# Prework Sesión 3

## Introducción

El proceso de transferencia de datos en cualquier infraestructura se efectúa por medio de la red. Si bien la nube quita mucho trabajo a la hora de configurar redes en comparación a redes on premise de dispositivos de marca Cisco por ejemplo, aún así se deben generar configuraciones de redes pues no siempre todos los dispositivos, servicios o instancias conectadas a una red deben tener comunicación con otros, todo dependerá de la arquitectura de la solución pero una arquitectura básica puede ser la basada en [DMZ](https://searchsecurity.techtarget.com/definition/DMZ).

Las redes requieren direcciones IP para funcionar, pero para nosotros como humanos no es cómodo trabajar con direcciones IP, requerimos algo más intuitivo, de esa necesidad surge el protocolo DNS y AWS cuenta con él.

Gracias a las redes y protocolos DNS es posible generar balanceadores de carga es decir servicios capaces de brindar una única cara al cliente para recibir peticiones para después redireccionarlas o despacharlas a un servidor específico de las decenas que pueden estar conectados con el balanceador de carga.

### 1. Objetivo 🎯

* Conocer las opciones que ofrece AWS para la gestión de redes y como interactuan con otros servicios.

### 2. Instrucciones 📋

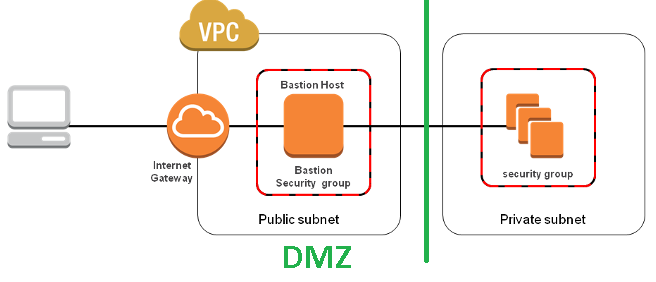
Conviene ver los siguientes videos para reafirmar algunos conceptos que serán útiles. - [HTTP Load Balancing with Nginx](https://www.youtube.com/watch?v=SpL_hJNUNEI)

-[¿Qué es un DNS?](https://community.icann.org/download/attachments/62395695/1-ICANN58-Mar2017-What-is-the-Domain-Name-System.pdf?version=1&modificationDate=1489502569000&api=v2)

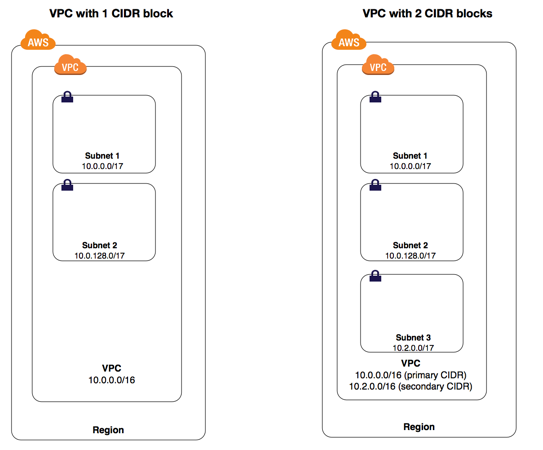
* [Understanding How DNS Works in Depth](https://www.youtube.com/watch?v=T-eghY-9WdE)
* [Nuevas Edge Locations](https://aws.amazon.com/es/about-aws/whats-new/2020/09/amazon-cloudfront-launches-in-two-new-countries-mexico-and-new-zealand/)
* [¿Qué es un CDN?](https://www.youtube.com/watch?v=52VSbXBlfdc)
* [¿Qué es DHCP?](https://www.ionos.es/digitalguide/servidores/configuracion/que-es-el-dhcp-y-como-funciona/)
* [¿Por qué es importante hacer subnetting?](https://www.ionos.com/digitalguide/server/know-how/subnetting-how-do-subnets-work/)

### 3. Desarrollo 📑

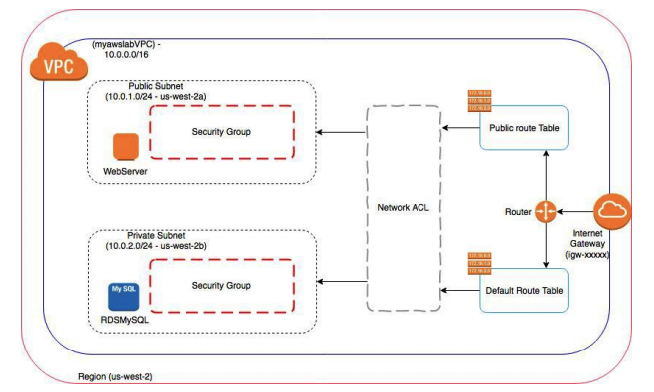
# Amazon Virtual Private Cloud (VPC)

AWS VPC es una red definida por software (red virtual) aislada de otras redes virtuales optimizada para el transporte de masivas cantidades de paquetes de red hacia, desde y entre regiones de AWS. En una VPC es posible conectar recursos de AWS como instancias EC2, como toda red esta red puede ser dividida en subredes para mejor administración del espacio de direcciones IP. Hay una red virtual VPC generada en cada región por defecto, cuenta con conexión hacia y desde internet con al finalidad de conectar servicios de AWS rápidamente. Dependiendo de la arquitectura de las soluciones a desplegar no siempre será deseado que los recursos puedan ser accedidos desde internet directamente, típicamente el ejemplo es la conexión a base de datos, las bases de datos normalmente se encuentran asiladas del acceso desde internet directamente siendo accedidas por medio de un servidor web en las llamadas DMZ. 

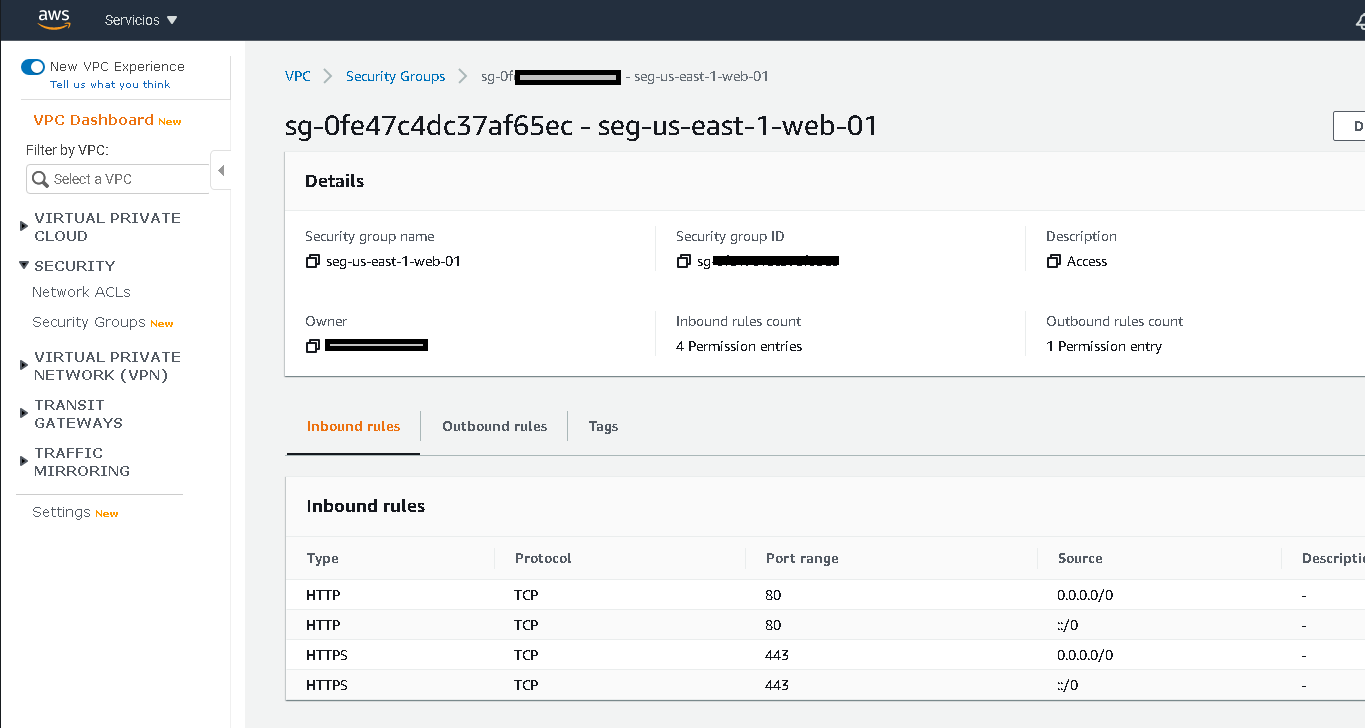
Las redes virtuales VPC se conforman de un gran bloque de direcciones IP, la VPC por defecto cuenta con una máscara de subred /16 (es posible conectar 65,534 dispositivos con este espacio de direcciones IP), se cuenta con la suficiente flexibilidad para generar redes VPC con máscara de subred de tamaño 28.

Para mejorar el tráfico de las redes y mejorar la administración del espacio de direcciones IP las VPC pueden ser divididas en espacios de direcciones IP mas pequeños, estos espacios son llamados subredes, la división del espacio de direcciones IP en VPC corresponde a la notación [CIDR](https://www.youtube.com/watch?v=z07HTSzzp3o) (Classless Inter-Domain Routing) tradicionalmente usada a la hora de administrar cualquier red. En el siguiente ejemplo se ver como un espacio de direcciones IP puede ser manejado en redes mas pequeñas. 

Para la mejora de la seguridad entre subredes es posible definir reglas que dicten que tipo de tráfico puede fluir entre ellas tanto de entrada como de salida por medio de ACL (Access Control List), las ACL se encuentran definidas a nivel de VPC y pueden ser asociadas con una o mas subredes. Para manejo mas granular de la seguridad de tráfico de red es posible definir reglas que regulen el tipo de tráfico no solo entre subredes si no entre instancias EC2 (o instancias RDS o Lambdas por ejemplo) por medio de los Security Groups, pueden ser vistos como un firewall muy básico asociado a una instancia, así es posible definir un servidor Web EC2 con cara hacia internet, en el Security Group asociado a esta instancia EC2 se permite solo el acceso desde internet por los puertos 80 y 443.



e23cf7b9e8d458ba8d749a9747cd2f42.png

A continuación observa el detalle de un Security Group definiendo reglas de entrada de tráfico a la instancia solo a tráfico http y https: 

Para conectar las redes VPC a internet se hace uso de un Internet Gateway, un IG es un componente de VPC de escala horizontal, redundante y de alta disponibilidad sin riesgos de disponibilidad ni limitaciones de ancho de banda.

Al hablar de subredes es necesario hablar de routeo, AWS define tablas de routeo que dictan como será dirigido el tráfico desde y hacia internet, por defecto todas las subredes de una VPC pueden comunicarse con otras subredes sin definir explícitamente reglas de routing, las reglas de routing aplican para el trafico entrante y saliente de la subred. Una red privada es definida si no tiene routing de su tráfico hacia un Internet Gateway sin embargo muchas veces es necesario tener acceso para descargar software o actualizaciones en las instancias, para lo cual hay un mecanismo que permite conectar estas instancias en redes privadas hacia internet solamente, el mecanismo es un llamado NAT Gateway, con él se puede definir en la tabla de routeo que si una instancia desea conectarse a internet desde una red privada en lugar de hacerlo por un Internet Gateway de la VPC (El internet Gateway es bidireccional) lo haga por medio de un NAT Gateway, con ellos es posible tener las instancias privadas conectadas hacia internet.

Para dejar mas claro los conceptos tocados, analicemos el siguiente diagrama de la Universidad de Cornell. Diagrama

Podemos ver que la configuración tiene una VPC con mascara 22, lo que significa es posible tener disponibles 1022 IP para host, esta VPC tiene su Internet Gateway para comunicar las redes publicas hacia internet y desde internet. Ellos configuran 2 redes públicas (que pueden ser accedidas desde internet) y dos privadas (no pueden ser accedidas desde internet directamente), hay que notar que cada par esta alojado en zonas de disponibilidad distintas, esto para brindar cierta tolerancia a fallos. Se observan también las subredes, cada una con un espacio de direcciones IP de 252 hosts ya que cada una tiene una mascara 24, con lo que se pueden tener 252 dispositivos que necesiten una IP, no necesariamente tienen que ser 252 instancias, puede haber instancias que sean configuradas con 2 direcciones IP o más. Otro elemento a destacar es el NAT gateway que siempre reside en una red pública, con él las redes privadas tendrán acceso hacia internet, muchas veces para configurar un servidor se requiere acceso a repositorios de software externos. Finalmente VPG (Virtual Private Gateway) da la flexibilidad de conectar por medio de una VPN las instalaciones de la propia universodad con AWS.

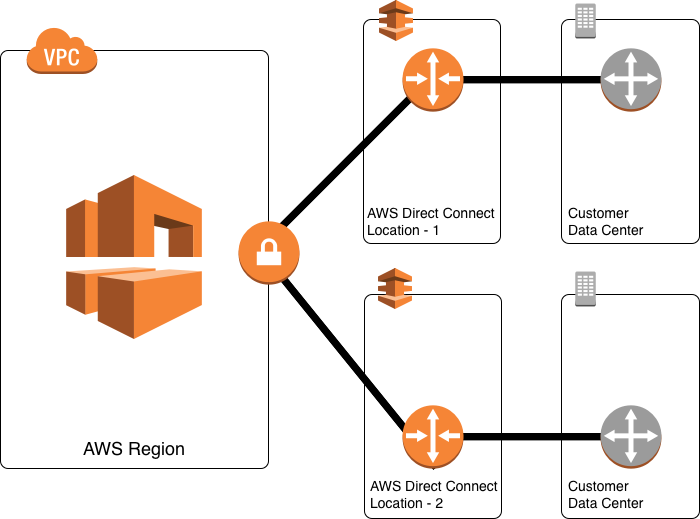
¿Qué hacer y tomar en cuenta a la hora de construir una VPC? Las VPC tienen un límite, AWS por defecto limita a 5 VPC por región, este límite se puede superar pero solo hasta 100 VPCs, de la misma forma, el número de bloques de direcciones IP permitidos en cada VPC es un máximo de 50 sin poder aumentar este límite. Al establecer el tamaño del bloque VPC y de las subredes se tiene que tomar en cuenta que este valor ya no puede ser modificado, la VPC y las subredes ya se quedan con el numero de IP definidas desde la creación, así que si se estima un crecimiento rápido de recursos en una VPC es mejor ser generoso con el bloque de direcciones IP. Tomar en cuenta que las subredes pertenecen a una zona de disponibilidad. No todas las redes deben tener acceso directo desde internet, hay que ser cuidadoso a la hora de configurar el routing para asegurar que las redes definidas como privadas en realidad lo sean. En la mayoría de los casos es buena práctica manejar múltiples capas de acceso en la infraestructura, ej: se tiene una VPC para producción subdividida en una red pública destinada a los servidores web y otra subred privada para las bases de datos.

# Conectividad con VPNs y Amazon Direct Connect

# TODO Completar

Una red pública es básicamente el tipo de red que proporciona un servicio de conexión o telecomunicaciones a cambio del pago de una cuota de servicio. Existen redes privadas en organizaciones con multitud de dispositivos ubicados dentro de un recinto, conectados a switchs a través de cables. En estas redes privadas, un usuario que sea externo a esta red no podrá conectarse a ella para acceder a Internet, el acceso solamente esta restringido a los usuarios que estén en el interior de ella, en la mayoría de los casos claro.

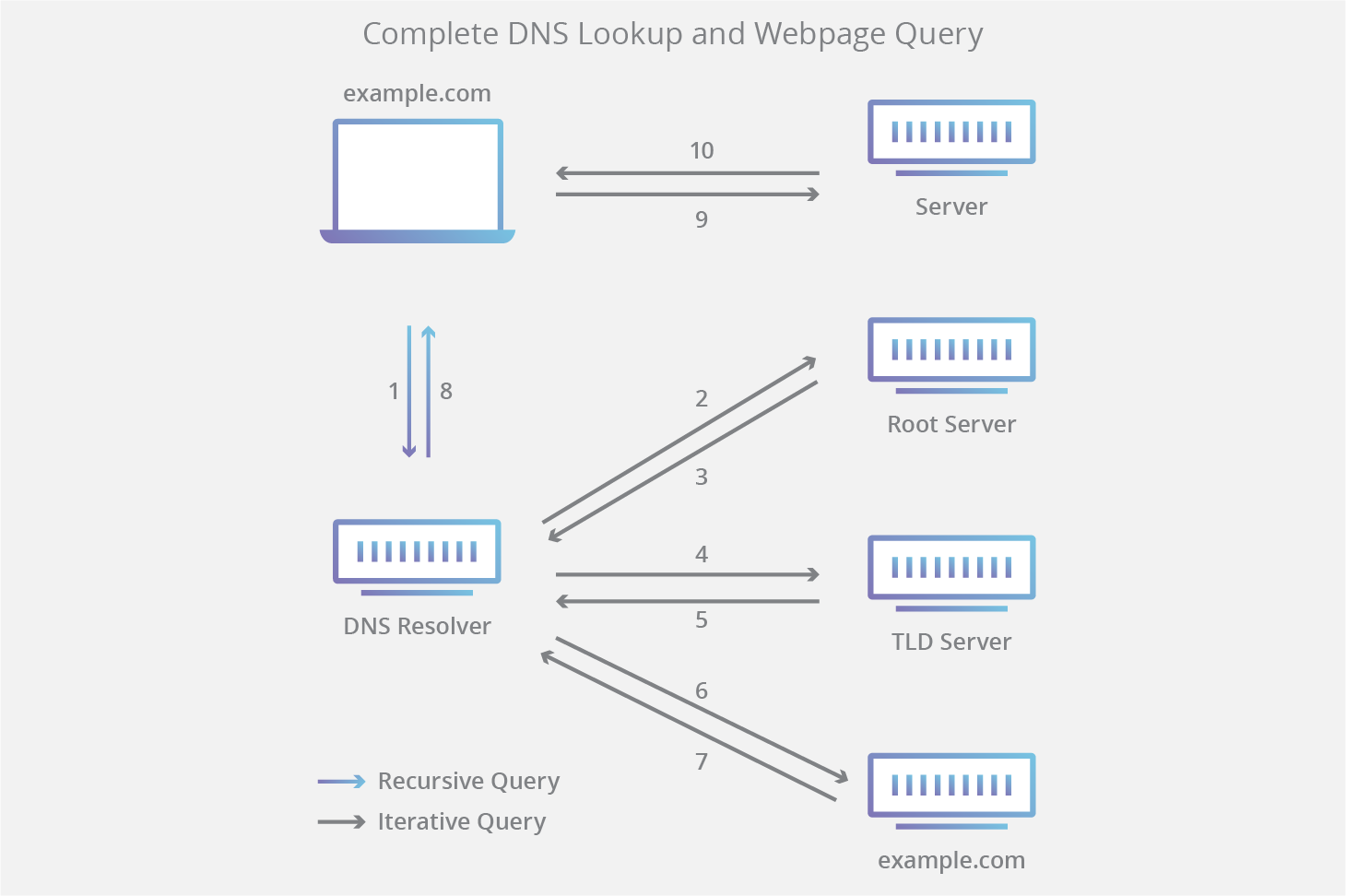
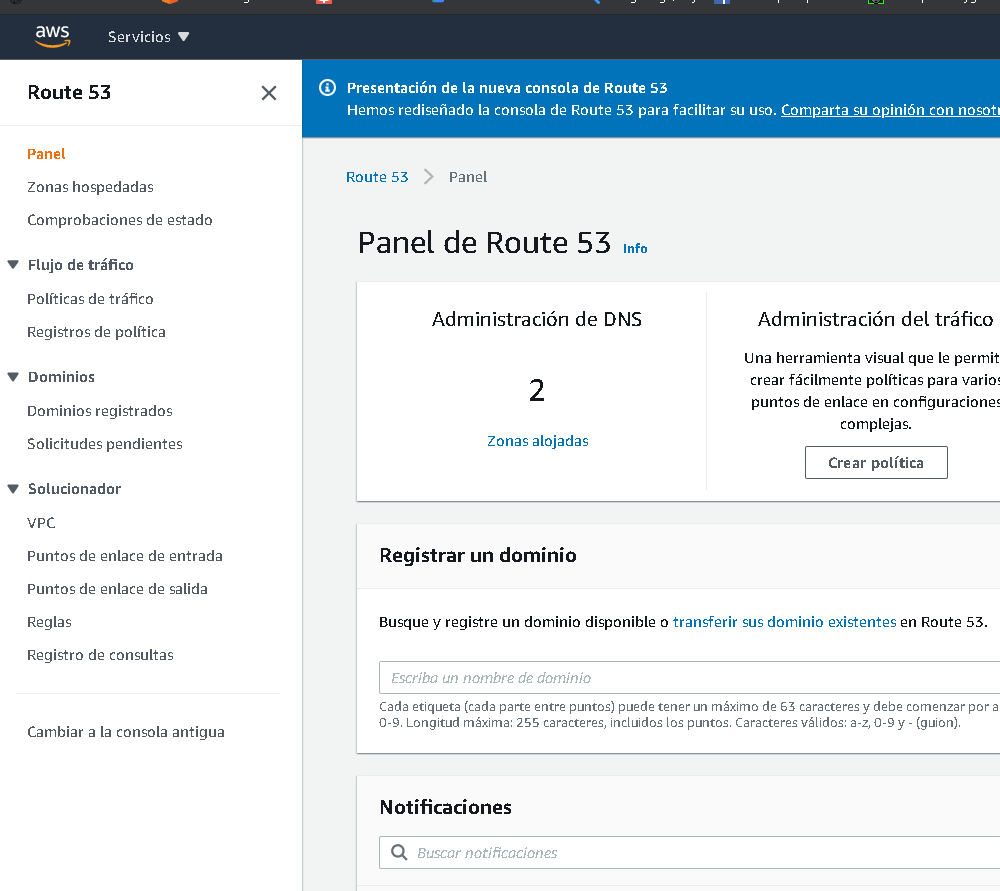
Es la solución de AWS para conectar la red on-premises a la red de AWS en un enlace dedicado.

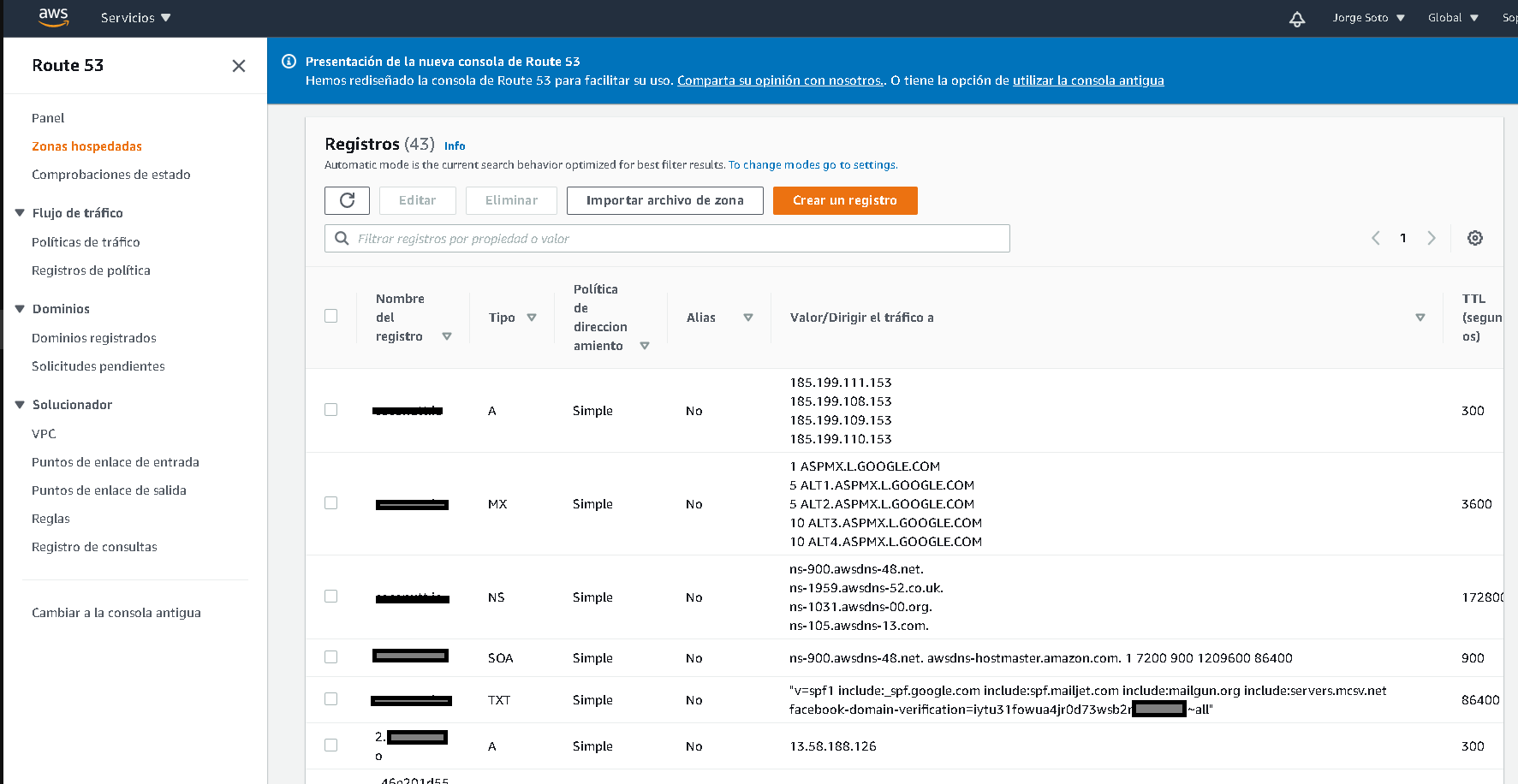


bc0712eb21de5e97c7963e8573afb3fa.png

# Amazon Route 53

Amazon Route 53 proporciona una facilidad para registrar nombres de dominio, un servicio de Sistema de Nombres de Dominio (DNS ) para que los nombres de dominio se traduzcan a direcciones IP.

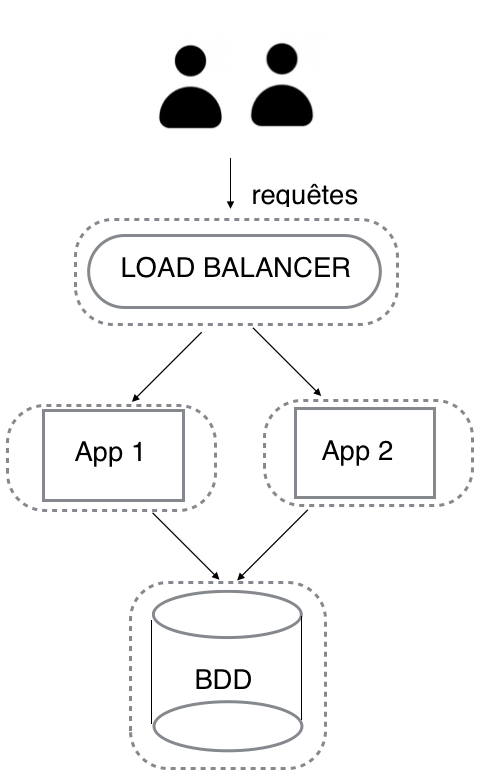
Al teclear example.com un servidor debe responder a la petición, pero los dispositivos de red funcionan con direcciones IP, para traducir example.com a una dirección IP con la que routers puedan dirigir la petición al servidor correcto el DNS buscará en su propia tabla la correspondencia de dominio y dirección IP, si el no la tiene preguntará a otro servidor hasta encontrar la equivalencia.  En Route53 es posible dar de alta múltiples dominios. En el siguiente ejemplo son 2. 

Al ingresar al detalle de la zona (una zona es un dominio DNS manejado por AWS) se visualizan los distintos tipos de registros DNS, MX, TXT, A..  Al querer generar un nuevo registro en el DNS (tal ves se requiere agregar un nuevo subdominio para acceder a una nueva página web), será necesario seleccionar entre varias de política de redireccionamiento. 

El routing simple redirige las peticiones a un servidor. La opción ponderado dirige el tráfico a varios recursos en las proporciones diferentes, 80% peticiones al servidor1.com y 20% al servidor2.com. Geolocalización permite dirigir tráfico a recursos y/o servidores basados en la localización física del cliente. Por latencia verifica la petición y la redirige a los recursos AWS con la región con menos latencia. Por conmutación por error ayuda a implementar una estrategia de fail-over de tipo activo/pasivo. Respuesta de varios valores redirige el tráfico de forma pseudo aleatoria entre hasta 8 direcciones IP.

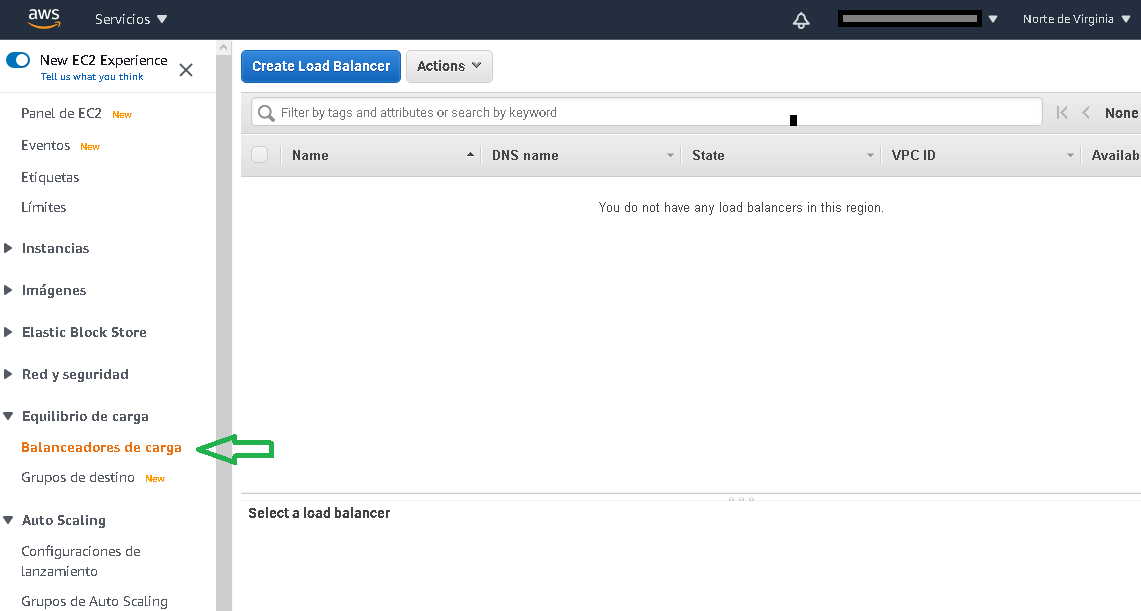
Si se selecciona un registro simple se pueden agregar valores estándar del protocolo DNS como CNAME, TXT, A, pero se vuelve interesante la integración de AWS con servicios como Elastic Beanstalk, Balanceadores de carga, Buckets S3 configurados para hosting web, AWS API Gateway, AWS Cloud Front, lo que da mucha facilidad a la hora de definir las reglas para resolver direcciones IP.

# Elastic Load Balancing

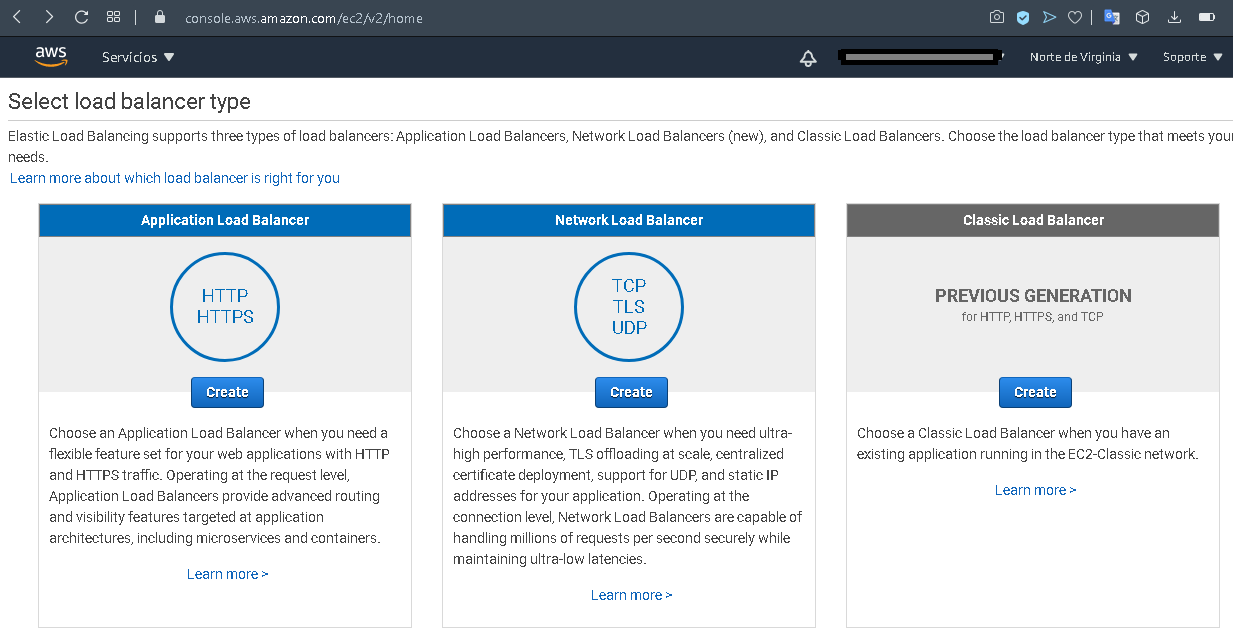
Al tener cargas de trabajo grandes surge la necesidad de distribuir la carga de trabajo entre varios servidores, la distribución puede ser hecha por medio de un balanceador de carga. Imaginar un e-commerce en época navideña, las peticiones al e-commerce subirán y un solo servidor posiblemente no tenga lo necesario para atender todos los posibles compradores que se conectan a él, es posible optar por una estrategia de escalamiento horizontal para generar múltiples instancias del e-commerce (siempre y cuando la arquitectura del e-commerce lo permita). Al contar con múltiples servidores operando se debe hacer uso de un balanceador de carga encargado de redirigir las peticiones a un servidor dado, de cara al cliente solo verá una sola dirección IP pero en realidad son múltiples servidores los que están atendiendo las peticiones. 

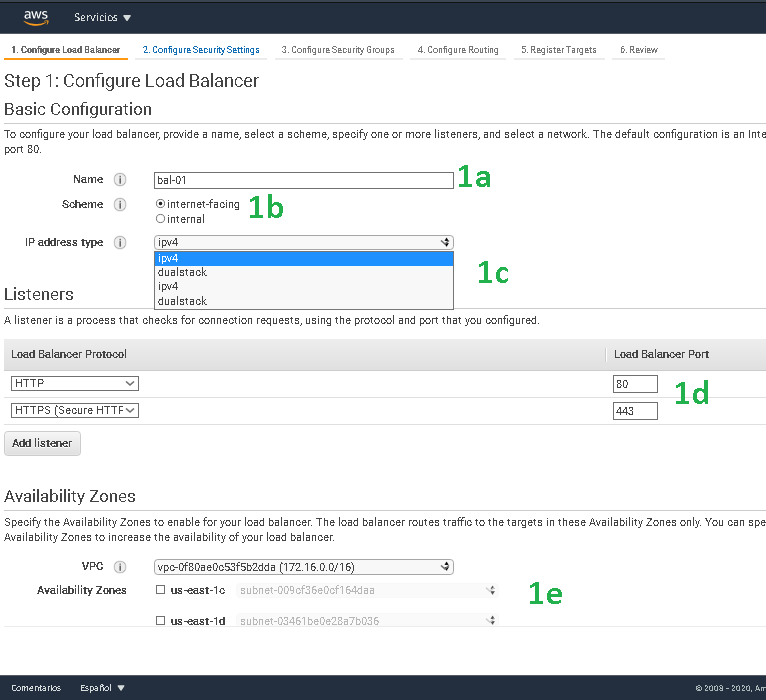
AWS cuenta con balanceadores con modelo PaaS con el servicio Elastic Load Balancing. Elastic Load Balancing cuenta con tres tipos de balanceadores de carga: Application Load Balancers, Network Load Balancers y Classic Load Balancers.  
Application Load Balancers se utilizan para redireccionar el tráfico HTTP / HTTPS (o Capa 7). Los Balanceadores de carga de red y los Balanceadores de carga clásicos se utilizan para redireccionar el tráfico TCP (o Capa 4).

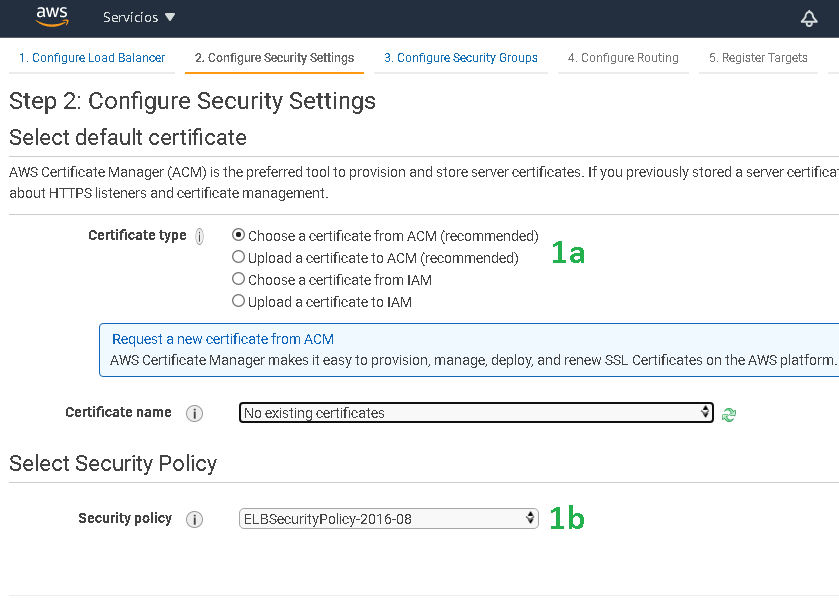
Para generar un balanceador de carga en AWS ir a la seccioné EC2 y seleccionaremos Balanceadores de carga



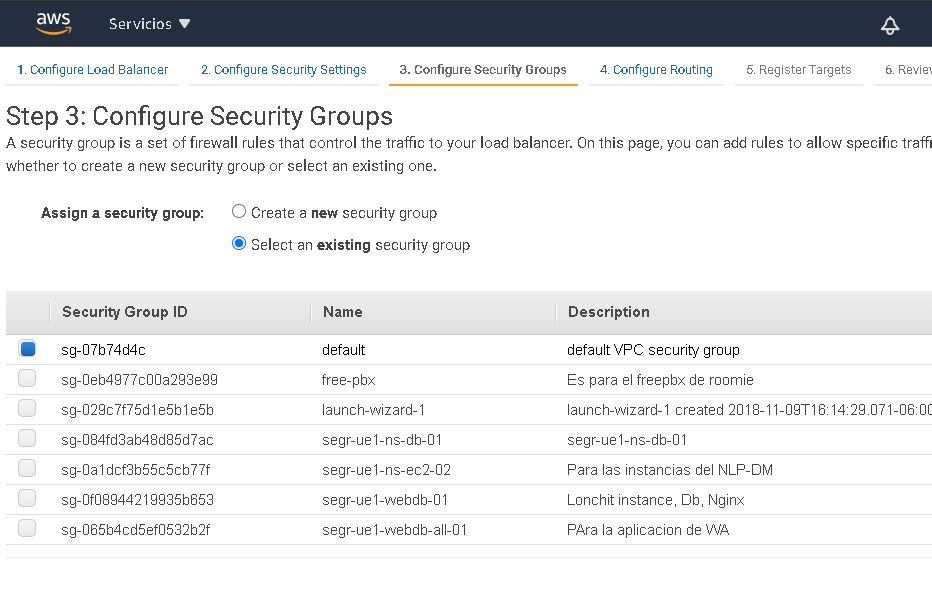
c462c5a456e53527d62a2f5948a8bb14.png

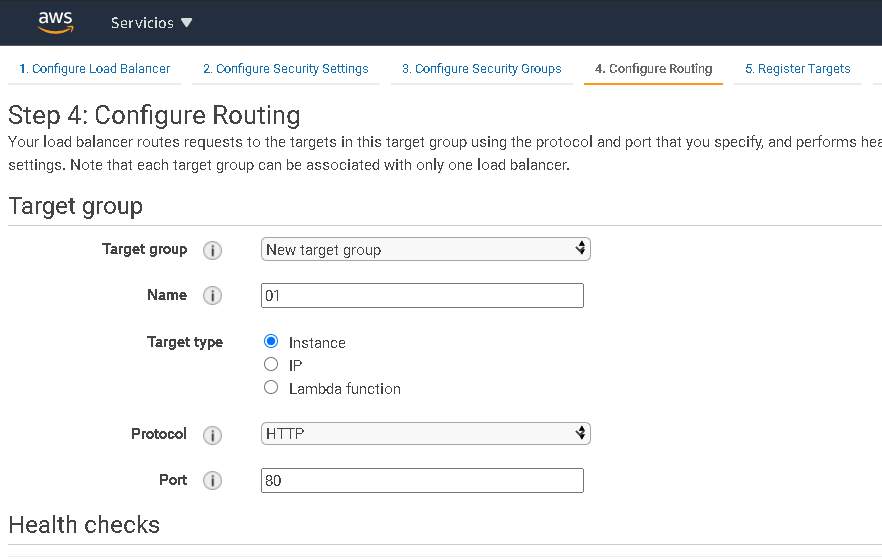
Seleccionar el tipo de balanceador de carga, ya dependiendo del tipo seleccionado estarán disponibles diferentes opciones de configuración. 

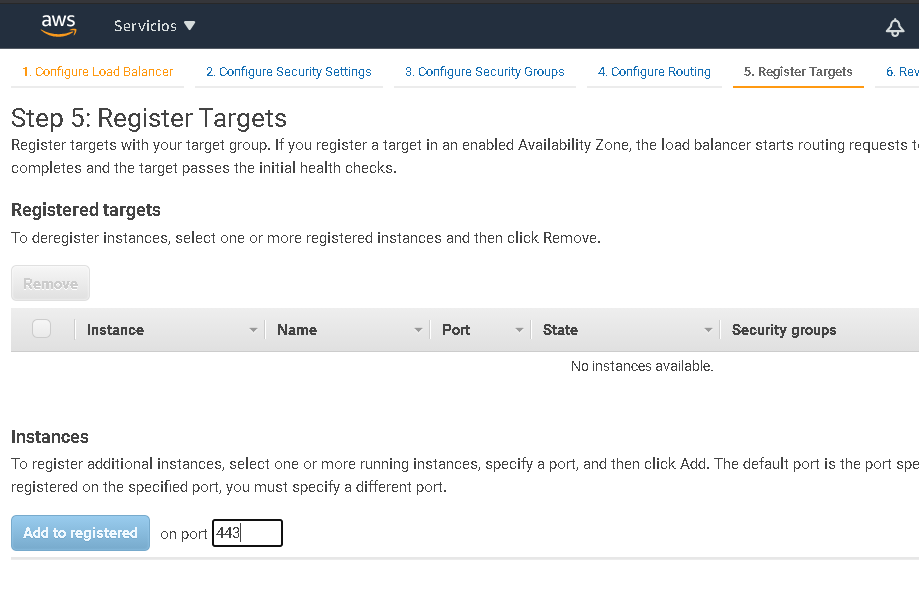
En caso de seleccionar Application Load Balancer se tiene que especificar un nombre para el balanceador de carga (1a), si será un balanceador de carga interno para manejar tráfico con origen y destino entre subredes o externo para manejar tráfico de origen desde internet (1b), se debe especificar si se requiere trabajar con ipv6 además de ipv4 (1c), especificar los puertos de escucha (1d), por el momento solo se pueden manejar los protocolos http y https con posibilidad para seleccionar un número de puerto no estándar, por último seleccionaremos la VPC dentro de la que se habilitará el balanceador teniendo que seleccionar las zonas de disponibilidad y las subredes entre las que habrá balanceo de carga, muy importante tomar en cuenta a la hora de diseñar el balanceo de carga, solo se puede escoger una subred por zona de disponibilidad. 

En la siguiente sección se debe seleccionar como se debe manejar el cifrado de la conexión, se brinda la opción de subir un certificado SSL propio en caso de ya contar con uno de un proveedor externo o se puede escoger alguno ya hecho en AWS Certificate Manager (1a), después seleccionar los protocolos de cifrado que se usarán para establecer el canal seguro (1b) 

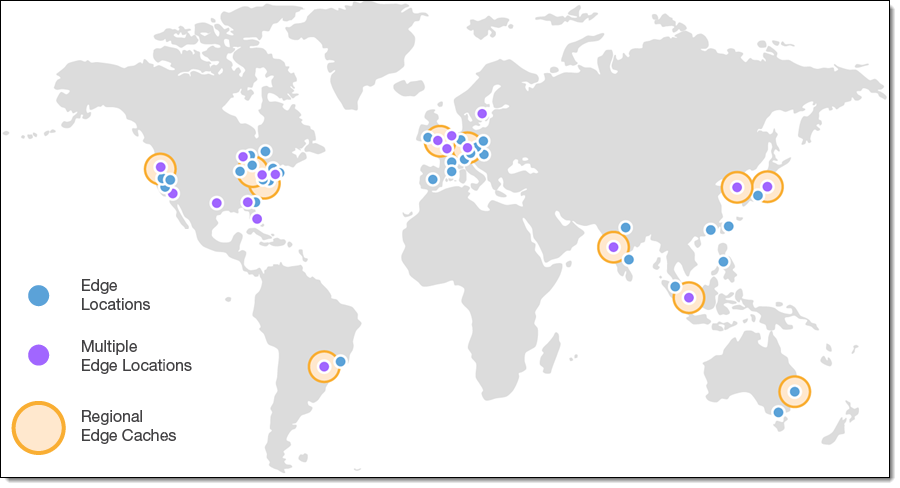
Dependiendo de security policy seleccionada serán los protocolos de cifrado que dispondrá el balanceador de carga para negociar con los clientes que se conectarán. 

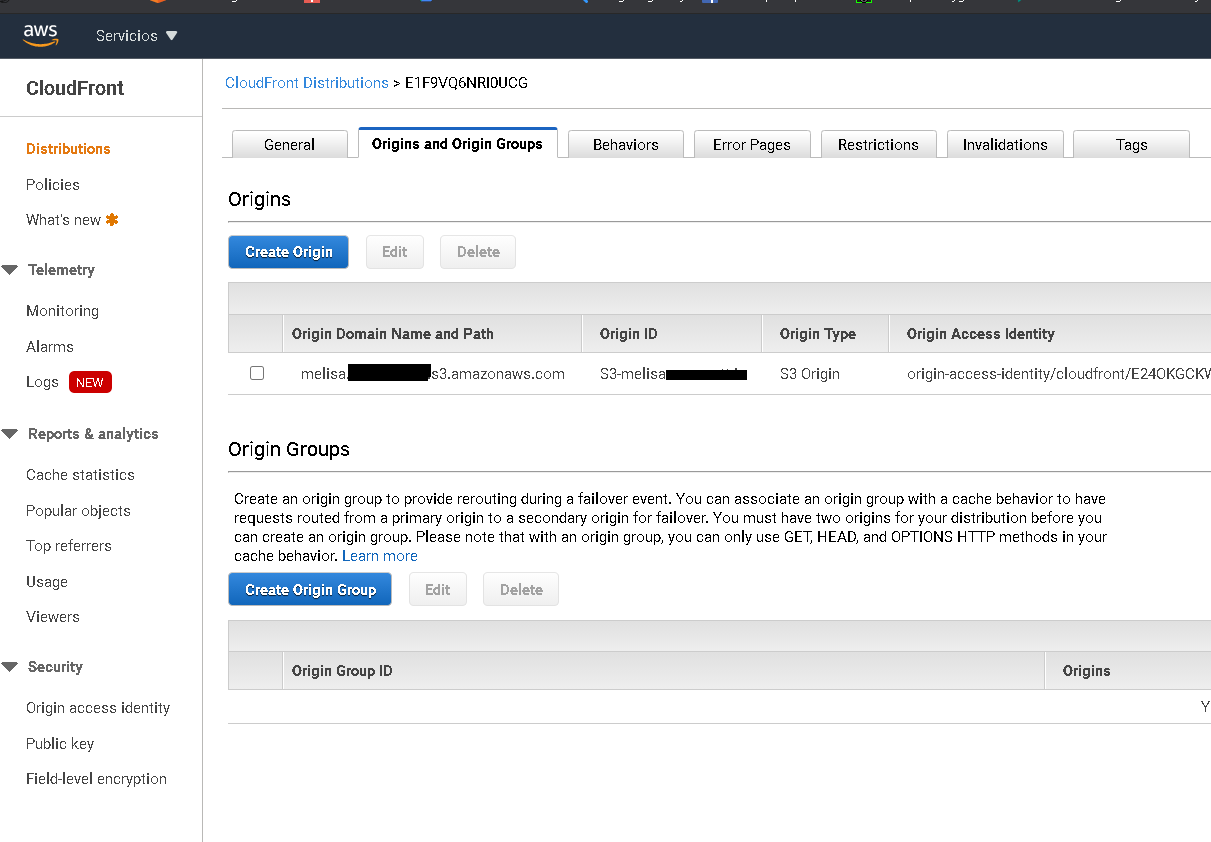
En la siguiente fase seleccionar entre grupos de seguridad, ello da control sobre las direcciones IP de origen y destino permitidas para el balanceador de carga 

Es necesario seleccionar el grupo de servidores hacia donde serán transferidas las peticiones. 

Seleccionaremos las instancias (se pueden seleccionar direcciones IP o funciones de AWS Lambda también dependiendo lo seleccionado en Target Type en el paso anterior) que atenderán las peticiones redirigidas por el balanceador. 

# Amazon CloudFront

Es el servicio para la entrega de contenido (Content Delivery Network) para distribución global de vídeo, datos, aplicaciones de baja latencia. Para poder entregar el contenido con baja latencia, Amazon cuenta con los llamados *edge locations*, cuando un usuario solicita un contenido es dirigido a un edge location que provee la latencia mas baja de forma que el contenido sea entregado con el mejor performance posible. Se cuenta con 200 puntos de entrega de contenido distribuidos en 34 países y 77 ciudades  Soporta todo el contenido que pueda ser provisto mediante HTTP o HTTPS, incluyendo HTML, JS, CSS, audio, video, media y binarios. CloudFront se integra con AWS Shield para mitigación de ataques DDoS, también con AWS WAF para mitigación de ataques SQL Injection, XSS entre otros, para brindar transporte de datos seguro con https se integra con AWS Certificate Manager. Al momento de configurar el servicio CloudFront se debe especificar el origen de los datos para replicar a la edge location, el origen puede ser un Bucket S3 configurado en modo web hosting, un balanceador de carga o una instancia EC2.

El siguiente ejemplo muestra una distribución con origen de datos desde un Bucket S3 configurado como web hosting. 

Los casos de uso destacados son: Servir a los usuarios finales un sitio web estático con posibilidad de uso de https y dominio propio, distribución de paquetes de software, en general es recomendado si los usuarios finales se encuentran distribuidos geográficamente por todo el mundo.