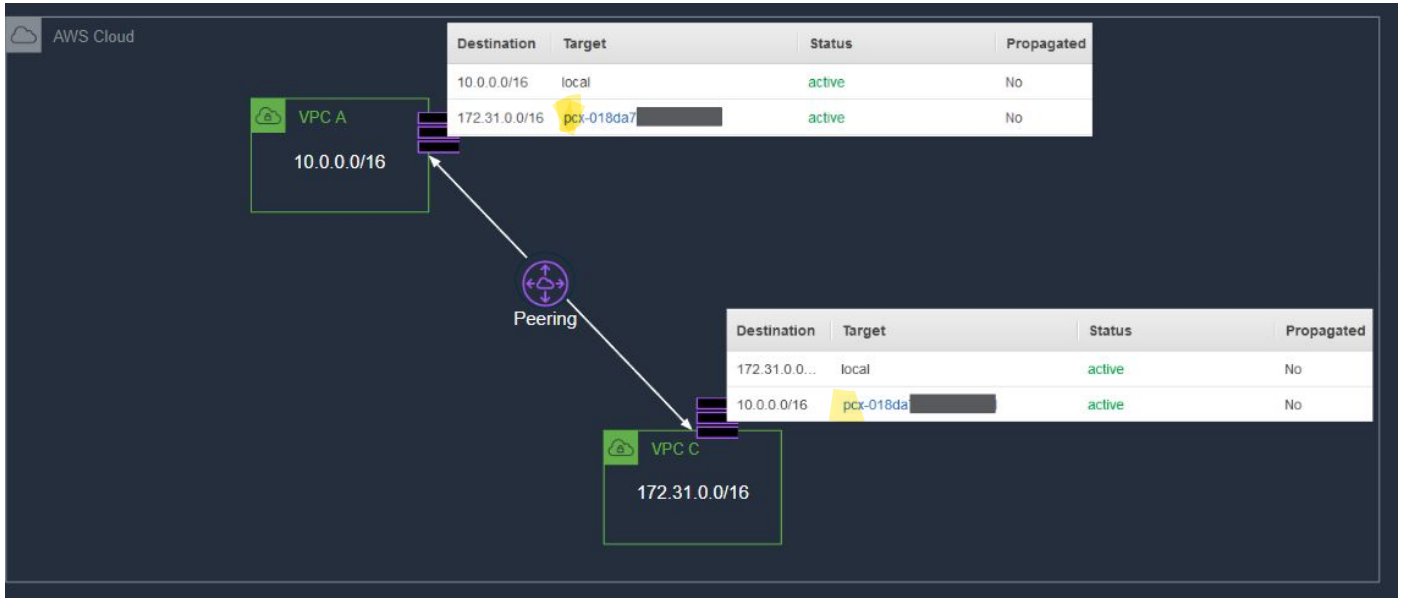
VPC Peering

- Las VPC por default están aisladas de otras redes privadas
- Cuando tenemos múltiples VPC podemos conectarlas entre sí:



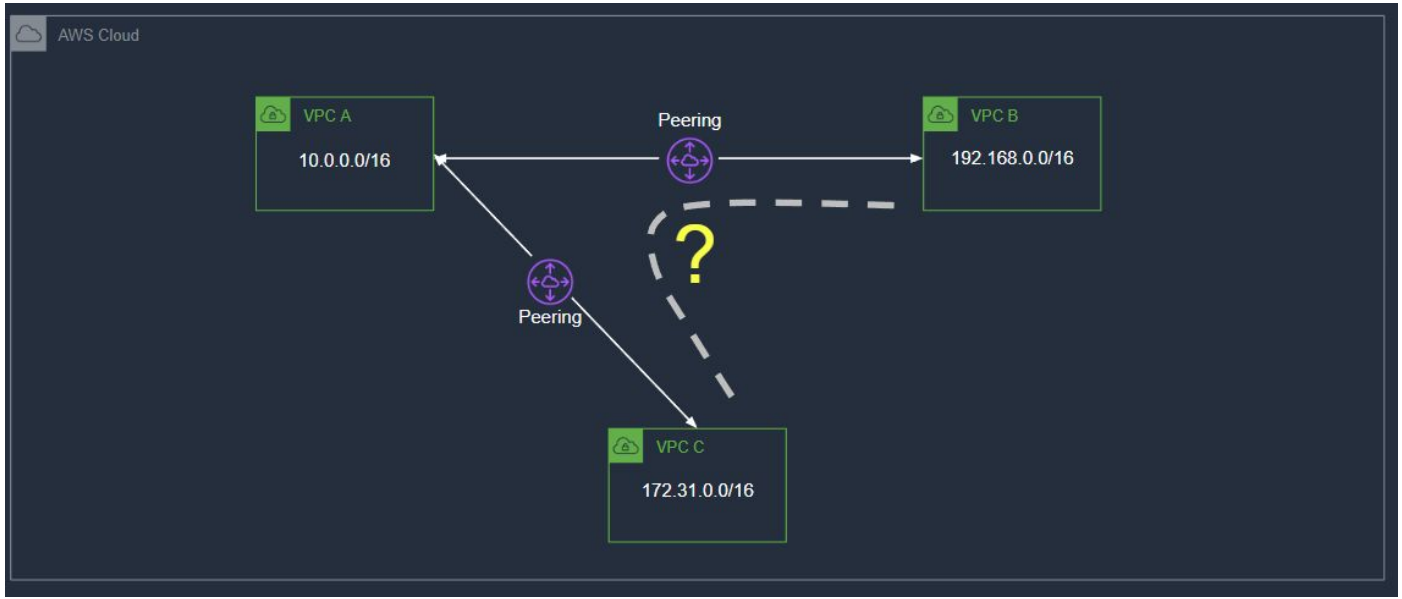
VPC Peering

- Se crea una tabla de ruteo con destino a equipo que hace el Peering



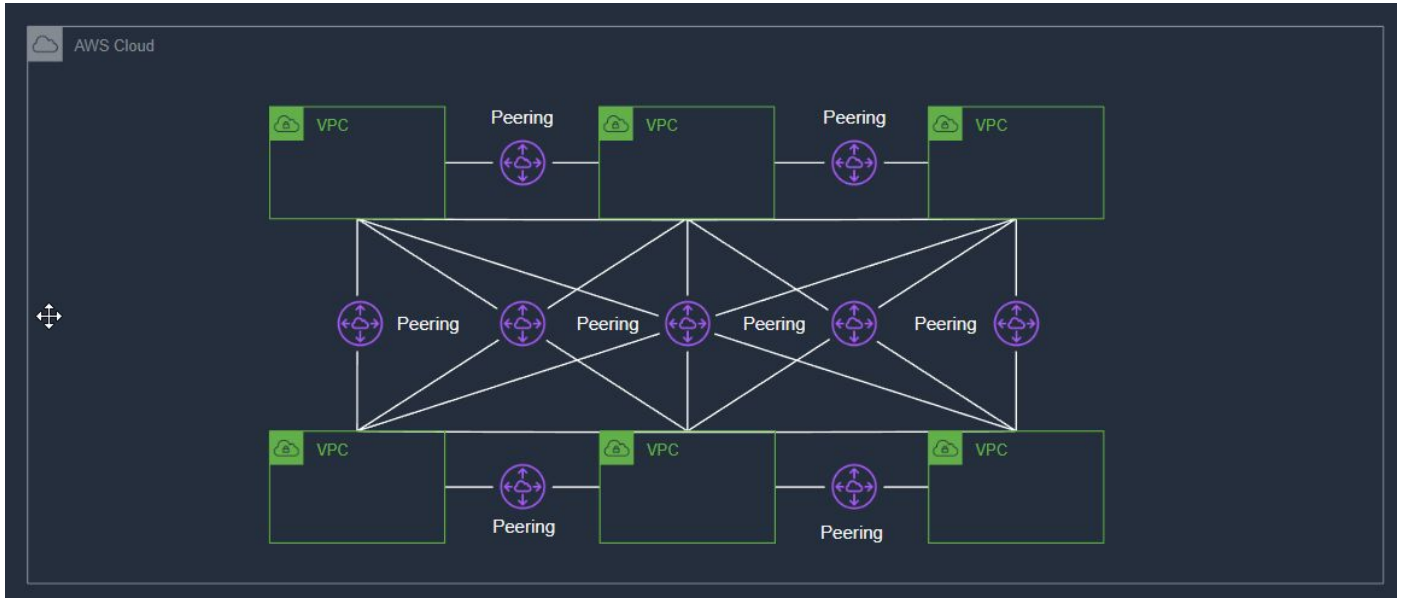
VPC Peering - Transit

- Cuando se hace más complejo se debe decidir si se hace de Transito o no



Transit Gateway

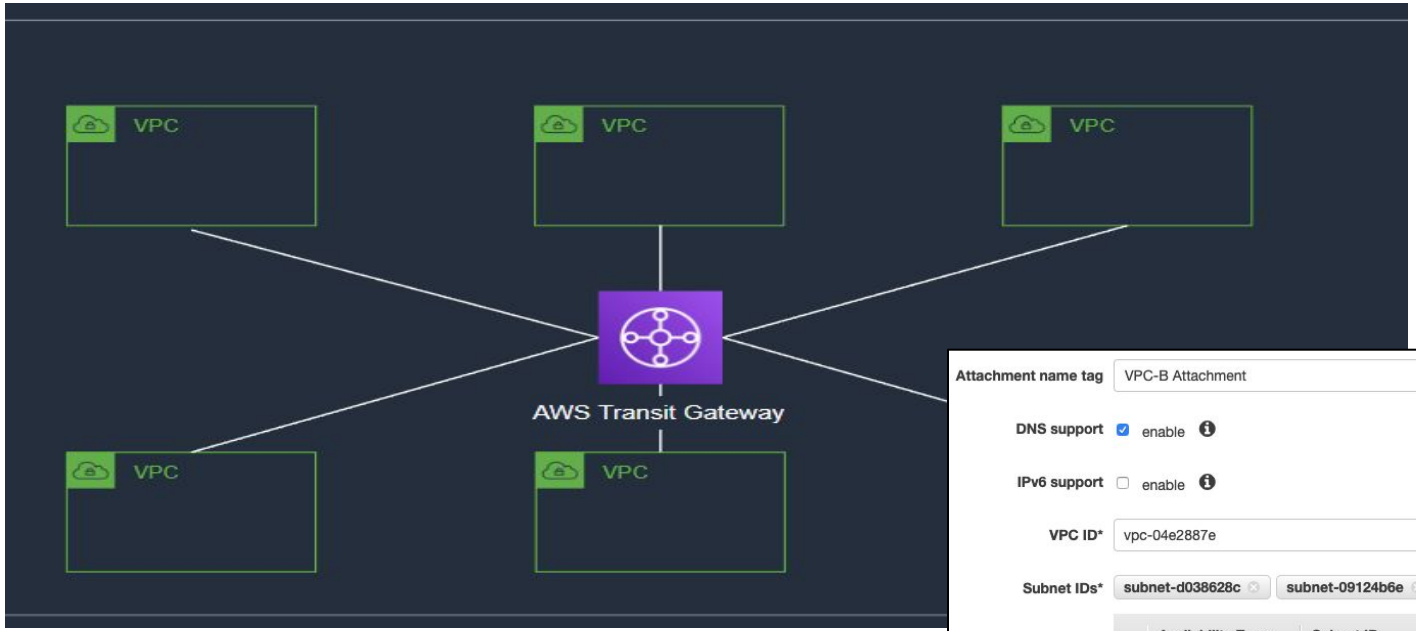
- Actúa cómo un router virtual regional
- Simplifica la conexión de:
  - Múltiples peering
  - Centro de Cómputos On-Premise

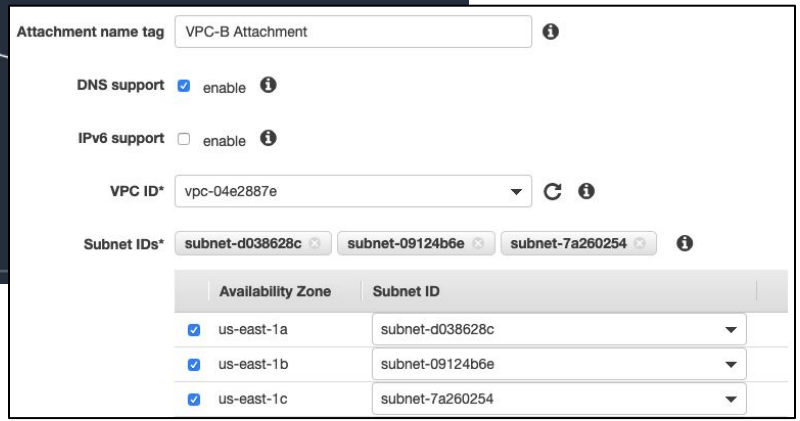


Transit Gateway

- Se declaran los ID de todas las VPC

El transit gateway configura todo automaticamente para no tener que hacer N peerings.





Definiciones AWS

- Internet Gateway: Entrada y salida de datos de Internet
- NAT Gateway: Salida de datos hacia Internet
- Virtual Private Gateway: Rutea Tráfico de VPN sitio a sitio ó conexiones dedicadas (AWS Direct Connect)
- Transit Gateway: Permite redes de tipo estrella (hub and spoke)

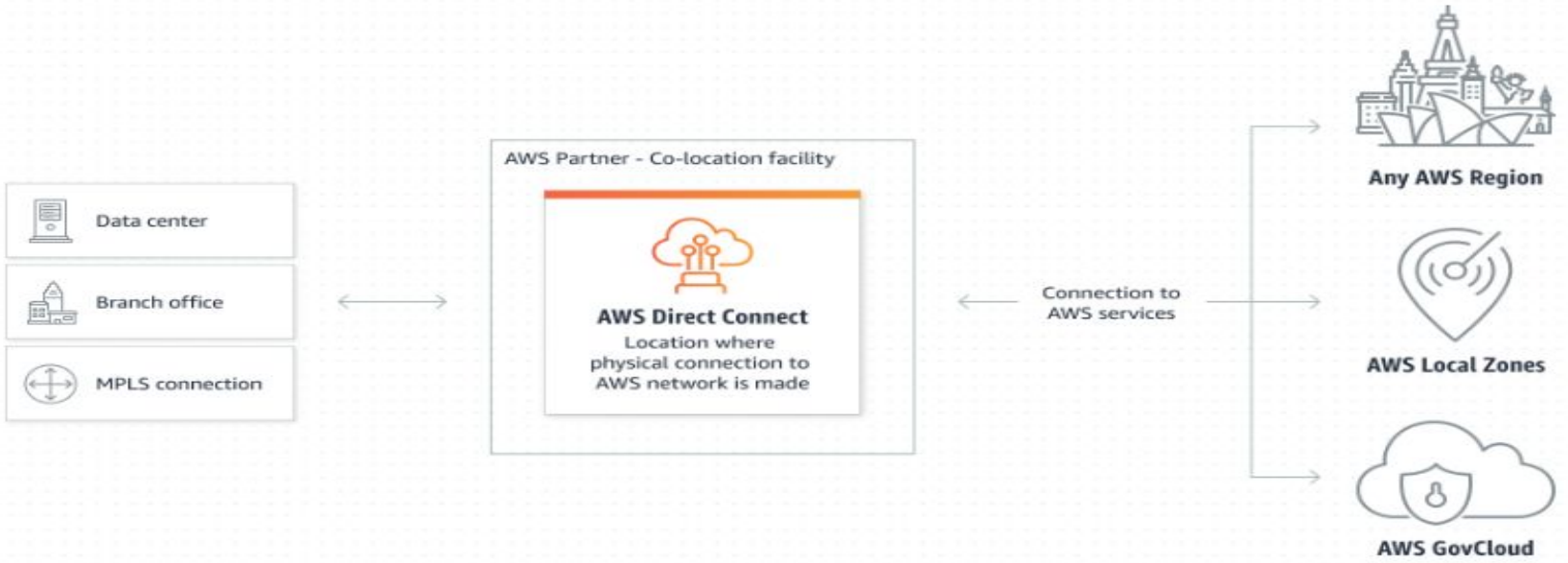
AWS Direct Connect es comprar una conexion dedicada a AWS. Amazon nos tira un cable a nuestras oficinas, y vamos derecho a la red de amazon sin pasar por internet.

Armar una red en AWS

1) Crear VPC
2) Definir subnets dentro de la VPC (al menos una subnet para cada AZ)
3) Armar las tablas de ruteo
4) Provisionar Internet Gateway/NAT Gateway (si quiero acceder a internet)

AWS Direct Connect

- Permite conexión directa a la red de AWS
  - Colocación: Llevar equipamiento (router) a AWS
  - Partner: Usar un partner que ya tenga equipos en AWS



AWS Direct Connect

- Se accede directo a la Región asociada (sin pasar por Internet)
- El equipamiento on-premise debe soportar 802.1q VLAN
- El Router debe soportar BGP

https://aws.amazon.com/es/directconnect/locations/

Asia Pacific | Canada | China | EU, Africa | Middle East / Israel | **Mexico, South America** | United States

AWS Direct Connect Location	Also accessible from:	Associated AWS Region	1G	10G	100G
Equinix BG1, Bogota, Colombia (1)		South America (Sao Paulo)	✔	✔M	
Equinix RJ2, Rio de Janeiro, Brasil (1)		South America (Sao Paulo)	✔	✔M	✔M
Equinix SP4, Sao Paulo, Brasil (1)		South America (Sao Paulo)	✔	✔M	✔M
Tivit, Sao Paulo, Brasil (1)		South America (Sao Paulo)	✔	✔M	✔M
KIO Networks QRO 1, Queretaro, Mexico (1)		US East (Virginia)	✔	✔M	
Cirion BNARAGMS, Buenos Aires, Argentina		South America (Sao Paulo)	✔	✔M	
Cirion LIMAPUCD, Lima, Peru (1)		South America (Sao Paulo)	✔	✔M	
Sonda Quilicura Q2, Santiago, Chile (1)		South America (Sao Paulo)	✔	✔M	

El private link hay que configurarlo aparte, no es automatico

AWS Private Link

- Permite conexión directa entre una VPC y otro servicio sin pasar por Internet

AWS tiene sus cables submarinos por todo el mundo. Ellos te ofrecen llegar de un servicio a otro sin pasar por internet en ningun momento (por mas que tenga un servicio en US y otro en JAPAN)

- Se crea un VPC endpoint para el servicio externo
- Esto genera una Elastic Network Interface (ENI) en nuestra subnet con una IP privada, que sirve de punto de entrada para el tráfico destinado al servicio
- Se puede utilizar para servicios de AWS (ej: S3) o externos
- Útil para compliance

Si yo tengo una instancia que quiere leer un archivo en S3, entonces puedo acceder en forma interna sin pasar por internet.

Compliance porque hay leyes internacionales (como en la EU) que piden que los datos de los clientes se guarden en X pais

Zero Trust Policy

- Modelo conceptual de seguridad
- Basado en la idea de que el acceso no depende solamente de la ubicación en la red
- No invalida el modelo tradicional, sino que se adiciona al mismo
- Identity-Aware Networks

Zero Trust Policy

- Principios:

(lo que seria el modelo tradicional, que son controles y bloqueos a nivel red y puerto)

  - El rol de los controles en la red y los perímetros siguen siendo importantes para la arquitectura de seguridad general
  - El significado de Zero Trust puede variar según el contexto
  - Los conceptos deben ser aplicados de acuerdo al valor asignado a los sistemas y datos que estamos protegiendo

Por ejemplo: si yo accedo de determinado lugar, tengo que poder acceder al servicio, y si accedo de otro lugar no

Ejemplos de Zero Trust en AWS

- La API de AWS
  - Accesible a todo el mundo desde Internet (entonces no la estoy segurizando a nivel firewall)
  - Cada request es autenticado y autorizado cada vez (con un token)
- Service-linked roles
  - Utilizados cuando un servicio de AWS necesita interactuar con otro

  Un ejemplo es IAM. Cuando un servicio tiene que acceder a otro, en vez de definirle un usuario y clave, directamente le doy un permiso.
- IoT
  - El tráfico entre dispositivos IoT y los servicios de AWS IoT utiliza Transport Layer Security (TLS) y autenticación, incluyendo certificados