AWS

AWS Cloud Practitioner (CLF-C02)

Elastic Load Balancing, Auto Scaling Groups & Virtual Private Cloud

HARSHAL MORE
harshalmore.it@gmail.com
in https://www.linkedin.com/in/harshal-more-patil/

Elastic Load Balancing & Auto Scaling Groups

Scalability & High Availability

- Scalability means that an application / system can handle greater loads by adapting.
- There are two kinds of scalability:
  - Vertical Scalability
  - Horizontal Scalability (= elasticity)
- Scalability is linked but different to High Availability

Vertical Scalability

- Vertical Scalability means increasing the size of the instance
- For example, your application runs on a t2.micro
- Scaling that application vertically means running it on a t2.large
- Vertical scalability is very common for non distributed systems, such as a database.
- There's usually a limit to how much you can vertically scale (hardware limit)

Horizontal Scalability

- Horizontal Scalability means increasing the number of instances / systems for your application
- Horizontal scaling implies distributed systems.
- This is very common for web applications / modern applications
- It's easy to horizontally scale thanks the cloud offerings such as Amazon EC2

High Availability

- High Availability usually goes hand in hand with horizontal scaling
- High availability means running your application / system in at least 2 Availability Zones
- The goal of high availability is to survive a data center loss (disaster)

High Availability & Scalability For EC2

- **Vertical Scaling**: Increase instance size (= scale up / down)
  - From: t2.nano - 0.5G of RAM, 1 vCPU
  - To: u-12tb1.metal – 12.3 TB of RAM, 448 vCPUs

- **Horizontal Scaling**: Increase number of instances (= scale out / in)
  - Auto Scaling Group
  - Load Balancer

- **High Availability**: Run instances for the same application across multi AZ
  - Auto Scaling Group multi AZ
  - Load Balancer multi AZ

Scalability	Elasticity	Agility
ability to accommodate a larger load by making the hardware stronger (scale up), or by adding nodes (scale out)	once a system is scalable, elasticity means that there will be some "auto-scaling" so that the system can scale based on the load. This is "cloud-friendly": pay-per-use, match demand, optimize costs	(not related to scalability - distractor) new IT resources are only a click away, which means that you reduce the time to make those resources available to your developers from weeks to just minutes.

What is load balancing?

- Load balancers are servers that forward internet traffic to multiple servers (EC2 Instances) downstream.

Why use a load balancer?

- Spread load across multiple downstream instances
- Expose a single point of access (DNS) to your application
- Seamlessly handle failures of downstream instances
- Do regular health checks to your instances
- Provide SSL termination (HTTPS) for your websites
- High availability across zones

Why use an Elastic Load Balancer?

- An ELB (Elastic Load Balancer) is a managed load balancer
  - AWS guarantees that it will be working
  - AWS takes care of upgrades, maintenance, high availability
  - AWS provides only a few configuration knobs
- It costs less to setup your own load balancer but it will be a lot more effort on your end (maintenance, integrations)
- 3 kinds of load balancers offered by AWS:
  - Application Load Balancer (HTTP / HTTPS only) – Layer 7
  - Network Load Balancer (ultra-high performance, allows for TCP) – Layer 4
  - Classic Load Balancer (slowly retiring) – Layer 4 & 7

What's an Auto Scaling Group?

- In real-life, the load on your websites and application can change
- In the cloud, you can create and get rid of servers very quickly
- The goal of an Auto Scaling Group (ASG) is to:
  - Scale out (add EC2 instances) to match an increased load
  - Scale in (remove EC2 instances) to match a decreased load
  - Ensure we have a minimum and a maximum number of machines running
  - Automatically register new instances to a load balancer
  - Replace unhealthy instances
- Cost Savings: only run at an optimal capacity (principle of the cloud)

Auto Scaling Groups Scaling Strategies

- Manual Scaling: Update the size of an ASG manually
- Dynamic Scaling: Respond to changing demand
    - **Simple / Step Scaling**
        - When a CloudWatch alarm is triggered (example CPU > 70%), then add 2 units
        - When a CloudWatch alarm is triggered (example CPU < 30%), then remove 1
    - **Target Tracking Scaling**
        - Example: I want the average ASG CPU to stay at around 40%
    - **Scheduled Scaling**
        - Anticipate a scaling based on known usage patterns
        - Example: increase the min. capacity to 10 at 5 pm on Fridays
    - **Predictive Scaling**
        - Uses Machine Learning to predict future traffic ahead of time
        - Automatically provisions the right number of EC2 instances in advance
        - Useful when your load has predictable time - based patterns

ELB & ASG Summary

- High Availability vs Scalability (vertical and horizontal) vs Elasticity vs Agility in the Cloud
- Elastic Load Balancers (ELB)
    - Distribute traffic across backend EC2 instances, can be Multi-AZ
    - Supports health checks
    - 3 types: Application LB (HTTP – L7), Network LB (TCP – L4), Classic LB (old)
- Auto Scaling Groups (ASG)
    - Implement Elasticity for your application, across multiple AZ
    - Scale EC2 instances based on the demand on your system, replace unhealthy
    - Integrated with the ELB

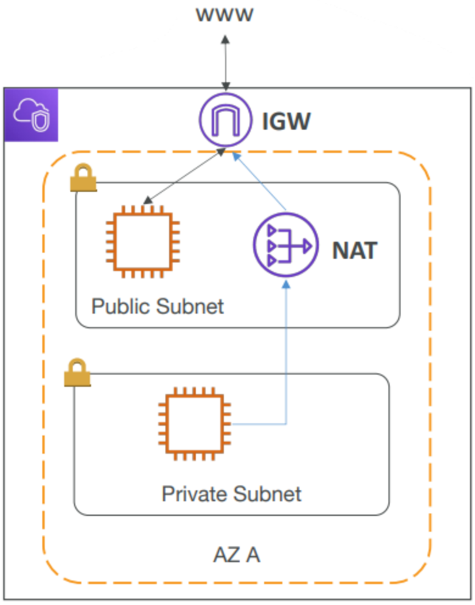
Virtual Private Cloud (VPC)

VPC & Subnets Primer

- VPC -Virtual Private Cloud: private network to deploy your resources (regional resource)
- Subnets allow you to partition your network inside your VPC (Availability Zone resource)
- A public subnet is a subnet that is accessible from the internet
- A private subnet is a subnet that is not accessible from the internet
- To define access to the internet and between subnets, we use Route Tables.

Internet Gateway & NAT Gateways

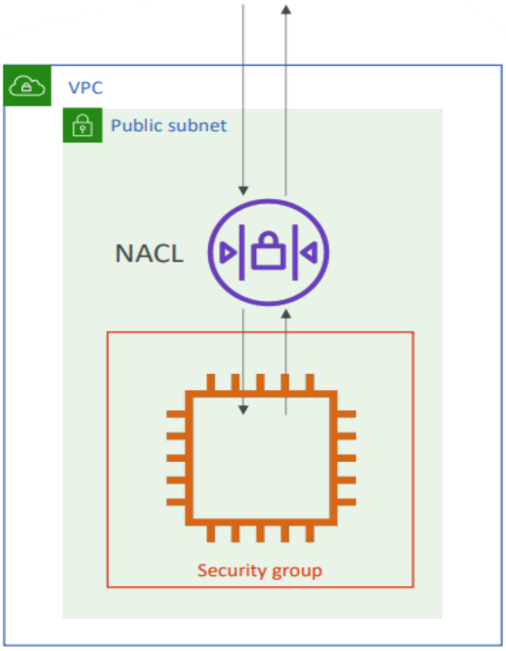
- Internet Gateways helps our VPC instances connect with the internet
- Public Subnets have a route to the internet gateway.
- NAT Gateways (AWS-managed) & NAT Instances (self-managed) allow your instances in your Private Subnets to access the internet while remaining private



Network ACL & Security Groups

- NACL (Network ACL)
  - A firewall which controls traffic from and to subnet
  - Can have ALLOW and DENY rules
  - Are attached at the Subnet level
  - Rules only include IP addresses

- Security Groups
  - A firewall that controls traffic to and from an ENI / an EC2 Instance
  - Can have only ALLOW rules
  - Rules include IP addresses and other security groups



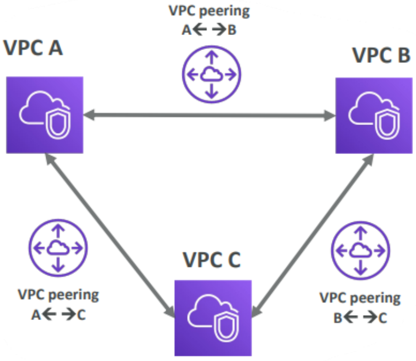
Network ACLs vs Security Groups

Feature	Network ACLs (NACLs)	Security Groups
Level of Operation	Operates at the subnet level	Operates at the instance level
Statefulness	Stateless: Responses to allowed inbound traffic must be explicitly allowed for outbound traffic	Stateful: Automatically allows responses to inbound traffic
Rules	Rules are evaluated in numerical order (lowest to highest)	All rules are evaluated before allowing or denying traffic
Allow/Deny	Can explicitly allow or deny traffic	Can only allow traffic, no explicit deny rules
Default Behavior	Allows all inbound and outbound traffic by default, unless otherwise specified	Denies all inbound traffic and allows all outbound traffic by default
Evaluation Process	Stateless: Each packet is checked against the rules list without context	Stateful: Once a connection is established, traffic is automatically allowed in both directions
Number of Rules	Limited to 20 rules per NACL	Up to 60 rules per security group (can be increased)
Association	Can be associated with multiple subnets; each subnet can only have one NACL	Associated with instances; an instance can have multiple security groups

https://docs.aws.amazon.com/vpc/latest/userguide/VPC_Security.html

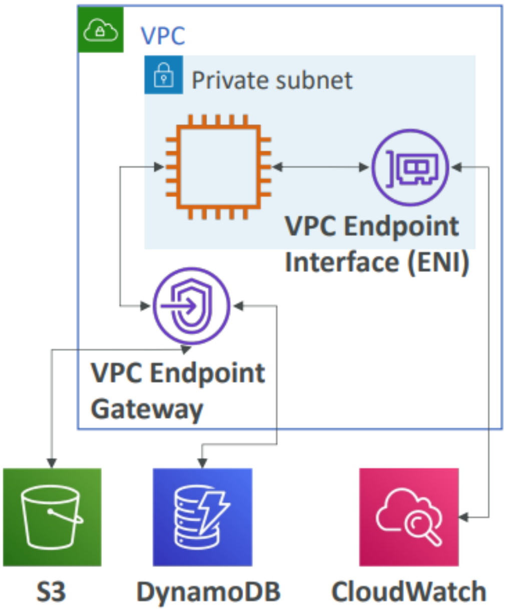
VPC Flow Logs

- Capture information about IP traffic going into your interfaces:
  - VPC Flow Logs
  - Subnet Flow Logs
  - Elastic Network Interface Flow Logs
- Helps to monitor & troubleshoot connectivity issues. Example:
  - Subnets to internet
  - Subnets to subnets
  - Internet to subnets
- Captures network information from AWS managed interfaces too: Elastic Load Balancers, ElastiCache, RDS, Aurora, etc…
- VPC Flow logs data can go to S3 / CloudWatch Logs



VPC Peering

- Connect two VPC, privately using AWS' network
- Make them behave as if they were in the same network
- Must not have overlapping CIDR (IP address range)
- VPC Peering connection is not transitive (Must be established for each VPC that need to communicate with one another)

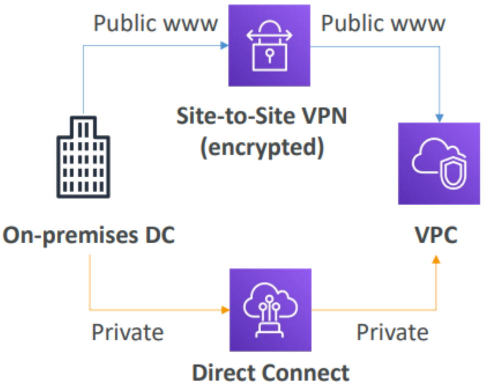


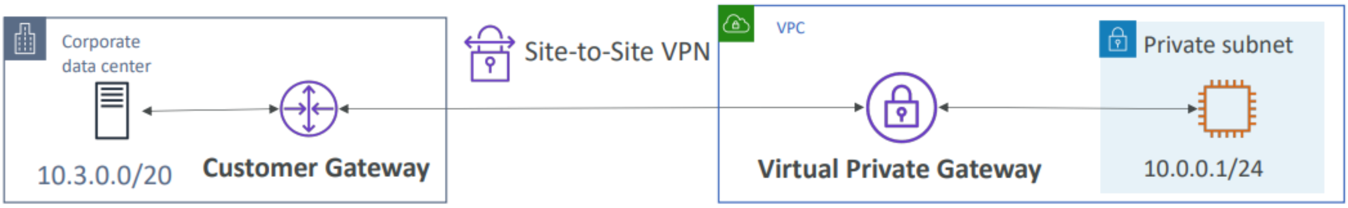
VPC Endpoints

- Endpoints allow you to connect to AWS Services using a private network instead of the public www network
- This gives you enhanced security and lower latency to access AWS services
- VPC Endpoint Gateway: S3 & DynamoDB
- VPC Endpoint Interface: the rest

Site to Site VPN & Direct Connect

- Site to Site VPN
    - Connect an on-premises VPN to AWS
    - The connection is automatically encrypted
    - Goes over the public internet
    - On-premises: must use a Customer Gateway (CGW)
    - AWS: must use a Virtual Private Gateway (VGW)
- Direct Connect (DX)
    - Establish a physical connection between on-premises and AWS
    - The connection is private, secure and fast
    - Goes over a private network
    - Takes at least a month to establish



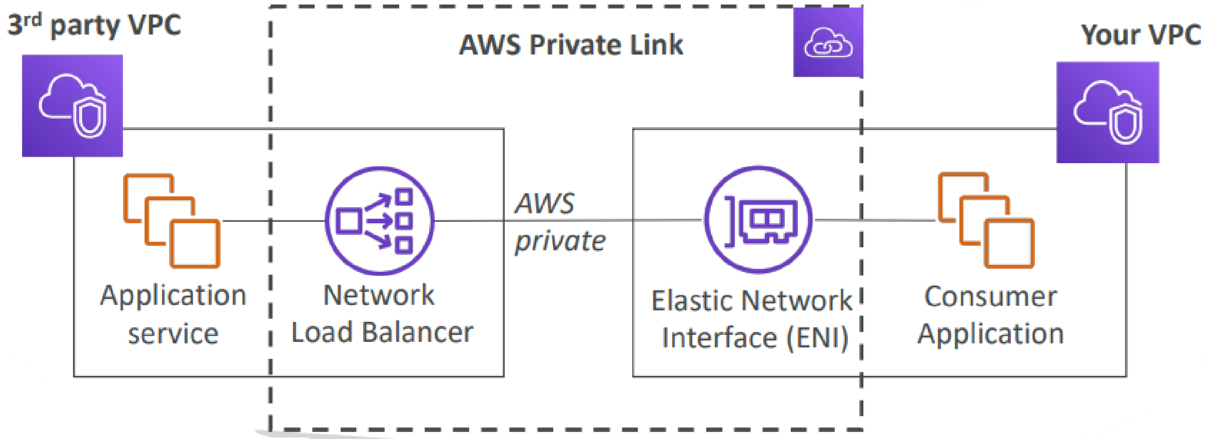


Transit Gateway

- For having transitive peering between thousands of VPC and on-premises, hub-and-spoke (star) connection
- One single Gateway to provide this functionality
- Works with Direct Connect Gateway, VPN connections

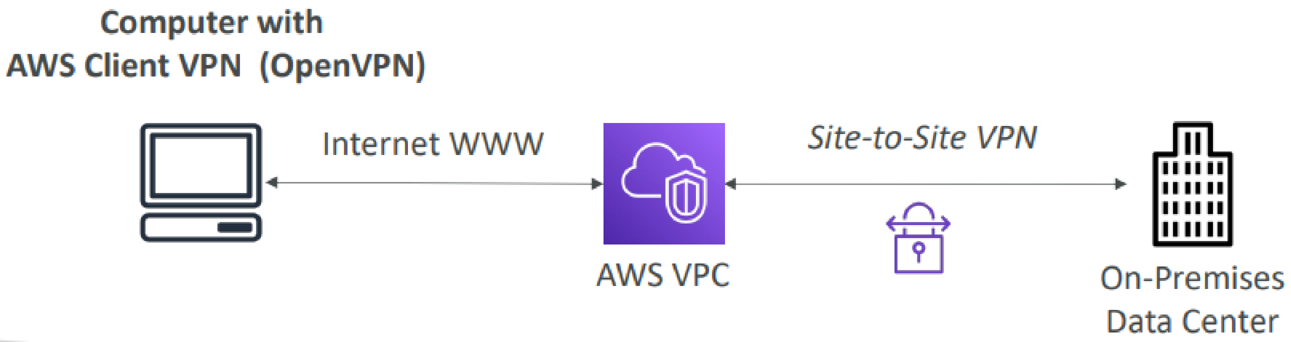
AWS PrivateLink (VPC Endpoint Services)

- Most secure & scalable way to expose a service to 1000s of VPCs
- Does not require VPC peering, internet gateway, NAT, route tables…
- Requires a network load balancer (Service VPC) and ENI (Customer VPC)



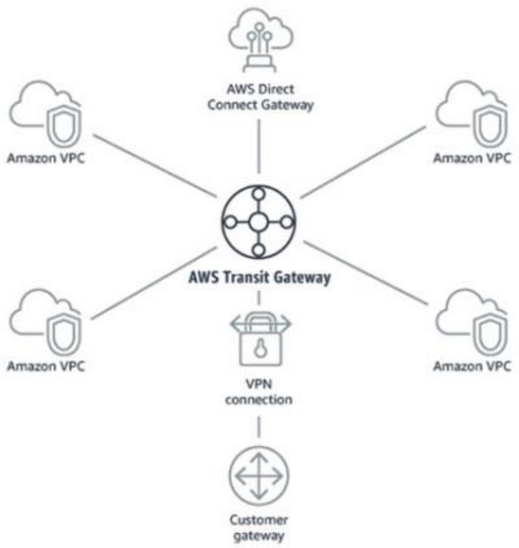
AWS Client VPN

- Connect from your computer using OpenVPN to your private network in AWS and on-premises
- Allow you to connect to your EC2 instances over a private IP (just as if you were in the private VPC network)
- Goes over public Internet



Transit Gateway

- For having transitive peering between thousands of VPC and on-premises, hub-and -spoke (star) connection
- One single Gateway to provide this functionality
- Works with Direct Connect Gateway, VPN connection



VPC Summary

- VPC: Virtual Private Cloud
- Subnets:Tied to an AZ, network partition of the VPC
- Internet Gateway: at the VPC level, provide Internet Access
- NAT Gateway / Instances: give internet access to private subnets
- NACL: Stateless, subnet rules for inbound and outbound
- Security Groups: Stateful, operate at the EC2 instance level or ENI
- VPC Peering: Connect two VPC with non overlapping IP ranges, nontransitive
- VPC Endpoints: Provide private access to AWS Services within VPC
- VPC Flow Logs: network traffic logs
- Site to Site VPN: VPN over public internet between on-premises DC and AWS
- Direct Connect: direct private connection to AWS
- Transit Gateway: Connect thousands of VPC and on-premises networks together