

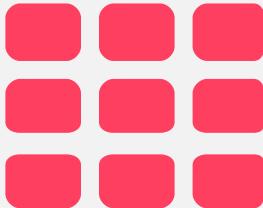
This Material is NOT for Copying or Distribution without
the prior written consent of DolfinED™

©

This document and its contents is the sole property of DolfinED© and is protected by the federal law and international treaties. This is solely intended to be used by DolfinED© students enrolled into the DolfinED's AWS Certified Solutions Course, it is not for anyone else be it a user, business, or any other commercial or non-commercial entity. You are strictly prohibited from making a copy or modification of, or from or distributing this document without the prior written permission from DolfinED© public relations, except as may be permitted by law.

Not for copy, modification or Redistribution –
Please report any breach to info@dolfined.com



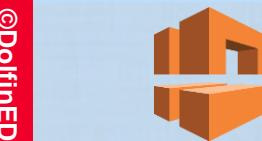


AWS VIRTUAL PRIVATE CLOUD (VPC)



YOU CAN DO IT TOO!

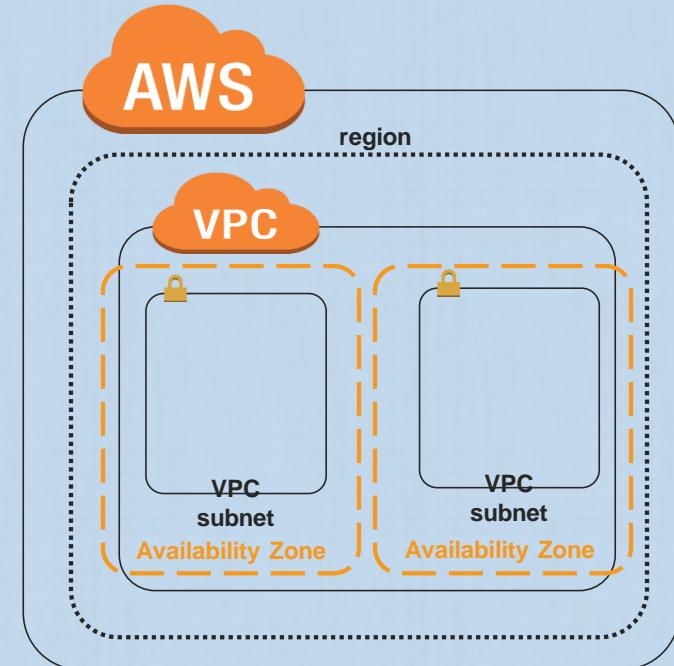




AWS Virtual Private Cloud (VPC)

What is it?

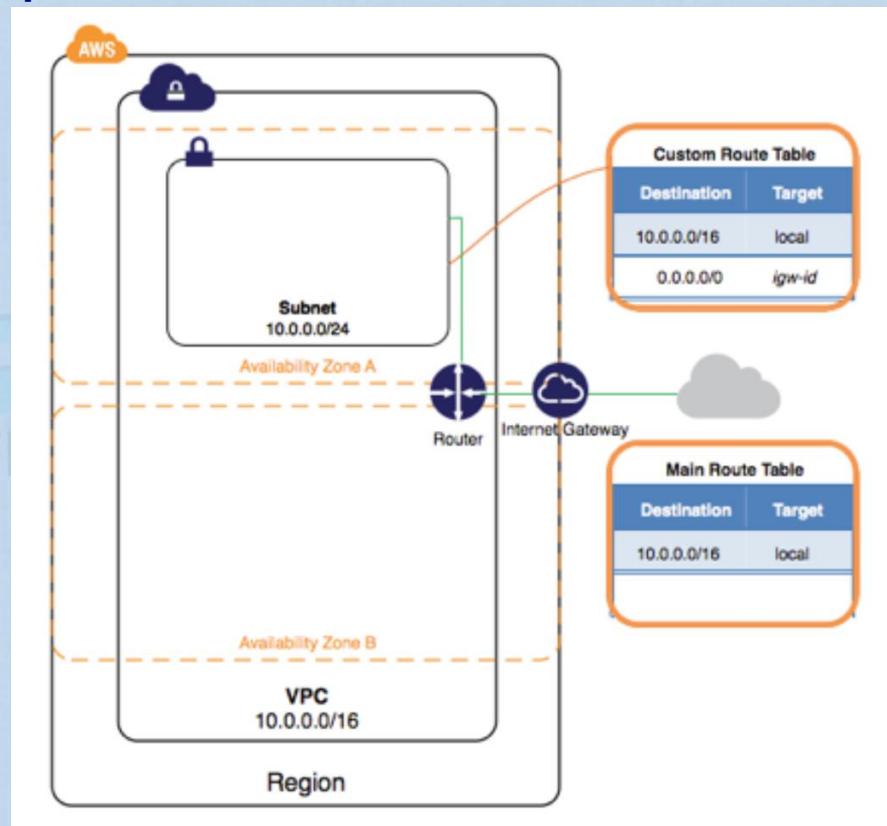
- Is a virtual network or data center inside AWS for one client, or a department in an enterprise
- The AWS client has full control over resources & virtual compute instances (virtual servers) hosted inside that VPC
- Is similar to having your own data center inside AWS
- Logically isolated from other VPCs on AWS
- You can have one or more IP address subnets inside a VPC
- A VPC is confined to an AWS region and does not extend between regions



AWS Virtual Private Cloud (VPC)

VPC Components

- CIDR and IP address subnets
- Implied Router
- Route tables
- Internet gateway
- Security Groups
- Network Access Control Lists (N. ACLs)
- Virtual Private Gateway (VGW)



AWS Virtual Private Cloud (VPC)

VPC Types

- A Default VPC
 - Created in each AWS region when an AWS account is created
 - Has default CIDR, Security Group, N ACL, and route table Settings
 - Has an Internet Gateway by default
- A Custom (non-default) VPC
 - Is a VPC an AWS account owner creates
 - AWS user creating the custom VPC can decide the VPC CIDR block
 - Has its own default security group, N ACL, and Route tables
 - Does not have an Internet Gateway by default, one needs to be created if needed



Implied Router / Route Tables



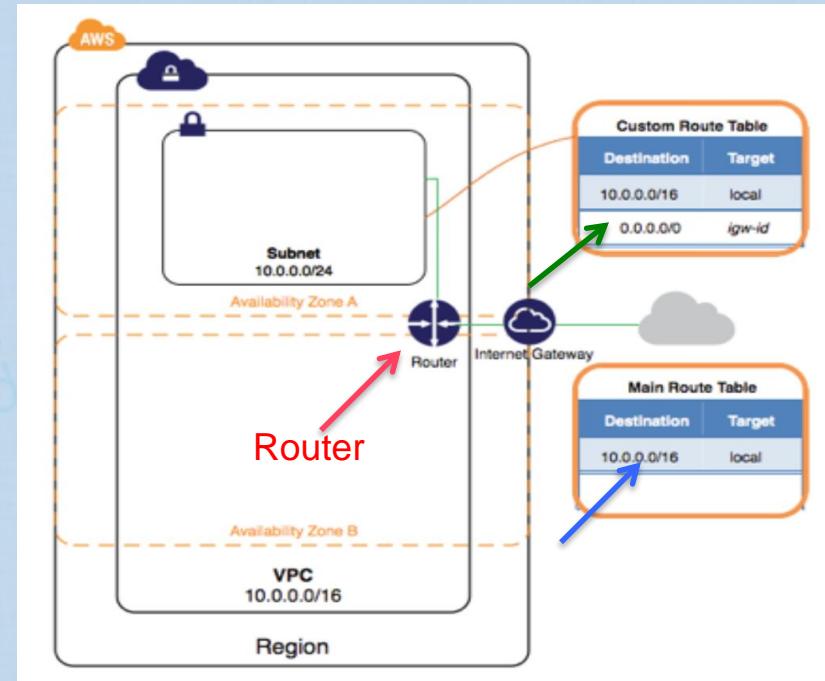
Solutions Architect - Associate



AWS Virtual Private Cloud (VPC)

Implied Router

- It is the central VPC routing function,
- It connects the different AZ's together and connects the VPC to the Internet Gateway (and Virtual Private Gateway when configured)
- Each subnet will have a route table that the router uses to forward traffic within the VPC
- The route tables can also have entries to external destinations



AWS Virtual Private Cloud (VPC)

Route Tables

- Each Subnet MUST be associated with only one route table at any given time
- If you do not specify a subnet-to-route-table association, the subnet (when created) will be associated with the main (default) VPC route table
- Please reference the AWS website for updated Quotas for the different AWS services
- https://docs.aws.amazon.com/general/latest/gr/aws_service_limits.html
- For VPC -> <https://docs.aws.amazon.com/general/latest/gr/vpc-service.html>



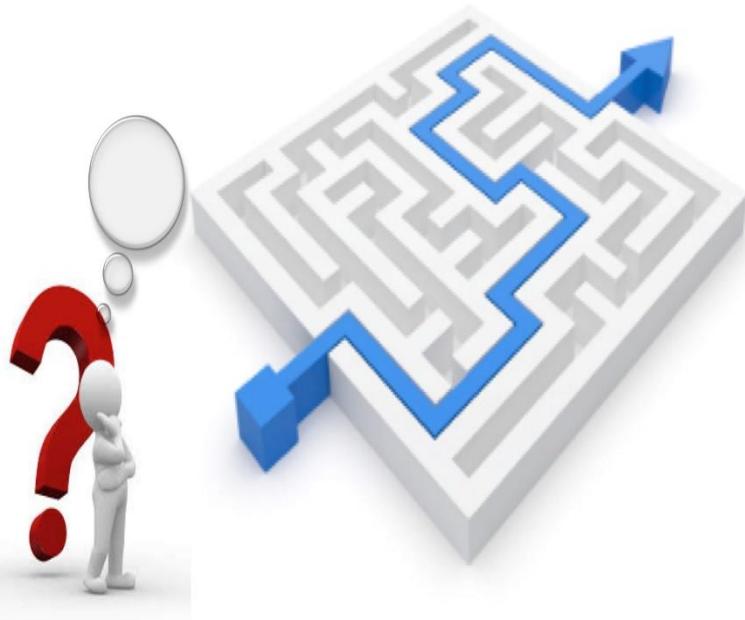
AWS Virtual Private Cloud (VPC)

Route Tables

- You can change the subnet association to another route table when/as needed
- You can also edit the main (default) route table if you need, but you can NOT delete the Main (default) route table
 - However, you can make a custom route table manually become the main route table, then you can delete the former main, as it is no longer a main route table
- Every route table in a VPC comes with a default rule that allows all VPC subnets to communicate with one another
 - You can NOT modify or delete this rule



VPC IP Addressing



AWS Virtual Private Cloud (VPC)

VPC IP Addressing

- The CIDR block is the range of IP addresses that you choose for the VPC when you create it
- Once the VPC is created, you can NOT change its main CIDR block range
 - But you can expand the VPC CIDR block by adding additional CIDR blocks
 - Some restrictions apply
- If you need a different main CIDR block range, create a new VPC
- The different subnets within a VPC can NOT overlap (basic TCP/IP rule)



AWS Virtual Private Cloud (VPC)

AWS Reserved IP's in each subnet

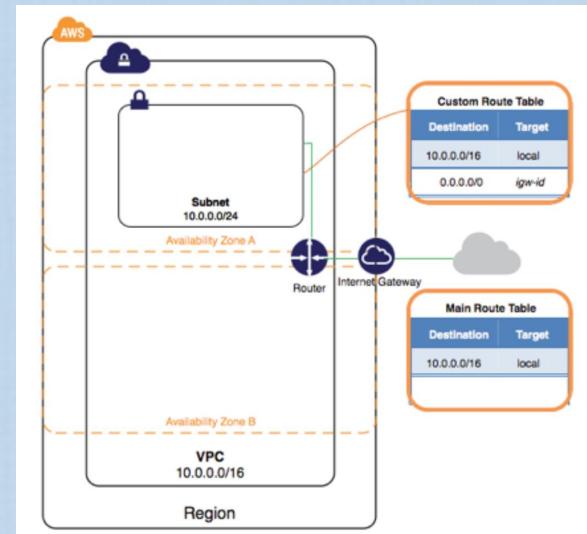
- First 4 IP addresses in each subnet and the last one are reserved by AWS
 - Ex. If the subnet is 10.0.0.0/24
 - 10.0.0.0 is the base network
 - 10.0.0.1 VPC router
 - 10.0.0.2 DNS related
 - 10.0.0.3 Reserved for future use
 - 10.0.0.255 last IP



AWS Virtual Private Cloud (VPC)

Internet Gateway

- Is the gateway through which your VPC communicates with the internet, and with other AWS services
- Is a horizontally scaled, redundant, and highly available VPC component
- It performs NAT (static one-to-one) between your Private IPv4 addresses in your VPC and the allocated Public (or Elastic) IPv4 addresses
- It supports both IPv4 and IPv6
- You can not SSH or connect to it, it is fully managed by AWS



AWS Virtual Private Cloud (VPC)

Public Subnet vs. Private Subnet

- Public Subnet means:
 - Its VPC has an Internet gateway attached to it
 - It is associated with a route table that has an entry for a default route pointing at the VPC's Internet gateway
 - Destination 0.0.0.0/0 Target: igw-1234
- Any subnet that does not satisfy either or both of the conditions above is considered a private subnet by AWS definition
 - Private subnet means, it is not accessible from the Internet since it has no Public Internet IP addresses configured.



AWS Virtual Private Cloud (VPC)

Elastic IP addresses

- Elastic IPs are Internet routable IP addresses that you can have allocated to your VPC, and will continue being allocated to your VPC until you decide to release them back to AWS
- Some AWS services (example NAT gateway) require an Elastic IP address to function
- You have 5 Elastic IP addresses per region (Soft limit that you can change by contacting AWS)
- Public IPv4 addresses on the other hand, are DHCP based (dynamically allocated) to your Compute, and are released back to AWS if you stop your compute instance.



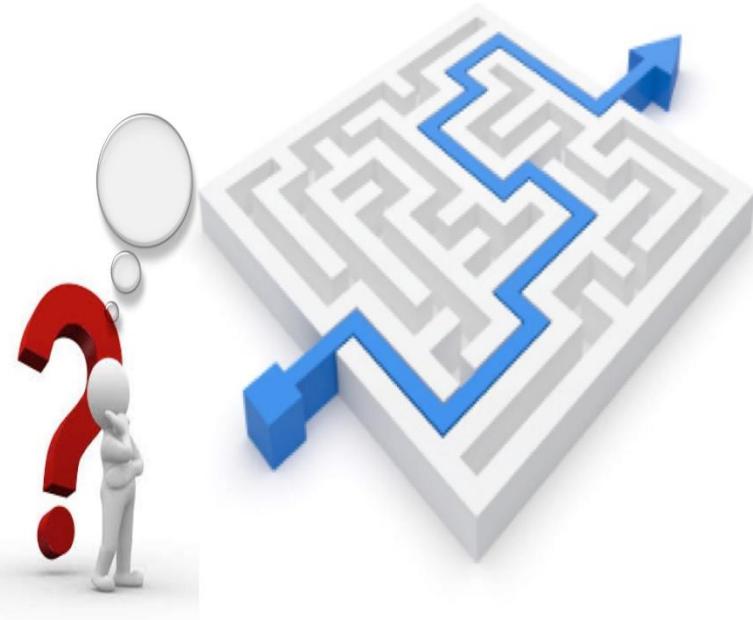
AWS Virtual Private Cloud (VPC)

Elastic IP addresses – AWS Charges

- You are not charged for a used Elastic IP, but if you are not using it, you will be charged.
- If you have attached more than one Elastic IP address to a running EC2 instance, you get charged for all except one (the one you are entitled to use for free)



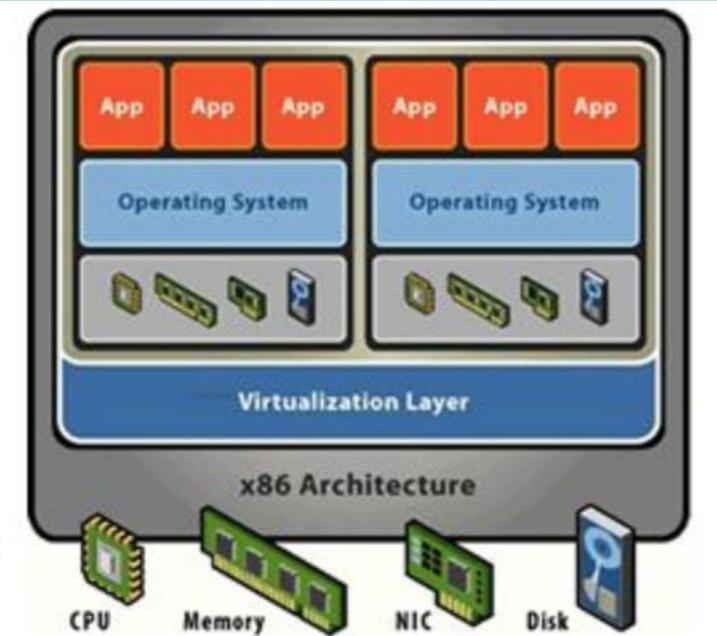
Security Groups



AWS Virtual Private Cloud (VPC)

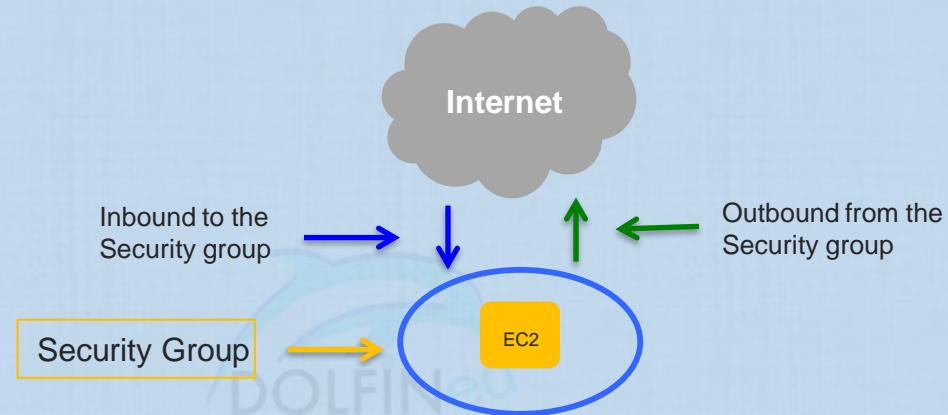
Security Groups

- A security group is a virtual firewall
- It controls traffic at the virtual server (EC2 Instance) level
 - Specifically at the virtual Network Interface level
- Up to 16 (5 is the default) security groups per EC2 instance interface can be applied
- **Stateful**, return traffic, of allowed inbound traffic, is allowed, even if there are no rules to allow it
- **Can only** have permit rules, **can NOT** have deny rules
- Implicit deny rule at the end
- Security Groups are associated with EC2 instances' elastic network interfaces (ENIs)
- All rules are evaluated to find a permit rule



AWS Virtual Private Cloud (VPC)

Security Groups



- You can use Security Group names as the source or destination in other security group rules
- You can use the security group name as the source in its own inbound security group rules
- Security groups are directional and can use allow rules only
- A security group set of rules ends with an implicit deny any



AWS Virtual Private Cloud (VPC)

Security Groups (Cont.)

- Any Virtual Server Instance(EC2) created without specifying a security group for it (during its creation), will be assigned the VPC default security group
- Each VPC created will have a default Security Group created for it, you can NOT delete a default Security group
- Security groups are VPC resources, hence, different EC2 instances, in different Availability Zones, belonging to the same VPC, can have the same security group applied to them
- **Changes to security groups take effect immediately**



AWS Virtual Private Cloud (VPC)

Default and non-Default Security Groups

- A default security group
 - Is the one created by AWS when the default VPC is created, or when you create your own Custom VPC and it will have (by default)
 - Inbound rules allowing Instances assigned the same security group to talk to one another
 - All outbound traffic is allowed
- A Custom (non-default) security group
 - Is the one you create under a default or non-default VPC, and by default it will have
 - No inbound rules – basically all inbound traffic is denied by default
 - All outbound traffic is allowed by default

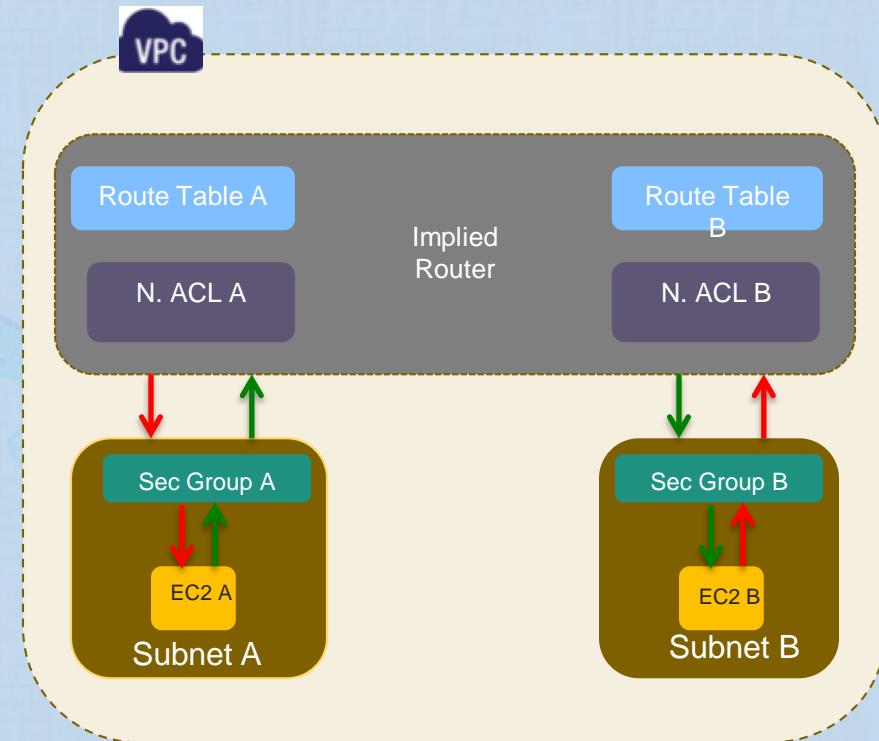
VPC Network ACLs



AWS Virtual Private Cloud (VPC)

Network Access Control Lists (N.ACLs)

- It is a function performed on the implied router (The implied VPC router hosts the Network ACL function)
- It functions at the Subnet Level
- N. ACLs are “Stateless”. Outbound traffic for an allowed inbound traffic, must be “explicitly” allowed too
- You can have “permit” and “deny” rules in a NACL
- NACL is a set of rules, each has a number



AWS Virtual Private Cloud (VPC)

Network Access Control Lists (NACLs) (Cont.)

- NACL rules are checked for a “permit” from lower numbered rules until either a permit is found, or an explicit/implicit deny is reached
- You can insert rules (based on the configured rule number spacing) between existing rules, hence, it is recommended to leave a number range between any two rules to allow for edits later.
- N. ACLs end with an **explicit** deny any, which you can NOT delete
- A subnet must be associated with a N. ACL, if you do not specify the N. ACL, the subnet will get associated with the default N. ACL automatically



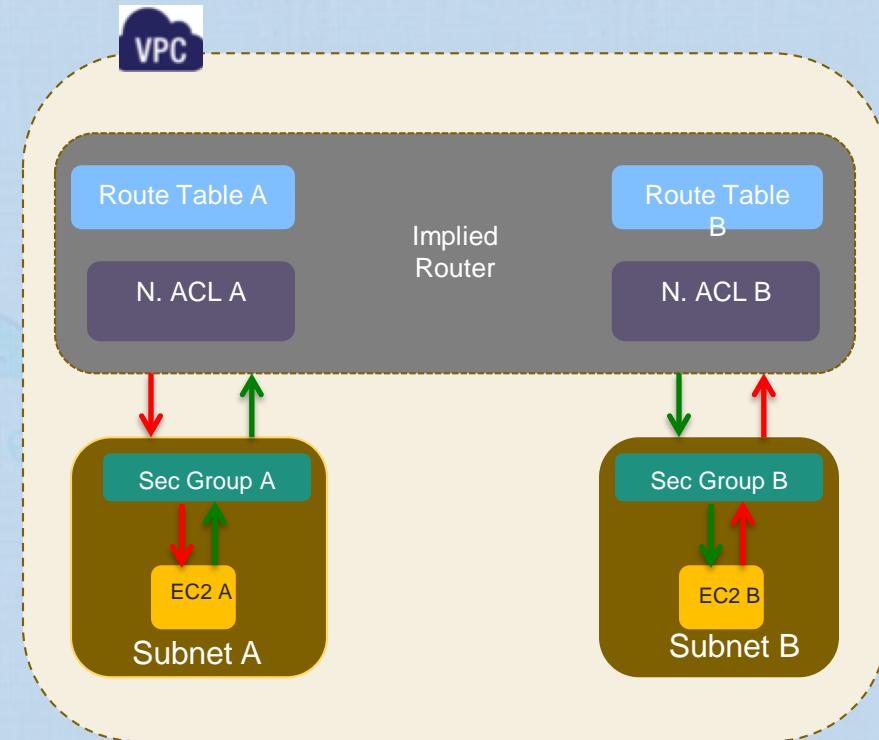
AWS Virtual Private Cloud (VPC) Network Access Control Lists (NACLs) (Cont.)

- You can create your own custom N. ACL, you do not have to use the default one
- A default N.ACL allows all traffic inbound and outbound
- A custom (non-default) N. ACL blocks/denies all traffic inbound and outbound by default.

Review Topic : NACLs

Network Control Access Lists (NACLs)

- For NACLs:
 - Inbound in NACL means coming from outside the subnet destined to the subnet.
 - Outbound means going out of the subnet.
- For Security Groups
 - Inbound for security group means inbound from outside the instance destined to the instance.
 - Outbound means going out of the instance's ENI.



Review Topic : VPC Security

NACLs vs. Security Groups

Security Group	Network ACL
Operates at the instance level (first layer of defense)	Operates at the subnet level (second layer of defense)
Supports allow rules only	Supports allow rules and deny rules
Is stateful: Return traffic is automatically allowed, regardless of any rules	Is stateless: Return traffic must be explicitly allowed by rules
We evaluate all rules before deciding whether to allow traffic	We process rules in number order when deciding whether to allow traffic
Applies to an instance only if someone specifies the security group when launching the instance, or associates the security group with the instance later on	Automatically applies to all instances in the subnets it's associated with (backup layer of defense, so you don't have to rely on someone specifying the security group)



Source: aws.amazon.com

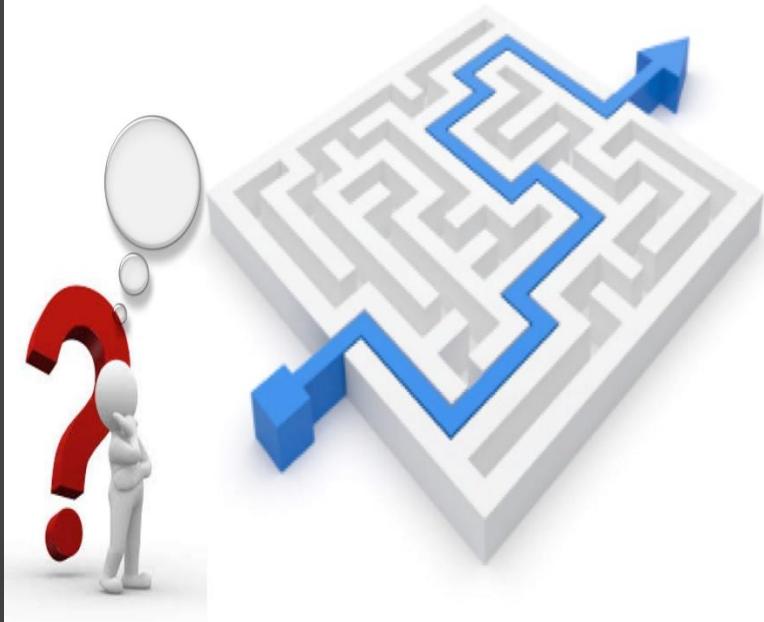
Review Topic : NACLs & Security Groups

Communication among Subnets

- Within a VPC, every single routing table created, or there by default, has an entry to allow traffic to be forwarded anywhere within the VPC without restriction from a routing perspective (not security)
 - This rule can NOT be edited, nor can it be deleted
- If you are facing any issues regarding communication between EC2 instances in a VPC, always look for the security setting of security groups and N ACLs relevant to the communication path (Source instance on which subnet and to Destination instance on which subnet).



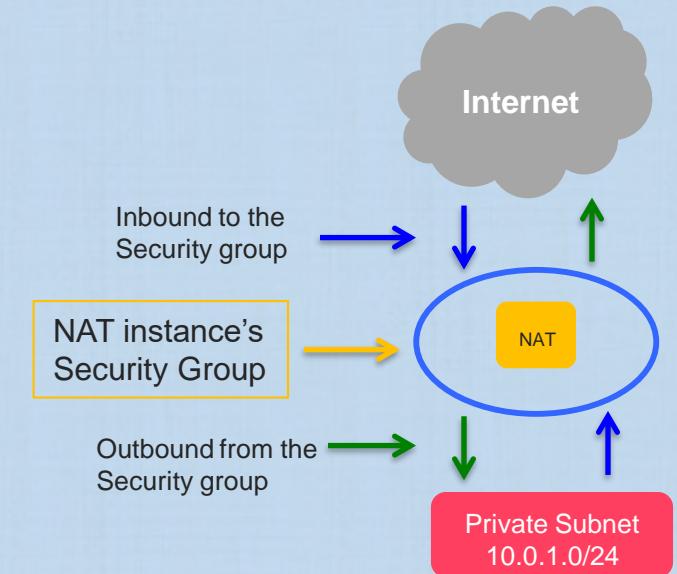
NAT instance and Security Groups



Review Topic : Security Groups

NAT Instance - Security Group Configurations

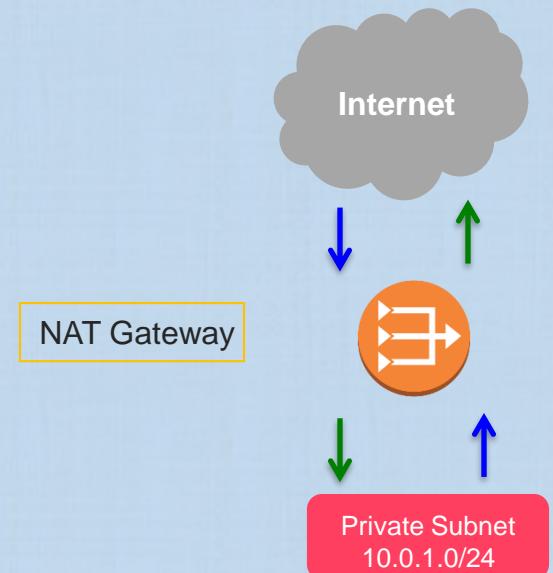
- NAT instance is required to enable the private subnet EC2 Instances to get to the internet
 - Hence, the NAT instance MUST be configured in a **public subnet**
 - EC2 instances with Public/Elastic IP addresses do not need to go through NAT instances to access the Internet
- NAT instance need to be assigned a security group
- No traffic initiated from the Internet can access the private subnet through the NAT instance
 - Only responses to traffic initiated from the private subnets are allowed through NAT instances
- Only admin SSH traffic can be allowed to the NAT instance (or RDP if Windows)



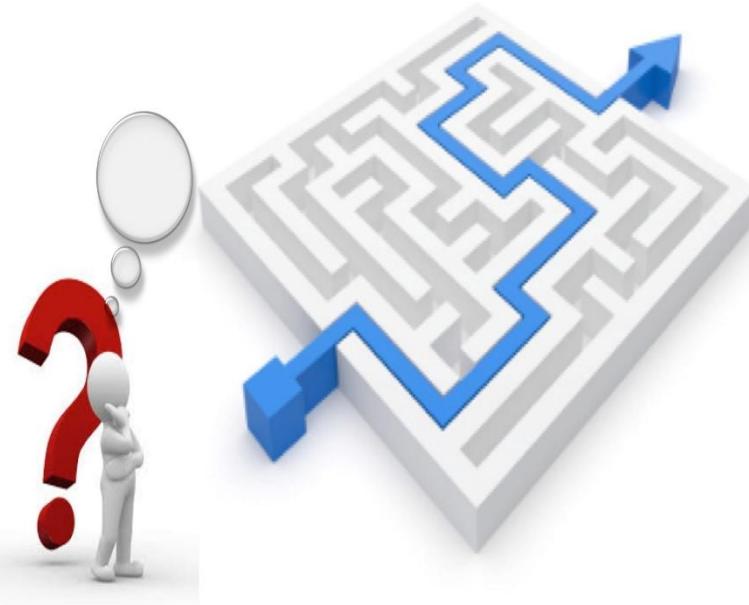
Review Topic : VPC

NAT Gateway

- Is an AWS managed service (Highly available, redundant..etc)
 - Customer does not need to worry about patching or OS updates
- Can not be assigned a security group
- AWS is responsible for its security/patching...etc
- Can scale to 10s of Gbps throughput
- Works only with an Elastic IP, can Not use a Public IP to do its function
 - NAT instances can work with Public and Elastic IP addresses



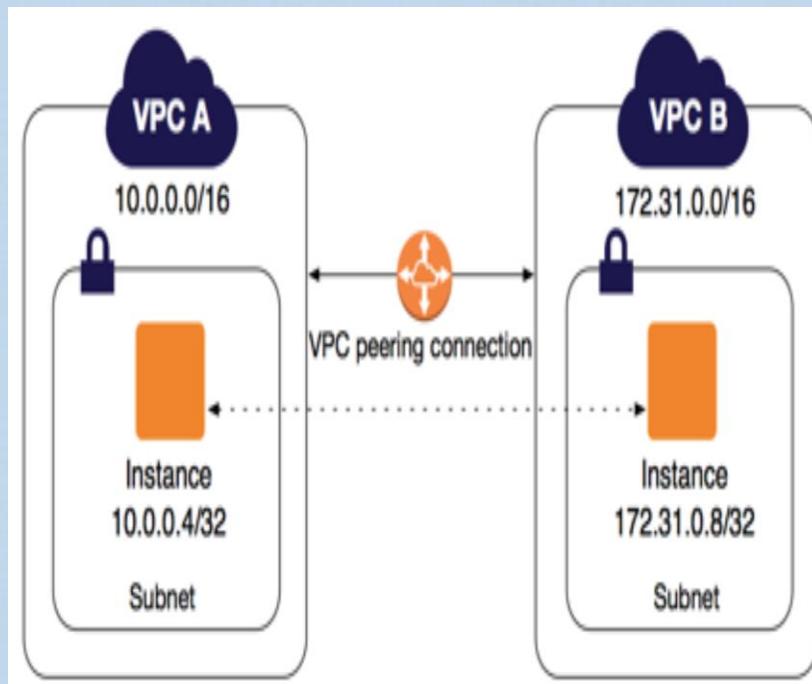
VPC Peering



Review Topic : VPC

VPC Peering

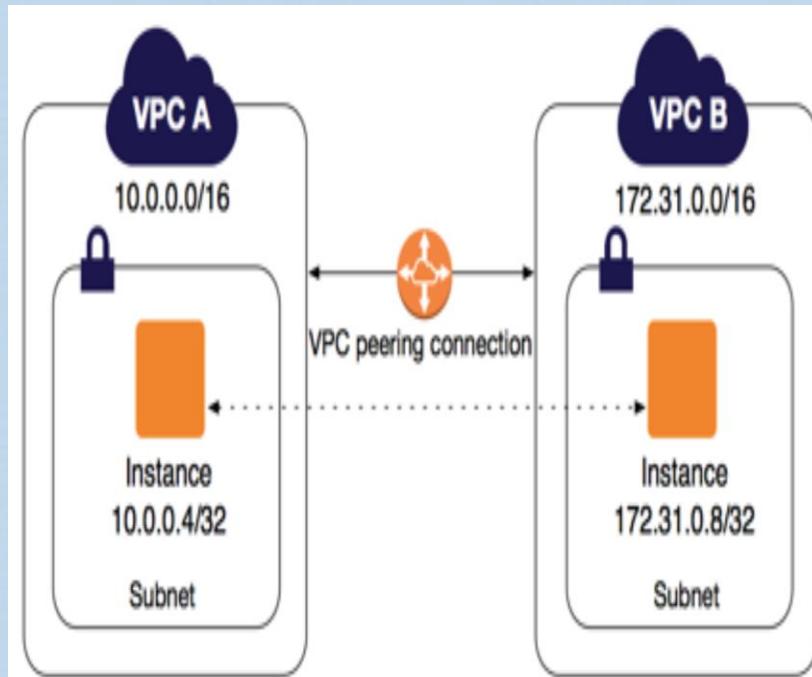
- By default a VPC can not communicate with any other VPC through Private IPv4 addresses, even if the other VPC belongs to the same AWS account.
- To allow two VPCs to communicate you need to configure a VPC Peering connection
- A VPC peering connection is a highly available networking connection between two VPCs that enables routing traffic between them using private IPv4 addresses or IPv6 addresses.
- Instances in either VPC can communicate with each other as if they are within the same network.



Review Topic : VPC

VPC Peering

- You can create a VPC peering connection between your own VPCs, or with a VPC in another AWS account within the same region, or between AWS regions.
- There is no single point of failure in the VPC Peering connection, so you need not worry about creating two VPC Peering connections for redundancy or high availability.
- For the connection to be established:
 - There MUST not be any overlapping IP ranges between the two VPCs
 - Request from a VPC has to be initiated
 - An accept from the other VPC has to be done
 - Routing and Security Groups/NACLs has to be updated to allow traffic both ways

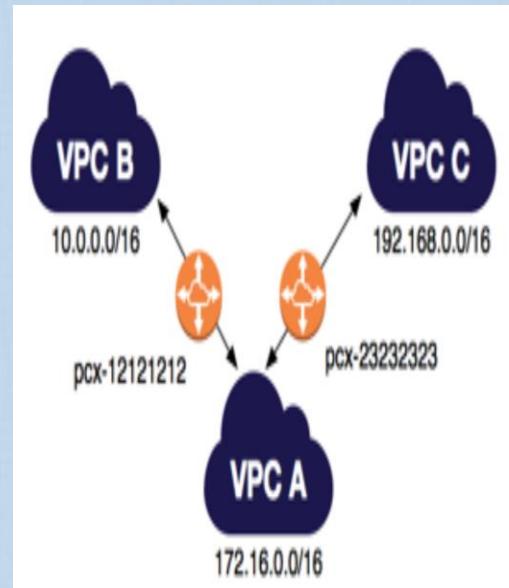


Source: aws.amazon.com

Review Topic : VPC

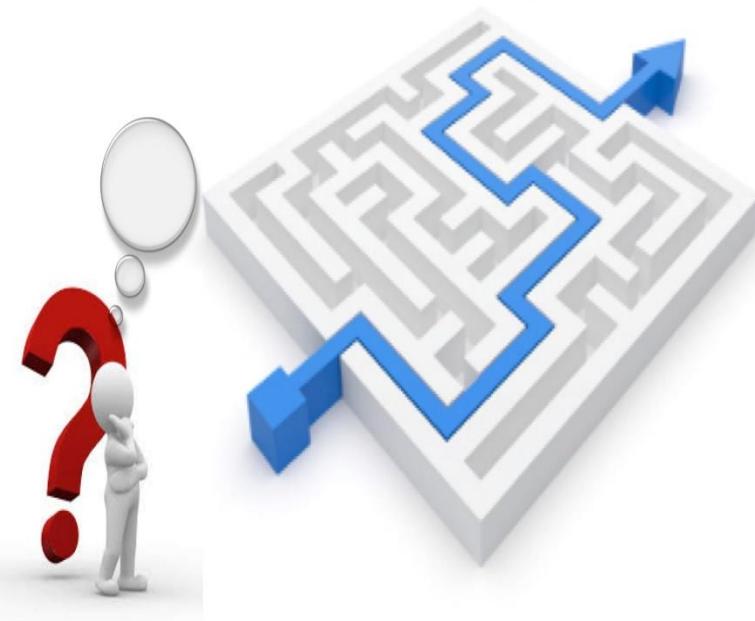
VPC Peering

- You can reference a security group from the peer VPC as a source or destination for ingress or egress rules in your security group rules.
- A VPC peering connection is a one to one relationship between two VPCs. You can create multiple VPC peering connections for each VPC that you own
- Transitive peering relationships are not supported:
 - You do not have any peering relationship with VPCs that your VPC is not directly peered with.
- You cannot have more than one VPC peering connection between the same two VPCs at the same time.



Source: aws.amazon.com
DOLFINED

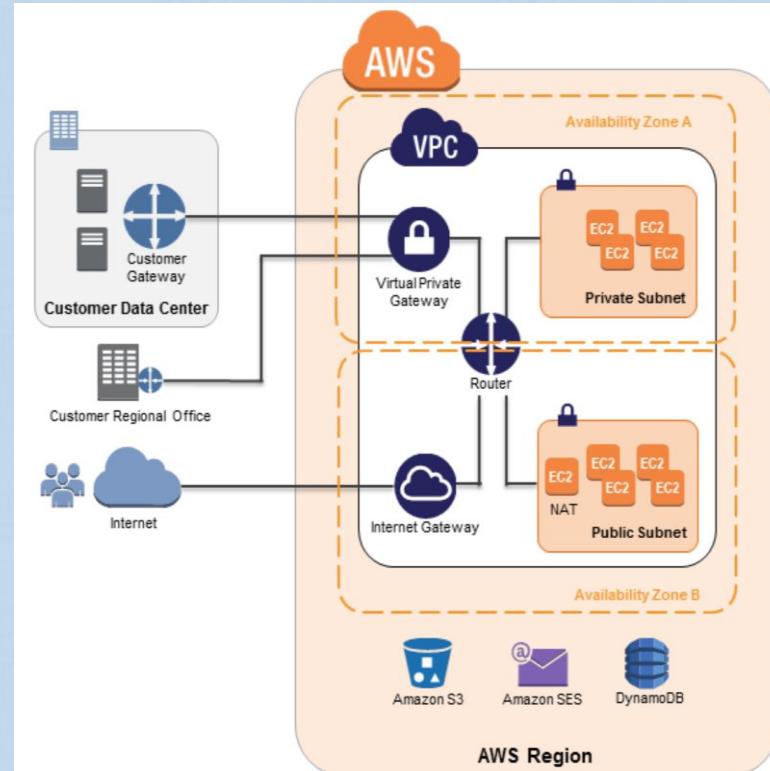
Virtual Private Networks (VPN)



Review Topic : VPC

Virtual Private Networks (VPN)

- A secure connection over the Internet or Direct Connect between On-Premise and AWS
- VPN connections are quick, easy to deploy, and cost effective
- A VGW is required on the VPC side, and a Customer gateway on the client's data center (locations) side
- An Internet routable IP address is required on your Customer gateway
- Two tunnels are configured for each VPN connection for redundancy
- You can NOT use the NAT gateway in your VPC through the VPN connection



Source: aws.amazon.com

AWS Virtual Private Cloud (VPC)

Enabling Dynamic Route Propagation

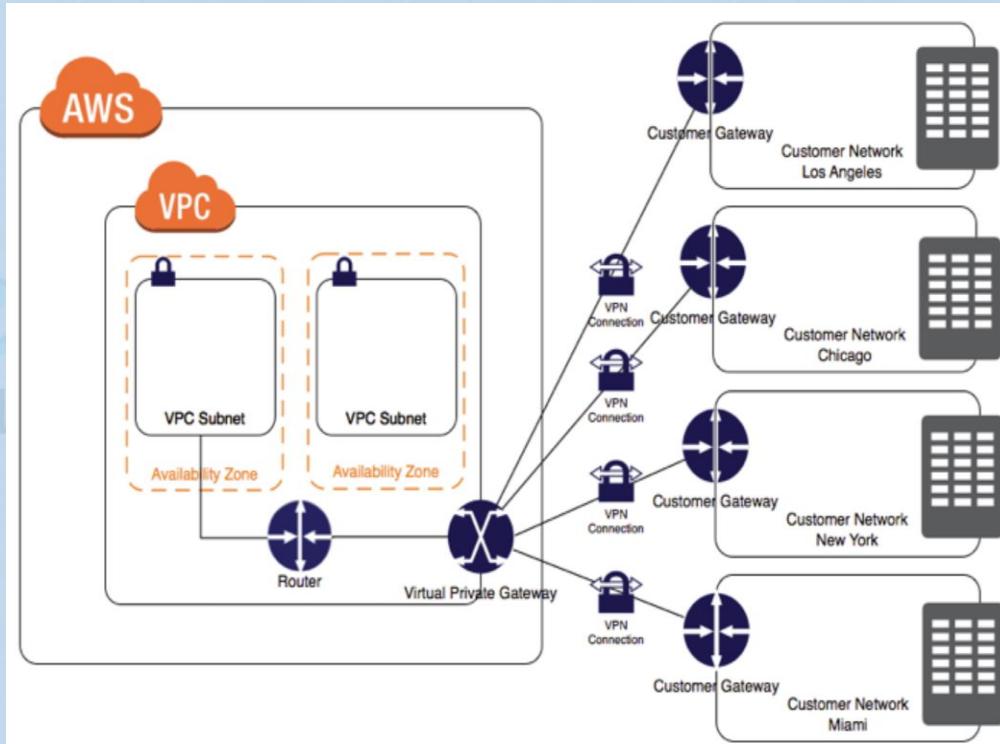
- To allow the VPC subnet(s) to communicate with the on premise subnets, you need to update the route table(s) of the subnet(s) in the VPC to point to the VGW
- Alternatively, you can enable route propagation in these route tables such that, routes the VGW learns over the VPN connection, are dynamically propagated to the route table pointing at the VGW as the next hop (Target)
 - Less manual tasks



AWS Virtual Private Cloud (VPC)

AWS VPN CloudHub

- You can **have up to 10 IPSec** connections per VGW (soft limit can be increased by contacting AWS)
- VPN based Hub and Spoke connectivity to a common VGW
- Can mix DX connections (explained next) with VPN connections
- Spokes can communicate with each other and with the VPC



Source: aws.amazon.com

AWS Virtual Private Cloud (VPC)

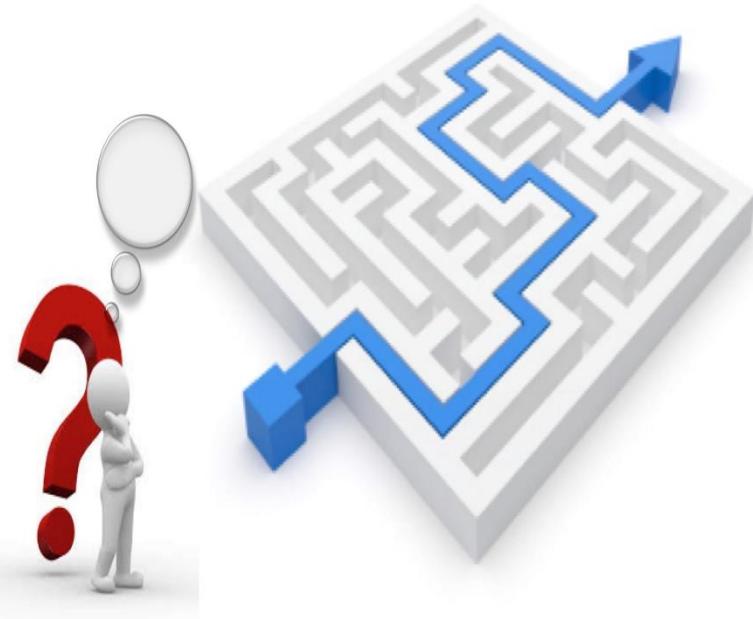
VPN – Allowed IP Prefixes

Which IP prefixes can receive/send traffic through the VPN connection?

- Only IP prefixes that are known to the virtual private gateway,
- VGW learns about these prefixes through Static or BGP routing
- VGW does not route any other traffic destined outside of the received BGP advertisements, static route entries, or its attached VPC CIDR
- You can NOT access Elastic IPs on your VPC side using the VPN tunnel established, Elastic IPs in AWS can only be accessed from the Internet
- You can NOT use a NAT gateway in your VPC over the VPN connection

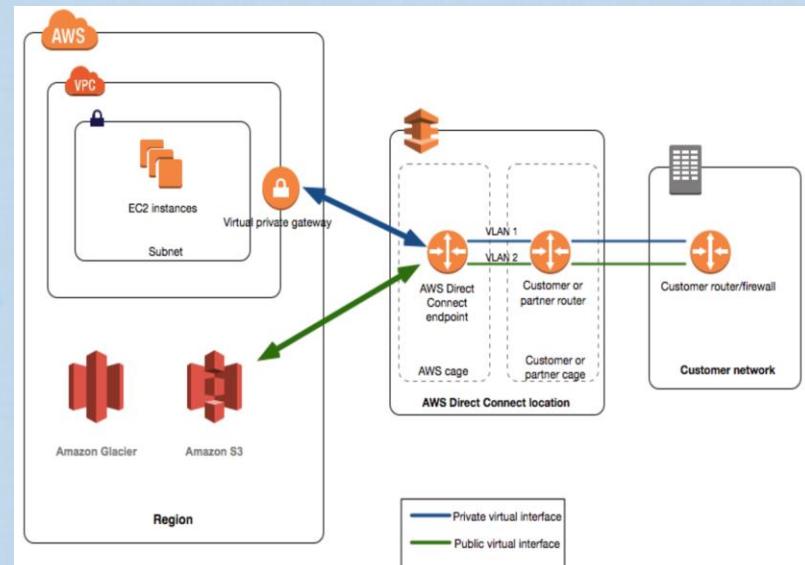


VPC Direct Connect (DX)



Review Topic : VPC AWS Direct Connect (DX)

- It is a direct connection (not internet based) and provides for higher speeds (bandwidth), less latency and higher performance than Internet
- A Virtual Interface (VIF) is basically a 802.1Q VLAN mapped from the customer router to the Direct Connect router
- You need one private VIF to connect to your private VPC subnets, and one public VIF to connect your AWS public services
- You can NOT establish layer 2 over your DX connection
- You can NOT use a NAT instance/gateway in your VPC over the direct connect connection

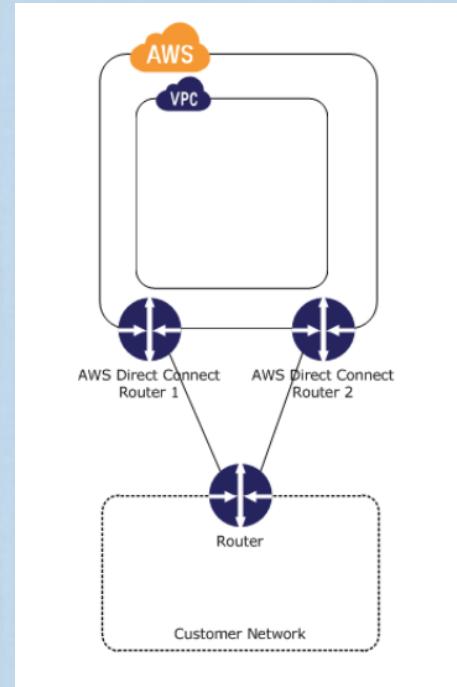


Source: aws.amazon.com

Review Topic : VPC

AWS Direct Connect

- High Availability with DX
 - You can have two dedicated links from the customer location to two DX routers, these routers will then connect through AWS infrastructure to your VPC VGW which also can be redundant hardware/VGW end points on AWS side.
- Over these two links you have a choice (via EBGP manipulation) of:
 - Active / Active (using eBGP multipath)
 - Active / Passive (using AS prepend failover)

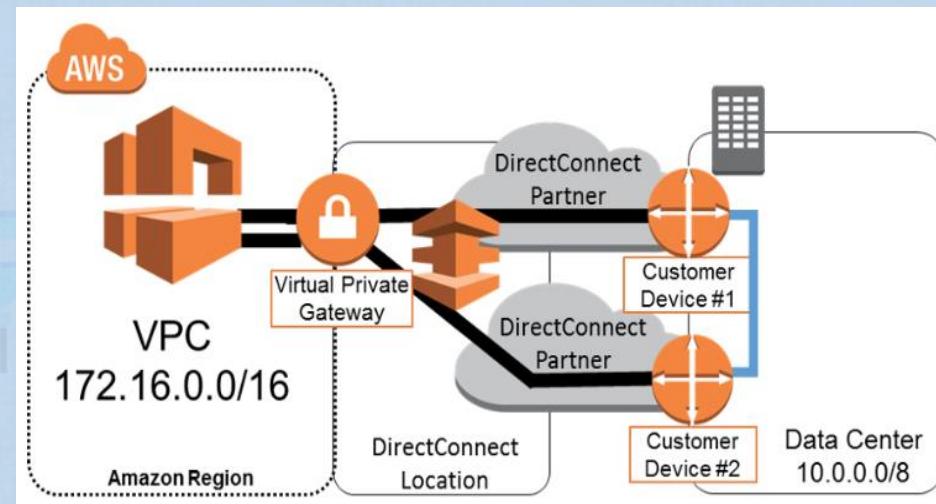


Source: aws.amazon.com

Review Topic : VPC

AWS Direct Connect – High Availability and Fault Tolerance

- Highest in terms of availability
- Two direct connect connections from two different providers
- Two Customer routers
- Two Direct connect routers
- Two AWS Direct Connect Locations
- eBGP routing and possibility of active/active or active/failover

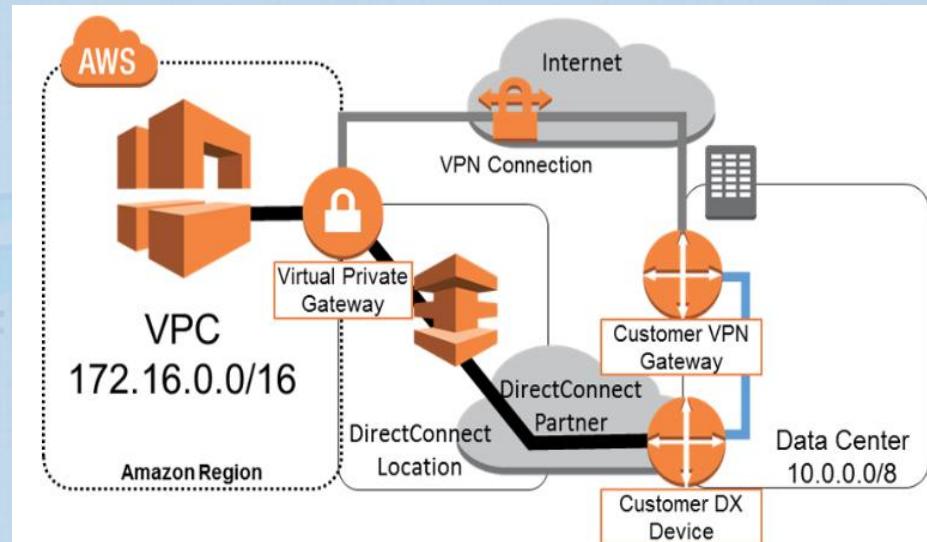


Source: aws.amazon.com

Review Topic : VPC

AWS Direct Connect – High Availability and Fault Tolerance

- One DX connection and a backup VPN connection
- Two Customer routers
- Primary is DX connection, fallback is VPN
- VPN is a cheaper backup connection

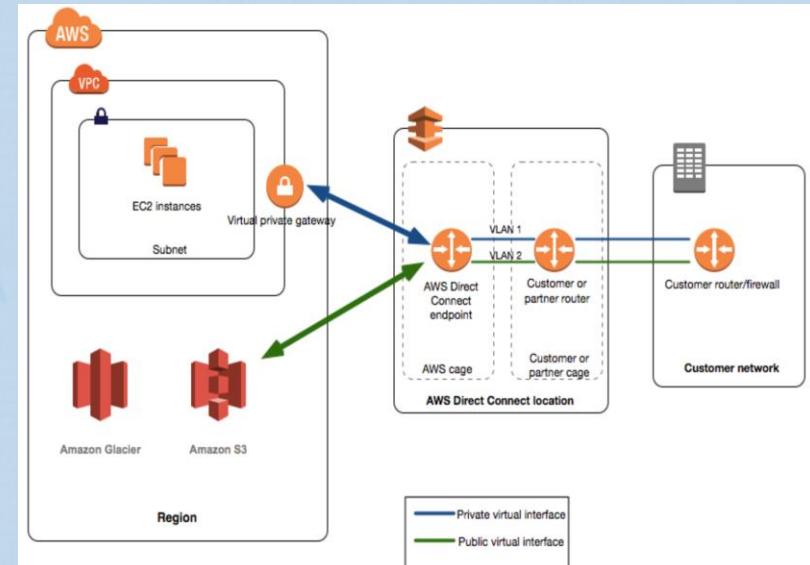


Source: aws.amazon.com

Review Topic : VPC

AWS Direct Connect (DX)

- Once connected via DX, you can access all availability zones in a region
 - And you can establish IPSec VPN tunnels over the Public VIF to connect to VPCs in remote regions as well

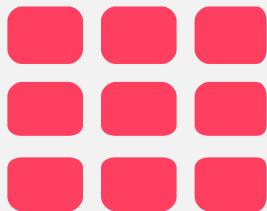


Source: aws.amazon.com

Review Topic : VPC Direct Connect

- You can only have one default route (0.0.0.0/0) per routing table
- Route propagation can be used to have the VGW send the Customer side routes to the respective VPC private subnets' route tables



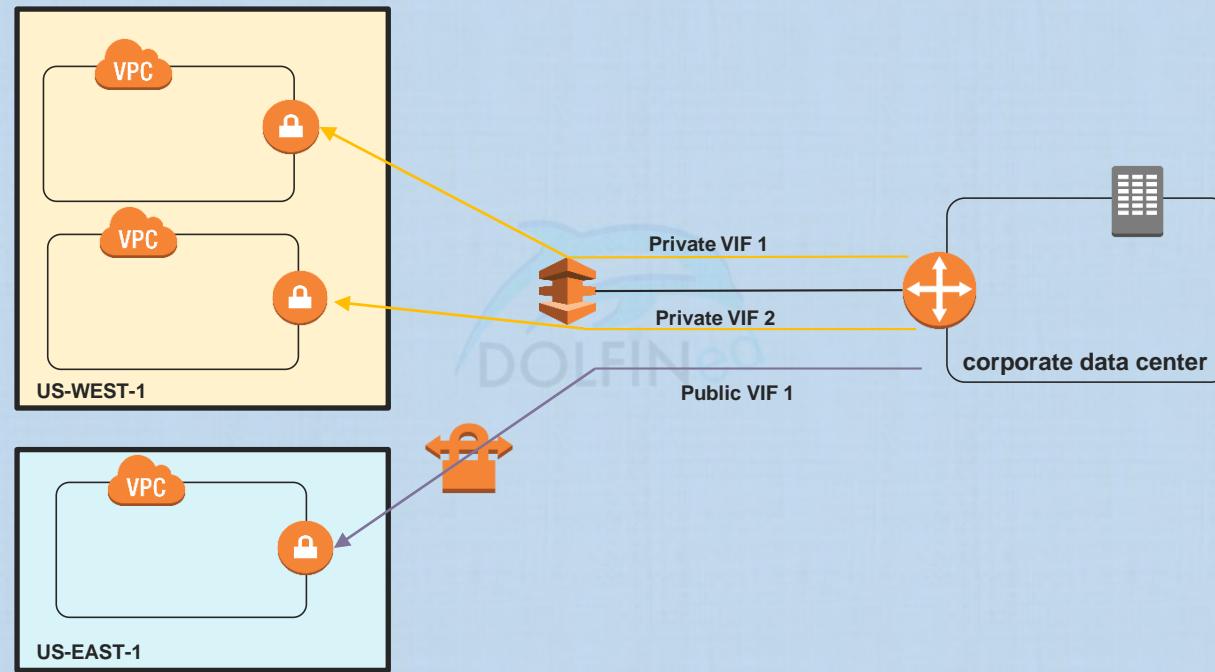


AWS DIRECT CONNECT GATEWAY

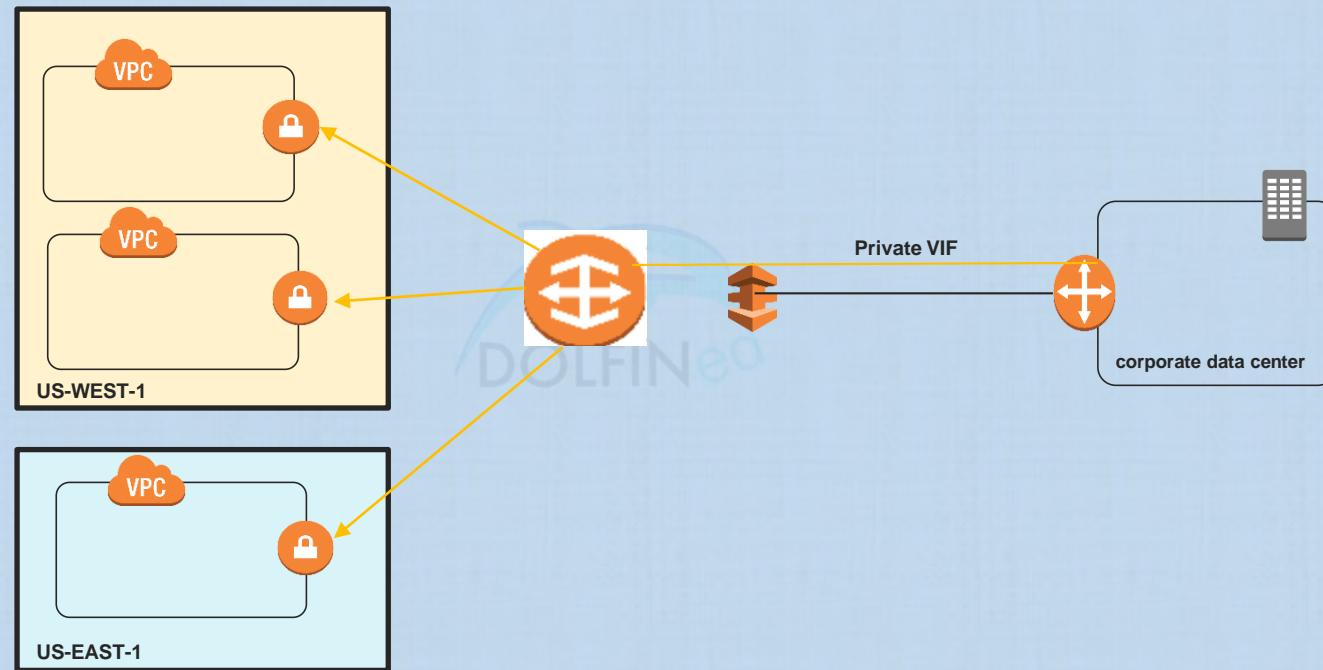
YOU CAN DO IT TOO!



AWS Direct Connect Gateway – Why do we need it?



AWS Direct Connect Gateway – Why do we need it?



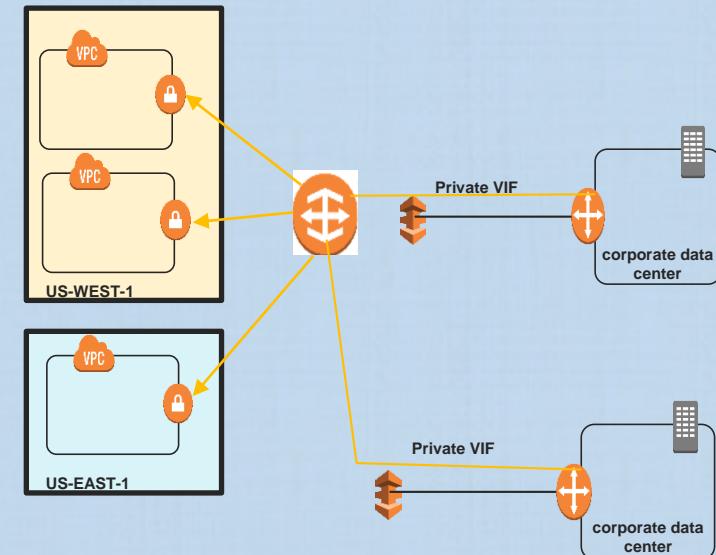
AWS Direct Connect Gateway

- A Direct Connect gateway is a globally available resource.
 - You can create the Direct Connect gateway in any public Region and access it from all other public Regions.
- Use AWS Direct Connect gateway to connect your VPCs.
- You associate an AWS Direct Connect gateway with either of the following gateways:
 - A virtual private gateway(s) of the VPC(s) you need to connect to
 - A transit gateway when you have multiple VPCs in the same Region
- Create a private virtual interface (Private VIF) for your AWS Direct Connect connection to the Direct Connect gateway.
 - You can attach multiple private virtual interfaces to your Direct Connect gateway (from different locations or from different DX links connected to different Customer Gateways).
- You can associate a VGW from one account with a Direct Connect Gateway from another account, by creating association proposal from the Direct Connect Gateway in one account to the VGW in the other account.



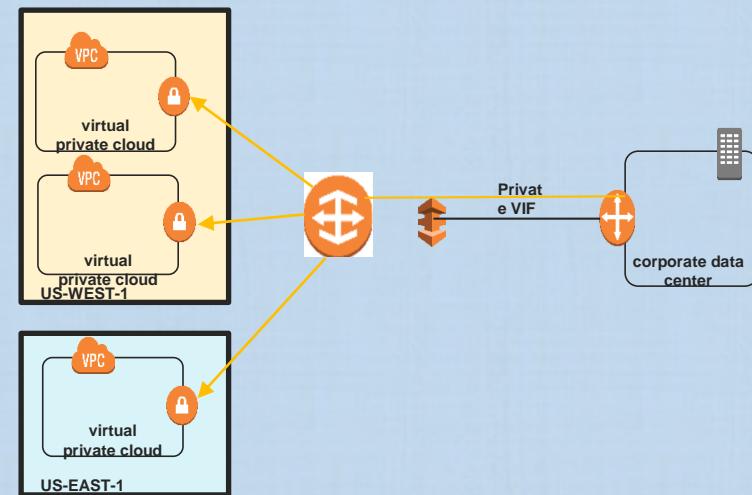
AWS Direct Connect Gateway - Limitations

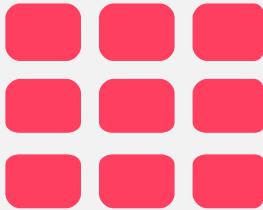
- The VPCs to which you connect through a Direct Connect gateway cannot have overlapping CIDR blocks.
 - If you add an IPv4 CIDR block to a VPC that's associated with a Direct Connect gateway, ensure that the CIDR block does not overlap with an existing CIDR block for any other associated VPC.
- You cannot create a public virtual interface to a Direct Connect gateway.
- A Direct Connect gateway supports communication between attached private virtual interfaces and associated virtual private gateways only. The following traffic flows are not supported:
 - Direct communication between the VPCs that are associated with the Direct Connect gateway.
 - Direct communication between the virtual interfaces that are attached to the Direct Connect gateway.
 - Direct communication between a virtual interface attached to a Direct Connect gateway and a VPN connection on a virtual private gateway that's associated with the same Direct Connect gateway.



AWS Direct Connect Gateway - Limitations

- You cannot associate a virtual private gateway with more than one Direct Connect gateway and you cannot attach a private virtual interface to more than one Direct Connect gateway.
- A virtual private gateway that you associate with a Direct Connect gateway must be attached to a VPC





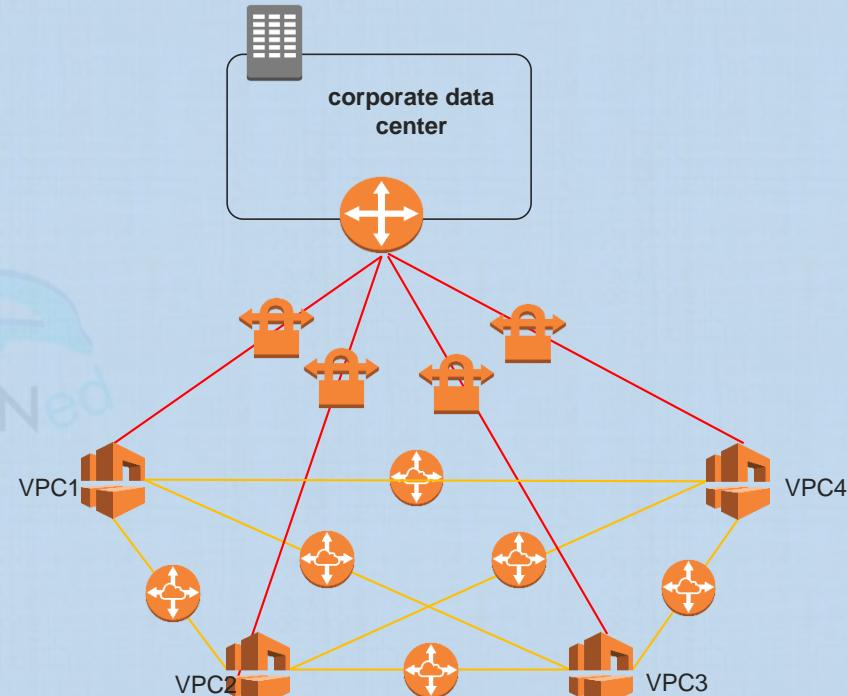
AWS TRANSIT GATEWAY

YOU CAN DO IT TOO!



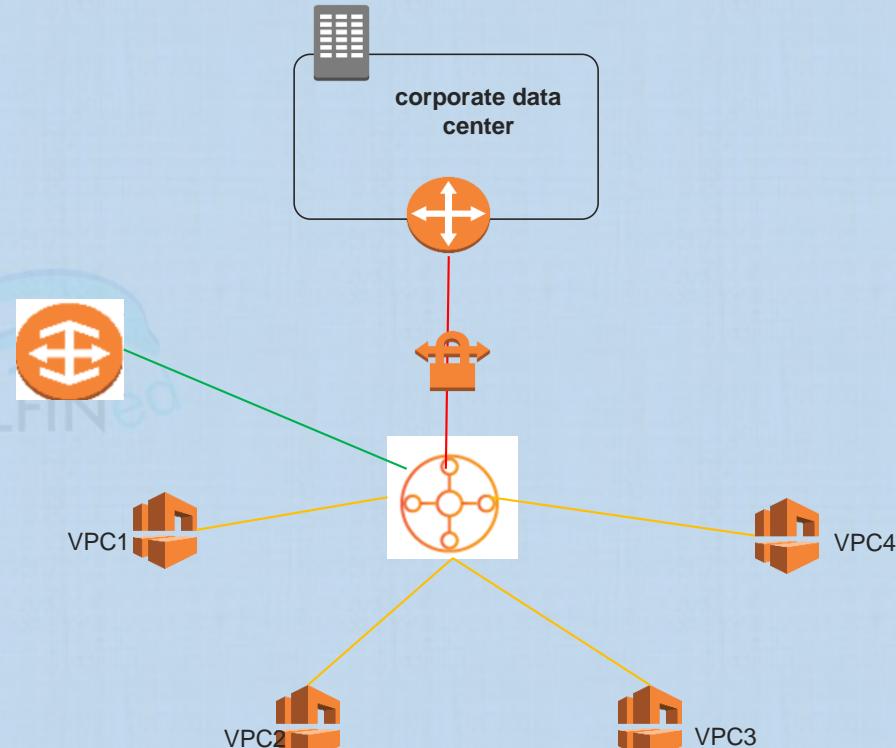
AWS VPC Peering - Limitations

- VPC Peering is non transitive
- To allow 4 VPCs to talk to each other,
 - You need a full mesh among them
 - That is $n(n-1) / 2$ VPC peerings
 - 6 peering connection are required
 - What if these were 20 or 30 VPCs



AWS Transit Gateway

- A *transit gateway* is a network transit hub that you can use to interconnect your virtual private clouds (VPC) and on-premises networks.
- It is a regional resource
- VPCs are allowed to communicate with one another, and with On-premise CIDR blocks by default.
- This can be changed by creating multiple route tables, associated different VPCs with different routing tables to limit/control who talks to whom
- A Transit Gateway can be associated across accounts.



AWS Transit Gateway Concepts

- **Attachment**
 - A VPC, an AWS Direct Connect gateway, or a VPN connection can be attached to a transit gateway.
- **Transit gateway route table**
 - A transit gateway has a default route table and can optionally have additional route tables.
 - A route table includes dynamic and static routes that decide the next hop based on the destination IP address of the packet.
 - The target of these routes could be a VPC or a VPN connection.
 - By default, the VPCs and VPN connections that are attached to a transit gateway are associated with the default transit gateway route table.

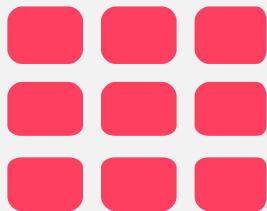
AWS Transit Gateway

- **Associations**
 - Each attachment is associated with exactly one route table.
 - Each route table can be associated with zero to many attachments.
- **Route propagation**
 - A VPC or VPN connection can dynamically propagate routes to a transit gateway route table.
 - With a VPC, you must create static routes to send traffic to the transit gateway.
 - With a VPN connection, routes are propagated from the transit gateway to your on-premises router using Border Gateway Protocol (BGP).

AWS Transit Gateway

- It supports AWS Resource Access Manager (RAM), hence, it can be shared between accounts
- It is enabled per AZ
- Regional resource
- It supports IPv6





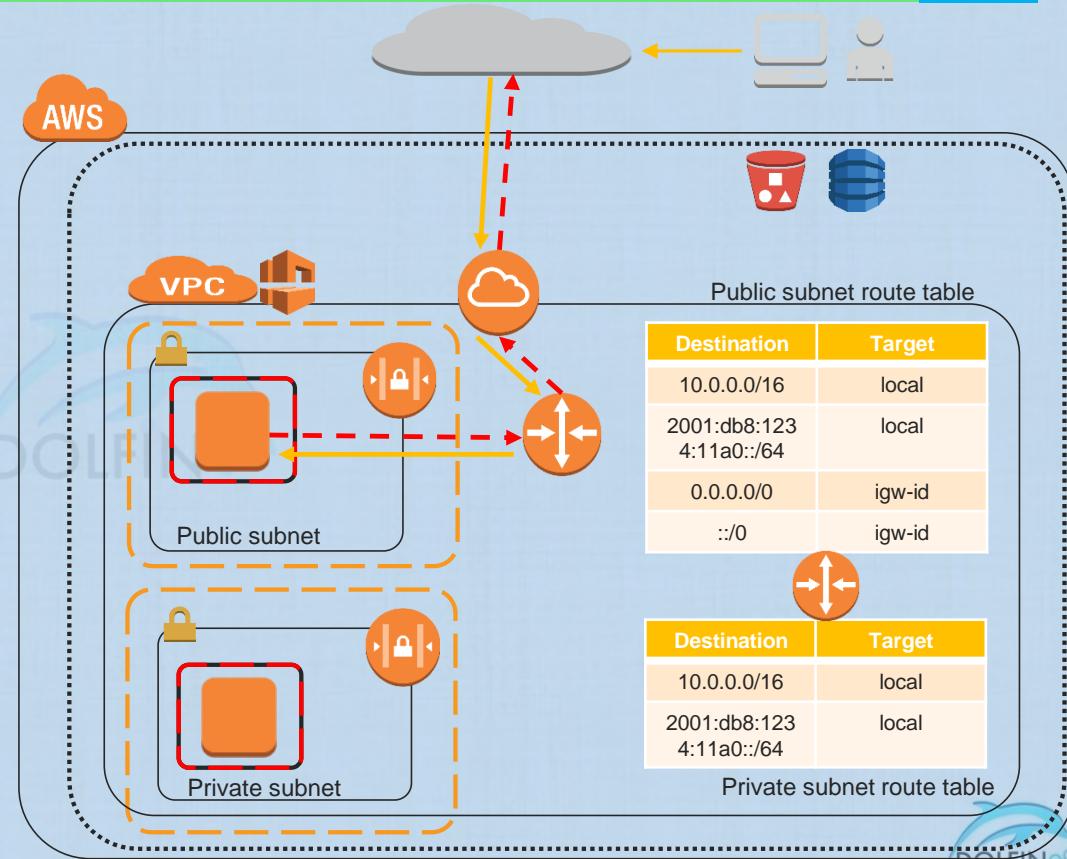
AWS IPV6 EGRESS ONLY GATEWAY

YOU CAN DO IT TOO!



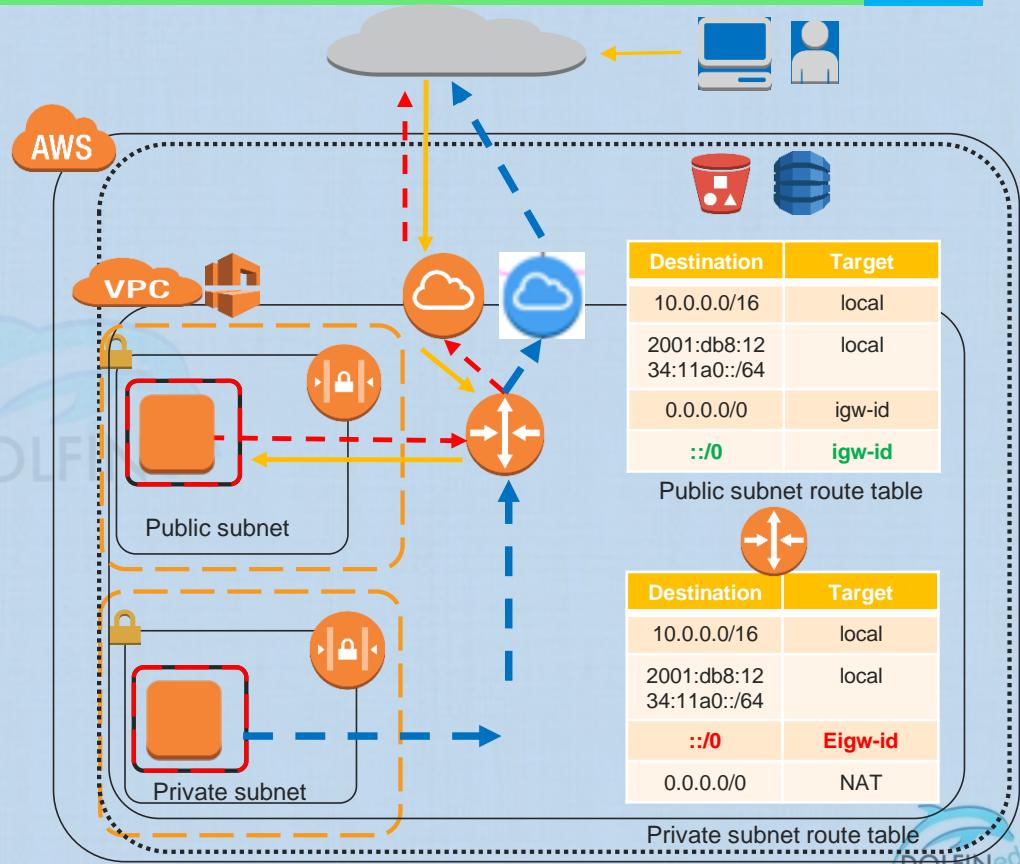
IPv6 and the Internet Gateway

- As in IPv4, an instance with an IPv6 address, in a public subnet, can connect to the Internet through the Internet gateway.
- Clients on the Internet can initiate a connection to such an instance as well.
- IPv6 addresses are globally unique, which means IPv6 addresses are public IP addresses (no Private address notion)



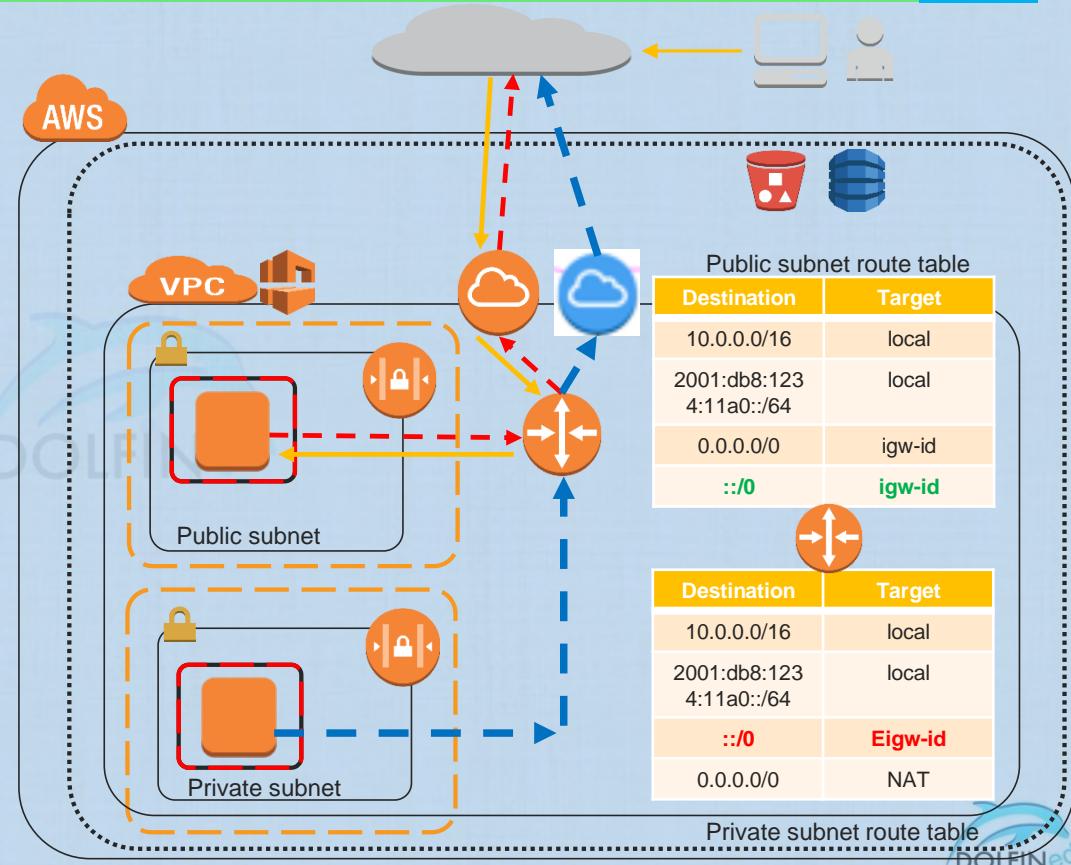
IPv6 and Egress-only Internet Gateway

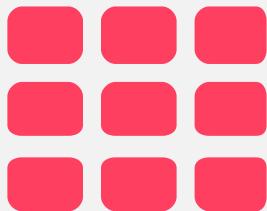
- To prevent initiating traffic to your IPv6 addressed instances from the internet, yet allow the instance to access the internet (initiate traffic)
 - An egress-only Internet gateway will be required.
 - Create an egress-only Internet gateway in the VPC,
 - Add a route to the respective route table, directing all ::/0 (All IPv6 traffic destined to the Internet) pointing to the egress-only Internet gateway.
- Any IPv6 traffic in this subnet (that has no other explicit routes in the route table) is then routed to the egress-only Internet gateway.
- The egress-only Internet gateway is stateful:
 - It forwards traffic from the instances in the subnet to the Internet or other AWS services,
 - It then sends the response back to the instances.



IPv6 and Egress-only Internet Gateway

- An egress-only Internet gateway has the following characteristics:
 - A security group can not be associated with an egress-only Internet gateway.
 - Protect the private subnet EC2 instances with security groups
 - Use a network ACL to control traffic to/from the subnet for which the egress-only Internet gateway routes traffic.





AWS VPC ENDPOINTS

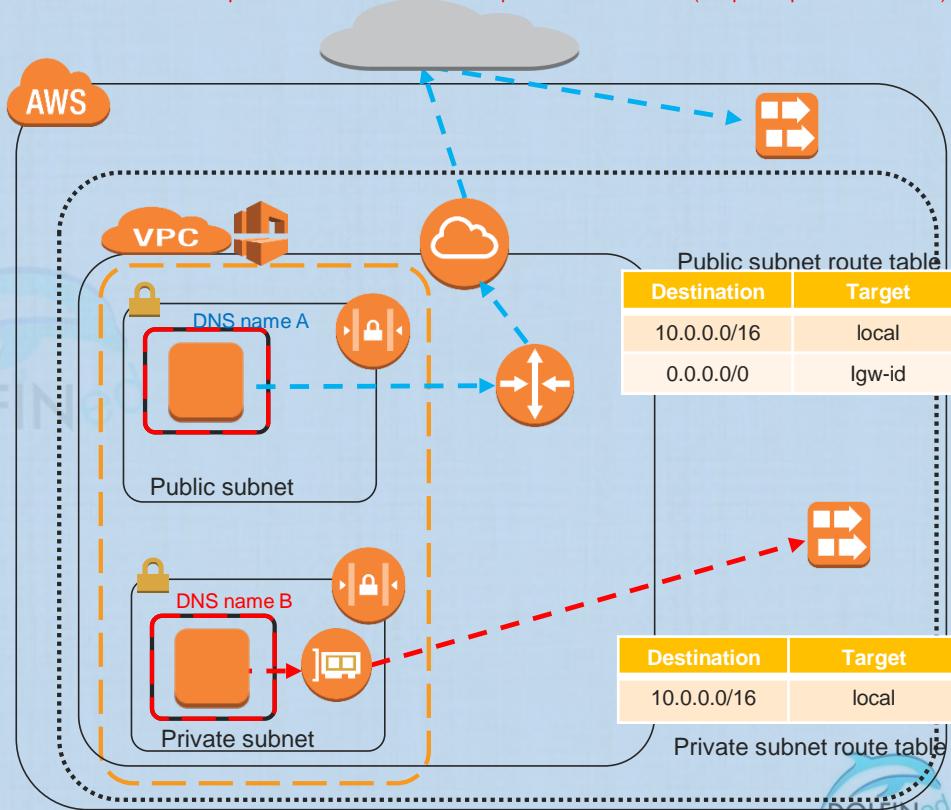
YOU CAN DO IT TOO!



Interface VPC Endpoints (AWS PrivateLink)

- VPC Endpoints allow the connection from within a VPC to AWS services (that are powered by AWS PrivateLink) privately without going over the public Internet.
- These services include:
 - Some AWS services,
 - Services hosted by other AWS customers and partners in their own VPCs (endpoint services), Supported AWS Marketplace partner services.
- An Interface endpoint is an ENI with a private IP address that serves as an entry point for traffic destined to a supported service.
- AWS will create an ENI per subnet you specify
 - Not highly available, you need to configure it in multiple subnets in multiple AZs
- The owner of the service is the service provider, and you, as the principal creating the interface endpoint, are the service consumer.
- Relies on DNS resolution and is not based on Route table entries.

DNS name A : Kinesis default hostname – kinesis.us-east-1.amazonaws.com
 DNS name B: vpce-123-ab.kinesis.us-east-1.vpce.amazonaws.com (endpoint specific hostname)



Interface VPC Endpoints (AWS PrivateLink)

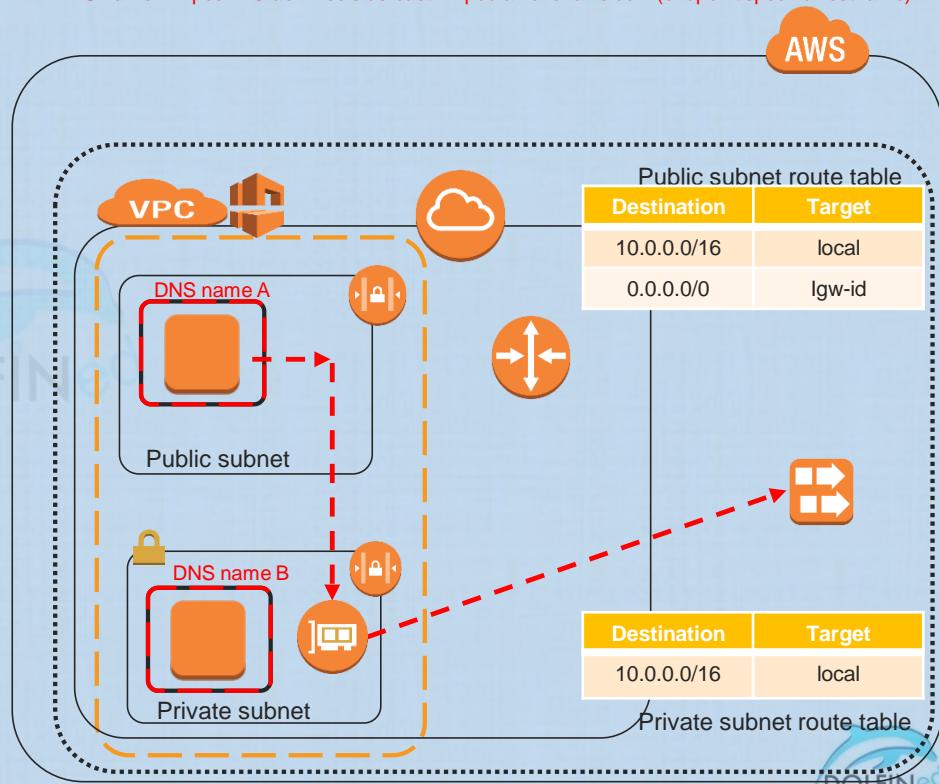
- Multiple DNS endpoints(URLs) are returned
 - Services cannot initiate requests to resources in the VPC through the endpoint.
 - An endpoint only returns responses to traffic initiated from resources in your VPC.
- Example of the supported services:
 - API Gateway, CloudFormation, CloudWatch , CloudWatch Events/Logs, EC2 API, KMS, Kinesis Data Streams, ELB, SNS, Systems Manager, Endpoint Services hosted by other AWS Accounts, STS, Codebuild, AWS Config, Service Catalogue, Secrets Manager, Amazon SageMaker among other services

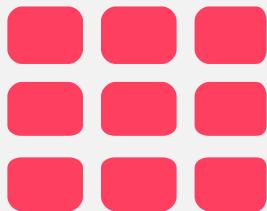
Interface VPC Endpoints – Private DNS

- Instead of using the generated Interface Endpoint's specific DNS hostnames to access the respective service, and to avoid changing the applications (usually they do use the AWS public default DNS hostname for the services), you can enable Private DNS feature (default is enabled for AWS services and Marketplace partner services).
- It associates a private hosted zone with your VPC.
 - The hosted then will have a record set for the default Service's DNS name that resolves to the private IP addresses of the interface Endpoints created in the VPC.
 - This allow to use the service's default DNS hostname instead of the endpoint-specific DNS hostnames to make requests to the service.
 - This way applications in the VPC that were configured to use the default services DNS hostnames, they can continue to use that and be routed to the Interface endpoints.

DNS name A : Kinesis default hostname – kinesis.us-east-1.amazonaws.com

DNS name B: vpce-123-ab.kinesis.us-east-1.vpce.amazonaws.com (endpoint specific hostname)



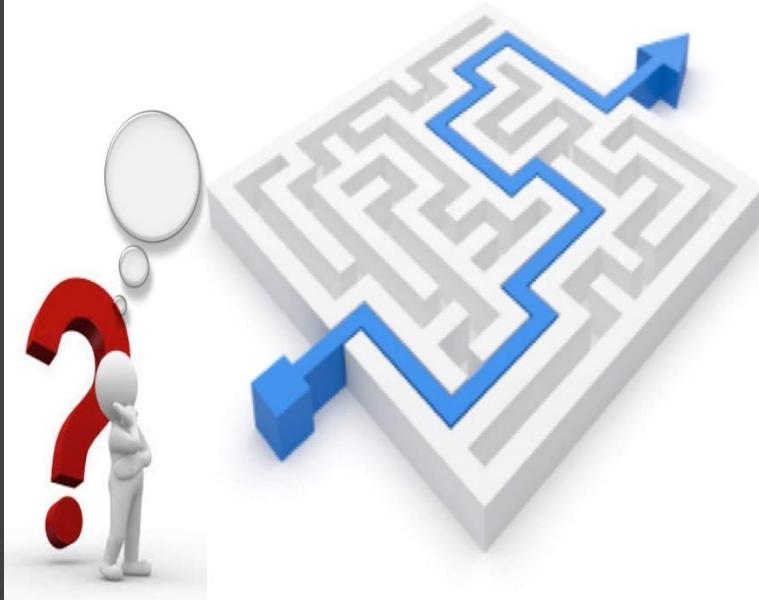


AWS ELASTIC COMPUTE CLOUD (EC2)

YOU CAN DO IT TOO!

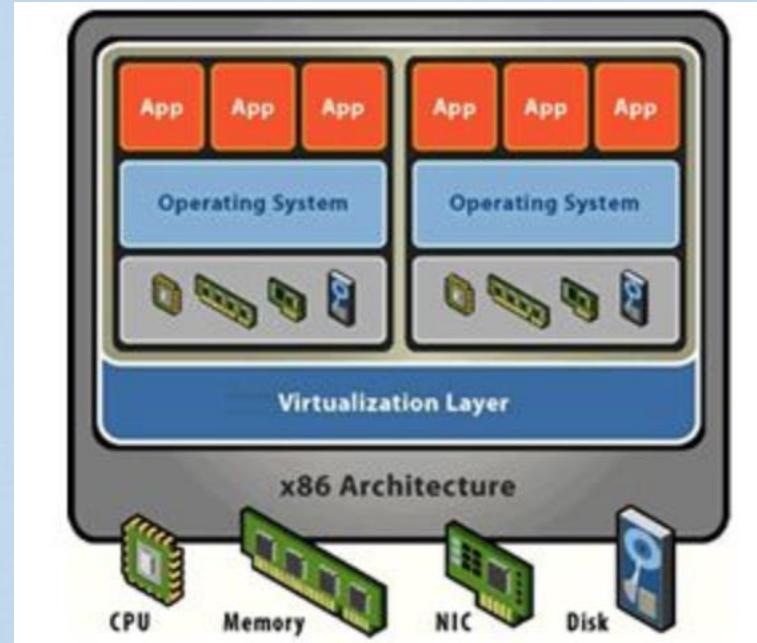


AWS Elastic Compute Cloud (EC2)



Review Topic : Elastic Compute Cloud (EC2)

- EC2 service provides resizable compute capacity in the cloud
- You have root access to each of your EC2 instances
- You can stop, restart, reboot, or terminate your instance
- You can provision your EC2 instances on shared or dedicated hosts (physical servers)



Review Topic : Elastic Compute Cloud

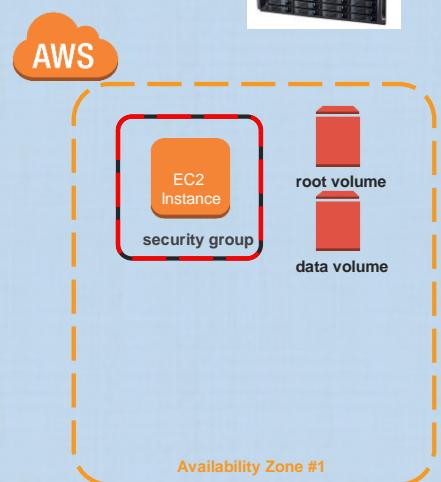
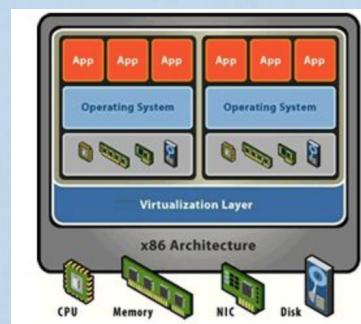
EC2 – Instance Access

- To access an instance you need a key and key pair name
 - When you launch a new EC2 instance, you can create a public/private key pair
 - You can download the private key only once
 - Save it in a safe place so you won't lose it
 - The public key is saved by AWS to match it to the key pair name, and private key when you try to login to the EC2 instance
 - If you launch your instance without a key pair, you will not be able to access it (via RDP or SSH)

Review Topic : Elastic Compute Cloud

EC2

- There is a 20 EC2 instances soft limit per account, you can submit a request to AWS to increase it
- Two types of Block store devices are supported:
 - Elastic Block Store (EBS)
 - Persistent
 - Network attached virtual drives
 - Instance-store
 - Basically the virtual hard drive on the host allocated to this EC2 instance



Review Topic : Elastic Compute Cloud

EC2 – Root/Boot Volume

- EC2 instance root/boot volumes can be EBS or Instance Store volumes
- EBS-Backed EC2 instance
 - It has an EBS root volume
- Instance-store backed EC2 instance
 - It has an Instance-store root volume



Review Topic : Elastic Compute Cloud

EC2 - Charges

- You are charged on EC2 service in hourly based or per second based pricing (depending on which Instance you launch and which AMI (OS or Image) is used)
- A reboot of an EC2 instance is considered as the instance is still running
- If the instance is stopped, you are not charged if it remains stopped
- You are also charged for data transfer in/out of EC2 instance (if sent to outside the AWS region)



Review Topic : Elastic Compute Cloud

EC2 Instance Families

- **General Purpose**
 - Balanced memory and CPU
 - Suitable for most applications
 - Ex. M3, M4, T2
- **Compute Optimized**
 - More CPU than memory
 - Compute & HPC intensive use
 - Ex. C2, C4
- **Memory Optimized**
 - More RAM/memory
 - Memory intensive apps, DB, and caching
 - Ex. R3, R4



Review Topic : Elastic Compute Cloud

EC2 Instance Families

- **GPU compute instances**
 - Graphics Optimized
 - High performance and parallel computing
 - Ex. G2
- **Storage Optimized**
 - Very high, low latency, I/O
 - I/O intensive apps, data warehousing, Hadoop
 - Ex. I2, D2



AWS Elastic Compute Cloud (EC2)

EC2 Block Store Types



Review Topic : Elastic Compute Cloud

EBS types

- **General Purpose (gp2)**
 - SSD-Backed (Solid State Drives)
 - Are better for transactional workloads and Dev/Test environments where performance is highly dependent on IOPS
 - Use cases include, System boot volumes, Virtual desktops, Low-latency interactive apps, and Development and test environments
- **Provisioned IOPS (io1)**
 - SSD-backed
 - Highest-performance SSD volume for mission-critical low-latency or high-throughput workloads (Critical business application or Production DBs)
 - Provides sustainable IOPS performance & Low latency
 - Max IOPS/Volume , 64,000 IOPS



Review Topic : Elastic Compute Cloud

EBS types

- **Throughput Optimized HDD (not SSD) (st1)**
 - Ideal for streaming, big data, log processing, and data warehousing
 - Can NOT be used as a boot volume
 - Use for frequently accessed, throughput intensive workloads
- **Cold HDD (sc1)**
 - Ideal for less frequently accessed workloads
 - Throughput-oriented storage for large volumes of data that is infrequently accessed
 - Scenarios where the lowest storage cost is important
 - Cannot be a boot volume

More can be found here

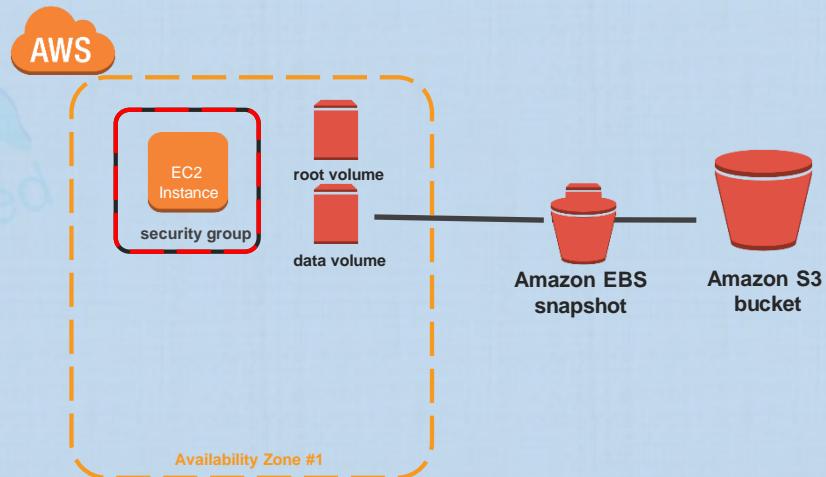
<https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ebs-volume-types.html>



Review Topic : Elastic Compute Cloud

EBS

- Elastic block store (EBS) has 99.999% availability
- You can create point-in-time snapshots of your EBS volumes
- You can resize your EBS volume size up but not down
 - You can do this by creating a new volume of a snapshot of the volume



AWS Elastic Compute Cloud (EC2)

- EC2 Optimized Instances and Enhanced Networking
- Placement Groups



Review Topic : Elastic Compute Cloud

EC2 – EBS optimized instances

- EBS optimized EC2 instances enable the full use of an EBS volume's provisioned IOPS
 - They deliver dedicated performance between EC2 instances and their attached EBS volumes
 - Are designed to work with all EBS volume types
 - This is all about high performance data transfer between EC2 instances and their attached EBS volumes
- For supported instance types, SR-I/OV provides:
 - Higher packet per second (PPS) performance for data transfers
 - Lower latency
 - Very low network Jitter

Review Topic : Elastic Compute Cloud

EC2 – Enhanced Networking

- Takes advantage of SR-I/OV on supported EC2 Instance types to provide:
 - Higher inter-instance PPS rates & Low latency
- EC2 enhanced networking can be enabled on EBS-backed or Instance Store-backed instances
- EC2 enhanced networking can function across Multi-AZ
- To use enhanced networking, the EC2 Instance needs to:
 - Support SR-I/OV
 - Should be created from HVM(Hardware Virtual Machine) AMI
 - Be launched in a VPC
- Using Enhanced Networking does not cost extra

Review Topic : Elastic Compute Cloud

EC2 – Placement Group Types

- **Cluster Placement groups**
 - Clustering of EC2 instances in a single availability zone
 - EC2 instances in the cluster can use the full 10Gbps speeds, and 100Gbps aggregate speed without any oversubscription
 - Use for application that require low latency and/or high throughput between nodes
 - Use SR-I/OV (Single Root I/O Virtualization) based enhanced networking instances for placement groups
- If you try to add instances to the placement group, and you can't due to availability reasons, try to stop and start all instances, this may result in migration to other hosts that have Availability
- It can also be created across a VPC peering connections

Review Topic : Elastic Compute Cloud

EC2 – Placement Group Types

- **Partition Placement Groups:**

- AWS tries to launch your group instances into different logical entities called partitions
- Partition is launched in a separate rack to minimize the impact of a failure
- Partition placement groups can be in a single or multiple availability zones in the same region
- Maximum of 7 partitions per AZ
- Ideal for Hbase, HDFS, and Cassandra
- You get visibility into which Instances are in which partitions
- If the request fails because of insufficient capacity, try again later

Review Topic : Elastic Compute Cloud

EC2 – Placement Group Types

- **Spread Placement Groups:**
 - Launched each instance in the group in a different rack
 - Can be in a single or multiple Availability zones in the same region
 - Maximum of 7 running instances per AZ per group
 - If the request fails because of insufficient capacity, try again later

More on placement groups

<https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/placement-groups.html>



AWS Elastic Compute Cloud (EC2)

EC2 Monitoring and Status Checks



Review Topic : Elastic Compute Cloud

EC2 Status checks

- By default, AWS EC2 service performs automated status checks **each minute**
 - This is done on every running EC2 instance to identify any hardware or software issues
 - Each status check returns either a pass or a fail status
 - If one or more status checks return a “fail”, the overall EC2 instance status is changed to “impaired”
- You can configure CloudWatch to initiate actions (Reboot or Recovery) on impaired EC2 instance (i.e for failed status checks)

Review Topic : Elastic Compute Cloud

EC2 Status checks

- Once EC2 instance(s) status changes to impaired because of a Host Hardware or Software problem, AWS will schedule a stop/start for the EBS backed Instances to relocate them to a different host
- You can also do this manually
- AWS EC2 Service Status checks are very important for Auto Scaling Groups too, to determine EC2 Instance status

Source: aws.amazon.com



Review Topic : Elastic Compute Cloud

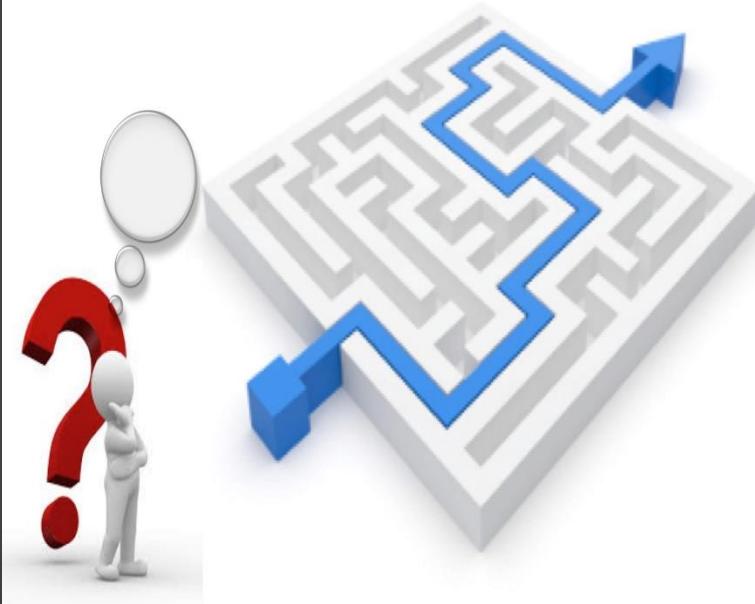
EC2 Monitoring

- EC2 service can send its metric data to AWS Cloudwatch every 5 minutes (enabled by default)
 - This is free of charge
 - It is called basic monitoring
- You can choose to enable detailed monitoring while launching the instance (or later) where the EC2 service will send its metric data to AWS Cloudwatch every 1 minute
 - Chargeable
 - It is called detailed monitoring
- You can set CloudWatch alarm actions on EC2 instance(s) to :
 - Stop, Restart, Terminate, or Recover your EC2 instance
 - You can use Stop or Terminate actions to save cost
 - You can use the reboot and recover to move your EC2 instance to another host



AWS Elastic Compute Cloud (EC2)

EC2 Instance States and Termination



Review Topic : Elastic Compute Cloud

EC2 – Stopping an EC2 instance

- When you stop an **EBS backed instance**, any data in any Instance-store volumes is lost
 - Even though the instance can be re-started, all instance store data will be gone
- When you stop an EBS-Backed EC2 instance
 - Instance performs a shutdown
 - State changes from running -> Stopping -> Stopped
 - EBS volumes remain attached to the instance
 - Any data cached in RAM or Instance Store volumes is lost
 - Most probably, when restarted again, it will restart on a new physical host
 - Instance retains its private IPv4 address, any IPv6 address
 - Instance releases its public IPv4 address back to AWS pool
 - Instance retains its Elastic IP address
 - You will start to be charged for un-used Elastic IP



Review Topic : Elastic Compute Cloud

EC2 – Instance Termination

- By default, EBS root device volumes (created automatically when the instance is launched) are deleted automatically when the EC2 instance is terminated
- Any additional (non boot/root) volumes attached to the instance (those you attach to the instance during launch or later), by default, persist after the instance is terminated
- You can modify both behaviors by modifying the “DeleteOnTermination” attribute of any EBS volume during instance launch or while running



Review Topic : Elastic Compute Cloud

EC2 Termination Protection

- This is a feature you can enable such that an EC2 instance is protected against accidental termination through API, Console, or CLI
- This can be enabled for Instance-store backed and EBS-Backed Instances
- CloudWatch can ONLY terminate EC2 instances if they **do not** have the termination protection enabled



AWS Elastic Compute Cloud (EC2)

EC2 Instance Metadata and User Data



Review Topic : Elastic Compute Cloud

EC2 – Instance Meta Data

- Instance Meta Data:
 - This is instance data that you can use to configure or manage the instance
 - Examples are IPv4 address, IPv6 address, DNS Hostnames, AMI-ID, Instance-ID, Instance-Type, Local-hostname, Public Keys, Security groups...
 - Meta data can be only viewed from within the instance itself
 - i.e you have to logon to the instance
 - **Meta data is not protected by encryption** (cryptography), anyone that has access to the instance can view this data



Review Topic : Elastic Compute Cloud

EC2 – Instance Meta Data

- To view an EC2 Instance's Meta Data (from the EC2 instance console):

GET `http://169.254.169.254/latest/meta-data/`

OR

Curl `http://169.254.169.254/latest/meta-data/`

To view a specific metadata parameter, example to view local hostname

GET `http://169.254.169.254/latest/meta-data/host-name/`



Review Topic : Elastic Compute Cloud

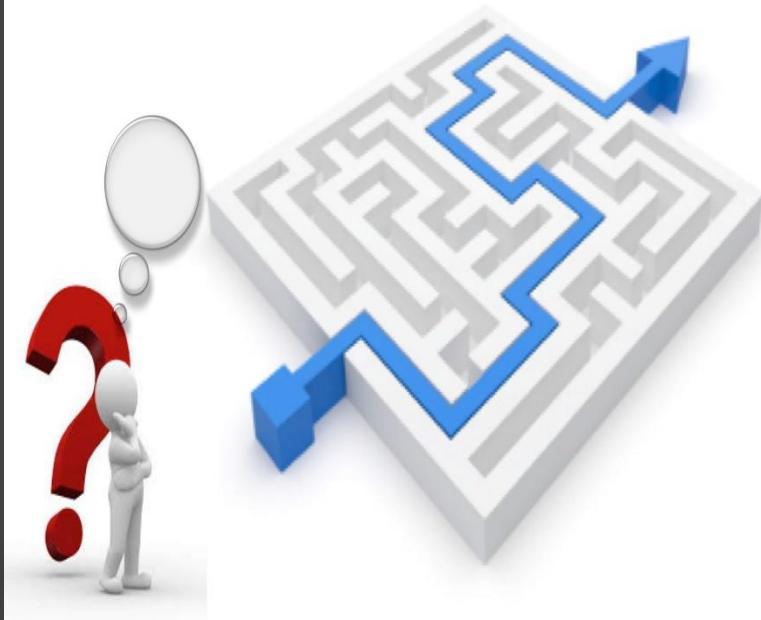
EC2 – Instance User Data

- Instance user data:
 - Is data supplied by the **user at instance launch** in the form of a script to be executed during the instance boot
 - **User data is limited to 16KB**
 - User data can only be viewed from within the instance itself (logon to it)
 - **You can change user data**
 - To do so, you need to stop the instance first (EBS backed)
 - Instance -> actions -> Instance-settings -> View/Change user data
 - **User data is not protected by encryption**, do not include passwords or sensitive data in your user data (scripts)
 - You are not charged for requests to read user data or metadata



AWS Elastic Compute Cloud (EC2)

EC2 IAM Role



Review Topic : Elastic Compute Cloud

EC2 – IAM Roles

- In General, for an AWS services to have permission to read or write to another service, an IAM role is required to be attached to the first AWS Service with rights/permissions on the second AWS service
- Drawing on that, for an EC2 instance to have access to other AWS services (example S3) you need to configure an IAM Role, which will have an IAM policy attached, under the EC2 instance.
 - Applications on the EC2 instance will get this role permission from the EC2 instance's metadata
- You can add an IAM to an EC2 instance during or after it is launched



AWS Elastic Compute Cloud (EC2)

EC2 Bastion Hosts



Review Topic : Elastic Compute Cloud

EC2 – Bastion Hosts (for Linux instances) (Remote Desktop for Windows Instances)

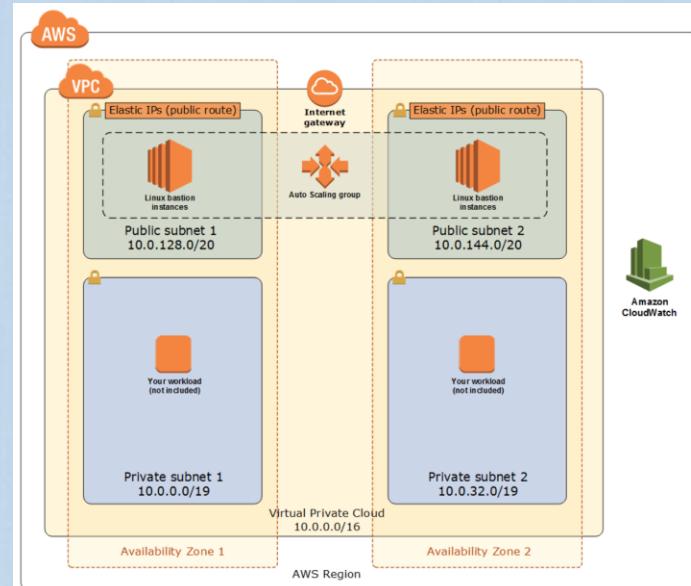
- For inbound, secure, connectivity to your VPC to manage and administer public and/or private EC2 instances, you can use a bastion host (or a jump box/stone).
 - The Bastion host is an EC2 instance, whose interfaces will have a security group allowing inbound SSH access for Linux EC2 instances or inbound RDP access for windows instances
 - Bastion hosts can have auto-assigned public IP addresses or Elastic IP addresses (Elastic IPs are better for security reasons and to fix the IP address)
 - Using Security groups you can further limit which IP CIDRs can access the Bastion Host.
 - Once logged to the Bastion host, you can connect via RDP (Windows) or SSH (Linux) to the EC2 instance(s) you desire to manage



Review Topic : Elastic Compute Cloud

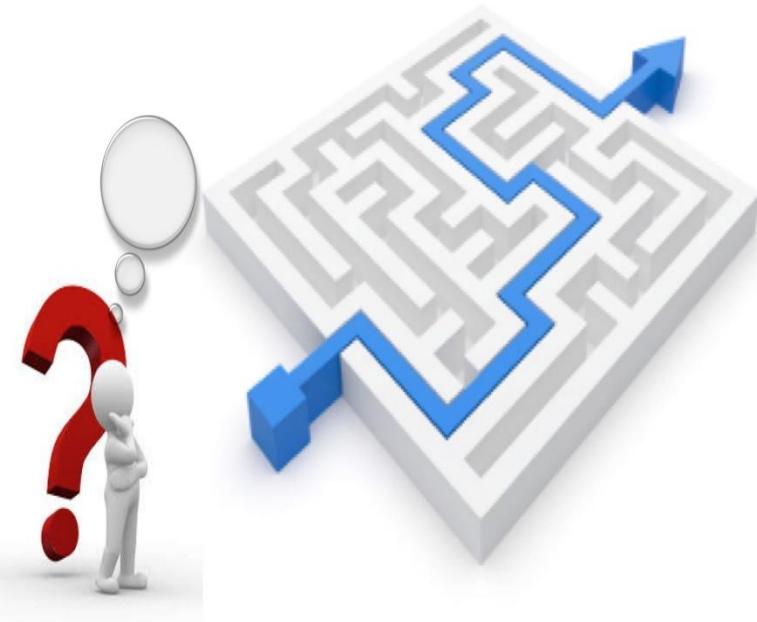
EC2 – Bastion Hosts

- To configure a bastion host in high availability, you can use auto scaling groups as follows:
 - Create the ASG with desired capacity of 2, choose multiple Availability zones (2) using Elastic IPs on each
 - This is the recommended HA way
- (Not an HA but saves on cost) Create an ASG with desired capacity 1, min 1, max 1, such that if the bastion instance fails, or gets terminated, the ASG will launch another one.
 - Downside is, you have only one at a time, and you may have a down time until ASG launches another one,
 - But since this is for management & administration, a downtime can be acceptable



AWS Elastic Compute Cloud (EC2)

EC2 Purchasing Options



Review Topic : Elastic Compute Cloud

EC2 – Instance Purchasing Options

- Reserved Instances
 - 1- or 3-years commitments, high savings, can be zonal (per AZ) or Regional scoped.
- Scheduled instances
 - Upfront purchase instance capacity for a recurring schedule
- Spot Instances
 - Request AWS unused EC2 instances, highest savings, availability not guaranteed when you need it
- Dedicated hosts
 - Pay for a fully dedicated physical host
- Dedicated Instances
 - Pay by the hour for instances that run on single-tenant hardware
- On-Demand
 - Pay by the second for instances that you launch
- Savings plans
 - 1 or 3 years usage commitment for a consistent amount of usage, in USD per hour.



Review Topic : Elastic Compute Cloud

EC2 – Spot instances use cases

- Use it if you are flexible regarding the time you want to run your applications
- Use if your applications can be interrupted (in case AWS terminates the instances)
- Suitable for
 - Data analysis
 - Batch jobs
 - Background processing
 - Optional tasks
- I/O optimized I series Instance(s) are not available as a spot instance



Review Topic : Elastic Compute Cloud

EC2 – Strategies for using Spot Instances

- Use spot instances to augment your reserved and on-demand compute capacity, since spot instances' availability is not guaranteed
- Always guarantee the minimum level of compute using RIs and On-demand and add spot to add compute capacity and save costs



AWS Elastic Compute Cloud (EC2)

EC2 ENI



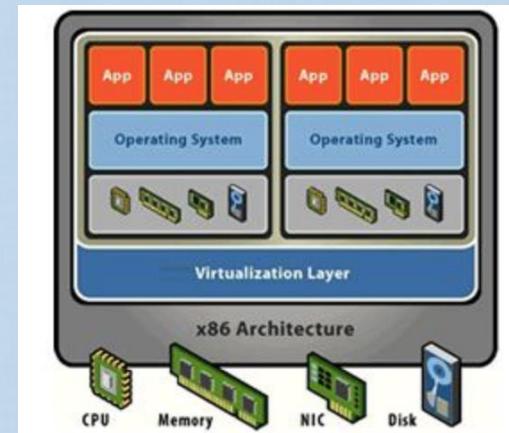
Solutions Architect - Associate



Review Topic : Elastic Computer Cloud (EC2)

Elastic Network Interfaces (ENI)

- Eth0 is the Primary network interface,
 - You can't move/detach the primary (eth0) interface from an instance
- By default, Eth0 is the only ENI created with an EC2 instance when launched
- You can add more interfaces to your EC2 instance (number of additional interfaces is determined by the instance family/type)
- An ENI is bound to an Availability Zone
- You can specify which subnet/AZ you want the additional ENI be added in



Review Topic : Elastic Computer Cloud (EC2)

Elastic Network Interfaces (ENI) - Attributes

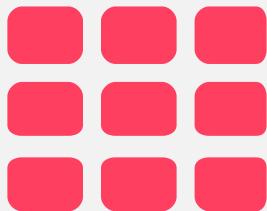
- Each Elastic Network Interface can have up to:
 - A description
 - One Primary IPv4 addresses
 - **One or more secondary IPv4 addresses**
 - Secondary IPv4 addresses can be re-assigned to another instance in failure scenarios if you allow it
 - One Elastic IP address corresponding to each IPv4 address (via NAT)
 - One Public IPv4 address (automatically assigned)
 - One or more IPv6 addresses
 - **Up to 5 Security groups**
 - A MAC address
 - **A source/destination check flag**



Review Topic : Elastic Computer Cloud (EC2)

Secondary IP addresses - Benefits

- You can configure secondary IPv4 addresses to your EC2 instance's Interfaces and ENIs
- To attach a network interface (ENI) in a subnet to EC2 instance in another subnet, **they both "MUST" be in the same AWS Region and same AZ**
- It can be useful to assign multiple IP addresses to an EC2 instance in your VPC to do the following:
 - Hosting multiple websites on a single server (multiple SSL certificates each associated with one IP address)
 - Security and network appliances use in your VPC
 - *Redirecting internal traffic to a standby EC2 instance in case your primary EC2 instance fails,*
 - *This can be achieved by moving (reassigning) the secondary IPv4 address from the failed instance to the standby one*



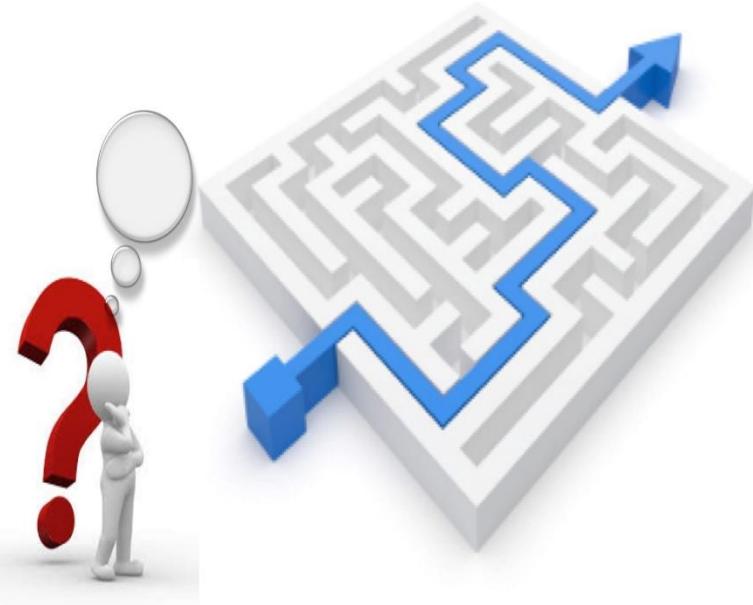
AWS EC2 ELASTIC BLOCK STORE (EBS)

YOU CAN DO IT TOO!



AWS Elastic Block Store

EBS



Review Topic : Elastic Block Store

EBS Device Volumes

- EBS volumes behave like raw, unformatted, external block storage devices that you can attach (mount) to your EC2 instances
- EBS volumes are block storage devices suitable for database style data that requires frequent reads and writes
- An EBS volume data is replicated by AWS across multiple servers in the same AZ to prevent data loss resulting from any single AWS component failure
- EBS volumes are attached to your EC2 instances through the AWS network, like virtual hard drives
- An EBS volume can attach to a single EC2 instance only at a time
- Both EBS volume and EC2 instance MUST be in the same AZ



Review Topic : Elastic Compute Cloud

EC2 – Root/Boot Volume

- EC2 instance root/boot volumes can be EBS or Instance Store volumes
- EBS-Backed EC2 instance
 - It has an EBS root volume
 - It can have other EBS or Instance store data volumes (non root volumes)
- Instance-store backed EC2 instance
 - It has an Instance-store root volume
 - You can EBS volumes after launch

Review Topic : EBS

EBS Persistence

- EBS is persistent and can retain the data it has even when the EC2 instance is stopped or restarted,
 - If configured, it can persist after the EC2 instance is terminated
 - You can change this by changing the “DeleteOnTermination” attribute of the instance’s block store (EBS) volume(s)
 - » An EBS volume (root or not) with a DeleteOnTermination attribute set to “false”, will not be deleted when the instance is terminated
 - » This comes in handy when you want to keep the EBS volume for future use while you terminate the EC2 instance

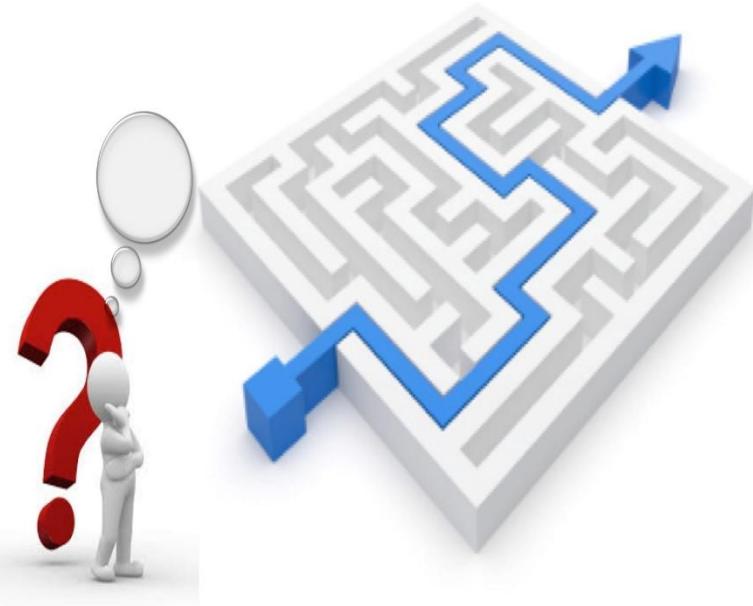
Review Topic : EBS

IOPS Performance – Instance Store vs. EBS

- **Use Instance Store instead of EBS if very high IOPS rate is required**
 - Instance store, although can not provide for data persistence, but it can provide for much higher IOPS compared to, network attached, EBS storage



AWS EBS Snapshots



Review Topic : EBS

EBS Snapshots

- EBS Snapshots are point-in-time Images/copies of your EBS volume(s)
 - Can be taken from root or non-root (Data) EBS volumes and will only include the data that was on the volume when the snapshot process started (but not the data written afterwards)
- Any data written to the volume after the snapshot process is initiated, will NOT be included in the resulting snapshot (but will be included in future, incremental, updates)
- EBS Snapshots are stored on S3, however you can NOT access them directly. You can only access them through EC2 APIs
 - This is unlike Instance-Store AMIs (where you specify a S3 bucket of your own)
- Snapshots can be automated (scheduled using Amazon Data Lifecycle Manager) or user-initiated manually
- If the instance is running, you can create a snapshot and access the volume simultaneously



Review Topic : EBS

EBS Snapshots - Characteristics

- Snapshots are Region specific
 - To migrate an EBS from one AZ to another, create a snapshot (region specific), and create an EBS volume from the snapshot in the intended (other) AZ
- You can only create/restore an EBS volume (from a snapshot) of the same or larger size than the original volume size, from which the snapshot was initially created
- You can NOT create a snapshot of an instance store volume, however you can:
 - Backup individual files on the instance store volume (to S3 for example)
 - Copy the data to a new EBS volume attached to the same EC2 instance (then create a snapshot of the EBS volume)



Review Topic : EBS

EBS Snapshots' Consistency

- To take a complete (consistent) snapshot of your non-root (not the boot) EBS volume:
 - Pause file writes to the desired volume for enough time to take a snapshot, your snapshot will be complete (freeze I/Os).
 - If you can't pause file writes, you need to un-mount (Detach) the volume from the EC2 instance, take the snapshot, and then re-mount the volume to ensure a consistent and complete snapshot.
 - You can re-mount the volume while the snapshot status is pending (being processed)
 - This means the volume does not have to be detached until the snapshot completes
- To create a snapshot for a root (boot)EBS volume, you should stop the instance first then take the snapshot.
 - Be careful if you have any Instance-Store volumes on the EC2 instance, their data will be lost once you stop the instance



Review Topic : EBS

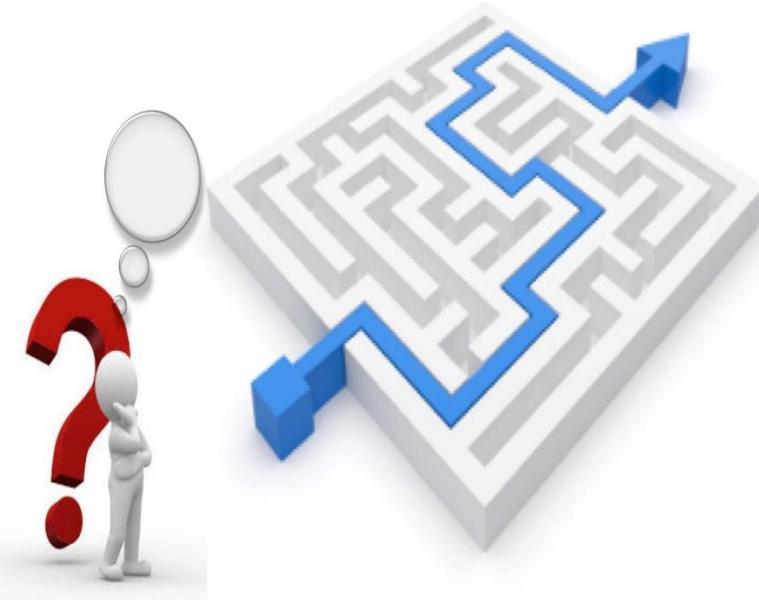
EBS – Incremental Snapshots

EBS snapshots are stored incrementally

- For low cost storage on S3, and a guarantee to be able to fully restore data from the snapshot,
 - What you need is a single snapshot, then further snapshots will only carry the changed blocks (incremental updates)
 - Therefore, you do not need to have multiple full/complete copies of the snapshot (less storage, faster updates).
 - The latest incremental snapshot is enough to re-create your EBS volume, you can go ahead and delete the other. Unless it is required to maintain them for audit reasons.
- EBS snapshots are asynchronously created
 - This refers to the fact that, updates or changes to snapshots do not have to happen immediately when the respective volume data changes



AWS EBS Encryption



Review Topic : EBS

EBS Encryption Support

- Encryption operation is done on the servers hosting the EC2 instances
- Uses AWS Key Management Service (KMS) Customer Master Keys (CMKs)
- EBS Encryption is supported on **all EBS volume types**,
- EBS Encryption is supported on majority of instance families but **NOT on all EC2 instance types within an instance family**
- Please consult AWS documentation website for latest updates
- <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/EBSEncryption.html>
- Snapshots of encrypted volumes are also encrypted.
- Creating an EBS volume from an encrypted snapshot will result it an encrypted volume.
- You can expect the same IOPS performance on encrypted volumes as on unencrypted volumes, with a minimal effect on latency.

Review Topic : EBS

EBS – Data Encryption at Rest

- Data encryption at rest means, encrypting data while it is stored on the data storage device
- There are many ways you can encrypt data on an EBS volume at rest, while the volume is attached to an EC2 instance
 - Use 3rd party EBS volume (SSD or HDD) encryption tools
 - Use encrypted EBS volumes
 - Use encryption at the OS level (using data encryption plugins/drivers)
 - Encrypt data at the application level before storing it to the volume
 - Use encrypted file system on top of the EBS volume



Review Topic : EBS

EBS Encryption – Data In-Transit

- Remember that the EBS volumes are not physically attached to the EC2 instance, rather, they are virtually attached through the AWS infrastructure
 - This means when you encrypt data on an EBS volume, data is actually encrypted on the server hosting the EC2 instance then transferred, encrypted, to be stored on the EBS volume
 - This means data in transit between EC2 and Encrypted EBS volume is also encrypted
 - Data at rest in the EBS volume is also encrypted
 - Encrypted is handled transparently, where encrypted volumes are accessed exactly like unencrypted ones
 - You can attach an encrypted and unencrypted volumes to the same EC2 instance



Review Topic : EBS

EBS Encryption

There is no direct way to change the encryption state of a volume.

To change the state (indirectly) you need to follow one of the following ways:

1. Attach a new, encrypted, EBS volume to the EC2 instance that has the data to be encrypted then,
 - Mount the new volume to the EC2 instance
 - Copy the data from the un-encrypted volume to the new volume
 - Both volumes MUST be on the same EC2 instance
2. Volumes that you create from an unencrypted snapshot that you own or have access to can be encrypted on-the-fly.
3. Create a snapshot of the un-encrypted volume,
 - Copy the snapshot and choose encryption for the new copy, this will create an encrypted copy of the snapshot
 - Use this new copy to create an EBS volume, which will be encrypted too
 - Attach the new, encrypted, EBS volume to the EC2 instance
 - You can delete the one with the un-encrypted data



Review Topic : EBS

EBS Volume/Snapshot - Encryption Keys

- To encrypt a volume or a snapshot, you need an encryption key, these keys are called Customer Master Keys (CMKs) and are managed by AWS Key Management Service (KMS)
- When encrypting the first EBS volume, AWS KMS creates a **default CMK key**,
 - This key is used for your first volume encryption, encryption of snapshots created from this volumes, and subsequent volumes created from these snapshots
- After that, each newly encrypted volume is encrypted with a unique/separate AES256 bits encryption key
 - This key is used to encrypt the volume, its snapshots, and any volumes created of its snapshots



Review Topic : EBS

Changing Encryption Keys

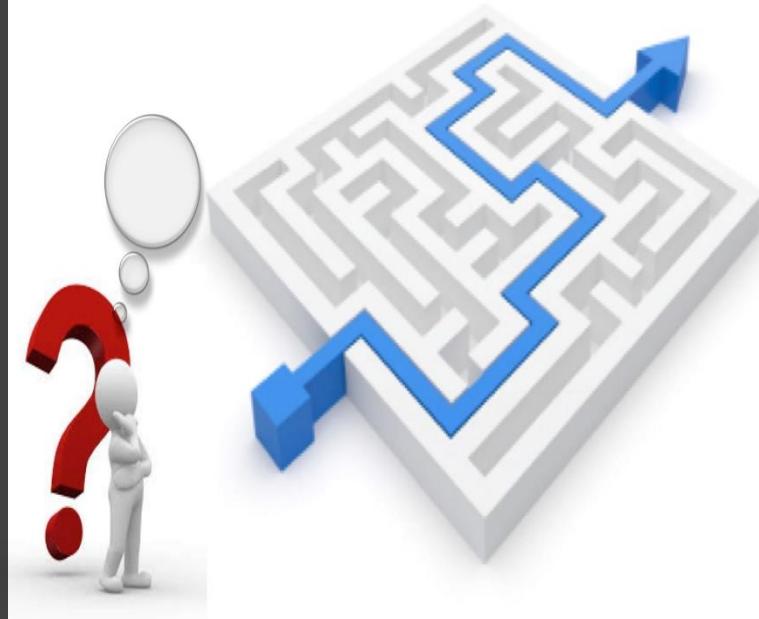
- You can NOT change the encryption (CMK) key used to encrypt an existing encrypted snapshot or encrypted EBS volume
 - If you want to change the key, create a copy of the snapshot, and specify, during the copy process, that you want to re-encrypt the copy with a different key
 - This comes in handy when you have a snapshot that was encrypted using your default CMK key, and you want to change the key in order to be able to share the snapshot with other accounts

Review Topic : EBS

EBS Volume Migration

- **EBS volumes are AZ specific** (can be used in the AZ where they are created only)
 - To move/migrate **your EBS volume to another AZ** in the same region,
 - Create a snapshot of the volume,
 - Use the snapshot to create a new volume in the new AZ
 - Snapshots are Region specific
 - To move/migrate **you EBS volume to another region**,
 - You need to create a snapshot of the volume,
 - Copy the snapshot and specify the new region where it should be,
 - In the new region, create a volume out of the copied snapshot

AWS Sharing/Copying EBS Snapshots



Review Topic : EBS

Sharing EBS Snapshots

- By default, only the account owner can create volumes from the account snapshots
- You can share your unencrypted snapshots with the AWS community by:
 - **Making them public (modifying the snapshot permissions to public) or,**
 - Share your unencrypted snapshots with a selected AWS account(s), **by making them private** then selecting the AWS accounts to share with
- You **can NOT** make your encrypted snapshots public
 - However, you can share it with other AWS accounts if needed, but you need to mark it “private” then share it.
 - You need to also provide the other accounts permissions on the CMK key used to encrypt the snapshot
 - You can NOT share encrypted snapshots that were encrypted with the default CMK key



Review Topic : EBS

Copying Snapshots

- The following is possible for your own snapshots:
 - You can copy an un-encrypted snapshot
 - During the copy process, you can request encryption for the resulting copy
 - You can also copy an encrypted snapshot
 - During the copy process, you can request to re-encrypt it using a different key
- Snapshot copies receive a new Snapshot ID, different from that of the original snapshot



Review Topic : EBS

Copying a Snapshot

- You can copy a snapshot within the same region, or from one region to another
- To move a snapshot to another region, you need to copy it to that region
- You can only make a copy of the snapshot when it has been fully saved to S3 (its status shows as “complete”), and not during the snapshot’s “pending” status (when data blocks are being moved to S3)
- Amazon S3’s Server Side Encryption (SSE) protects the snapshot data in-transit while copying (Since the snapshot and the copy are both on S3)

Review Topic : EBS

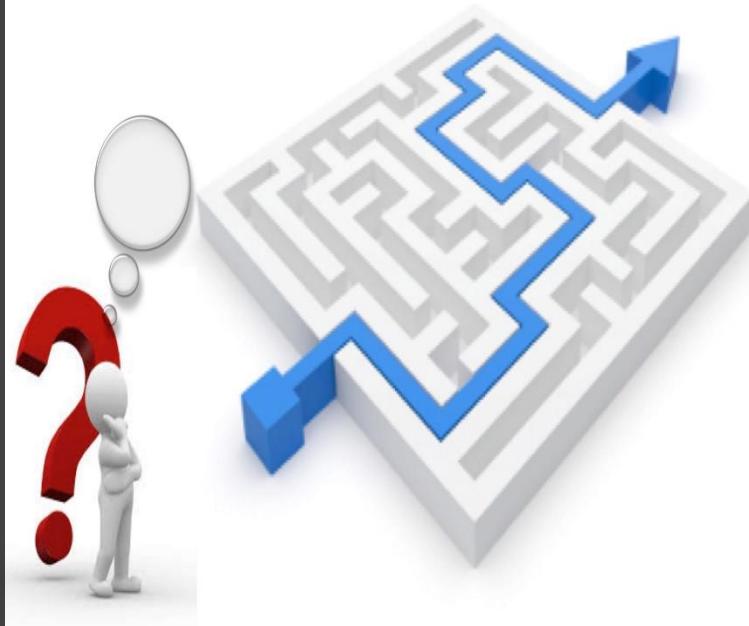
Use Cases - Copying a Snapshot

Use cases for Copying a snapshot

- Geographic expansion
- Disaster Recovery : backing up your data and logs in another region, restoring from that data/logs in case of a disaster
- Migration to another region
- Encryption (of unencrypted volumes)
- Data retention and auditing requirements
 - Copy data and logs to another AWS account for auditing
 - This also protects against account compromise



RAID in AWS



Review Topic : EBS

Redundant Array of Independent Disks (RAID)

- When you need to increase the I/O performance/throughput of your EC2 instance you can do so by:
 - Using EBS optimized EC2 instances
 - Use a RAID array of your EBS volumes
- RAID array is a collection of drives (EBS volumes in our case)
- RAID is accomplished at the OS level, EBS volumes are supported in any RAID array type
- The EC2 instance throughput/bandwidth can handle the resulting array I/O performance to get the best I/O performance
- Use EBS optimized instances or instances with 10Gbps network Interface



Review Topic : EBS

Redundant Array of Independent Disks (RAID)

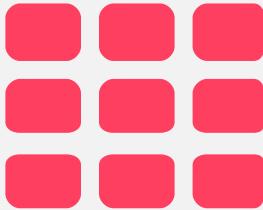
- **Stripping** means, distributing the data to be written over the array disks and writing to multiple disks in parallel (faster writing) without redundancy
- **Mirroring** means, writing the same data to multiple array disks for redundancy
- As a rule of thumb,
 - An EC2 instance's max bandwidth need to be **greater or equal to** the total I/O of EBS volume (or RAID array)
- It is not recommended to use a RAID array as the root/boot volume of an instance



Review Topic : EBS

Redundant Array of Independent Disks (RAID)

- RAID array types:
 - RAID 0:
 - Has stripping and no mirroring
 - Provides the best I/O performance among RAID types
 - Resulting I/O is the sum of the individual disks I/Os
 - Failure of one EBS volume means failure of the entire array
 - RAID 1:
 - Redundancy (writing the same data to multiple drives), no Stripping
 - No I/O performance enhancement
 - RAID 10:
 - Redundancy and Stripping (Combines both RAID 0 and RAID 1)
 - Good performance and Redundancy



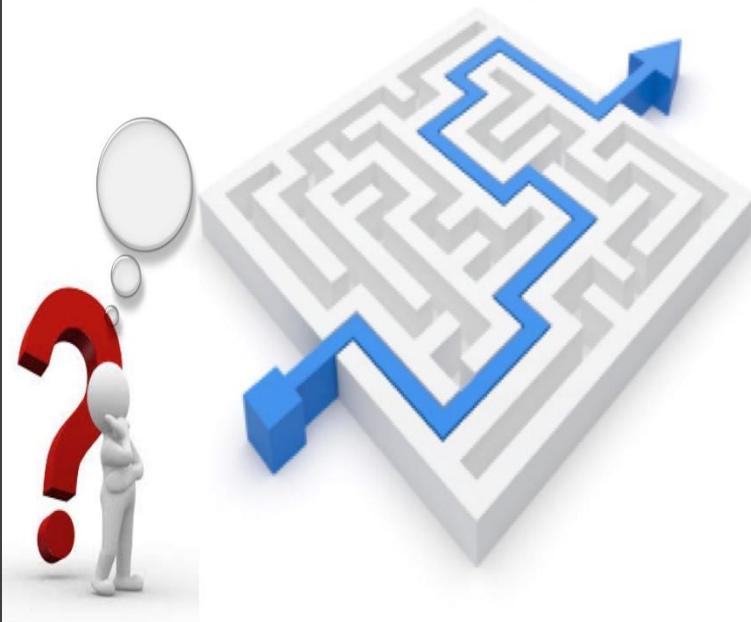
AWS ELASTIC LOAD BALANCING (ELB)

You Can Do It Too!



AWS Elastic Load Balancer (ELB)

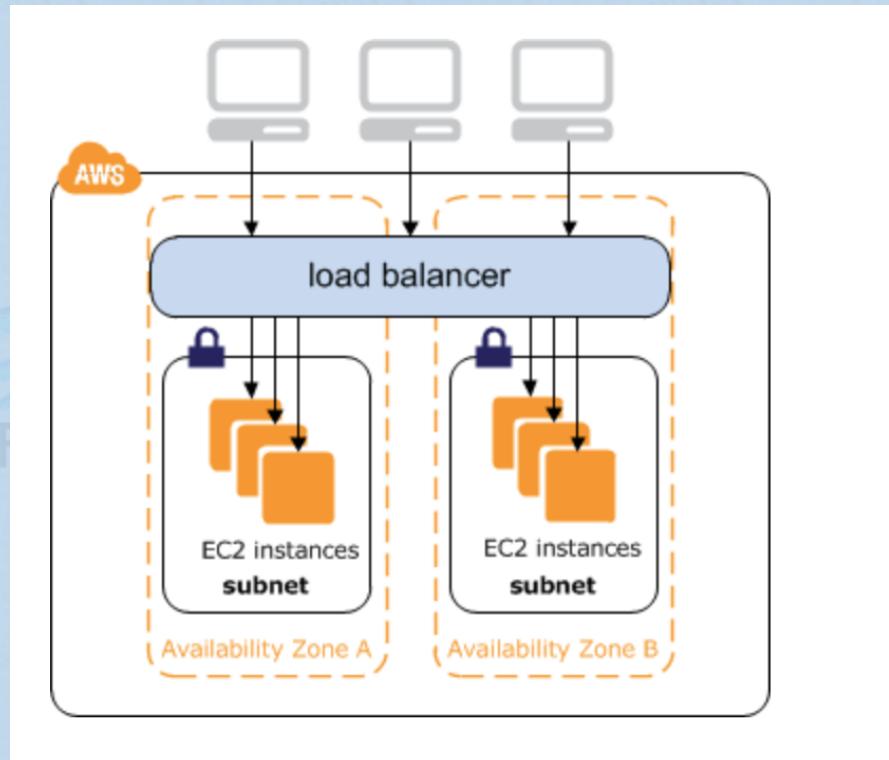
ELB Introduction



Review Topic : Elastic Load Balancer

ELB - Overview

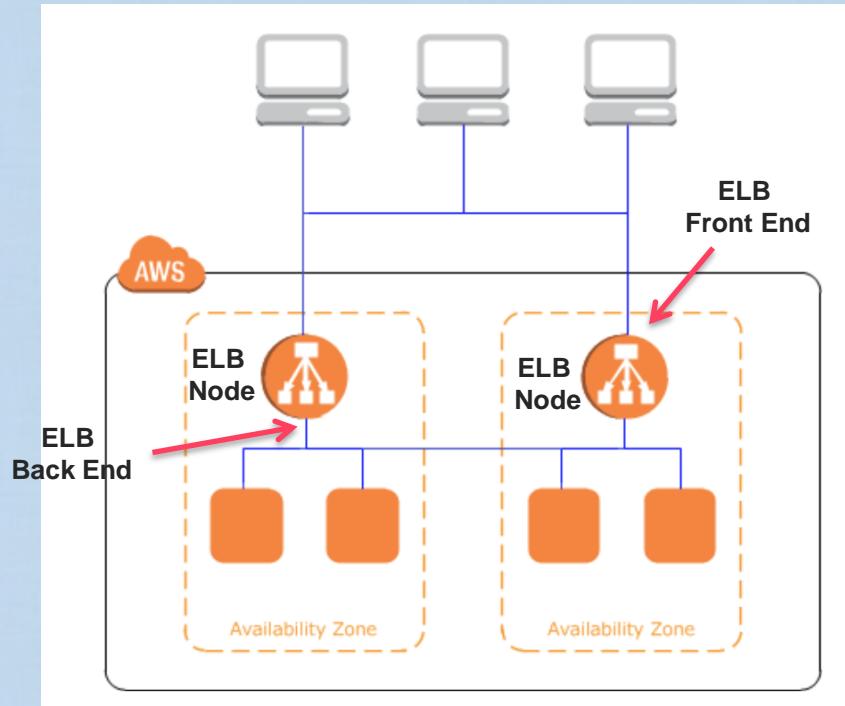
- ELB service is region specific
- An ELB can be launched as internet-facing or Internal in a VPC
- An Internet-facing load balancer has a publicly resolvable DNS name
- Domain names for content on the EC2 instances served by the ELB, is resolved by the Internet DNS servers to the ELB DNS name (and hence IP address(es))
- This is how traffic from the Internet is directed to the ELB front end



Source: aws.amazon.com

Review Topic : Elastic Load Balancer

ELB - Overview



Source: aws.amazon.com



Review Topic : Elastic Load Balancer

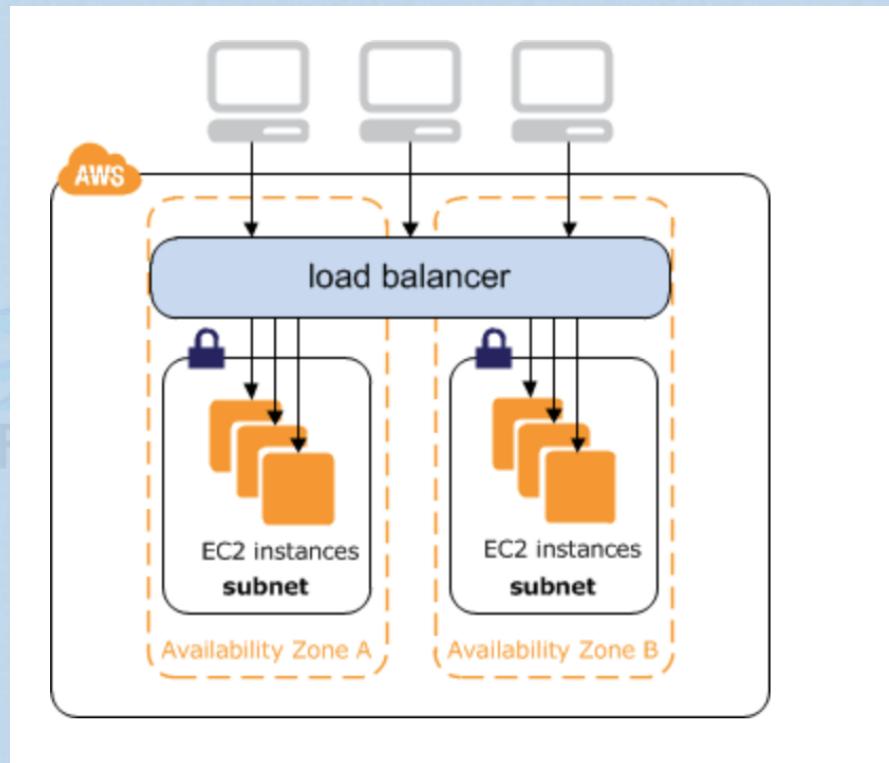
ELB Load Balancer Types in AWS

- There are multiple types of Load balancers in the AWS offerings,
 - Classical load balancer – CLB (focus of this section)
 - Application (layer7) load balancer – ALB (Will be explained in the next section)
 - Network Load Balancer – NLB (Will be explained in detail later)
- Classical load balancer (ELB) service supports:
 - HTTP, HTTPS, TCP, SSL (but not HTTP/2)
 - Protocols ports supported are : 1 -> 65535
 - It supports IPv4, IPv6 and Dual stack



Review Topic : Elastic Load Balancer

ELB – White Boarding Discussion

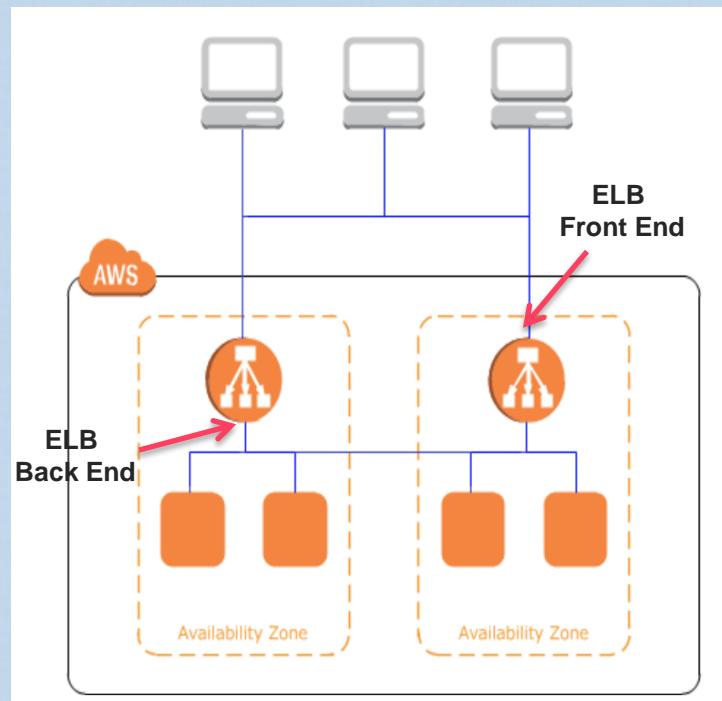


Source: aws.amazon.com

Review Topic : Elastic Load Balancer

ELB - Listeners

- An ELB Listener, is the process that checks for connection requests
- You can configure the protocol/port on which your ELB listener listens for connection requests
- Frontend listeners check for traffic from clients to the ELB
- Backend listeners are configured with protocol/port to check for traffic from the ELB to the EC2 instances



Review Topic : Elastic Load Balancer

ELB

- It may take some time for the registration of EC2 instances under the ELB to complete
- Registered EC2 instances are those that are defined under the ELB
- ELB has “Nothing” to do with the outbound traffic that is initiated/generated from the registered EC2 instances destined directly to the internet, or to any other instances within the VPC.
- ELB has to do only with Inbound traffic destined to the EC2 registered instances (as the destination), and the respective return traffic
- You start to be charged hourly (also for partial hours) once your ELB is active
 - You are also charged for GBs transferred through your Classical Load Balancer
 - You are charged for Load Balancer Capacity Units (LCUs) per hour in case of ALB and NLB



Review Topic : Elastic Load Balancer

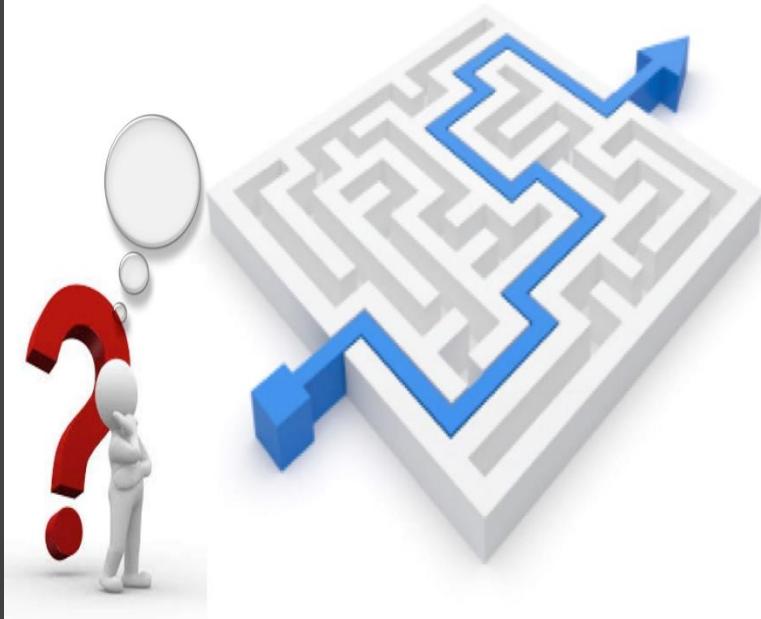
ELB – How it works

- Deleting the ELB does not affect, or delete, the EC2 instances registered with it
- ELB forwards traffic to **eth0** of your registered instances
- In case the EC2 registered instance has multiple **IP addresses on eth0**, ELB will route the traffic to its **primary IP address**



AWS Elastic Load Balancer (ELB)

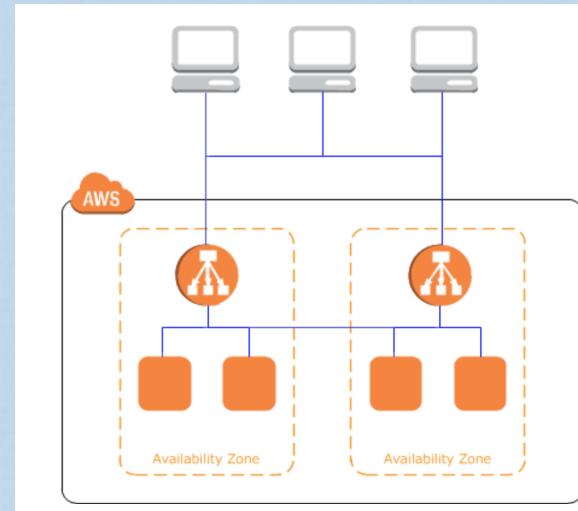
Classical Load Balancer (CLB) How it Works



ELB Scenario

IP Addressing Design & ELB Scaling

- CLB needs to be enabled in at least two AZ's in the region where they are used
- To ensure that the **ELB service can scale ELB nodes** in each AZ, ensure that the subnet defined for the **load balancer is at least /27 in size, and has at least 8 available IP addresses** the ELB nodes can use to scale
 - ELB service uses these IP addresses to open/connect with your registered EC2 instances (remember this in your Security Group and N. ACL settings for ELB environment in your VPC)



For fault tolerance, it is recommended to distribute your registered EC2 instances across multiple AZs within the VPC region

Source: aws.amazon.com



Review Topic : Elastic Load Balancer

ELB – Health Checks

- The **load balancer monitors the health of its registered instances** and ensures that it routes traffic only to healthy instances
 - A healthy instance shows as “**In-Service**” under the ELB
- When the ELB detects an unhealthy instance, it stops routing traffic to that instance
 - An un-healthy instance shows as “**Out-of-Service**” under the ELB
- When the ELB service detects the EC2 instance is back healthy, it resumes traffic routing to the Instance again



Review Topic : Elastic Load Balancer

ELB – Health Checks

- By default
 - AWS console uses ping HTTP (port 80) for health checks
 - AWS API uses Ping TCP (port 80) for health checks
- Registered instances must respond with a HTTP “200 OK” message within the timeout period, else, it will be considered as unhealthy
- Response timeout is 5 seconds (range 2 – 60 seconds)
- Health check interval:
 - Period of time between health checks
 - Default 30 (range 5 – 300 sec)



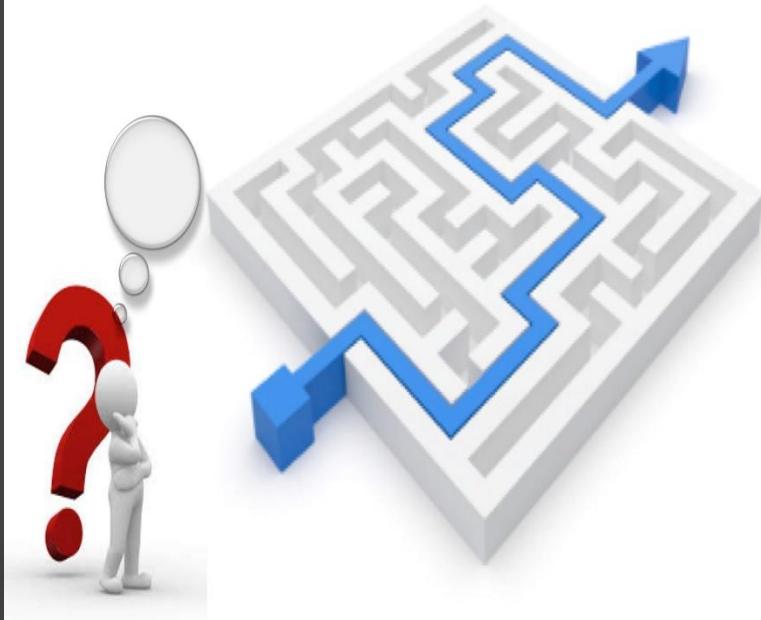
Review Topic : Elastic Load Balancer

ELB – Health Checks

- Unhealthy Threshold:
 - Number of consecutive failed health checks that should occur before the instance is declared unhealthy
 - Range 2-10
 - » Default 2
- Healthy Threshold:
 - Number of consecutive successful health checks that must occur before the instance is considered healthy
 - Range 2-10
 - » Default 10

AWS Elastic Load Balancer (ELB)

ELB Cross Zone Load Balancing



Review Topic : Elastic Load Balancer

ELB – Cross-Zone Load Balancing

- **Cross-Zone load balancing:**
 - The CLB will distribute traffic evenly between registered EC2 instances in the different AZ's it load balances to
 - This is to ensure that each registered /healthy instance gets an equal share of traffic from the CLB
 - If you have 5 EC2 instances in one AZ, and 3 in another, cross-zone load balancing will ensure that each registered EC2 instance will be getting around the same amount of traffic load from the ELB

Review Topic : Elastic Load Balancer

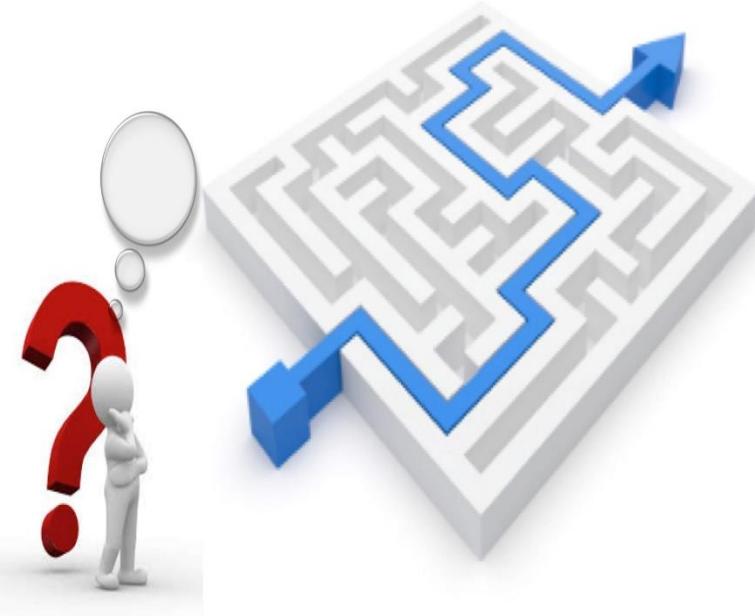
CLB

- To define your CLB in an AZ, you can select one subnet in that AZ
 - Only one subnet can be defined for the CLB in an AZ
- If you register instances in an AZ with the ELB but do not define a subnet in that AZ for the CLB,
 - These instances will not receive traffic from the CLB
- The CLB is most effective if there is one registered instance at least in each CLB defined AZ



Clastic Load Balancer (CLB)

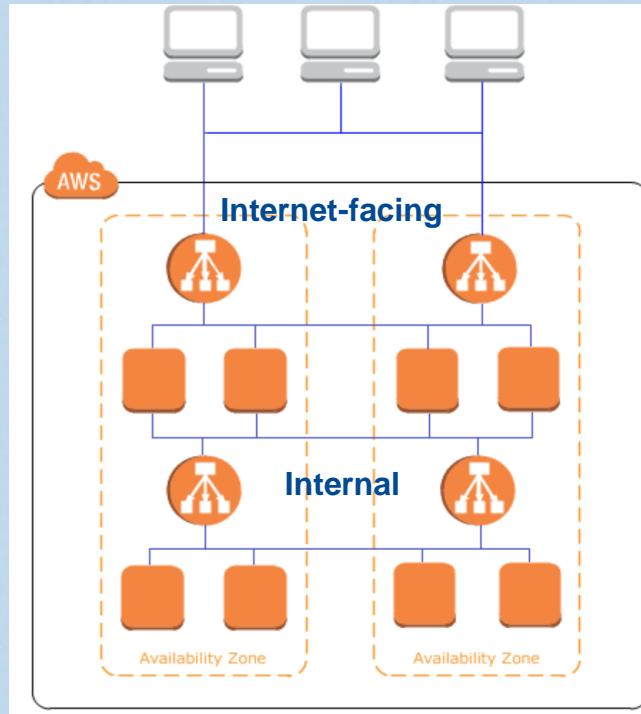
Positioning



Review Topic : Elastic Load Balancer

An ELB can be **Internet facing** or **Internal load balancer**

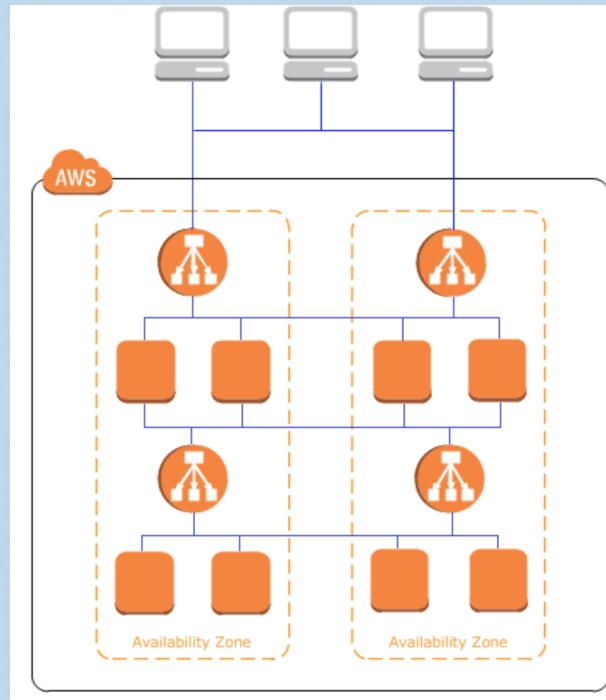
- **Internet facing:**
 - ELB nodes will have public IP addresses,
 - DNS will resolve the ELB DNS name to these IP addresses
 - It routes traffic to the private IP addresses of your registered EC2 instances,
 - Hence, why your instances do not have to have public IP addresses
 - You need one “Public” subnet in each AZ where the internet facing ELB will be defined, such that the ELB will be able to route internet traffic
 - Define this subnet in the ELB configuration



Review Topic : Elastic Load Balancer

ELB - Internal

- **Internal ELB:**
 - ELB nodes will have private IP addresses, to which the DNS resolves ELB DNS name
 - It routes traffic to the Private IP addresses of your registered EC2 instances



Review Topic : Elastic Load Balancer

CLB – Security Groups

- If you create your CLB in a **default VPC**, you can't choose an existing security group for your CLB, or create a new one
 - You must assign a security group to your ELB
 - This will control traffic that can reach your ELB's front end listeners
 - It must also allow health check protocol/ports & listener protocol/port (actual traffic) to reach your registered EC2 instances in the backend
 - You must also ensure that the subnets' N ACLs allow traffic to/from the ELB both ways (on the front and backend side)

Classic Load Balancer (CLB)

ELB Listeners



Review Topic : Elastic Load Balancer

ELB – TCP/SSL Support

TCP/SSL Load Balancer

Use Case	Front-End Protocol	Front-End Options	Back-End Protocol	Back-End Options	Notes
Basic TCP load balancer	TCP	NA	TCP	NA	<ul style="list-style-type: none"> Supports the Proxy Protocol header
Secure website or application using Elastic Load Balancing to offload SSL decryption	SSL	SSL negotiation	TCP	NA	<ul style="list-style-type: none"> Requires an SSL certificate deployed on the load balancer Supports the Proxy Protocol header
Secure website or application using end-to-end encryption with Elastic Load Balancing	SSL	SSL negotiation	SSL	Back-end authentication	<ul style="list-style-type: none"> Requires SSL certificates deployed on the load balancer and the registered instances Does not insert SNI headers on back-end SSL connections. Does not support the Proxy Protocol header.

Review Topic : Elastic Load Balancer

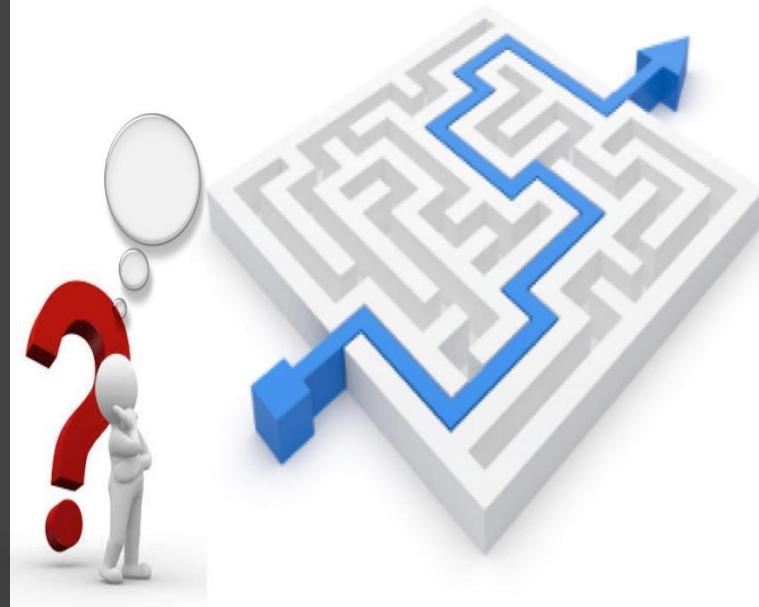
ELB HTTP/HTTPS – Support

HTTP/HTTPS Load Balancer

Use Case	Front-End Protocol	Front-End Options	Back-End Protocol	Back-End Options	Notes
Basic HTTP load balancer	HTTP	NA	HTTP	NA	<ul style="list-style-type: none">Supports the X-Forwarded-For header
Secure website or application using Elastic Load Balancing to offload SSL decryption	HTTPS	SSL negotiation	HTTP	NA	<ul style="list-style-type: none">Supports the X-Forwarded-For headerRequires an SSL certificate deployed on the load balancer
Secure website or application using end-to-end encryption	HTTPS	SSL negotiation	HTTPS	Back-end authentication	<ul style="list-style-type: none">Supports the X-Forwarded-For headerRequires SSL certificates deployed on the load balancer and the registered instances

Classical Load Balancer (CLB)

ELB Sticky Sessions

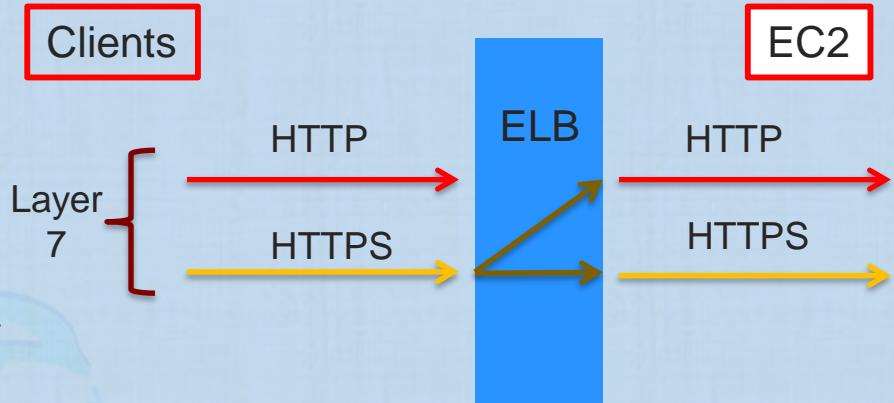


Review Topic : Elastic Load Balancer

CLB – HTTP/HTTPS Session Stickiness

Session Stickiness (Session Affinity)

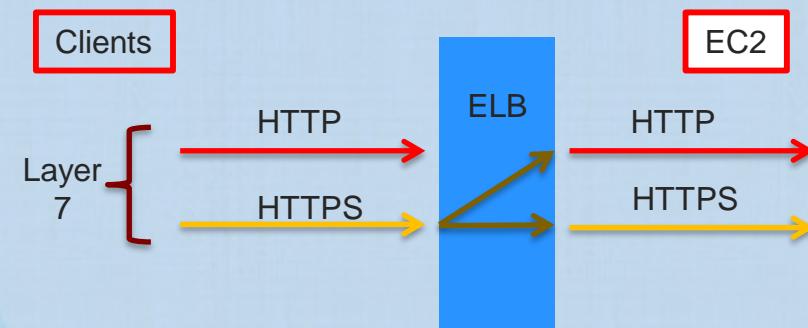
- Whereby the CLB binds a client/user session/requests to a specific backend EC2 instance
- It is not fault tolerant (in case the backend EC2 instance fails)
- It requires SSL termination (SSL Off-loading) on the CLB,
 - This in turn, requires an X.509 (SSL Server) certificate configured on the CLB



Review Topic : Elastic Load Balancer

ELB – HTTP/HTTPS Listeners

- Session Stickiness (Session Affinity)
 - You can upload the X.509 certificate if you have one using IAM, to be loaded on the ELB
 - The X.509 certificate MUST be in the same AWS Region as the ELB
- The duration of the session stickiness is either ELB duration based, or Application based.
- In either case the CLB requires a cookie inserted in the response of the first request, in order to be able to bind future requests from the same client to the same backend EC2 instance.



Review Topic : Elastic Load Balancer

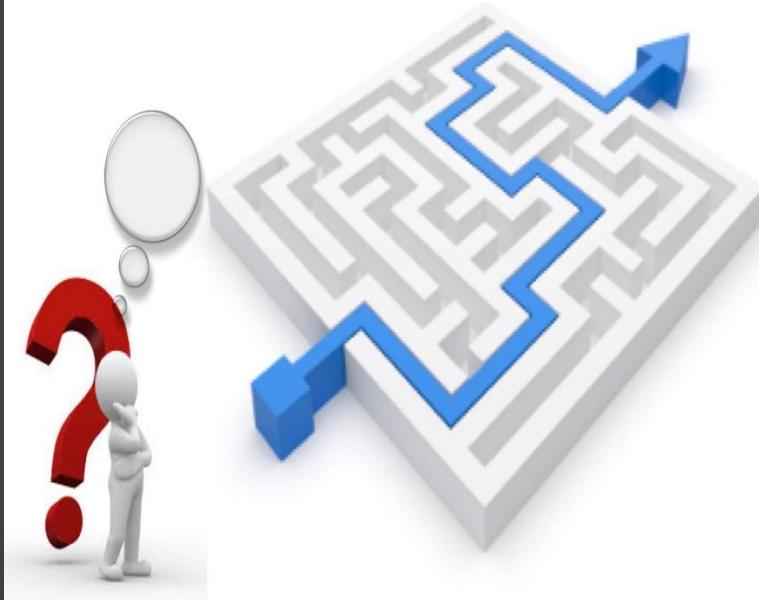
Session Stickiness Duration

- **Application-Controlled Session Stickiness**
 - The load balancer uses a special cookie to associate the session with the instance that handled the initial request, but the duration of stickiness follows the lifetime of the application cookie specified in the policy configuration.
- **CLB duration-based stickiness:**
 - If the application does not have or provide its own cookies, then the ELB can be configured to create one and determine the stickiness duration
 - The ELB inserts a cookie in the response to bind subsequent requests from the user to the same backend instance
 - The cookie helps the ELB identify which user/session should be sticky to which backend instance



Classic Load Balancer (ELB)

CLB Security policy for SSL / HTTPS sessions

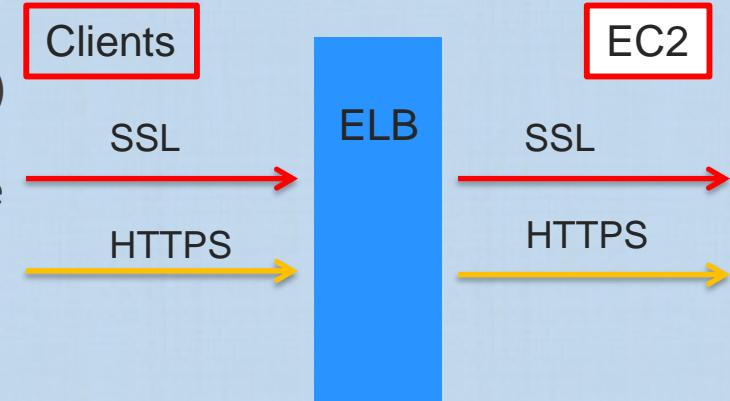


Review Topic : Elastic Load Balancer

ELB – Security Policy for SSL/HTTPs sessions

SSL Session Negotiation

- Elastic Load Balancing uses a Secure Socket Layer (SSL) negotiation configuration, known as a security policy, to negotiate SSL connections between a client and the load balancer.
- The “Security Policy” includes the encryption protocol version, Ciphers to be used...etc
- For Front end (Client to ELB) [HTTPS/SSL]
 - You can define your custom security policy or use the ELB pre-defined security policies
- For backend, encrypted, connections
 - Pre-defined security policies are always used



Review Topic : Elastic Load Balancer

ELB – Security Policy Components

Security Policy Components:

- SSL protocols
 - SSL or TLS, are cryptographic protocols
- SSL Ciphers (a set of ciphers is called a cipher suite)
 - Encryption algorithms
 - SSL can use different ciphers to encrypt data
- Server Order Preference
 - Enabled by default, the first match in the ELB cipher list with the Client list is used



Review Topic : Elastic Load Balancer

SNI and CLB

- Classic Load Balancer **does not support Server Name Indication (SNI)**,
- The ELB supports a single X.509 certificate
- For **multiple SSL certificates** (Like the case of multiple websites each with its own certificate):
 - Create multiple ELB instances (Expensive, Costly, Not scalable, Admin hassle), OR
 - Use TCP Passthrough configuration on the CLB where:
 - Front and back ends are configured with TCP listeners
 - The load balancer passes the request through, with the SNI certificate as is.
 - You then handle the HTTPS termination (SNI certificates) from the EC2 instance itself.
 - You loose X-Forwarded-For option once you go away from HTTP/HTTPS



Review Topic : Elastic Load Balancer

ELB – HTTPS/SSL X.509 Certificates

- CLB does not support Client side certificates with HTTPS (Client side certificates are used to confirm the Identity of the client – or a two way authentication)



AWS Elastic Load Balancer (ELB)

ELB Connection Draining & DNS Failover



Review Topic : Elastic Load Balancer

ELB – Connection Draining

- When identifying unhealthy instances, the CLB will wait for a period of 300 seconds (by default), for the in-flight sessions to this EC2 back end instance to complete
 - If the in-flight sessions are not completed before the maximum time (300 seconds - configurable between 1 – 3600 seconds), the ELB will force termination of these sessions
 - During the connection draining, the Back end instance state will be “InService : Instance Deregistration Currently In Progress”
 - AWS Auto-Scaling would also honor the connection draining setting for unhealthy instances
 - During the connection draining period, ELB will not send new requests to the unhealthy Instance



Review Topic : Elastic Load Balancer

DNS Failover for your CLB

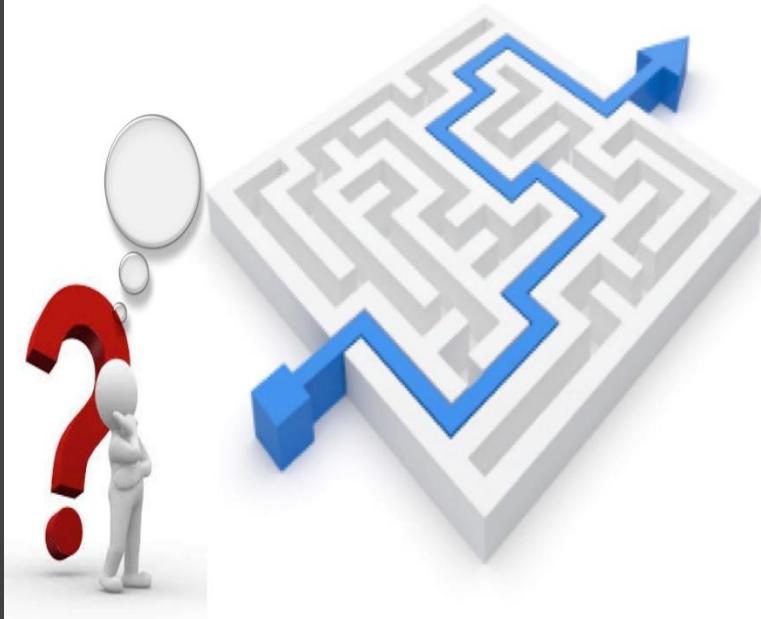
For fault tolerance across regions:

- You can use AWS Route53 to achieve DNS failover between two CLBs in two different AWS regions



AWS Classic Load Balancer

ELB Monitoring, Scaling & testing



Review Topic : Elastic Load Balancer

ELB – Monitoring

ELB monitoring can be achieved by:

- AWS Cloud Watch:
 - AWS ELB service sends ELB metrics to cloud watch every “One minute”
 - ELB service sends these metrics only if there are requests flowing through the ELB
 - AWS Cloud Watch can be used to trigger an SNS notification if a threshold you define is reached
- Access Logs:
 - Disabled by default
 - You can obtain request information such as requester, time of request, requester IP, request type...etc
 - Optional (disabled by default), you can choose to store the access logs in an S3 bucket that you specify



Review Topic : Elastic Load Balancer

ELB – Monitoring

- **Access Logs (Cont.):**
 - You are not charged extra for enabling access logs
 - » You pay for S3 storage
 - You are not charged for data transfer of access logs from ELB to the S3 bucket
- **AWS Cloud Trail:**
 - You can use it to capture all API calls for your ELB
 - You can store these logs in an S3 bucket that you specify

Review Topic : Elastic Load Balancer

ELB Scaling & Load Testing your Applications

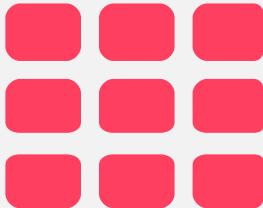
For efficient load testing of your ELB or applications hosted on backend instances

- Use multiple testing instances of client testing & try to launch the tests at the same time
 - You can also use global testing sites if possible
- If using a single client for testing, ensure your testing tool will enforce the Re-resolving of DNS with each testing/request initiated for testing
 - This will ensure that as ELB service launches new ELB nodes, the new nodes will be leveraged through DNS re-resolution

An alternative approach would be to implement a DNS round robin (e.g., with Amazon Route 53). In this case, DNS responses return an IP address from a list of valid hosts in a round robin fashion. While easy to

implement, this approach does not always work well with the elasticity of cloud computing. This is because even if you can set low time to live (TTL) values for your DNS records, caching DNS resolvers are outside the control of Amazon Route 53 and might not always respect your settings.





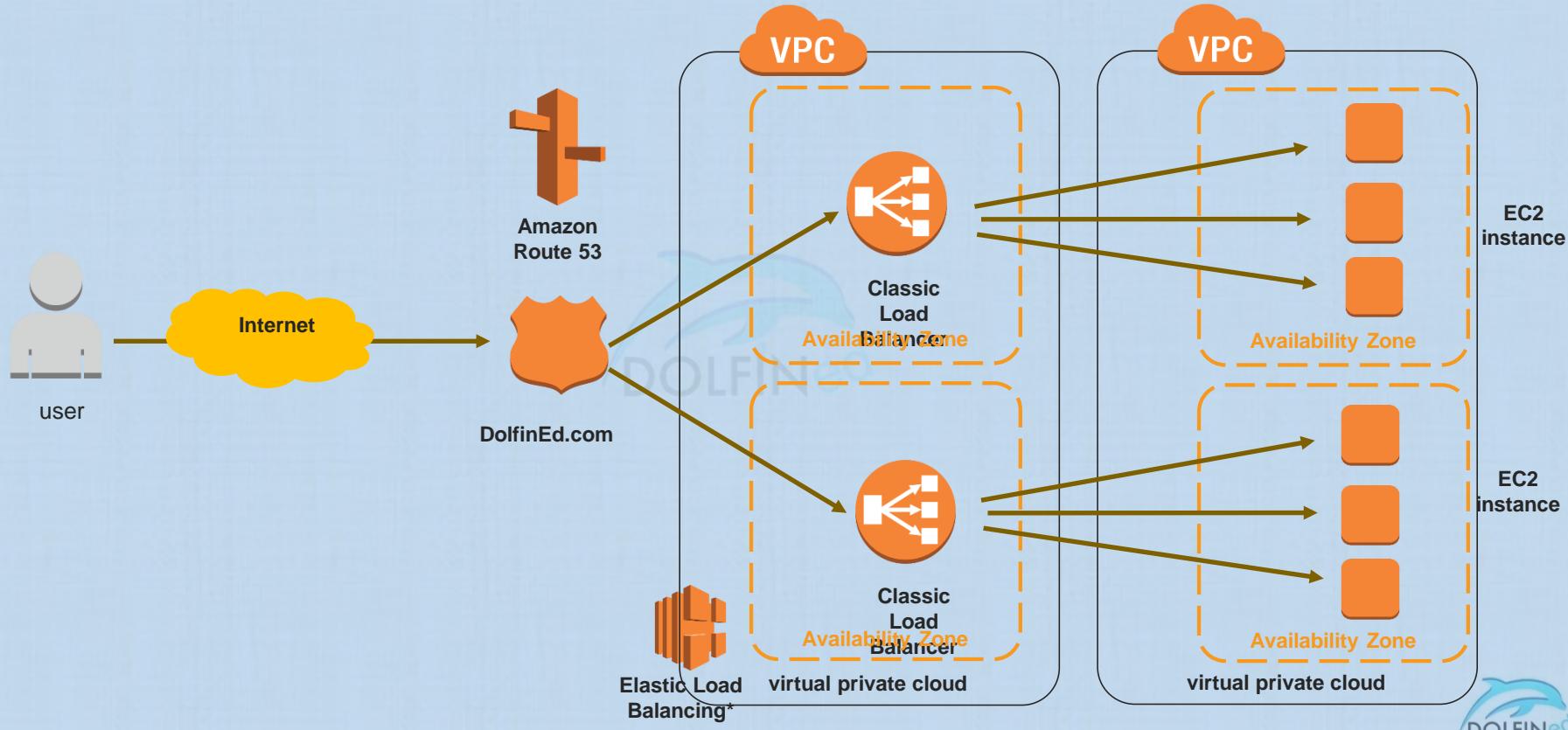
APPLICATION LOAD BALANCER (ALB)

You Can Do It Too!



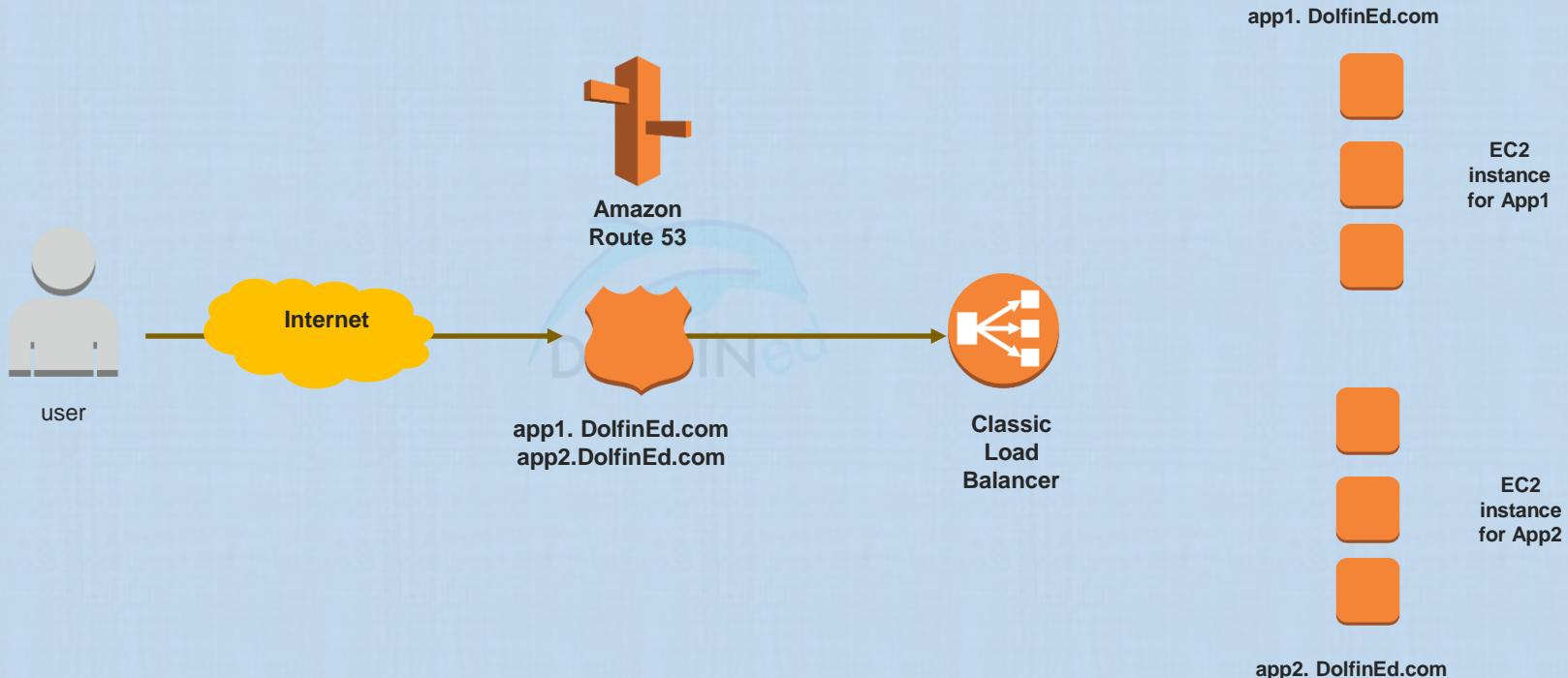
AWS Application Load Balancer

AWS – Classic Load Balancer – High Availability



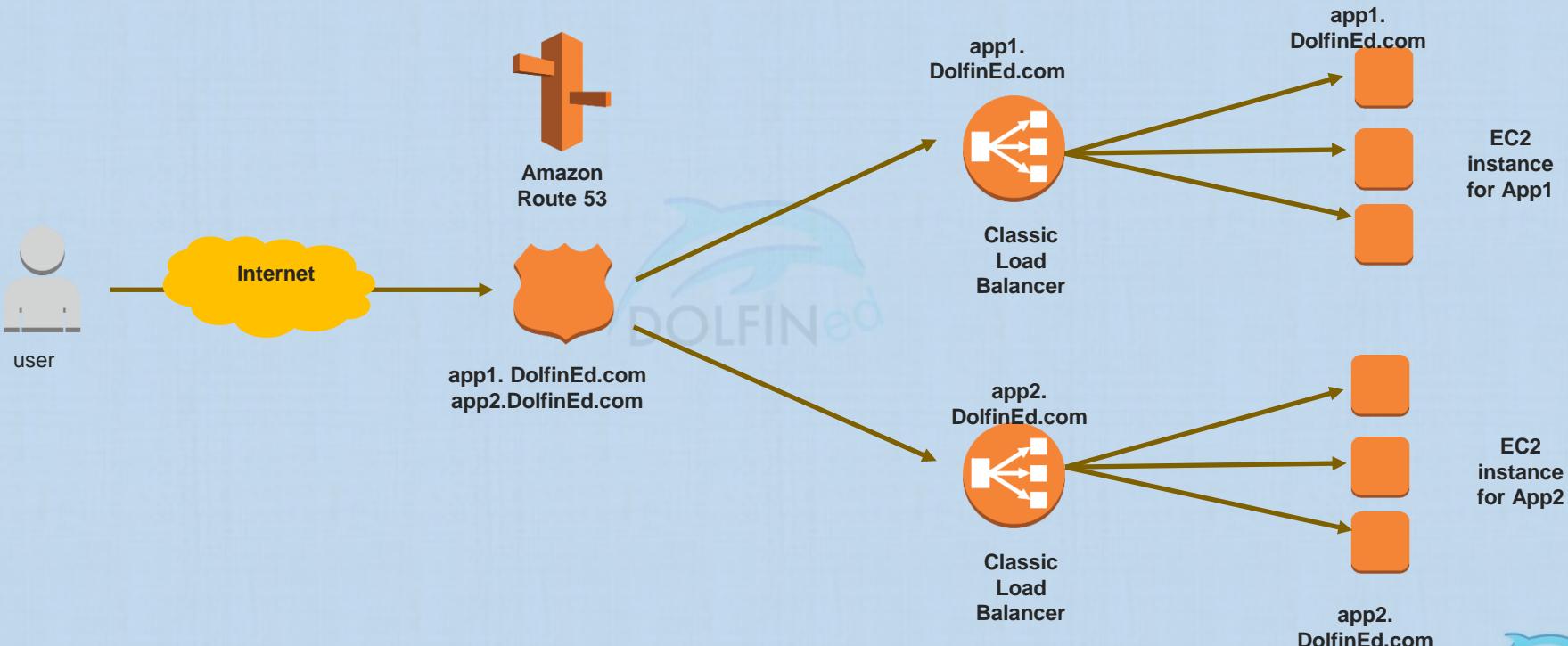
AWS Application Load Balancer

AWS – Classic Load Balancer – Splitting EC2 instance Fleet Limitation



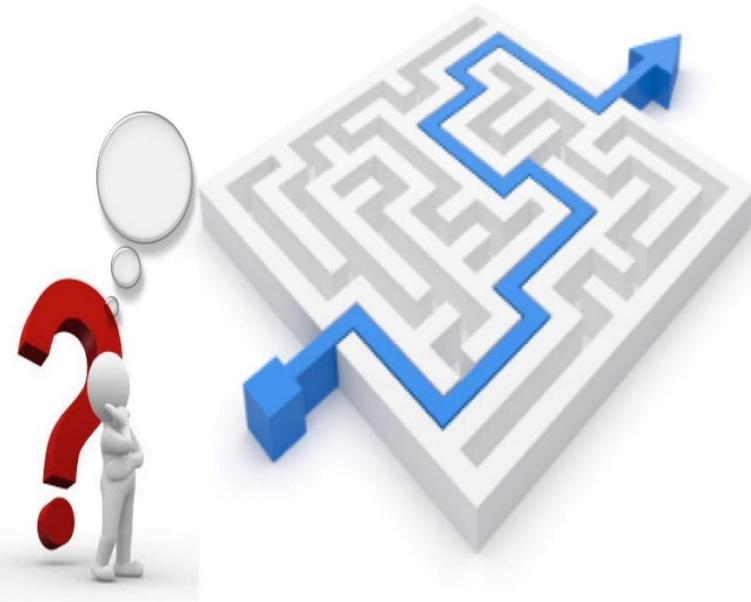
AWS Application Load Balancer

AWS – Classic Load Balancer – Splitting EC2 instance Fleet Limitation



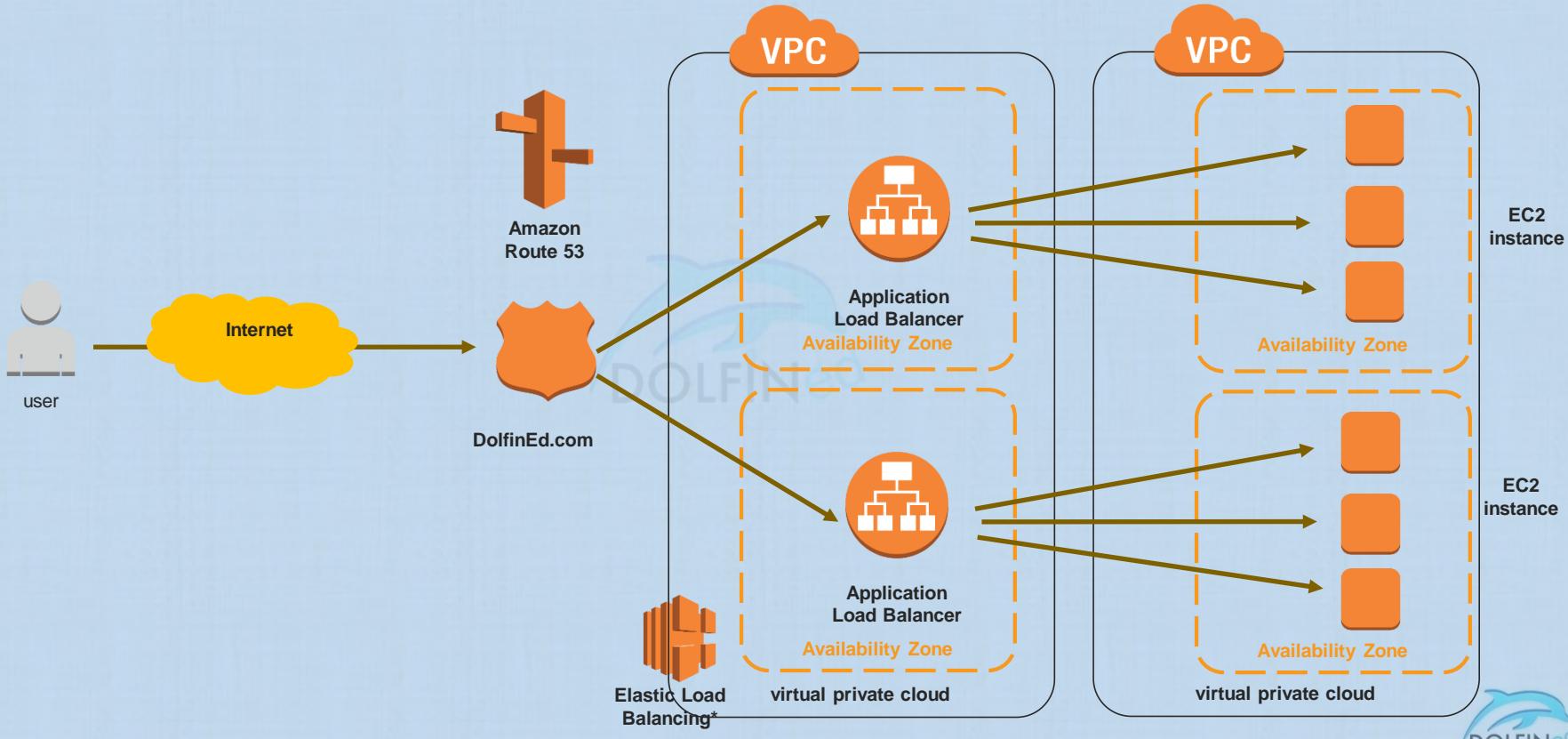
AWS ALB

AWS Application Load Balancer (ALB) Introduction



AWS Application Load Balancer

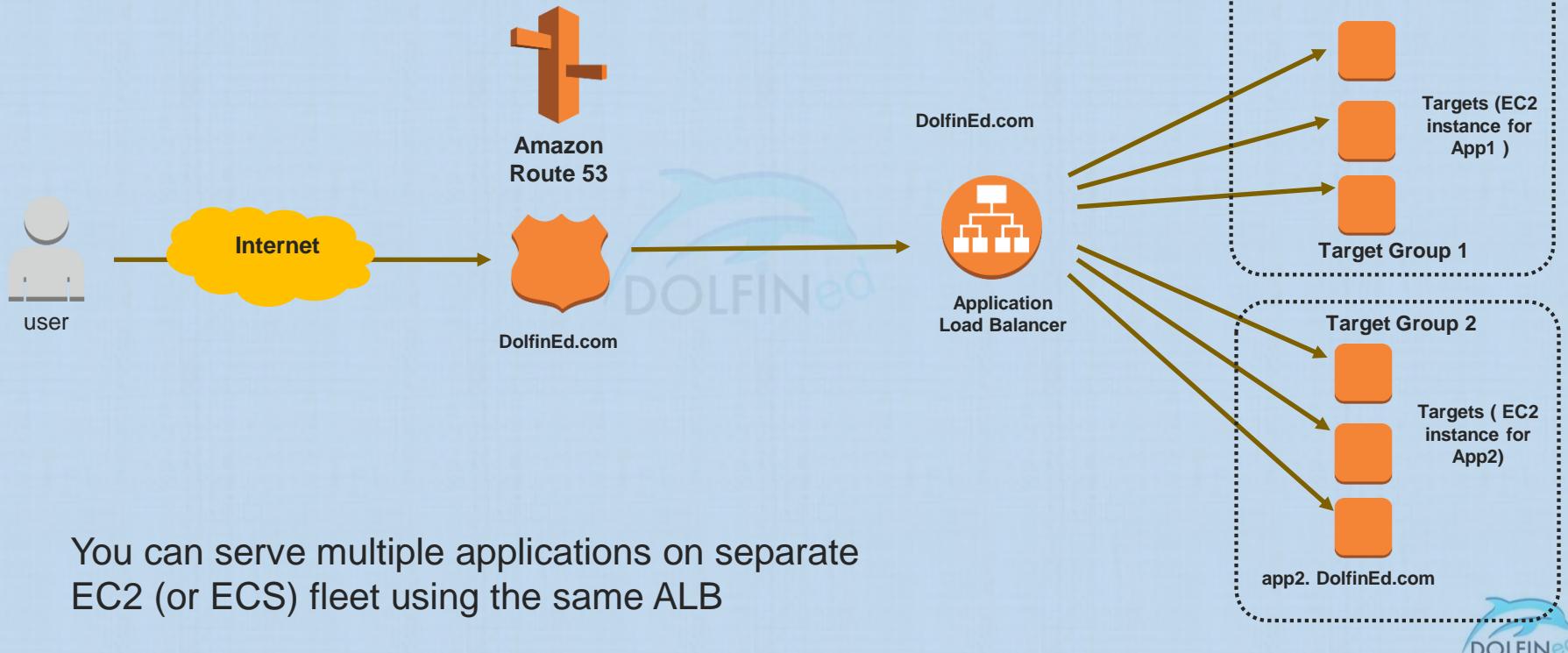
AWS – Application Load Balancer – High Availability



AWS Application Load Balancer

DOLFINED ©

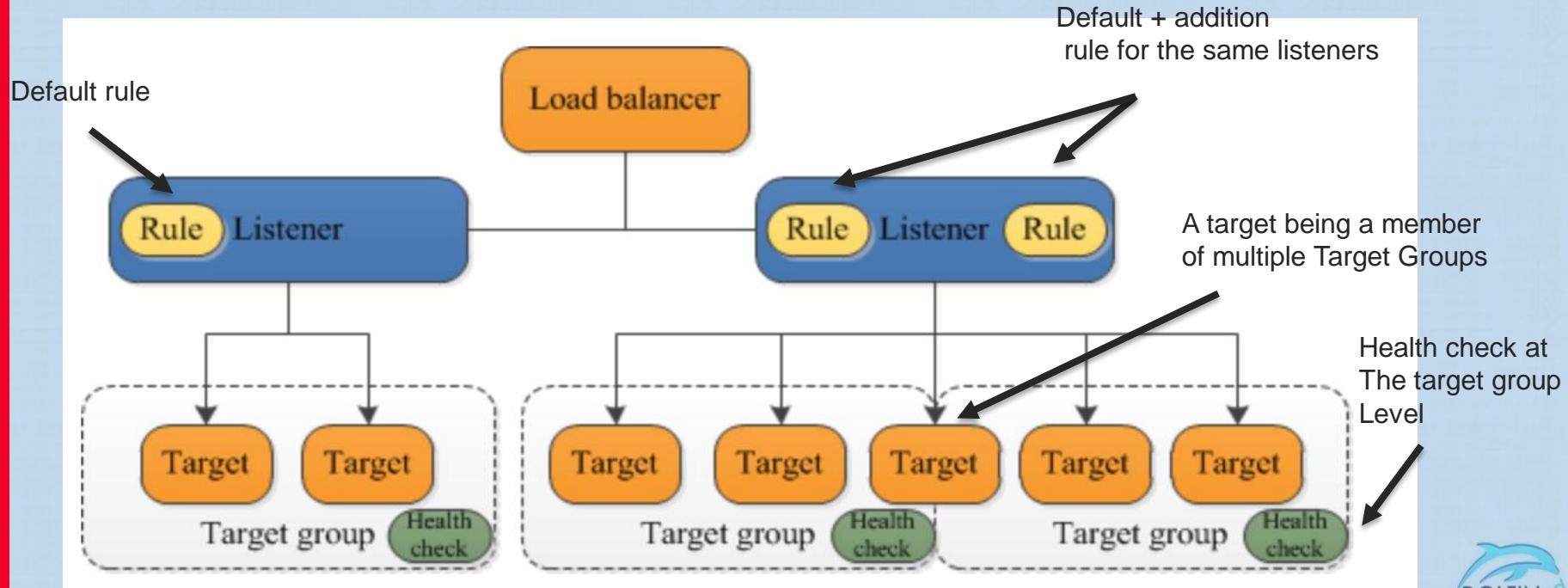
AWS – Application Load Balancer (ALB) – One ALB, Multiple Applications



AWS Application Load Balancer

AWS ALB

- An Application Load Balancer functions at the application layer, the seventh layer of the Open Systems Interconnection (OSI) model.
 - It supports HTTP, HTTPS, HTTP/2, and WebSockets



AWS Application Load Balancer

AWS ALB Components

- ALB
- Listeners
- Target Groups
- Targets
- Rules (Condition, Action, Priority)



AWS Application Load Balancer

AWS ALB Components – Target Groups

Target Groups:

- Are regional constructs (confined to a region)
- A Target Group is a logical grouping of Targets
- Each target group can be associated **with only one load balancer.**
- AS Groups can scale each target group individually
- The **target group** is used to route requests to registered **targets** as part of an action for a rule.
- Health checks can be configured per target group.
- An ALB can route to multiple target groups.
- You define **one Protocol and one port per target group** which will be used to route/forward traffic to the registered targets.
- They can exist independently from the ALB



AWS Application Load Balancer

AWS ALB - Targets

Targets:

- **Targets** specify the endpoints and are registered with the ALB as part of a target group.
- Targets can be EC2 instances, a Microservice, and Application on an ECS Container, or IP addresses
 - You can't specify public internet-routable IP addresses as targets.
- An EC2 instance can be registered with the same target group multiple times using multiple ports
- You can register a target with multiple target groups.
- You can add and remove targets from your load balancer as your needs change, without disrupting the overall flow of requests to your application.



AWS Application Load Balancer

AWS ALB – Targets – Type : IP address

- You can use IP addresses as targets to register:
 - Instances in a peered VPC,
 - AWS resources that are addressable by IP address and port (for example, databases),
 - On-premises resources linked to AWS through AWS Direct Connect or a VPN connection.
- You can register each EC2 instance or IP address with the same target group multiple times using different ports, which enables the load balancer to route requests to microservices.
- If you specify targets using an instance ID, traffic is routed to instances using the primary private IP address specified in the primary network interface for the instance.
- If you specify targets using IP addresses, you can route traffic to an instance using any private IP address from one or more network interfaces.
 - This enables multiple applications on an instance to use the same port.



AWS Application Load Balancer

AWS ALB Components – Target Groups & Targets

- You CAN NOT mix targets of different types in one target group, i.e you can not mix EC2 with ECS and/or IP targets in one target group
 - You need to keep the endpoint type homogenous in each group
- You can configure health checks on a per target group basis.
 - Health checks are performed on all targets registered to a target group that is specified in a listener rule for your load balancer.
- By default, the load balancer sends requests to registered targets using the port and protocol that you specified for the target group.
 - You can override this port when you register each target with the target group.

AWS Application Load Balancer

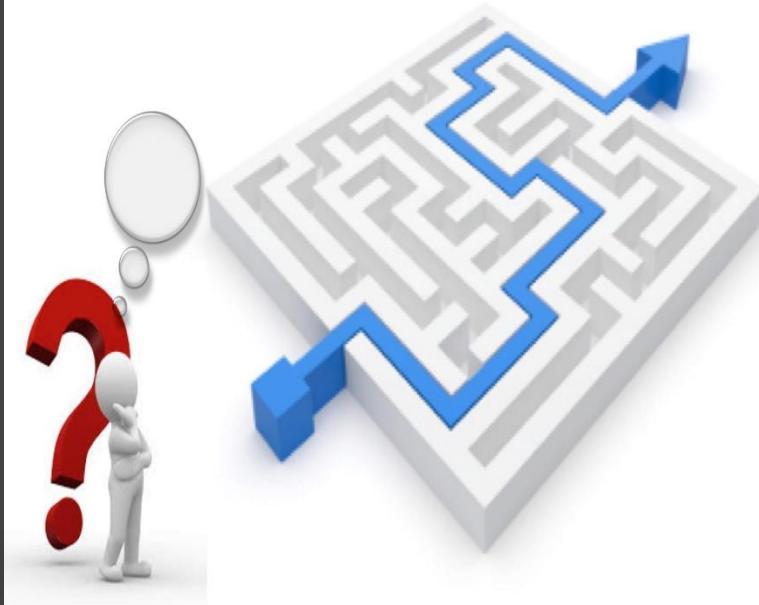
AWS ALB Components – Target Groups & Targets

- You can delete a target group if it is not referenced by any actions.
- Deleting a target group does not affect the targets registered with the target group. If you no longer need a registered EC2 instance, you can stop or terminate it.



AWS ALB

AWS Application Components – Routing Rules & Request Routing in ALB



AWS Application Load Balancer

AWS ALB Components – Rules (or Routing Rules)

Rules provide a link between listeners and target groups and consist of conditions and actions.

- Rules determine what action is taken when a rule matches a client request.
- Rules are defined on listeners
- Each rule consists of a priority, action, optional host condition, and optional path condition
- Each rule specifies a (optional) **condition**, **target group**, **action**, and a **priority**.
 - When the condition is met, the traffic is forwarded to the target group.
- You must define a default rule for each listener, and you can add rules that specify different target groups based on the content of the request (**also known as *content-based routing***).
- If no rules are found, the request will follow the default rule, which forwards the request to the default target group.



AWS Application Load Balancer

AWS ALB – Listener Rules

Rule Priority

- Each rule has a priority.
- Rules are evaluated in priority order, from the lowest value to the highest value.
- The default rule is evaluated last.
- You can change the priority of a non-default rule at any time.
- You cannot change the priority of the default rule.

Rule Actions

- Each rule action has a type and a target group.
- You can change the target group for a rule at any time.



AWS Application Load Balancer

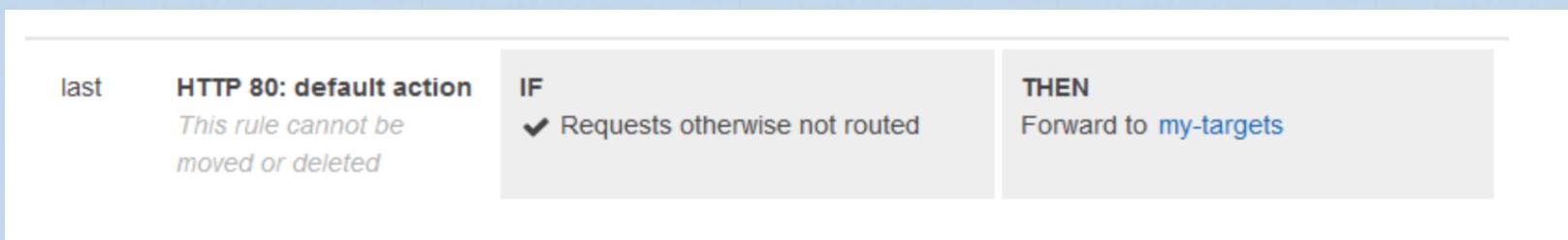
AWS ALB – Default Listener Rule

Listener Rules

- Each listener has a default rule, and you can optionally define additional rules.

Default Rules (can not be deleted)

- When you create a listener, you define an action for the default rule.
- Default rules can't have conditions.
- You can delete the non-default rules for a listener at any time. When you delete a listener, all its rules are deleted.
- If no conditions for any of a listener's rules are met, then the action for the default rule is taken.



AWS Application Load Balancer

AWS ALB Components – Rules Conditions

Rule Conditions

- There are two types of rule conditions: host and path. When the conditions for a rule are met, then its action is taken.
- Each Rule can have up to 2 conditions, 1 path condition and 1 Host condition
- (optional) **condition** is the path pattern you want the ALB to evaluate in order for it to route requests.



AWS ALB

AWS Application Load Balancer (ALB) – Content-based Routing



AWS Application Load Balancer

AWS ALB – Content-based Routing

- **Content-Based Routing**
Application Load Balancer can route a request to a service based on the content of the request.
- Two types of content routing are supported on the ALB, they are host-based and path-based
- **Host-based (Domain name based) Routing**
You can create ALB rules to route a client request based on the domain name Host field of the HTTP header allowing you to route to multiple domains from the same load balancer.
 - HTTP request – Host field:
 - Requests to blog.dolfinEd.com can be sent to a target group, while requests to content.dolfinEd.com are sent to another.



AWS Application Load Balancer

AWS ALB – Path based Routing

Using Path-based Routing

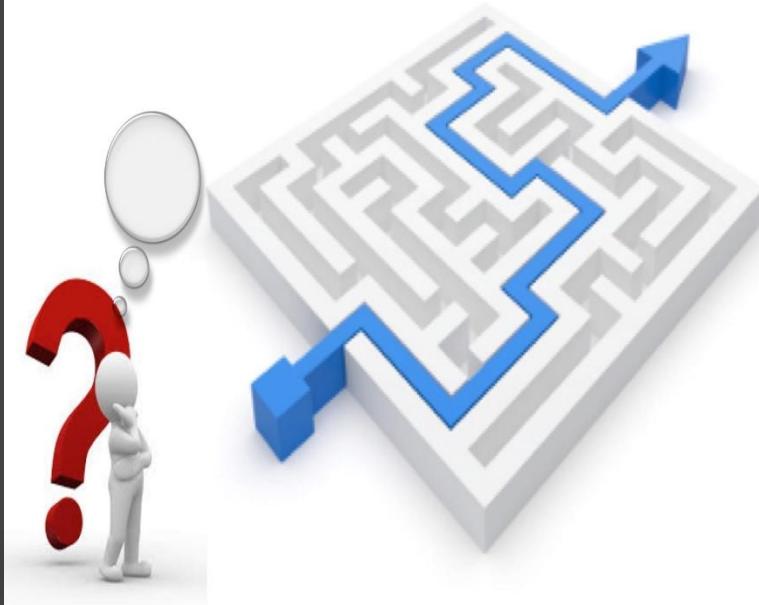
You can route a client request based on the URL path of the HTTP header.

- It routes incoming HTTP and HTTPS traffic based on the path element of the URL in the request.
- This path-based routing allows you to route requests to, for example, **/images** to one target group ,and **/videos** to another target group.
- Segmenting your traffic in this way gives you the ability to control the processing environment for each category of requests.
 - Perhaps **/images** requests are best processed on a specific type of EC2 instances, while **/videos** requests are best handled by Graphics Optimized instances.
- You can also create rules that combine host-based routing and path-based routing.
 - This would allow you to route requests to **images.example.com/thumbnails** and **images.example.com/production** to distinct target groups.



AWS ALB

AWS Application Load Balancer (ALB) – Features and Comparison to CLB



AWS Application Load Balancer

AWS ALB Components - Listeners

- WebSockets protocol support:
 - Application Load Balancers provide native support for Websockets.
 - You can use WebSockets with both HTTP and HTTPS listeners.
 - Websockets allow for full duplex communication
 - Websockets protocol support is enabled by default
 - **CLB does not support it**
- **HTTP/2 Support:**
 - HTTP/2 allow multiple requests at the same time
 - HTTP/2 is supported by default



AWS Application Load Balancer

AWS ALB – Features

- Cross zone load balancing is enabled by default
- Supports enhanced health checks and enhanced CloudWatch metrics
- ALB provides health check improvements that allow detailed error codes from 200-399 to be configured
- ALB provides additional information in Access Logs compared to CLB
- Internet facing ALB supports IPv4 and DualStack
 - However, the ALB will communicate with the Targets using IPv4
- Internal ALB uses IPv4 only (no dual stack support yet)



AWS Application Load Balancer

AWS ALB

- **Web Application Firewall (WAF) Support**
 - You can use AWS WAF with your Application Load Balancer to allow or block requests based on the rules in a web access control list (web ACL).



AWS Application Load Balancer

AWS ALB

- **ALB does not support backend server authentication**
 - Back-end Server Authentication enables authentication of the instances.
 - Load balancer communicates with an instance only if the public key that the instance presents to the load balancer matches a public key in the authentication policy for the load balancer.
 - **CLB does**
- A nice read about this in the following two links
 - <https://kevin.burke.dev/kevin/amazons-albs-insecure-internal-traffic/>
 - <https://kevin.burke.dev/kevin/aws-alb-validation-tls-reply/>



AWS Application Load Balancer

AWS ALB - SNI

- ALB supports Server Name Indication (SNI) certificates.
- You can serve multiple TLS secured applications (multiple domains) by the ALB, each with its own certificate.
- You can bind multiple certificates to the same secure listener of the ALB
- Integrates with AWS ACM
- ALB will choose the right certificates depending on the client request
- If the hostname indicated by a client matches multiple certificates, the load balancer determines the best certificate to use based on multiple factors including the capabilities of the client.



AWS Application Load Balancer

AWS ALB

Deregistration Delay (Connection Draining)

- Elastic Load Balancing stops sending requests to targets that are deregistering.
- By default, Elastic Load Balancing waits 300 seconds before completing the deregistration process, which can help in-flight requests to the target to complete.



AWS Application Load Balancer

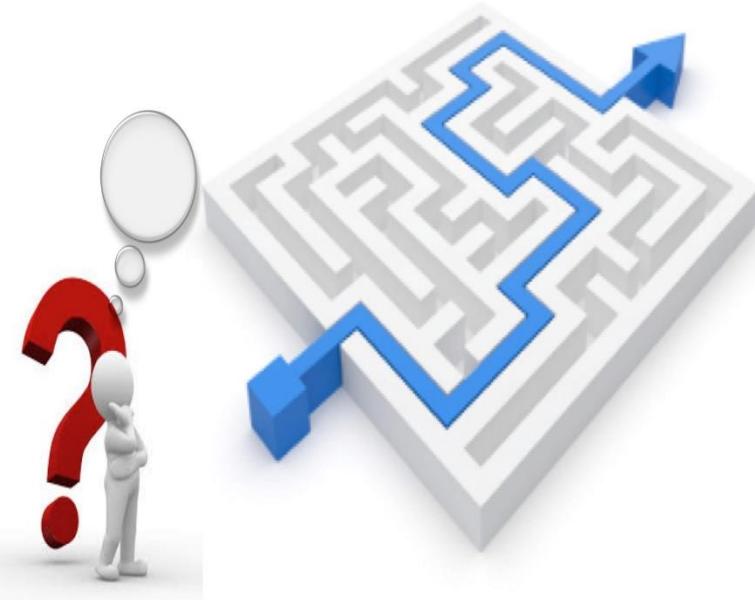
AWS ALB – Sticky Sessions

- To use sticky sessions, the clients must support cookies.
- Application Load Balancers support load balancer-generated cookies only.
 - The name of the cookie is AWSALB.
 - The contents of these cookies are encrypted using a rotating key.
 - You cannot decrypt or modify load balancer-generated cookies.
- WebSockets connections are inherently sticky.
 - After the WebSockets upgrade is complete, cookie-based stickiness is not used.
- You enable sticky sessions at the target group level.



AWS ALB

AWS Application Load Balancer (ALB) - Monitoring



AWS ALB - Monitoring

The following features can be used to monitor your load balancers, analyze traffic patterns, and troubleshoot issues with your load balancers and targets.

- **CloudWatch metrics**
 - Published every 1 minute if there are requests flowing through the ALB
- **Access logs**
 - You can use access logs to capture detailed information about the requests made to your load balancer and store them as log files in Amazon S3. You can use these access logs to analyze traffic patterns and to troubleshoot issues with your targets.

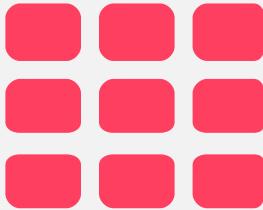


AWS ALB - Monitoring

Monitor Your Application Load Balancers

- Request tracing
 - You can use request tracing to track HTTP requests.
 - The load balancer adds a header with a trace identifier to each request it receives.
- CloudTrail logs
 - You can use AWS CloudTrail to capture detailed information about the calls made to the Elastic Load Balancing API and store them as log files in Amazon S3.
 - You can use these CloudTrail logs to determine which calls were made, the source IP address where the call came from, who made the call, when the call was made, and so on.





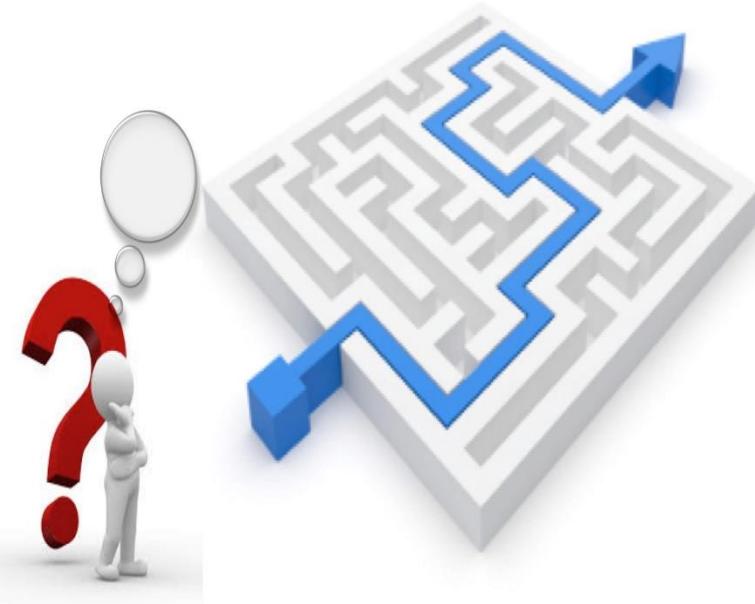
NETWORK LOAD BALANCER (NLB)

You Can Do It Too!



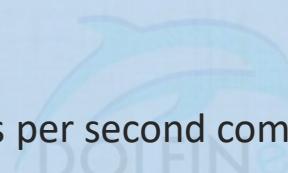
AWS NLB

AWS Network Load Balancer (NLB) – Features and How it Works



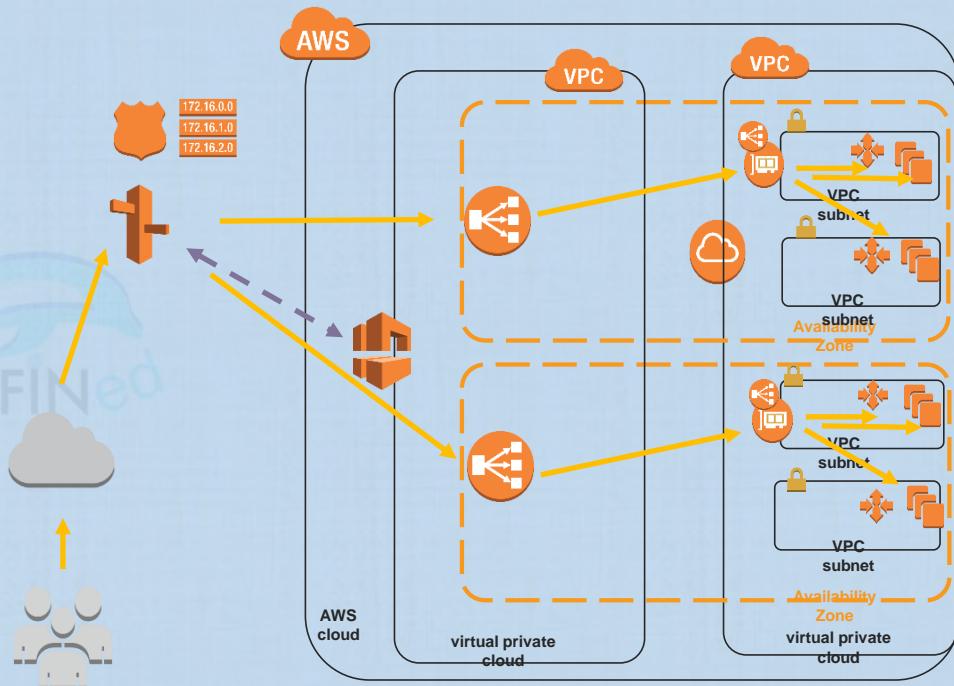
NLB Features and how it works

- A Network Load Balancer operates at the Transport layer (Layer 4) only of the OSI model.
- Supports TCP and TLS listeners for the client requests
- Supports UDP Traffic
- NLB has a higher connection rates per second compared to other ELBs, it can handle millions of requests per second.
 - Provides much lower latencies compared to other ELBs



NLB Features and how it works

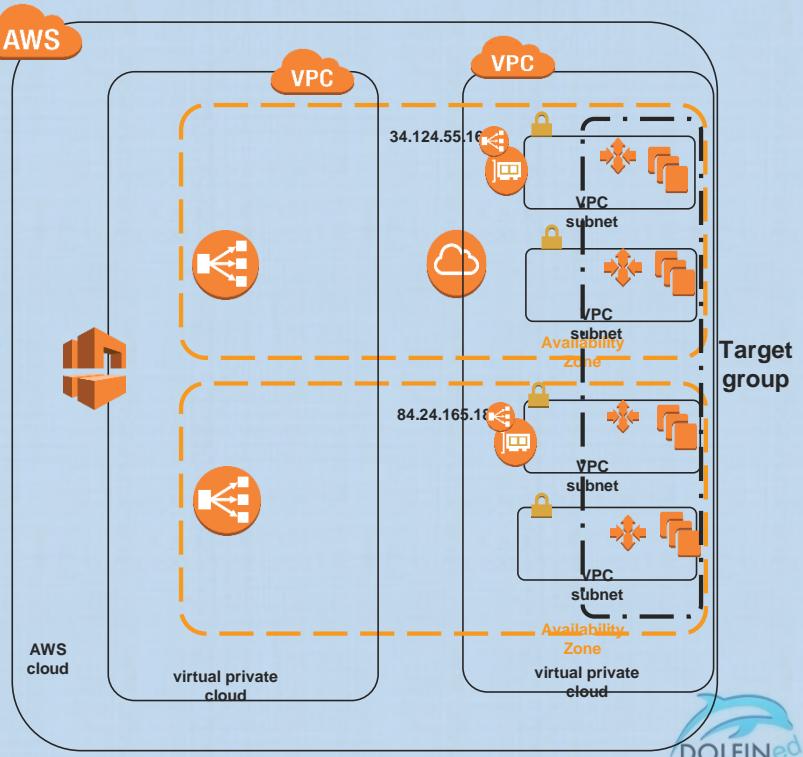
- After the load balancer receives a connection request, it selects a target from the target group for the default rule.
- It attempts to open a TCP connection to the selected target on the port specified in the listener configuration.



NLB Static IP (and EIP) support per AZ

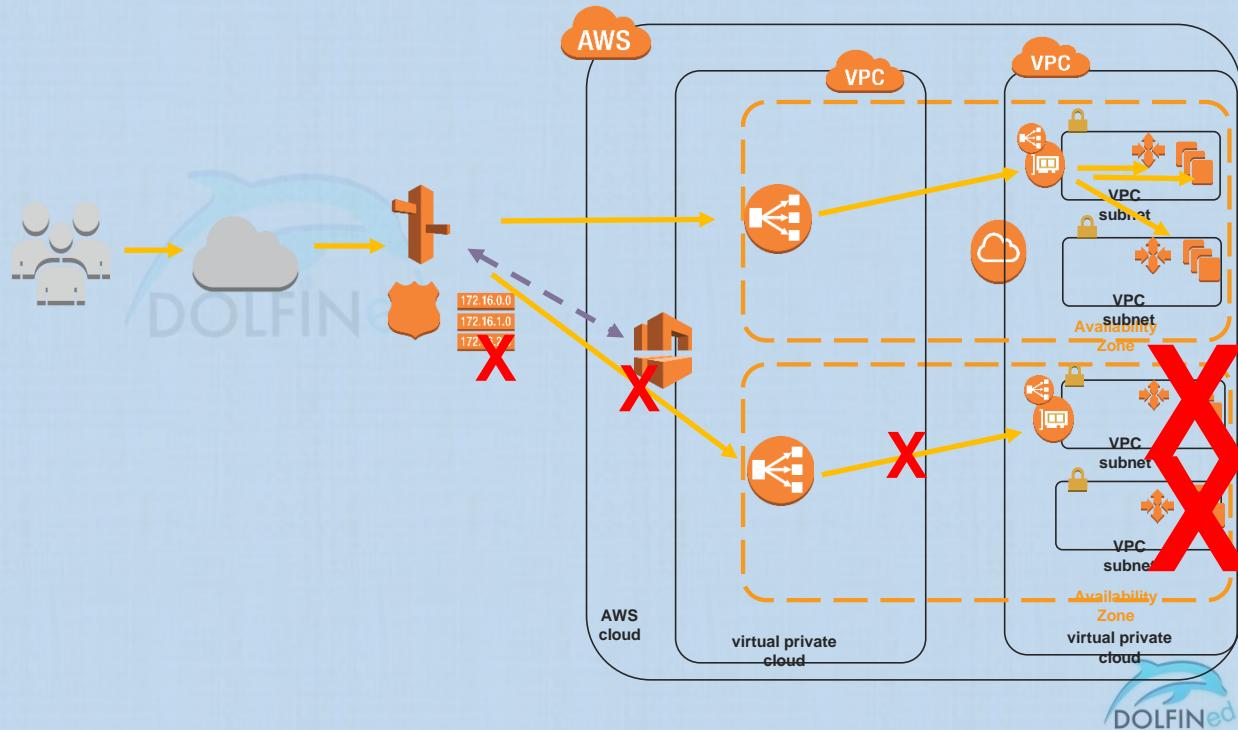
Support for **static IP addresses** for the load balancer.

- You can optionally associate one Elastic IP address per NLB enabled subnet, as a static IPv4 address for the NLB node in that subnet.
- Any connections/requests sent to the NLB's IP address will spread traffic across the instances in all the VPC subnets in the AZ.
- You can also specify an existing Elastic IP for each AZ for even greater control.
- can be used in situations where:
 - IP addresses need to be hard-coded into DNS records,
 - Customer firewall rules (whitelisting) or similar needs.



NLB and Multi-AZ

- If an AZ enabled for load balancing has no healthy targets, then its subnet will be removed from the DNS entries for that ELB instance, and no traffic will be resolving to that subnet.



NLB Features and how it works

- Unlike other ELB types, NLB supports TCP and UDP
- Support for routing requests to multiple applications on a single EC2 instance.
 - You can register each instance or IP address with the same target group using multiple ports.
- NLB Supports load balancing to ECS containers.
 - If you deploy multiple ECS services on an ECS instance, each receiving/listening on a different port, as in ALB, You can register the instance with the target group multiple times, each time with a different port from the ports used by the ECS services on the Instance.



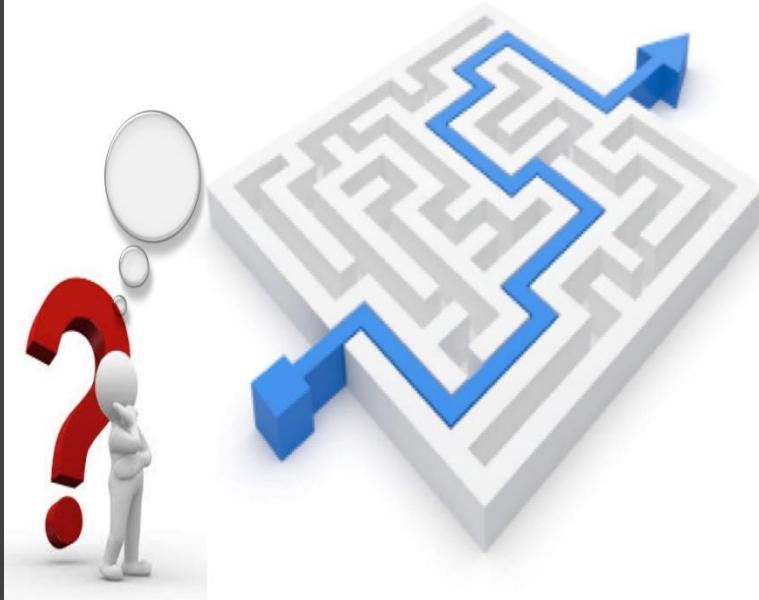
NLB Features and how it works

- NLB supports monitoring the health of each service independently,
 - Health checks are defined at the target group level
 - Many CloudWatch metrics are also supported and reported at the target group level.
- NLB can be used with Auto Scaling to achieve dynamic scaling of targets/services.
 - You can do this if you are registering targets by instance ID, not by IP address.
- Network Load Balancers support connections from clients over:
 - VPC peering,
 - AWS VPNs, and
 - Third-party VPN solutions.



AWS NLB

Multi-AZ, TLS Listeners, Target Types



NLB in Multi-AZ

- You enable one or more Availability Zones for your load balancer when you create it.
- You cannot enable or disable Availability Zones for a Network Load Balancer after you create it.
- Access logs, Delete protection, and Cross Zone load balancing are disabled by default on NLB

Hosted zone: Z26RNL4JYFTOTI

Creation time: March 28, 2019 at 11:18:57 AM UTC-7

Attributes

Deletion protection	Disabled
Cross-Zone Load Balancing	Disabled
Access logs	Disabled

Edit attributes



TLS Listeners

- If the listener protocol is TLS, you must deploy exactly one SSL server certificate on the listener.
 - The certificate can be from ACM, uploaded to ACM, or IAM
- You can use WebSockets with your listeners.
- When you create a listener, you specify a rule for routing requests. This rule forwards requests to the specified target group.



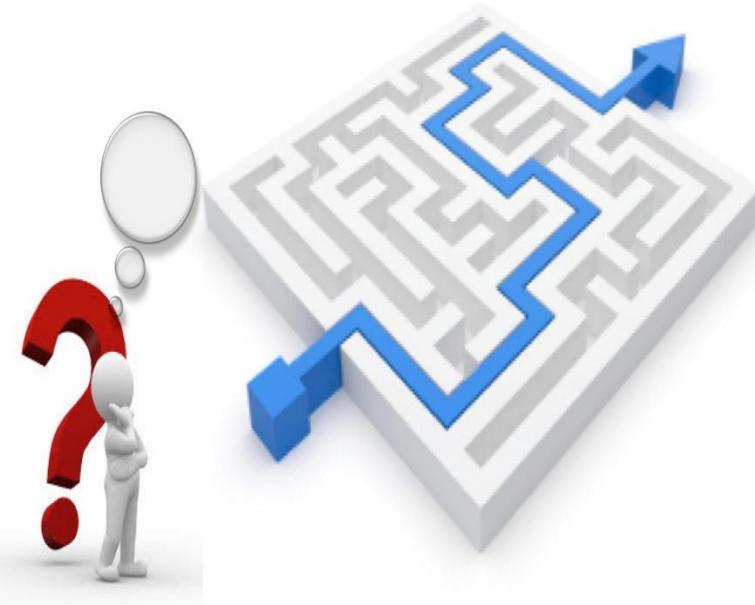
Target Types

- NLB supports the following target types:
 - IP Address: The Targets by IP address can be outside of the VPC.
 - Instance ID (Instances in a peered VPC must be referenced by IP not Instance ID)
 - You can either have all targets referenced by EC2 IDs or by IP Addresses, but not a mix and match in the same NLB
- If the target type is IP, it can be from one of the following CIDR ranges:
 - The subnets of the VPC for the target group
 - RFS1918 ranges: 10.0.0.0/8 , 172.16.0.0/12 , 192.168.0.0/16
 - 100.64.0.0/10 (RFC 6598)
- As in ALB, **target can not** be a publicly routable IP addresses.
- NLBs do not support the lambda target type,



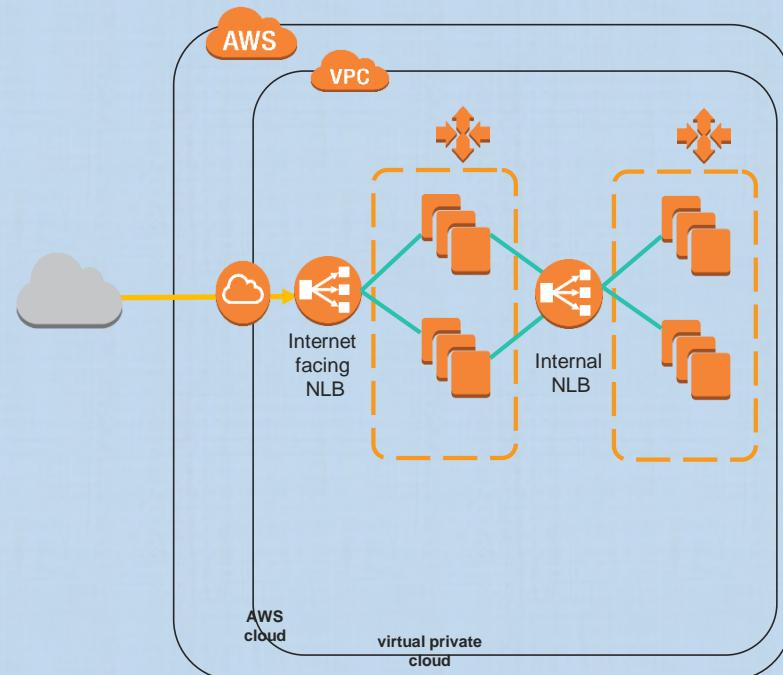
AWS NLB

Source IP address Preservation



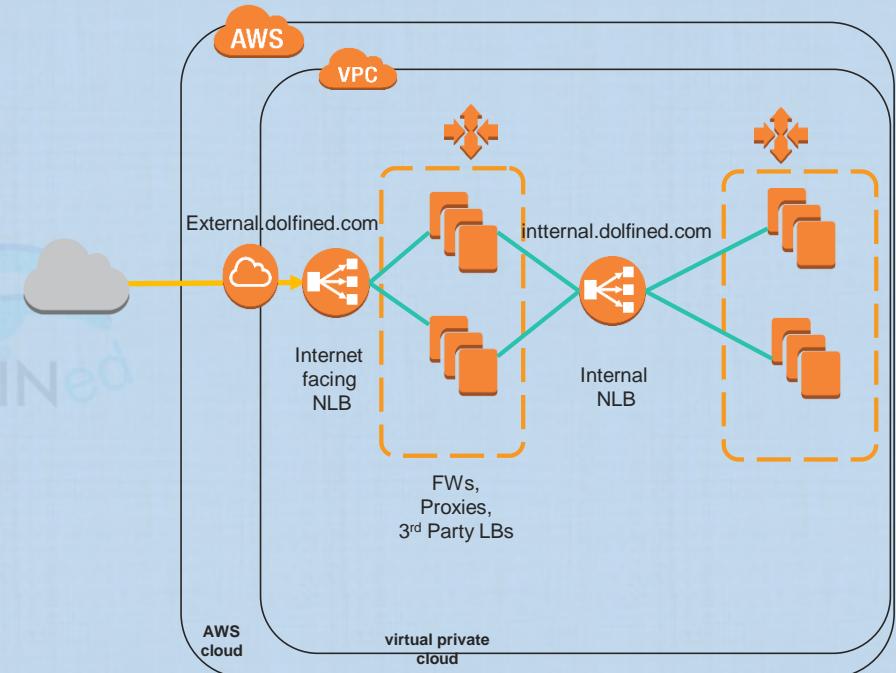
Client Source IP address Preservation

- If you use Instance ID as target type, NLB preserves the clients source IP addresses, and provides them to the targets.
- On the other hand, If you use the IP address as the target, the source IP addresses are the private IP addresses of the load balancer nodes.
 - In this case if your applications require the Clients' source IP address, you can configure Proxy Protocol on the NLB.



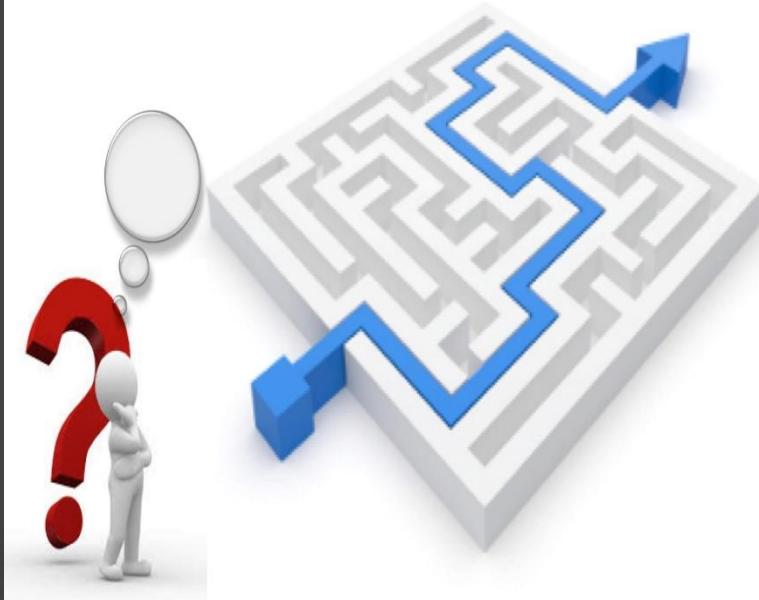
Client Source IP address Preservation

- Two load balancers with a services layer sandwiched between them
- The services layer can have NGFWs, Proxies, or 3rd Party load balancers, or even a web tier
- This way you can scale this layer while using fewer Elastic IP addresses (or static IPs)
- Preserving source IP addresses helps Geo-Location services if this is a FW layer (or 3rd Party LBs), and Whitelisting



AWS NLB

Proxy Protocol Support, Health Checks



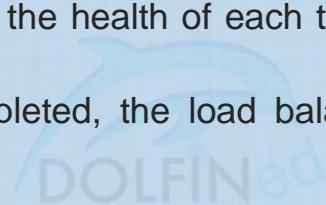
Proxy Protocol

- NLB supports Proxy Protocol v2
 - Is configurable at the Target group level. It is disabled by default
- This comes in handy in case targets are referenced by IP address (Target types) and the clients' source IP address is required for the applications.
 - The application must be able to parse that header from the packets.
- When enabled, the NLB prepends a proxy protocol header to the TCP data.
 - The NLB will not discard or overwrite any existing data, including any proxy protocol headers sent by the client or any other proxies, load balancers, or servers in the network path.
- If the traffic/requests are coming to the NLB from ELB service consumers through a VPC endpoint service,
 - The source IP addresses provided to your applications is the NLB nodes' private IP addresses.
 - Use Proxy protocol if your applications needs the IP addresses of the AWS service consumers



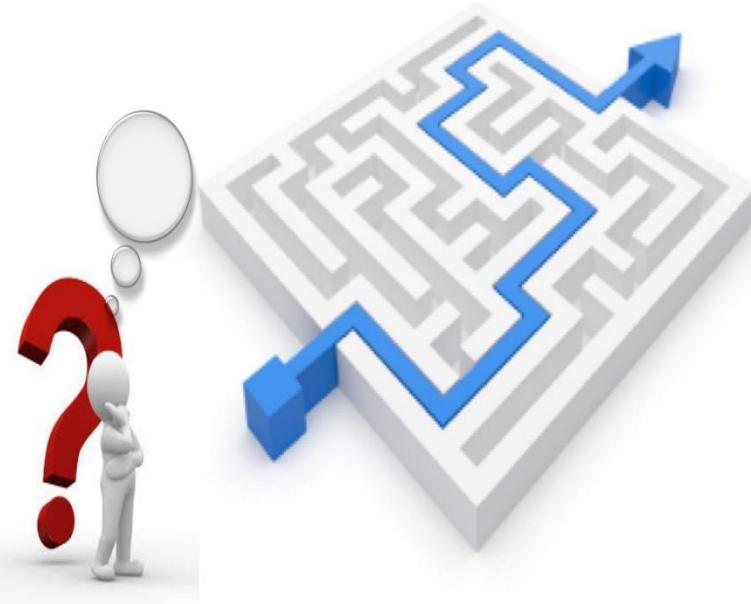
Health Checks

- Network Load Balancers use **active and passive (network) health checks** to determine whether a target is available to handle requests.
- With **active health checks**, the load balancer periodically sends a request to each registered target to check its status.
 - Each load balancer node checks the health of each target, using the health check settings for the respective target group.
 - After each health check is completed, the load balancer node closes the connection that was established for the health check.
- With **passive (Network) health checks**, the load balancer observes how targets respond to connections.
 - Passive health checks enable the load balancer to detect an unhealthy target before it is reported as unhealthy by the active health checks.
 - You cannot disable, configure, or monitor passive health checks.



AWS NLB

Monitoring the NLB



Monitoring the NLB – CloudWatch Metrics

- ELB service publishes data points to Amazon CloudWatch for load balancers and targets.
- CW metrics can be used to verify that the NLB is performing as expected.
 - And create alarms to monitor specific metrics
- ELB reports metrics to CloudWatch every one minute, but only when requests are flowing through the load balancer.
- Like the other ELBs, nothing is reported if there are no requests flowing through the load balancer, or no data for a metric.



Monitoring the NLB – VPC Flow Logs

- VPC Flow Logs can be used to capture detailed information about the traffic going to and from your NLB.
- Create a flow log for each network interface for your load balancer.
 - There is one network interface per load balancer subnet.
- If the target is registered by instance ID, the connection appears to the instance as a connection from the client.
 - Flow logs would also show NACL and Security Groups (Of the backend EC2 instances/Targets) Accept/Reject messages for the different load balancer traffic.
 - **Remember, NLB does not support security groups.**



Monitoring the NLB – Access Logs

- Access logs can be used to capture detailed information about TLS requests made to the NLB.
 - The log files are stored in Amazon S3 buckets.
 - Access logs can be used to analyze traffic patterns and to troubleshoot issues with your targets.
- Access logs are created only if the load balancer has a TLS listener and they contain information only about TLS requests.
- Access logs are disabled by default. You can enable it/disable it at anytime.
- There is no additional charge for access logs. Only storage charges apply.
- ELB publishes a log file for each load balancer node every 5 minutes.
 - Log delivery is eventually consistent.



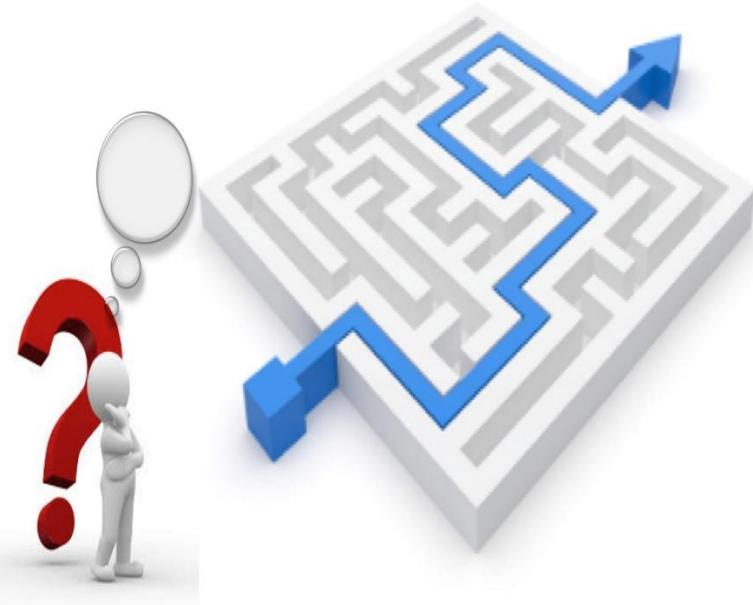
Monitoring NLB – Cloud Trail

- AWS CloudTrail can be used to capture detailed information about the calls made to the Elastic Load Balancing API and store them as log files in Amazon S3.
- These CloudTrail logs can be used to determine which calls were made, the source IP address where the call came from, who made the call, when the call was made, and so on.



AWS NLB

Perfect Forward Secrecy (PFS)

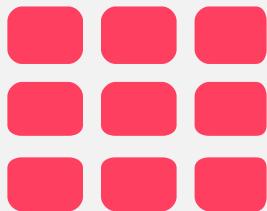


Review Topic : Elastic Load Balancing

Elastic Load Balancing and PFS

- Amazon ELB now offer advanced cipher suites that use the Elliptic Curve Diffie-Hellman Ephemeral (ECDHE) protocol.
- ECDHE allows SSL/TLS clients to provide Perfect Forward Secrecy, which uses session keys that are ephemeral and not stored anywhere.
 - This helps prevent the decoding of captured data by unauthorized third parties, even if the secret long-term key itself is compromised.





AWS AUTO SCALING

You Can Do It Too!



AWS Auto Scaling

AWS Auto Scaling Introduction



Review Topic : Auto Scaling

AWS Auto Scaling

- AWS Auto Scaling allows for the configuration of automatic scaling for the AWS resources that are part of your application very quickly.
- The AWS Auto Scaling console provides a single UI to use the automatic scaling features of multiple AWS services.
- Automatic scaling can be configured for individual resources or for whole applications.
- Scaling plans is how Auto scaling is configured and managed
 - Scaling plans uses dynamic scaling and predictive scaling to automatically scale the application resources.

Review Topic : Auto Scaling

AWS Auto Scaling

- Through scaling plan you can decide on the scaling strategies based on cost or availability optimization or both.
 - Alternatively, custom scaling strategies can be created
- AWS Auto Scaling is useful for applications that experience daily or weekly variations in traffic flow, including:
 - Cyclical traffic such as high use of resources during regular business hours and low use of resources overnight
 - On and off workload patterns, such as batch processing, testing, or periodic analysis
 - Variable traffic patterns, such as marketing campaigns with periods of spiky growth
- AWS Auto Scaling can be used to scale the following AWS resources:
 - EC2, Spot Instances, DynamoDB, Amazon Aurora, Amazon ECS



Review Topic : Auto Scaling

Application Auto Scaling

- Application Auto Scaling is a web service for developers and system administrators who need a solution for automatically scaling their scalable resources for individual AWS services beyond Amazon EC2.
- Application Auto Scaling allows you to configure automatic scaling for the following resources:
 - Amazon ECS services
 - Spot Fleet requests
 - Amazon EMR clusters
 - AppStream 2.0 fleets
 - DynamoDB tables and global secondary indexes
 - Aurora replicas
 - Amazon SageMaker endpoint variants
 - Custom resources provided by your own applications or services.
 - Amazon Comprehend document classification endpoints
 - Lambda function provisioned concurrency



Review Topic : Auto Scaling

Application Auto Scaling

- You can also use Application Auto Scaling and Amazon EC2 Auto Scaling in combination with AWS Auto Scaling to scale resources across multiple services.
- AWS Auto Scaling can help you maintain optimal availability and performance by combining predictive scaling and dynamic scaling (proactive and reactive approaches, respectively) together to scale your Amazon EC2 capacity faster.
- Application Auto Scaling allows you to automatically scale your scalable resources according to conditions that you define.
 - Target tracking scaling—Scale a resource based on a target value for a specific CloudWatch metric.
 - Step scaling— Scale a resource based on a set of scaling adjustments that vary based on the size of the alarm breach.
 - Scheduled scaling—Scale a resource based on the date and time.

EC2 Auto Scaling

AWS EC2 Auto Scaling Groups



Review Topic : Auto Scaling

EC2 Auto Scaling

- Is an AWS feature that allows your AWS compute needs (EC2 instances fleet) to grow or shrink depending on your workload requirements
- Auto scaling ensures that you have the right number of AWS EC2 instances for your needs at all times
- Auto Scaling helps you save cost by cutting down the number of EC2 instances when not needed, and scaling out to add more instances only when it is required

Review Topic : Auto Scaling Components

- **Launch Configuration:**



- Is the configuration template used to create new EC2 instances for the ASG, defines parameters like:
 - Instance family, instance type, AMI, Key pair, Block devices, and Sec groups are parameters defined in the launch configuration

- **AS Group:**



- Is a logical grouping of EC2 instances

- **Scaling Policy (Plan)**



- Determines when/if and how the ASG scales or shrinks (On-demand/Dynamic scaling, Cyclic/Scheduled scaling)



Review Topic : Auto Scaling

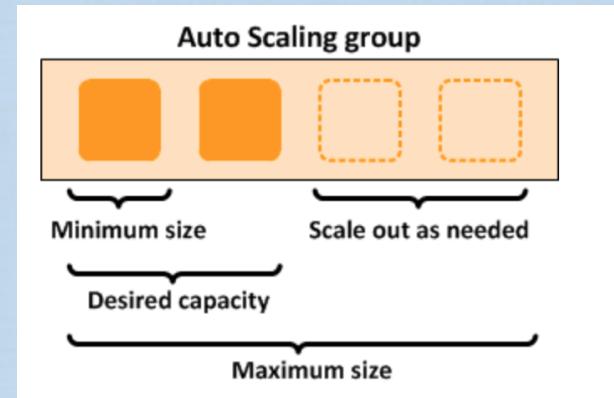
Auto Scaling Groups (ASG)

Auto Scaling components:

- AS Group:



- Is a collection of EC2 instances managed by an Auto Scaling Policy
- An ASG can have a minimum, maximum, and desired capacity of EC2 instances
- Can be edited after it is created



Source: aws.amazon.com



Review Topic : Auto Scaling

Components

Launch Configuration:

- Can be created from AWS Console or CLI
- You can create a launch configuration from scratch, **or**
- You can use an existing/running EC2 instance to create the launch configuration
 - This is provided that the AMI used to launch this instance does still exist on AWS
 - EC2 instance tags, and any additional block store volumes created after the instance launch will not be taken into account
- If you want to change your launch configurations, you have to create a new one, make the required changes, and use that with your auto scaling groups



Review Topic : Auto Scaling

EC2 Auto Scaling – Launch Templates

- Is similar to a launch configuration
- It specifies AMI ID, Instance type, Security group, tags, key pair among other parameters
- Launch templates allow you to have multiple versions of a template, which you can then reuse to create other templates or template versions.
- AWS recommend creating launch templates not to have access to all advanced EC2 features
- **Using launch templates, you can provision your capacity using on-demand and spot instances (which can not be done using launch configurations)**
 - This will allow you to achieve the desired scale, performance, and cost targets.
- You can create a template:
 - From scratch
 - Create a new version of an existing one
 - Copy parameters from a launch configuration, running instance, or another template
- Launch templates can not be changed/edited after it is created, but versions can be created
- Auto Scaling helps you save cost by cutting down the number of EC2 instances when not needed, and scaling out to add more instances only when it is required



Review Topic : Auto Scaling

Auto Scaling features

- Auto Scaling can span Multi-AZs within the same AWS region. But not across regions.
 - It can be used to create Fault Tolerant designs within a region in AWS
- Cost:
 - There is no additional cost for launching AS Groups. You pay for what you use of EC2 instances
- It works well with AWS ELB, Cloud Watch, and Cloud Trail
- AS is compliant with PCI DSS



Review Topic : Auto Scaling

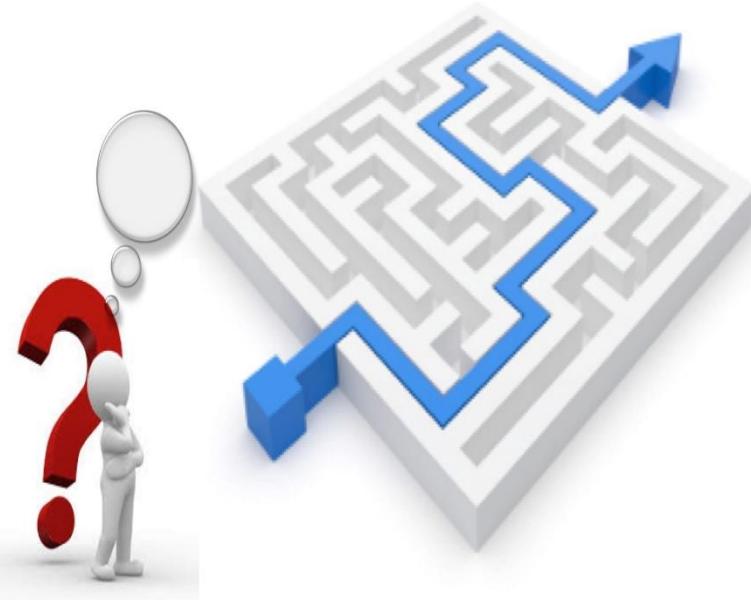
AS Features (Cont.)

- Auto Scaling can NOT span across multiple Regions
- You can determine which subnets will AS Groups use to launch new instances in each AZ
- **AZ Rebalance** - Auto Scaling service always tries to distribute EC2 instances evenly across AZs where it is enabled
 - If AS finds that the number of EC2 instances launched by an ASG into subject AZs is not balanced, AS will initiate a Re-Balancing activity
- If Auto Scaling fails to launch instances in an AZ (for AZ failure or capacity unavailability..etc), it will try in the other AZs defined for this AS Group until it succeeds



AWS Auto Scaling

**Auto Scaling & ELB Service,
Health Checks**



Review Topic : Auto Scaling

Adding an ELB to the ASG

- You can attach one or more ELBs (Classic, ALB, NLB are all supported) to your existing AS Group
 - The ELB(s) must be in the same region as the AS Group
 - Instances and the ELB(s) must be in the same VPC
- For CLB add the load balancer, for ALB & NLB add the Target group
- Once you do this, any EC2 instance existing or added by the AS Group will be automatically registered with the ASG defined ELB(s)
 - You do not need to register those instances manually on the ASG defined ELBs
 - The ELB(s) will then become the focal point for any inbound traffic destined to the ASG EC2 instances
- Auto Scaling honors connection draining configuration when de-registering an instance manually from an ELB.



Review Topic : Auto Scaling

ASG – Health Checks

- Auto Scaling classifies its EC2 instances health status as either Healthy or Unhealthy
- By default, AS uses EC2 Status Checks only to determine the health status of an Instance
- When you have one or more ELBs defined with the AS Group, you can configure Auto Scaling to use “both” the EC2 Health Checks and the ELB Health Checks to determine the Instances health status
- Health Check Grace period:
 - By default is 300 seconds
 - Is the time Auto Scaling waits from the time an Instance comes into service (become In-Service) before checking its health status
 - A value of “zero” means no grace period and the instance health is checked once it is In-service

Review Topic : Auto Scaling

ASG – Health Checks

- Until the Grace Period timer expires, any unhealthy status reported by EC2 status checks, or the ELB attached to the AS Group, will not be acted upon.
- After Grace Period expires, Auto Scaling would consider an Instance unhealthy in any of the following cases:
 - EC2 Status checks report to Auto Scaling an instance status other than running
 - If the instance status is impaired due to a host Hardware or Software Problem
 - If ELB health checks are configured to be used by the Auto Scaling, then if the ELB reports the Instance as “Out-of-Service”
 - If you have multiple ELBs attached to the AS Group, if any of them reports the EC2 instance status as “Out-of-Service”.
- One source reporting the instance as unhealthy is enough for Auto Scaling to mark it for replacement



AWS Auto Scaling

Auto Scaling Policies



Review Topic : Auto Scaling

ASG – Scaling Policies

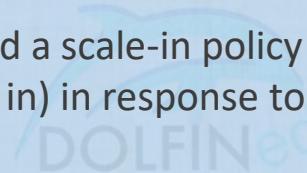
- Scaling Policies:
 - Maintain a current number of instances all the time
 - Manual Scaling
 - Manually change ASG's min/desired/max, attach/detach instances
 - Cyclic (schedule based) scaling
 - Predictable load change
 - On-demand/Dynamic (Event based) scaling
 - Scaling in response to an event/alarm
 - Predictive scaling
 - Combines AWS Auto Scaling with On Demand scaling (Proactive and reactive together)
- An ASG can have multiple policies attached to it at any time



Review Topic : Auto Scaling

ASG – Dynamic/On-demand Scaling

- Is scaling out, or in, in response to an alarm (demand)
- An alarm is an object that watches over a single metric (CPU utilization, memory, network in/out...etc)
- You need to have a scale-out and a scale-in policy configured, which will instruct Auto Scaling what to do (Scale out or in) in response to Alarms
- You can use Cloud Watch to monitor and generate the Alarms



Review Topic : Auto Scaling

Dynamic/On-Demand Types

Simple Scaling:

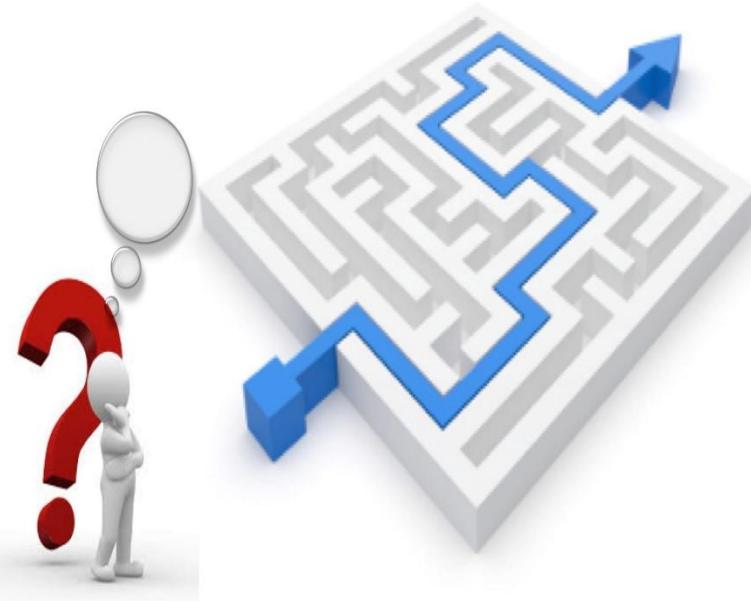
- Single adjustment (up or down) in response to an alarm
- Waits for a cool down timer to expire before responding to more alarms
 - Cool down Period:
 - Is the period of time auto scaling waits after a scaling activity (scale in or out) until the effect of the scaling activity becomes visible

Step Scaling:

- Multiple steps/adjustments
- Supports a warm-up timer
 - The time it will take a newly launched instance to be ready and contribute to the watched metric
 - Warm-up period:
 - The period of time before which a newly created EC2 instance by ASG, using step scaling, is not considered/ counted toward the ASG metrics
- **Target Tracking Scaling:** Increase or decrease the current capacity of the group based on a target value for a specific metric.

AWS Auto Scaling

Monitoring

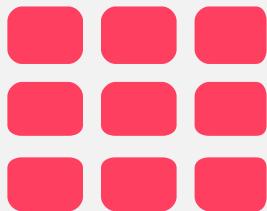


Review Topic : Auto Scaling

Monitoring Auto Scaling Groups

- AWS EC2 service sends EC2 metrics to cloud watch about the ASG instances
 - Basic Monitoring (Every 5 minutes enabled by default – free of charge)
 - You can enable detailed (every 1 minute – chargeable)





AWS RELATIONAL DATABASE SERVICE (RDS)

You Can Do It Too!



AWS RDS

Relational Database Service (RDS) - Introduction



Review Topic : Relational Database Service

Relational Databases

- Virtually all relational DBs use Structured Query Language (SQL)
- Are best suited for OLTP (On Line Transaction Processing)
 - OLTP facilitates and manages transaction oriented applications (typically used in data entry and retrieval)
 - An ATM machine transaction is an OLTP example
- Relational DBs are usually used in Enterprise applications/scenarios
 - Exception is MySQL which is used for web applications



Review Topic : Relational Database Service

AWS RDS

Is an AWS fully managed Relational DB Engine service where AWS is responsible for:

- Security and patching of the DB instances
- Automated backup for your DB instances (default setting)
- Software updates for the DB engine
- Easy scaling for storage and compute as required
- If selected, Multi-AZ with Synchronous replication between the active and standby DB instances
- Automatic failover if Multi-AZ option was selected
- Providing the ability to create DB read replicas for DB read scaling (intensive read deployments)
- AWS is NOT responsible for:
 - Managing DB Settings
 - Building a relational DB schema
 - DB performance tuning



Review Topic : Relational Database Service

Supported Relational DB engines

- MS SQL Server
- ORACLE (Two licensing models – bring your own license [BYOL] or License included)
- PostgreSQL
- MariaDB
- MySQL
- AWS Aurora (Explained in a separate section)



Review Topic : Relational Database Service

RDS instance storage

- Amazon RDS use EBS volumes (not Instance-Store) for DB and Logs storage
- General Purpose (gp2) RDS Storage:
 - Used for DB workloads with moderate I/O requirements
- Provisioned IOPS (io1) RDS Storage:
 - Used for High performance OLTP workloads
 - Provisioned IOPS storage is optimized for I/O intensive workloads that require low I/O latency and consistent throughput
 - Ex. Online Transaction Processing (OLTP) workloads that have consistent performance requirements.
- Magnetic RDS Storage:
 - Use for small DB workloads



Review Topic : Relational Database Service

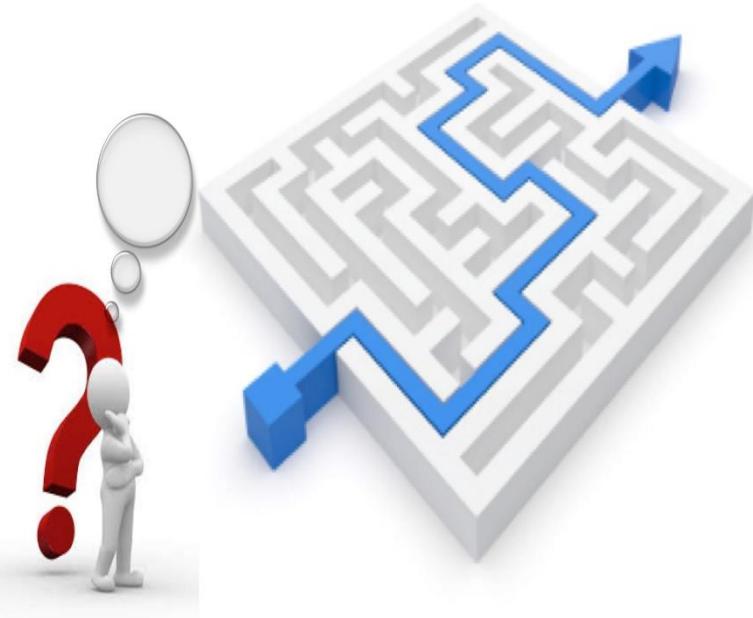
DB Subnet Group

- Is a collection of subnets in a VPC that you allocate for DB instances launched in the VPC
- Each DB subnet group must have at least one subnet in each AZ in a region
- AWS recommends, even if you are starting with standalone RDS instance, configure the subnet group with a subnet in each AZ in the region.
 - This will facilitate launching your standby instance in the subnet group when you opt for the multi-AZ deployment
- During creating your RDS instance you can select a preferred AZ, and specify which Subnet group, and subnet of that group, for your RDS DB instance
 - Then RDS service will allocate an IP address in that subnet to your RDS instance
 - And then RDS service will create an ENI, attach it to the RDS instance, and assign the above IP address to it



AWS RDS

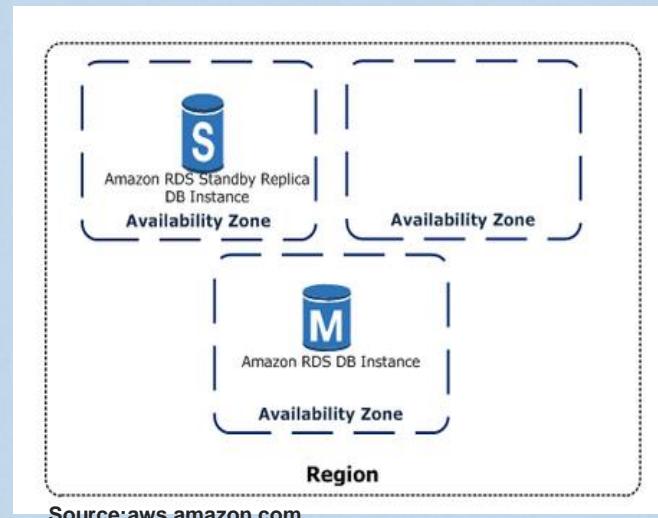
Relational Database Service (RDS) – Multi-AZ



Review Topic : Relational Database Service

Multi-AZ RDS option

- Multi-AZ for RDS provides high availability, data durability, and fault tolerance for DB instances
- Oracle, MySQL, PostgreSQL, Maria DB use AWS failover technology
- MS SQL Server uses SQL Server DB Mirroring (DBM)
- You can select the Multi-AZ option during RDS DB instance launch or modify an existing standalone RDS instance
- RDS service creates a standby instance in a different AZ in the same region, and configures “SYNCHRONOUS” replication between the primary and standby
- You can NOT read/write to the Standby RDS DB instance



Source:aws.amazon.com



Review Topic : Relational Database Service

Multi-AZ RDS – Event Notifications

- You will be alerted by a DB instance event when a failover occurs
- AWS RDS uses AWS SNS to send RDS events via SNS notifications
 - You can use API calls to the AWS RDS Service to list the RDS events in the past 14 days (DescribeEvents API), also this can be done via CLI
 - Using AWS Console, you can only view RDS events for the last 1 days (24 hours)



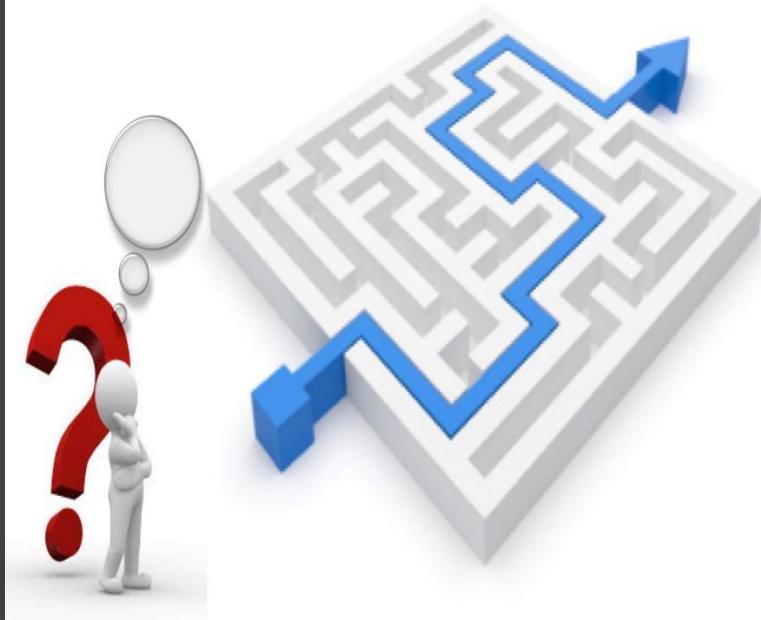
Review Topic : Relational Database Service

Processed that happen on Standby first

- The following procedures are done on standby first, then on primary
 - OS Patching
 - System upgrades
 - DB Scaling
- In Multi-AZ, Snapshots and Automated backups are done on Standby instance to avoid I/O suspension on Primary instance
- DB engine version upgrades happen on both primary and standby at the same time (causes an Outage)
- Maintenance sequence of events in Multi-AZ:
 - Maintenance on Standby is performed
 - Standby promoted to Primary
 - Maintenance performed on old primary (Current Standby)

AWS RDS

Relational Database Service (RDS) – DB backups - Automatic Backup



Review Topic : Relational Database Service

DB Automated Backups or Manual Snapshots

- There are the two methods to backup your RDS DB instances
 - AWS RDS automated backups
 - User initiated manual backups
- Either one creates a storage volume snapshot of your entire DB instance that gets saved in S3
 - Not just the individual databases
- You can make copies of automated backups and of manual snapshots. The resulting copy is considered a manual snapshot
- You can share manual snapshots, but not the automated ones
 - If you need to share an automated backup, make a copy first, then share the copy
- Retention period: AWS RDS keeps the automated backup for 7 days by default (retention period of 0 means no retention. It can be configured up to 35 days)



Review Topic : Relational Database Service

DB Automated Backups

- Automated backups are used for point-in-time DB instance recovery
- By default, RDS automatically backs up the DB instances daily, by creating a storage volume snapshot of your DB instance (full daily snapshot), including **the DB transaction logs (modifications)**,
 - You can choose when during the day this is done (Backup window)
 - No additional charge for RDS backing up your DB instances
 - Enabled by default, you can disable it by setting retention period to zero (0)
- The first snapshot is a full one, and then subsequent snapshots are incremental
- It can restore the DB up to 5 minutes in time using the DB transaction logs and the Automated snapshot



Review Topic : Relational Database Service

DB Automated Backups

- During your daily backup window, your I/O may be suspended (for standalone RDS deployments)
- For Multi-AZ deployment, backups are taken from the standby DB instance
- Automated backups are deleted when you delete your RDS DB instance
- An outage occurs if you change the backup retention period from Zero to Non-Zero value or the other way around
- For RDS MySQL and Maria DB, automated backups are currently only supported for InnoDB storage engine for MySQL
 - Use of automated backups with other DB storage engines, including ISAM, may lead to unpredictable behavior during restoration.



Review Topic : Relational Database Service

DB Manual Snapshots

- Are not used for point-in-time recovery
- Stored in Amazon S3
- They are not deleted automatically when you delete your RDS instance. Rather, they will stay on S3 until you go ahead and delete them
- It is recommended to take a final snapshot before deleting your RDS DB instance
- Can be shared with other AWS accounts directly



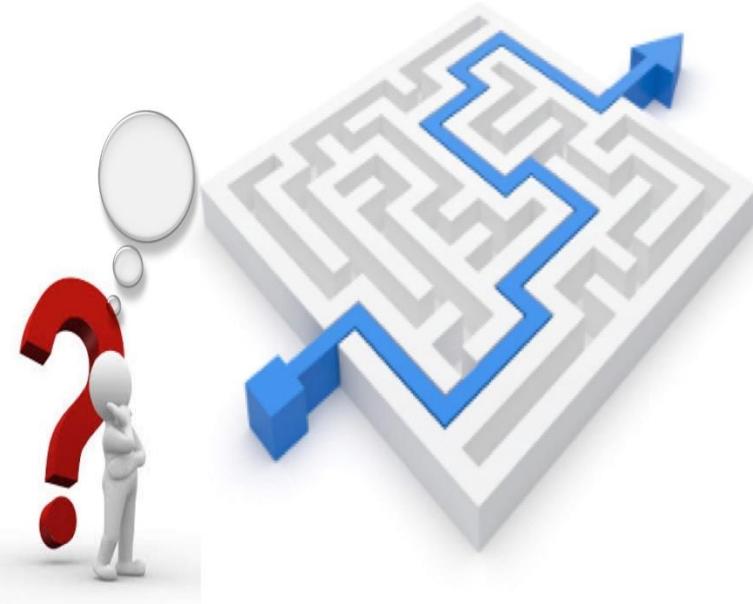
Review Topic : Relational Database Service

DB Automated Backups – Restore/Recovery

- You can specify a point-in-time restore to any given second during the retention period
- When you initiate a point-in-time recovery, transaction logs are applied to the most appropriate daily backup to restore your DB to that point-in-time
- You can not restore into an existing DB instance, it has to be a new DB instance with a new Instance/DB endpoint
- When you restore a DB instance, only the default DB parameters & Security groups are associated with the restored instance
 - Once the restore is complete, you need to associate/apply the customer DB parameters and security group settings
- You can change the Storage type (magnetic, Provisioned IOPS, General purpose) during a restore process

AWS RDS

Relational Database Service (RDS) – DB Read Replicas



Review Topic : Relational Database Service

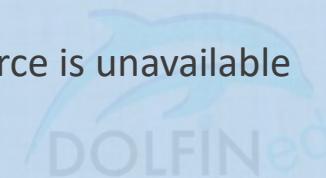
Read Replicas and Scaling DB read operations

- Remember that the standby DB instance in a Multi-AZ deployment **CAN NOT** be used for Read or Write.
- When the required read I/O capacity is reached but still more I/O capacity is required for heavy/intensive read applications, **RDS read replicas** can be helpful
- A read replica is a replica of the primary RDS DB instance that can only be used for read actions
- Amazon RDS uses the built-in replication functionality within MySQL, PostgreSQL, Oracle, and MariaDB to create the read replicas.
- Automatic backups must be enabled and remain enabled for read replicas to work

Review Topic : Relational Database Service

Read Replicas – Use cases

- Shifting read intensive applications such as Business (or Sales) reporting, or Data Warehousing to read from read replicas as opposed to overload the primary DB
- Scaling beyond the I/O capacity of your main DB instance for read-heavy workloads
- Service read traffic while the source is unavailable



Review Topic : Relational Database Service

Read Replicas

- Read replicas can be created in the same region as the master DB
 - In this case they must also be in the same VPC, not in a peered VPC in the same region
- They can also be created in a different region
 - If the read replica is in a different AWS region, Amazon RDS establishes a secure channel for replication between the master DB and the read replica
- Amazon RDS does not support a read replica on an EC2 instance (self managed) or on premises
- When initiated, RDS takes a snapshot from the master DB and creates the replica from that snapshot
- Primary DB instance becomes the source of the replication to the read replica
 - Using **asynchronous replication** (a time lag exists) data gets replicated to the read replica
 - If in Multi-AZ setup, the replication is done from the standby instance instead



Review Topic : Relational Database Service

Read Replicas

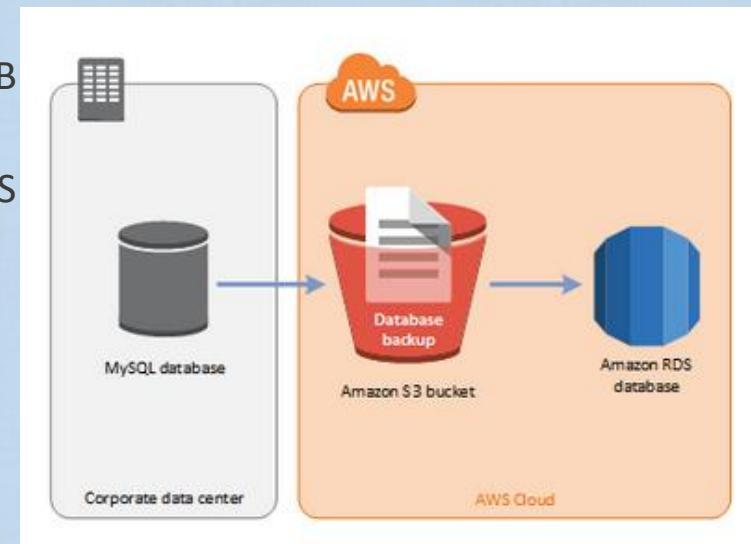
- Automatic backups must be enabled and remain enabled for read replicas to work
- RDS Multi-AZ can be combined with read replicas in one deployment
- For MySQL, MariaDB you can create a read replica from a read replica.
- Multi-AZ for read replicas:
 - It is also possible to create a standby read replica DB instances for your read replica instances. This is supported by MariaDB, PostgreSQL, and MySQL.
 - This is independent from whether the primary (source) DB instance was a standalone or Multi-AZ
- Is supported with transactional DB storage engines, and are supported on InnoDB engines not MyISAM (MySQL, MariaDB, PostgreSQL)
 - Each of these DB engines support up to 5 read replicas per source/primary DB instance



Review Topic : Relational Database Service

Restoring Backup into an Amazon RDS MySQL DB

- Amazon RDS supports importing MySQL databases by using backup files.
- Backup your on-premise or on EC2 instance MySQL DB to S3
- Use the backup file to restore into a new Amazon RDS MySQL DB



Review Topic : Relational Database Service

Creating Read Replicas

- The AZ where you want the read replica to be can be specified
- The read replica's storage type or instance class can be different from the source DB instance
- DB engine type CAN NOT be changed though, it is inherited from the source (primary) DB instance
- Connecting to the DB engine on the read replica is possible, the same way (DB console) you connect to the primary DB instance
- If you scale the source DB instance, you should also scale the Read Replicas.



Review Topic : Relational Database Service

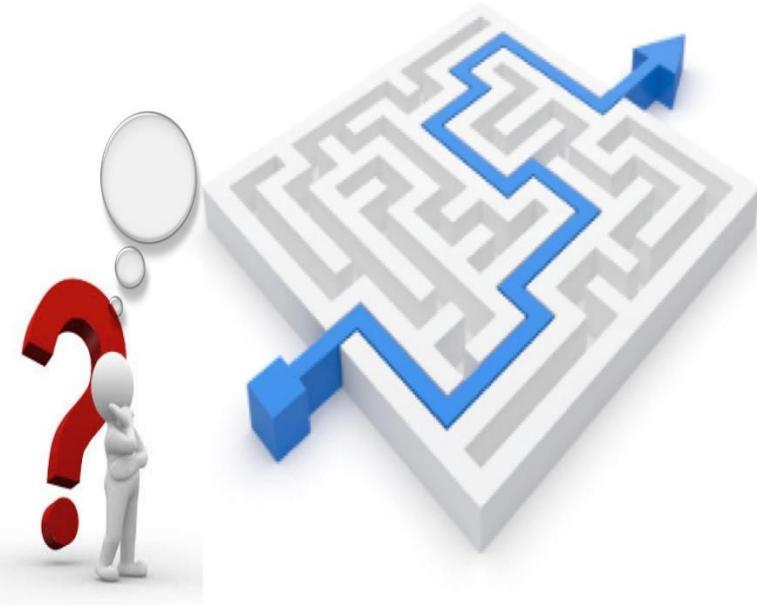
Promoting Read replicas to Standalone DB instance

- You can promote a read replica into a standalone/single AZ database instance
 - This is true for MySQL, MariaDB, PostgreSQL, and Oracle
 - Read replica will be rebooted before the standalone DB instance becomes available
- The promoted replica into a standalone DB instance will retain:
 - Backup retention period
 - Backup window
 - Option Group
 - DB parameter group of the formed read replica source (primary instance)



AWS RDS

RDS DB Security and Encryption



Review Topic : Relational Database Service

DB Instance encryption

- RDS Service supports **encryption at rest** (i.e data that is on the DB instance) for all DB engines using AWS KMS
- If you enable your RDS DB instance encryption at rest, underlying DB Storage, Logs, Snapshots, Read replicas, and automated backups will all be encrypted
- Amazon RDS uses AWS KMS for encryption keys
- RDS supports SSL encryption for communication between the DB clients and the RDS DB instances
 - RDS generates a certificate for the instance which is used to encrypt this communication



Review Topic : Relational Database Service

Encryption of Existing Unencrypted RDS Instances

- You can NOT enable encryption for an existing, un-encrypted database instance, alternatively to do that, you need to create an encrypted copy of that DB, this is how:
 - Create a snapshot of that DB
 - Copy the snapshot and choose to encrypt it during the copy process
 - Restore the encrypted copy into a **New DB**
- The resulting DB is an encrypted copy of your original, un-encrypted, DB
- You can't disable encryption on an encrypted DB

Review Topic : Relational Database Service

Transparent Data Encryption (TDE)

- Amazon RDS supports TDE for Oracle (Enterprise Edition) and MS SQL Server
- In TDE, data is automatically encrypted before it is written to DB storage, and automatically decrypted when it is read from storage.
- This can come in handy in scenarios where compliance requirements or the need to protect sensitive data is required
- Once TDE is enabled in an RDS option group it can't be disabled or turned off.
- TDE for SQL Server and Oracle can be simultaneously used with RDS encryption at rest, but this can impact the DB performance slightly



Review Topic : Relational Database Service

Amazon RDS encryption and Read Replicas

- A read replica of an Encrypted RDS instance is also encrypted
 - With the same RDS encryption key if the read replica is in the same region, or
 - With the read replica region's encryption key if in another region than the DB instance
- The read replica encryption status is like that of the original/Master DB,
 - Hence, you can't have an encrypted Read Replica of an unencrypted DB instance
 - Also, you can't have an unencrypted Read Replica of an encrypted DB instance.
- You can't restore an unencrypted backup or snapshot to an encrypted DB instance.
 - As discussed, to do this, make a copy, choose to encrypt it, then restore from the encrypted copy
- To copy an encrypted snapshot from one AWS Region to another, you must specify the KMS key identifier of the destination AWS Region. This is because KMS encryption keys are specific to the AWS Region that they are created in.



Review Topic : Relational Database Service

IAM DB Authentication for MySQL and PostgreSQL

- Is a mechanism that allows authentication to access the RDS DB instance
- IAM database authentication works with MySQL and PostgreSQL.
- With this authentication method, you don't need to use a password when you connect to a DB instance. Instead, you use an authentication token.
 - An *authentication token* is a unique string of characters that Amazon RDS generates on request.
 - Authentication tokens are generated using AWS Signature Version 4.
 - Each token has a lifetime of 15 minutes.
 - You don't need to store user credentials in the database, because authentication is managed externally using IAM.
- You can also still use standard database authentication.



Review Topic : Relational Database Service

IAM DB Authentication for MySQL and PostgreSQL - Benefits

- Network traffic to and from the database is encrypted using Secure Sockets Layer (SSL).
- You can use IAM to centrally manage access to your database resources, instead of managing access individually on each DB instance.
- For applications running on Amazon EC2, you can use profile credentials specific to your EC2 instance to access your database instead of a password, for greater security.
- Use this feature with care as currently it does not scale well for large number of DB connection requests per second



Review Topic : Relational Database Service

RDS DB Instances – Security Best Practices

- Use AWS IAM accounts to control access to RDS API actions
- Assign an individual IAM account to each person who manages RDS resources
- Grant the least permissions required by each user to perform the assigned duties
- Use IAM groups to manage/grant permission to multiple users at one time
- Rotate your IAM credentials regularly



AWS RDS

Relational Database Service (RDS) – Scaling, Billing and DB Instance Purchasing



Review Topic : Relational Database Service

Scaling

- You can scale (Up only, not down) the compute and storage capacity of your existing RDS DB instances
 - Scaling storage can happen while the RDS instance is still running (some performance impact)
 - Scaling compute will cause a downtime to your DB instance
- If you hit the largest RDS DB instance, and you still need to scale, you can:
 - Use partitioning and split your RDS DB over multiple RDS instances



Review Topic : Relational Database Service

RDS Billing – Standalone DB Instance

- No upfront costs
- You pay for:
 - DB instance hours (partial hours charged as full hours)
 - Storage GB/mo.
 - I/O requests/mo. - for Magnetic RDS storage Instance only
 - Provisioned IOPS/mo. – For RDS Provisioned IOPS SSD instance
 - Data transfer in and out of the DB instance from/to the internet or other AWS regions
 - Backup Storage (DB backups, and Active manual Snapshots)
 - This increases by increasing DB backups retention period
 - Backup storage for automated RDS backups (not the manual snapshots) up to the Provisioned RDS instance's EBS volume size (EBS volume) is free of charge



Review Topic : Relational Database Service

RDS Billing – Multi-AZ deployments

AWS will charge for the following (in addition to the single AZ DB instance charges)

- Multi-AZ DB hours
- Provisioned Storage (Multi-AZ)
- Double write I/Os (writing to the Active/Primary, and Replication from Primary to Standby)
- You are not charged for DB data transfer during replication from primary to standby
- Your DB storage does not change between Standalone and Multi-AZ deployments (same DB and same AWS Storage volume for that DB in multiple AZs for durability)

Review Topic : Relational Database Service

Size-Flexible Reserved DB Instances (RIs)

- Similar to EC2 Reserved Instances
 - One or three year term options
- DB RIs are “region” specific
- Each reservation must be specific in:
 - DB Engine
 - DB Instance class
 - Region
 - For RDS RI pricing to apply, an Exact RDS instance must be created on-demand, exact on all above (DB Engine, Instance class, **and** region)

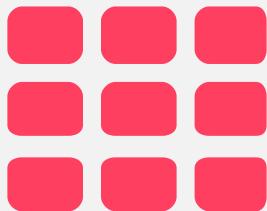


Review Topic : Relational Database Service

Reserved DB Instances (RIs)

- You can NOT move RDS RIs between regions
- You can move RDS RIs between AZs in the same region
- You can NOT cancel an RDS RI's reservation





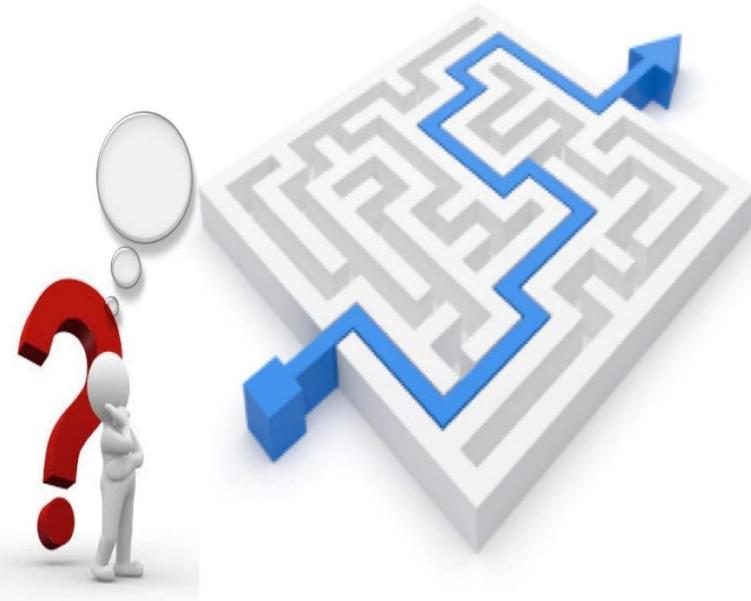
AWS AURORA

You Can Do It Too!



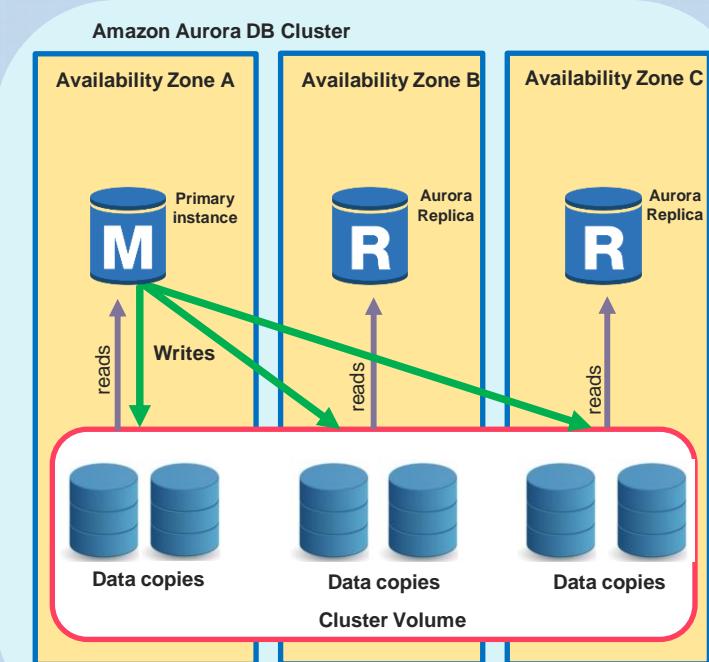
AWS AURORA

Introduction & Aurora Clusters



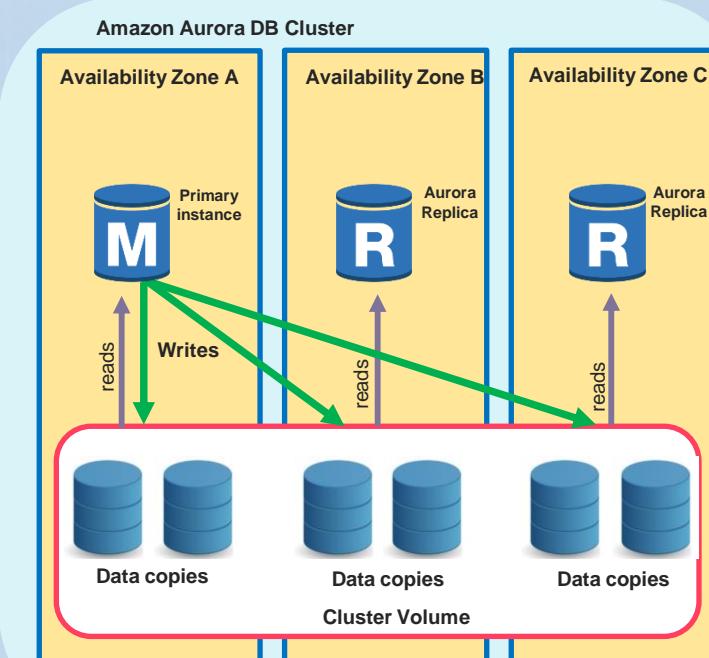
Amazon Aurora

- It is a fully managed, AWS proprietary, relational database engine that's compatible with MySQL and PostgreSQL.
 - The code, tools, and applications used today with existing MySQL and PostgreSQL databases can be used with Aurora.
- Aurora can deliver up to 5 times the throughput of MySQL and up to 3 times the throughput of PostgreSQL without requiring changes to most of your existing applications.
- Its MySQL and PostgreSQL compatible database engines are customized to take advantage of that fast-distributed storage.
- The underlying storage grows automatically as needed, up to 64 TiB.



Amazon Aurora DB Clusters

- An Amazon Aurora DB cluster consists of one or more DB instances and a cluster volume that manages the data for those DB instances.
- The Aurora cluster volume is a virtual database storage volume that spans multiple AZ's in one region.
 - Aurora maintains multiple (at least 2) copies of your data in three Availability Zones, in a single AWS region
- The Aurora cluster illustrates the separation of compute capacity and storage.
 - An Aurora configuration with only a single DB instance is still a cluster,
 - Since the underlying storage volume involves multiple storage nodes distributed across multiple Availability Zones (AZs).

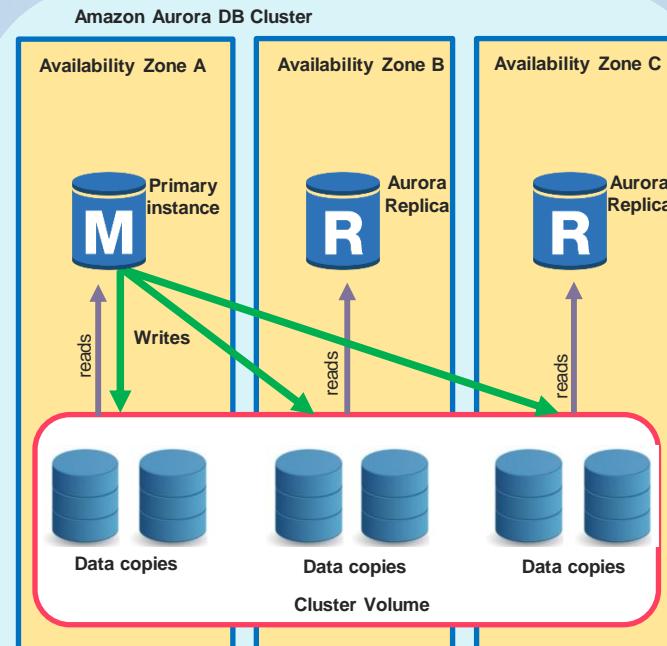


Amazon Aurora DB Clusters

Two types of DB instances make up an Aurora single Master DB cluster:

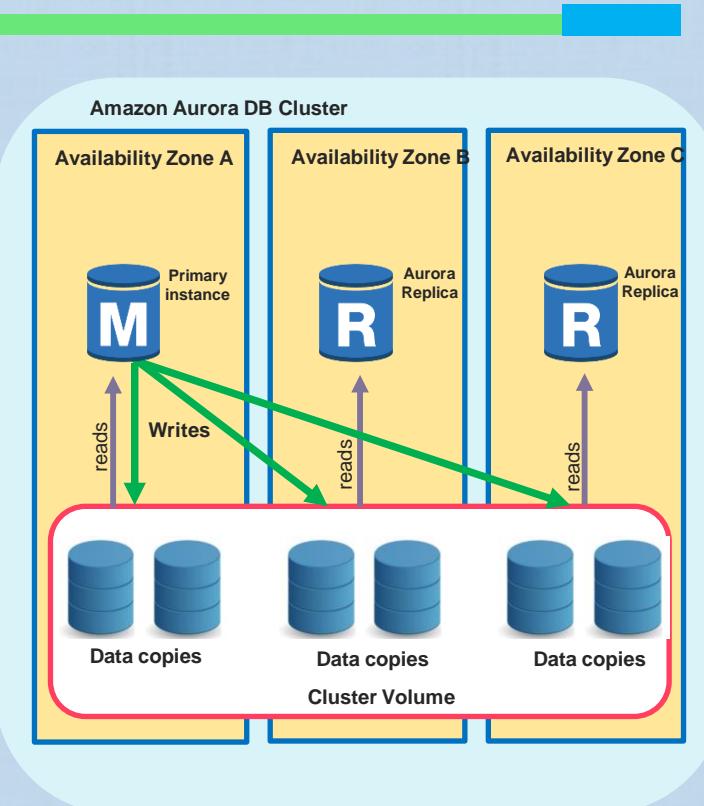
- **Primary DB instance**
 - Supports read and write operations and performs all of the data modifications to the cluster volume.
 - Each Aurora DB cluster has one primary DB instance.

- **Aurora Replica**
 - Connects to the same storage volume as the primary DB instance and supports only read operations (scaling read operations for the cluster).
 - Each Aurora DB cluster can have up to 15 Aurora Replicas in addition to the primary DB instance.
 - You can Maintain high availability by locating Aurora Replicas in separate Availability Zones.
 - Aurora automatically fails over to an Aurora Replica in case the primary DB instance becomes unavailable.
 - You can specify the failover priority for Aurora Replicas.



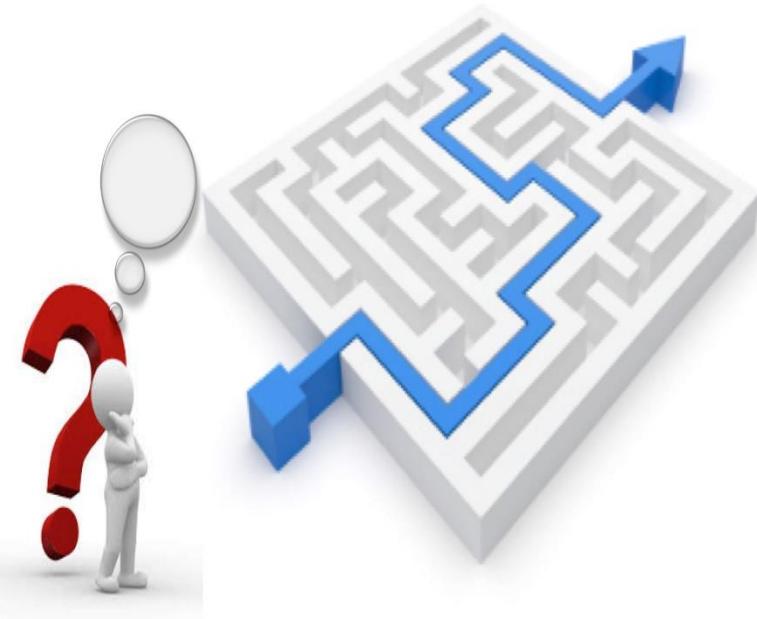
Amazon Aurora

- Data can be migrated from Amazon RDS for MySQL and Amazon RDS for PostgreSQL into Aurora, to do this
 - Create RDS snapshot and restore it to Aurora, or by setting up one-way replication.
- Push-button migration tools can be used to convert existing Amazon RDS for MySQL and Amazon RDS for PostgreSQL applications to Aurora.



AWS AURORA

- **Connection Management,**
- **End Points,**
- **Types of End Points**



Amazon Aurora – Connection Management & EndPoints

- When connecting to an Aurora cluster, you use URLs or hostnames known as Endpoints
 - Endpoints ensure that hardcoding all the hostnames or writing own logic for load-balancing and rerouting connections when some DB instances aren't available is not required.

Types of Aurora Endpoints:

- A **cluster endpoint** for an Aurora DB cluster that connects to the current primary DB instance for that DB cluster.
 - This endpoint is the only one that can perform write operations.
 - Example: mydbcluster.cluster-123456789012.us-east-1.rds.amazonaws.com:3306
- A **reader endpoint** for an Aurora DB cluster connects to one of the available Aurora Replicas for that DB cluster (or to the primary DB instance if no read replicas). Only one exists per cluster
 - Used for read only operations (can't be used for writes)
 - The reader endpoint provides load-balancing support for read-only connections to the DB cluster.
 - Example: mydbcluster.cluster-ro-123456789012.us-east-1.rds.amazonaws.com:3306



Amazon Aurora – Types of Aurora Endpoints (cont.)

- An **instance endpoint** connects to a specific DB instance within an Aurora cluster.
 - Each DB instance in a DB cluster has its own unique instance endpoint.
 - The instance endpoint provides direct control over connections to the DB cluster.
 - Example use case:
 - A client application might require more fine-grained load balancing based on workload type.
 - In this case, multiple clients can be configured to connect to different Aurora Replicas in a DB cluster to distribute read workloads.
 - Example: mydbinstance.123456789012.us-east-1.rds.amazonaws.com:3306
- A **custom endpoint** for an Aurora cluster represents a **set of DB instances** that you choose.
 - An Aurora DB cluster has no custom endpoints until you create one.
 - When you connect to the endpoint, Aurora performs load balancing and chooses one of the instances in the group to handle the connection.
 - Example: myendpoint.cluster-custom-123456789012.us-east-1.rds.amazonaws.com:3306



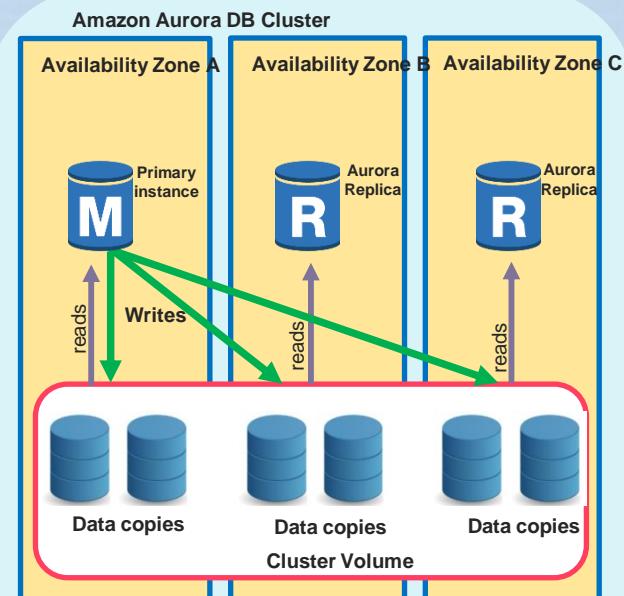
AWS AURORA

- Aurora Replicas
- Autoscaling
- Storage and Reliability
- Primary Failover



Amazon Aurora – Aurora Read Replicas

- The data in the cluster volume is represented as a single, logical volume to the primary instance and to Aurora Replicas in the DB cluster.
- Due to this cluster volumes architecture, Aurora Replicas can return the same data for query results with minimal replica lag
 - Usually much less than 100 milliseconds after the primary instance has written an update.
 - Because the cluster volume is shared among all instances, no additional work is required to replicate a copy of the data for each Aurora Replica.
- In contrast to Amazon RDS MySQL, Read Replicas must replay, on a single thread, all write operations from the master DB instance to their local data store.
 - This affects the ability of the Read Replicas to support large volumes of reads.



Auto Scaling with Aurora Replicas

- Aurora Auto Scaling dynamically adjusts the number of Aurora Replicas provisioned for an Aurora DB cluster using single-master replication.
- Aurora Auto Scaling enables Aurora DB clusters to handle sudden increases in connectivity or workload.
- When the connectivity or workload decreases, Aurora Auto Scaling removes unnecessary Aurora Replicas so that no change will be incurred for unused provisioned DB instances.
- Aurora Auto Scaling is available for both Aurora MySQL and Aurora PostgreSQL.
- Although Aurora Auto Scaling manages Aurora Replicas, the Aurora DB cluster must start with at least one Aurora Replica



Amazon Aurora – Primary Failover

- If the primary DB instance of a DB cluster fails, Aurora **automatically** fails over to a new primary DB instance.

It does so by either

- Promoting an Aurora Replica if one exists to be the primary instance.
- If there are no Aurora Replicas in the Aurora Cluster, then the cluster will be unavailable for the duration it takes the DB instance to recover or new DB instance gets created.

Notes

- Promoting an Aurora Replica is much faster than recreating the primary instance.
 - For high-availability scenarios, AWS recommends creating one or more Aurora Replicas of the same DB instance class as the primary instance and in different Availability Zones for your Aurora DB cluster.



AWS AURORA

- Aurora Security
- Aurora Encryption
- Aurora – Global DB
- Aurora with other AWS services



Amazon Aurora - Security

Security for Amazon Aurora is managed at different levels:

- IAM is used to control who can perform Amazon RDS management actions on Aurora DB clusters and DB instances.
 - With IAM database authentication, you authenticate to your Aurora MySQL DB cluster by using an IAM user or IAM role and an authentication token.
- Aurora DB clusters must be created in an Amazon Virtual Private Cloud (VPC).
 - An Amazon VPC endpoint for Amazon Aurora is a logical entity within a VPC that allows connectivity only to Amazon Aurora.
 - The Amazon VPC routes intra VPC requests to Amazon Aurora and routes responses back to the VPC through the VPC endpoint.
- To control which devices and Amazon EC2 instances can open connections to the endpoint and port of the DB instance for Aurora DB clusters in a VPC, a VPC security group is used.



Amazon Aurora - Encryption

- SSL can be used from the client application to encrypt a connection to a DB cluster running Aurora MySQL or Aurora PostgreSQL.
- Encrypt Amazon Aurora DB clusters and snapshots at rest by enabling the encryption option for Aurora DB clusters.
 - Data that is encrypted at rest includes the underlying storage for DB clusters, its automated backups, Read Replicas, and snapshots.
 - Database clients need not be modified to use encryption.
- Amazon Aurora encrypted DB clusters use the industry standard AES-256 encryption algorithm.
- You can't convert an unencrypted DB cluster to an encrypted one.
 - You can, however, restore an unencrypted Aurora DB cluster snapshot to an encrypted Aurora DB cluster.
 - DB clusters that are encrypted can't be modified to disable encryption.
- As of now, an encrypted Aurora Replica can't be created from an unencrypted Aurora DB cluster, and
 - An unencrypted Aurora Replica can't be created from an encrypted Aurora DB cluster.



Amazon Aurora Global Data Base

- An Aurora global database consists of **one primary AWS Region** where data is mastered, and **one read-only, secondary AWS Region**.
 - The Aurora cluster in the primary AWS Region performs both read and write operations.
 - The cluster in the secondary region can scale up to 16 Aurora replicas. It enables low-latency reads only.
- Aurora replicates data to the secondary AWS Region **with typical latency of under a second** using a dedicated infrastructure to do the replication.
- If you have an existing Aurora cluster, you can take a snapshot and restore it to a new Aurora global database
- You can manually activate the failover mechanism if a cluster in a different AWS Region is a better choice to be the primary cluster.



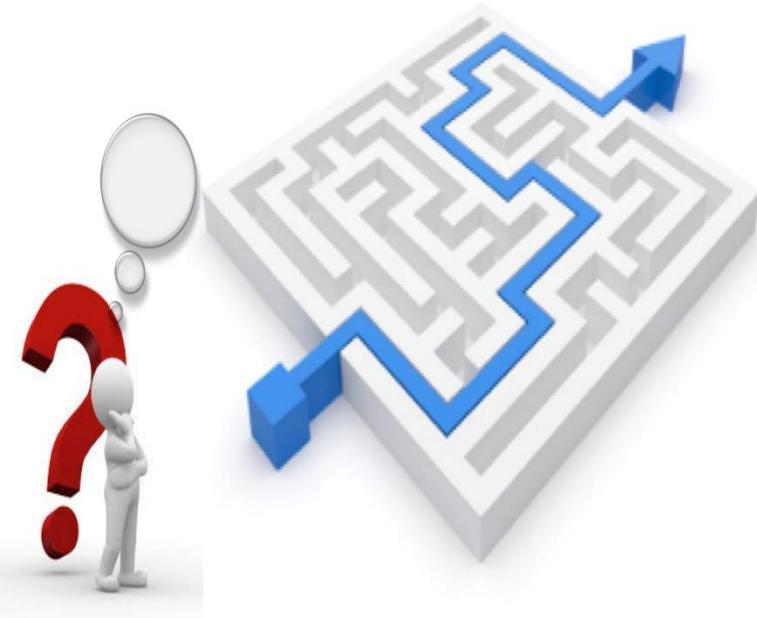
Amazon Aurora with other AWS Services

- You can use Aurora MySQL DB native function or stored procedure to invoke Lambda functions
- Loading data into a table from text files in an Amazon S3 bucket is available for Amazon Aurora MySQL
- It is possible to query data from an Amazon Aurora MySQL DB cluster and save it directly into text files stored in an Amazon S3 bucket.
 - This functionality can be used to skip bringing the data down to the client first, and then copying it from the client to Amazon S3.



AWS AURORA

- Aurora Replication
- Automatic backups and manual snapshots
- Sharing Aurora snapshots
- Aurora Backtrack feature



Amazon Aurora MySQL Replication - Across AWS Regions

- An Amazon Aurora MySQL DB cluster can be created as a Read Replica in a different AWS Region than the source DB cluster.
 - This approach can:
 - Improve your disaster recovery capabilities,
 - Allows for scaling read operations into an AWS Region that is closer to your users, and
 - Can make it easier to migrate from one AWS Region to another.
- You can create Read Replicas of both encrypted and unencrypted DB clusters.
 - The Read Replica must be encrypted if the source DB cluster is encrypted.
- For each source DB cluster, you can have up to five cross-region DB clusters that are Read Replicas.
- When creating the Read Replica, Amazon RDS takes a snapshot of the source cluster and transfers the snapshot to the Read Replica region.
- As a DR recovery mechanism, you can promote a read replica in another region to be come a standalone Aurora DB Cluster



Amazon Aurora – Automatic Backup and Manual Snapshots

- Aurora backs up your cluster **volume automatically and retains restore** data for the length of the backup retention period.
 - A backup retention period, from 1 to 35 days, can be specified when a DB cluster is created or modified.
- Aurora backups are **continuous and incremental** so you can quickly restore to any point within the backup retention period.
 - Recover your data to any given time during the retention period by creating **a new Aurora DB cluster** from the backup data that Aurora retains.
- No performance impact or interruption of database service occurs as backup data is being written.
- **Manual snapshots** of the data can also be created for cluster volume to retain a backup beyond the backup retention period,
 - A new DB cluster can be created from these snapshots

Amazon Aurora – Sharing DB Cluster Snapshots

- A **manual** DB cluster snapshot can be shared
- To share an **automated** DB cluster snapshot, create a **manual** DB cluster snapshot by copying the automated snapshot, and then share that copy.
- You can share a manual snapshot with up to 20 other AWS accounts.
- An unencrypted manual snapshot can be shared as public, it will be available to all AWS accounts.
- DB cluster snapshots that have been encrypted "at rest" can be shared,
 - The account this is shared with need to be given access (by sharing) the KMS encryption key used to encrypt the snapshot

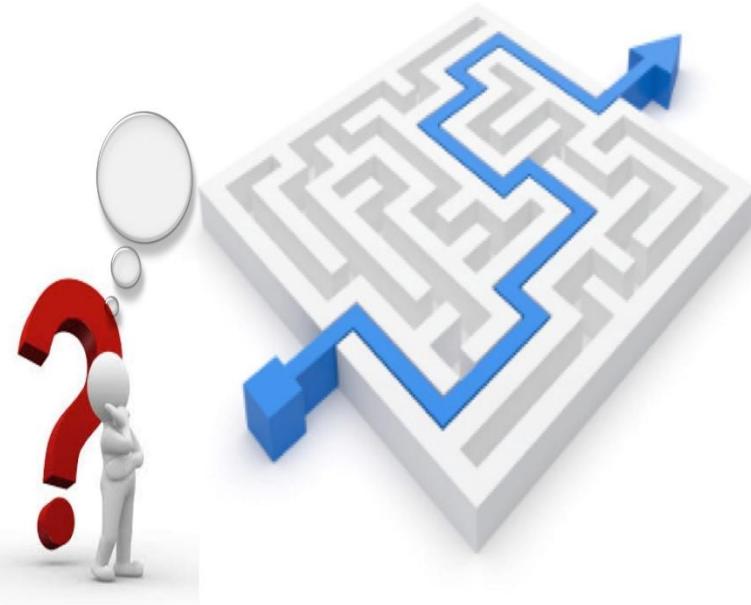
Amazon Aurora - Backtrack

- Backtracking "rewinds" the DB cluster to the time specified.
- Backtracking is not a replacement for backing up the DB cluster which allows for restoring it to a point in time.
- It does not rely on restoring from a backup or snapshot
- Backtracking provides the following advantages over traditional backup and restore:
 - You can easily undo mistakes. If you mistakenly perform a destructive action, you can backtrack the DB cluster to a time before the destructive action with minimal interruption of service.
 - You can backtrack a DB cluster quickly.
 - Unlike restoring the DB cluster to a point in time, Backtracking a DB cluster doesn't require a new DB cluster and rewinds the DB cluster in minutes.
 - It allows for exploring earlier data changes.
 - DB cluster can be backtracked back and forth in time to help determine when a particular data change occurred.



AWS AURORA

- Monitoring and Logging



Amazon Aurora – Monitoring and Logging

- **Amazon CloudWatch Alarms**
- **AWS CloudTrail Logs**
 - CloudTrail provides a record of actions taken by a user, role, or an AWS service in Amazon Aurora.
- **Enhanced Monitoring**
 - Amazon Aurora provides metrics in real time for the operating system (OS) that the DB cluster runs on.
- **Amazon RDS Performance Insights**
 - Performance Insights expands on existing Amazon Aurora monitoring features to illustrate the database's performance and help analyze any issues that affect it.
- **Database Logs**
 - Database logs can be viewed, downloaded, and watched
- **Amazon Aurora Event Notification**
 - Amazon Aurora uses the Amazon SNS to provide notification when an Amazon Aurora event occurs.



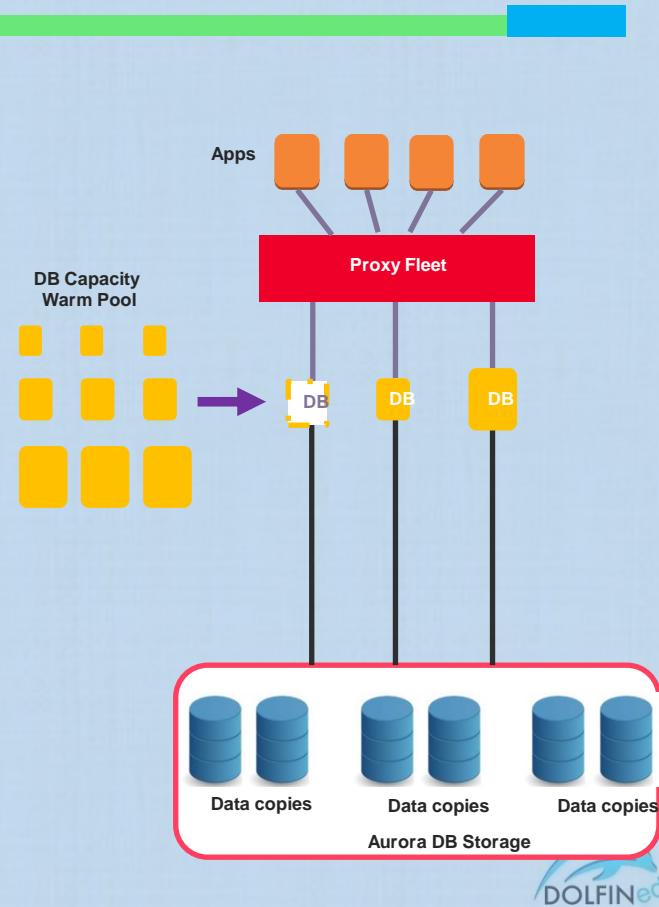
AWS AURORA

- Aurora Serverless

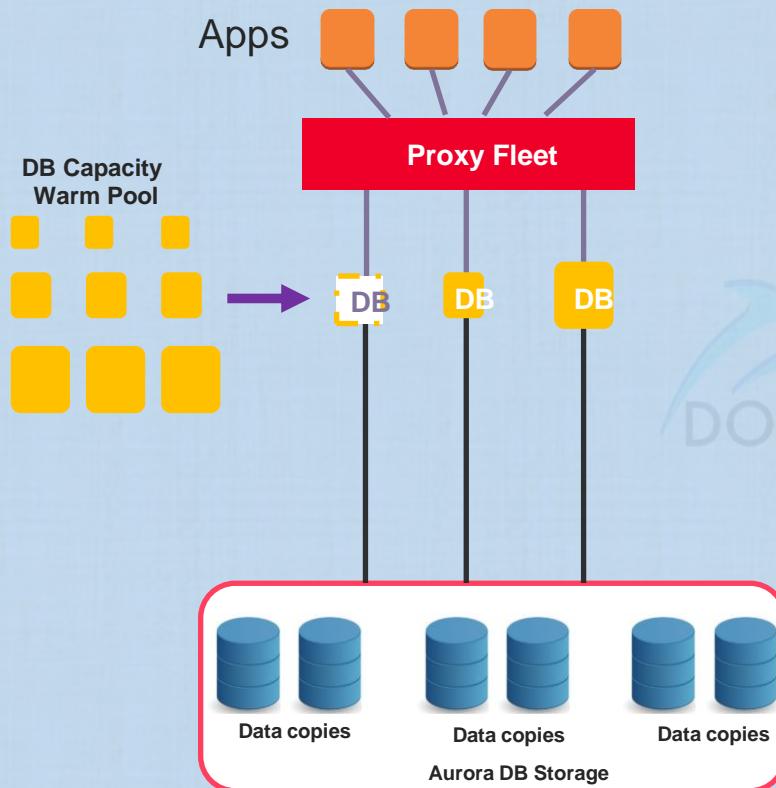


Amazon Aurora Serverless – What it is..

- It is an on-demand, autoscaling configuration for Amazon Aurora.
- Aurora Serverless provides a relatively simple, cost effective option for infrequent, intermittent, or unpredictable workloads.
 - It can provide this because it automatically starts up, scales compute capacity to match your application's usage, and shuts down when it's not in use.
- A non-Serverless DB cluster for Aurora is called a **provisioned DB cluster**.
- Aurora Serverless clusters and provisioned clusters both have the same kind of high-capacity, distributed, and highly available storage volume.
- You can connect to Aurora Serverless clusters using the TLS/SSL protocol



Amazon Aurora Serverless – How it works



- Aurora Serverless manages the warm pool of resources in an AWS Region to minimize scaling time.
- When new resources are added to the Aurora DB cluster, Aurora Serverless uses the proxy fleet to switch active client connections to the new resources.
- At any specific time, charges are only for the ACUs that are being used.
- It scales to zero capacity when there are no connections for a 5-minute period (default for Pause feature) actively used in your Aurora DB cluster.



Amazon Aurora Serverless – Use cases

- **Infrequently used applications**
 - Applications, such as **low-volume blog sites**, that are only used for a few minutes several times per day or week.
- **New applications**
 - When a **new application** is being deployed and required instance size is unknown
 - Aurora Serverless can create a database endpoint and have the database auto scale to the capacity requirements of the application.
- **Variable workloads**
 - A lightly used application, with peaks of 30 minutes to several hours a few times each day, or several times per year.
 - Such as **human resources, budgeting, and operational reporting** applications.
 - Provisioning to either peak or average capacity is no longer needed.
- **Unpredictable workloads**
 - When running workloads where there is database usage throughout the day, but also peaks of activity that are hard to predict.
 - An example is a traffic site that sees a surge of activity when it starts raining.



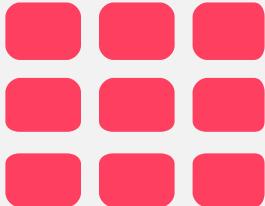
Amazon Aurora Serverless – Use cases

- **Development and test databases**
 - When developers use databases during work hours but don't need them on nights or weekends.
- **Multi-tenant applications**
 - With Aurora Serverless, individually managing database capacity for each application is no longer required.
 - Aurora Serverless manages individual database capacity for you.



Amazon Aurora Serverless – Snapshots

- You can create an Aurora Serverless cluster when restoring from snapshot of an Aurora Provisioned DB cluster
- You can take a snapshot of the Aurora Serverless DB Cluster
- The cluster volume for an Aurora Serverless cluster is always encrypted.
 - You can choose the encryption key, but not turn off encryption.
- To copy or share a snapshot of an Aurora Serverless cluster, you encrypt the snapshot using your own KMS key.

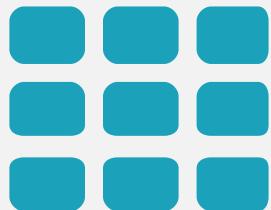


AWS MANAGEMENT, GOVERNANCE, AND NOTIFICATION SERVICES

SNS, CLOUD TRAIL & CLOUDWATCH

You Can Do It Too!





SIMPLE NOTIFICATION SERVICE (SNS)

You Can Do It Too!



AWS SNS



AWS Services

AWS SNS

- SNS is a fast, flexible, fully managed push notification service
- It is a web service that coordinates and manages the delivery or sending of messages (from the cloud) to subscribing endpoints or clients.
- It allows for sending individual messages or fan-out messages to a large number of recipients or to other distributed AWS (or non-AWS) services
 - Messages published to an SNS topics will be delivered to the subscribers immediately
- Instead of including a specific destination address in each message, a publisher sends a message to the topic. A publisher sends messages to topics that they have created or to topics they have permission to publish to.

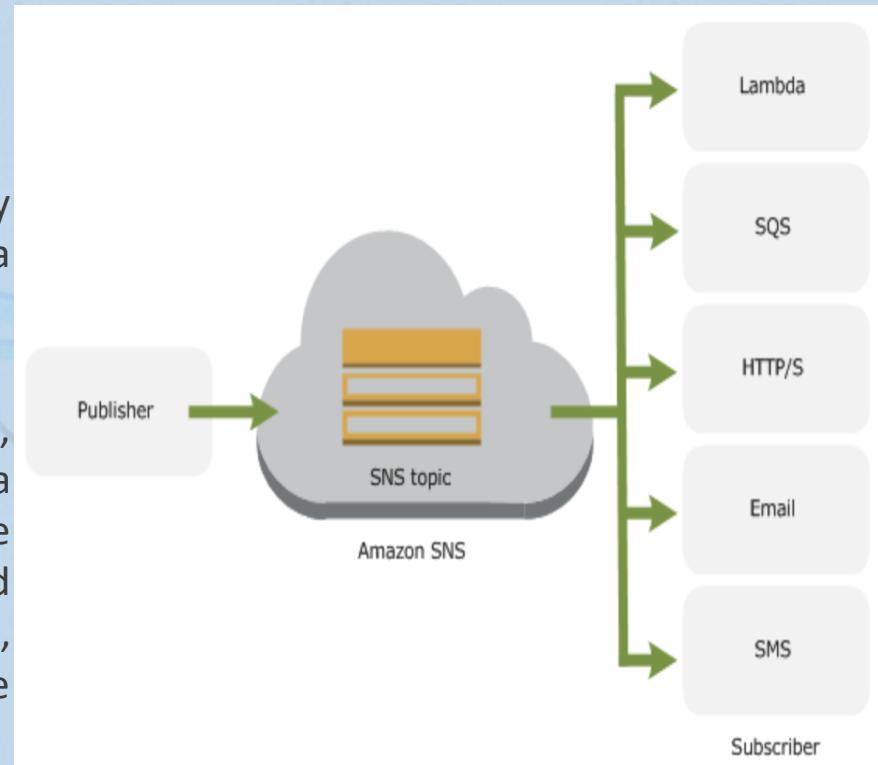


AWS Services

AWS SNS

In Amazon SNS, there are two types of clients—publishers and subscribers—also referred to as producers and consumers.

- Publishers communicate asynchronously with subscribers by producing and sending a message to a topic,
- Subscribers (web servers, email addresses, Amazon SQS queues, AWS Lambda functions) consume or receive the message or notification over one of the supported protocols (Amazon SQS, HTTP/S, email, SMS, Lambda, Application) when they are subscribed to the topic.



AWS Services

AWS SNS

Delivery formats / transport protocols (end points)

- SMS : Notification sent to registered phone number of the topic subscriber endpoint
- Email: Messages are sent as text email to registered email addresses (subscribed to the topic)
- Email – JSON : Messages/notifications are sent as JSON object to registered email addresses
 - It is meant for applications that can process emails
- HTTP/HTTPs: Subscribers specify a URL as part of their registration process. SNS messages/notifcations will be sent as “POST” to the URL
- SQS : SNS will en-queue messages in the specified SQS queue as the notification endpoint
- AWS Lambda

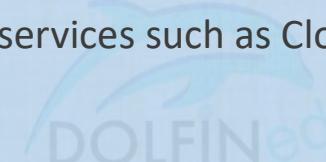


AWS Services

AWS SNS

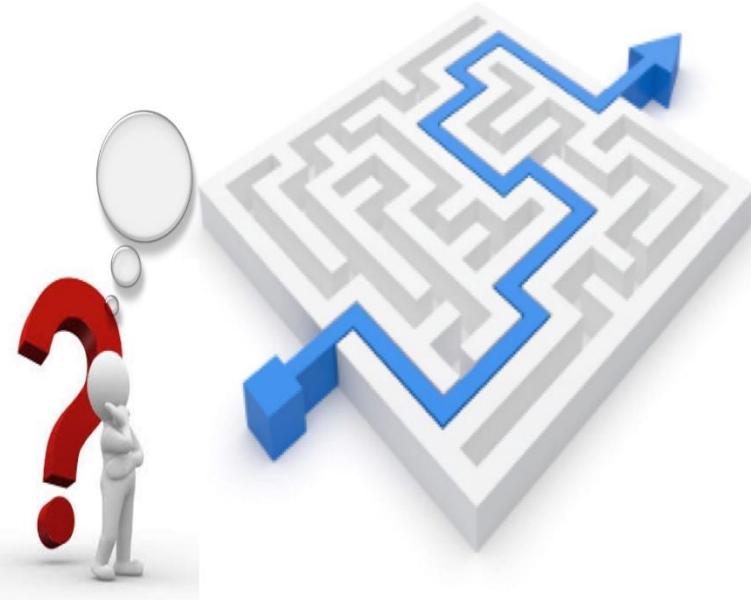
SNS allows you to:

- Send push messages (not Poll messages like SQS)
- Scale as your needs grow
- Engage audiences directly or all at once
- Deliver messages worldwide and across multiple transport protocols
- Easily connect with other AWS services such as CloudWatch, SQS, Lambda, S3
- Message delivery analysis
- Usage based pricing



AWS SNS

SNS Reliability and Security



Review Topic : AWS Services

SNS Reliability

- Amazon SNS stores all topic and message information within Amazon's proven network infrastructure and datacenters.
 - At least three copies of the data are stored across multiple availability zones,
 - This means that no single computer or network failure renders Amazon SNS inaccessible.



Review Topic : AWS Services

SNS Security

- Securing messages to topics:
 - All API calls made to Amazon SNS are validated for the user's AWS ID and the signature.
 - AWS recommends that users secure their data over the wire by connecting to the secure SSL end-points
- Authenticating API calls:
 - All API calls made to Amazon SNS will validate authenticity by requiring that:
 - Requests be signed with the secret key of the AWS ID account
 - And verifying the signature included in the requests.
- Amazon SNS requires publishers with AWS IDs to validate their messages by signing messages with their secret AWS key; the signature is then validated by Amazon SNS.



Review Topic : AWS Services

SNS Mobile Push Notifications

- SNS Mobile Push lets you use Simple Notification Service (SNS) to deliver push notifications to Apple, Google, Fire OS, and Windows devices
- With push notifications, an installed mobile application can notify its users immediately by popping a notification about an event, without opening the application.
- Push notifications can only be sent to devices that have your app installed, and whose users have opted in to receive them.
- SNS Mobile Push does not require explicit opt-in for sending push notifications, but iOS, Android and Kindle Fire operating systems do require it.
- In order to send push notifications with SNS, you must also register your app and each installed device with SNS.

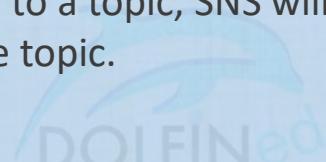


Review Topic : AWS Services

SNS Mobile Push Notifications

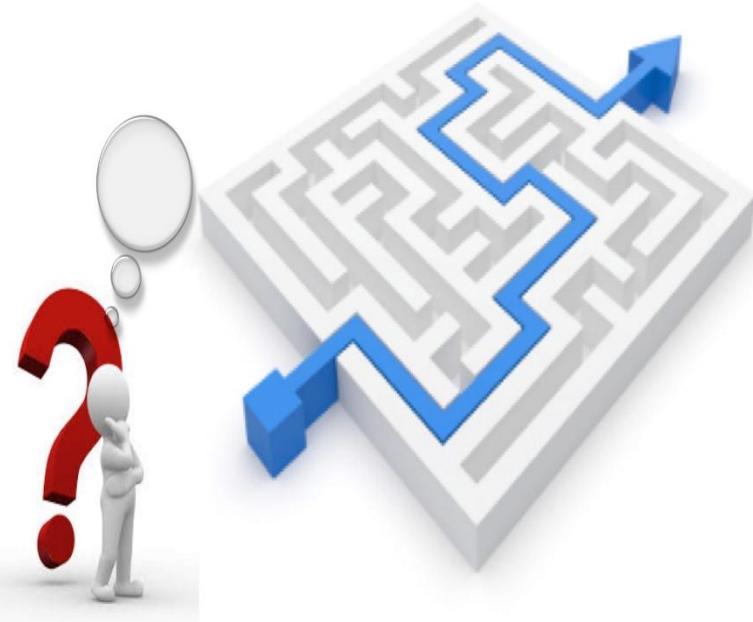
How does it work:

- SNS topics can have subscribers from any supported push notifications platform, as well as any other endpoint type such as SMS or email.
- When you publish a notification to a topic, SNS will send identical copies of that message to each endpoint subscribed to the topic.



AWS SNS

Billing and Logging



AWS Services

AWS SNS Billing

- With Amazon SNS, there is no minimum fee and you pay only for what you use.
- Users pay \$0.50 per 1 million Amazon SNS Requests, Plus
 - \$0.06 per 100,000 notification deliveries over HTTP,
 - \$2.00 per 100,000 notification deliveries over email
 - For SMS messaging, users can send 100 free notification deliveries,
 - For subsequent messages charges vary by destination country
- Amazon SNS also includes a Free Tier, where users can get started with Amazon SNS for free.
 - Each month,
 - Amazon SNS customers incur no charges for the first 1 million Amazon SNS requests,
 - No charges for the first 100,000 notifications over HTTP,
 - No charges for the first 100 notifications over SMS,
 - And no charges for the first 1,000 notifications over email.



AWS Services

AWS SNS Billing

- Amazon SNS currently allows a maximum limit of 256 KB for published messages.
- Each 64KB chunk of published data is billed as 1 request.
 - For example, a single API call with a 256KB payload will be billed as four requests.

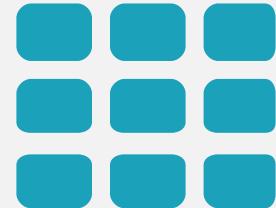


AWS Services

AWS SNS & AWS CloudTrail

- You can get the history for SNS API calls made to your account by enabling Cloudtrail
 - Cloudtrail will delivery log files for your SNS API Calls
- Cloudtrail logs will provide:
 - SNS API Caller
 - Source IP address
 - Time of the API call
 - Request parameters
 - Response elements returned by SNS
- This would be handy for security analysis, auditing , and operational/troubleshooting purposes
- Cloutrail logs for SNS are available for authenticated API calls only





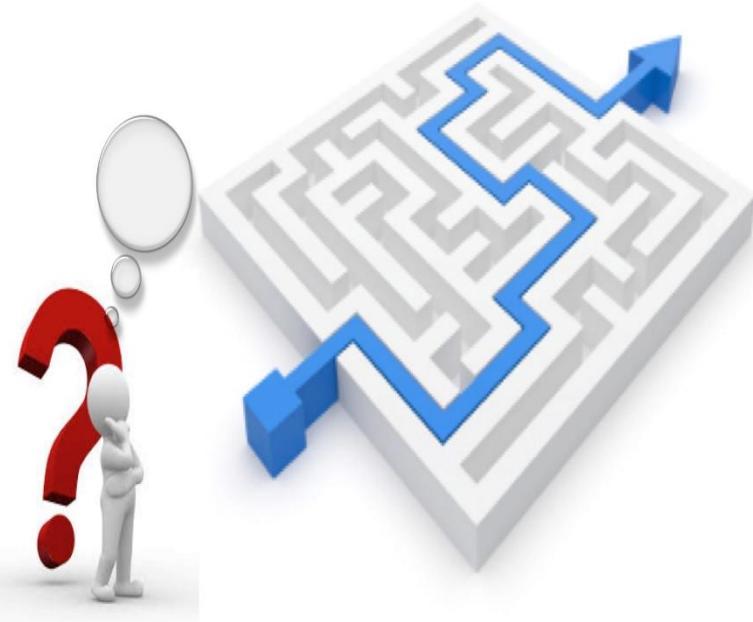
CLOUDTRAIL

You Can Do It Too!



AWS CloudTrail

Cloud Trail



AWS CloudTrail

- AWS CloudTrail is an AWS service that helps you enable governance, compliance, operational and risk auditing of your AWS account.
- Actions taken by a user, role, or an AWS service are recorded as events in CloudTrail.
 - Events include actions taken in the AWS Management Console, AWS Command Line Interface, and AWS SDKs and APIs.
- CloudTrail is enabled on your AWS account when you create it (but not CloudTrail Logging).
- When activity occurs in your AWS account, that activity is recorded in a CloudTrail event.
 - You can easily view recent events in the CloudTrail console by going to Event history.

AWS CloudTrail

Benefits

- You can identify:
 - Who or what took which action,
 - What resources were acted upon,
 - When the event occurred, and other details to help you analyze and respond to activity in your AWS account.

This can benefit in the following areas

- Security
 - Visibility into your AWS account activity is a key aspect of security best practices.
- Tracking changes in an AWS environment
 - You can use CloudTrail to view, search, download, archive, analyze, and respond to account activity across your AWS infrastructure.
- Compliance and auditing
- Operational and troubleshooting support



AWS CloudTrail

Event History and Trails

- Event history allows you to view, search, and download the past 90 days of activity in your account.
- You can create a **CloudTrail trail** to archive, analyze, and respond to changes in your AWS resources.
- A trail is a configuration that enables delivery of events to an Amazon S3 bucket that you specify.
 - You can also opt to send the logs to CloudWatch logs – log group instead
- CloudTrail logging, which is basically, sending the CloudTrail events to a bucket **is not enabled by default.**
 - You need to create a Trail and define a bucket, then CloudTrail will send events to this bucket, i.e will start logging the identified/selected events.

AWS CloudTrail

CloudTrail – Types of Trails

You can create two types of trails:

- **A trail that applies to all regions (recommended by AWS)**
 - When you create a trail that applies to all regions
 - CloudTrail records events in each region and delivers the CloudTrail event log files to an S3 bucket that you specify.
 - This is effectively like creating the trail in each of these regions
 - If a region is added after you create a trail that applies to all regions,
 - That new region is automatically included, and events in that region are logged.
- **A trail that applies to one region**
 - When you create a trail that applies to one region,
 - CloudTrail records the events in that region only.
 - It then delivers the CloudTrail event log files to an Amazon S3 bucket that you specify.
 - Single-region trails can deliver to the same or different S3 buckets
- For both types of trails, you can specify an Amazon S3 bucket from any region.



AWS CloudTrail

Advantages of All Regions Trails

A trail that applies to all regions has the following advantages:

- The configuration settings for the trail apply consistently across all regions.
- Receiving CloudTrail events from all regions in a single S3 bucket and, optionally, in a CloudWatch Logs log group.
- Managing trail configuration for all regions from one location.
- Immediately receiving CloudTrail events from a new region when launched.
- Ability to create trails in regions that you don't use often to monitor for unusual activity.
- If CloudTrail is configured to use an Amazon SNS topic for the trail, SNS notifications about log file deliveries in all regions are sent to that single SNS topic.



AWS CloudTrail

Encryption of CloudTrail Events' Logging

- By default, the log files delivered by CloudTrail to your bucket are encrypted by Amazon server-side encryption with Amazon S3-managed encryption keys (SSE-S3).
- To provide a security layer that is directly manageable, you can instead use server-side encryption with AWS KMS-managed keys (SSE-KMS) for your CloudTrail log files.

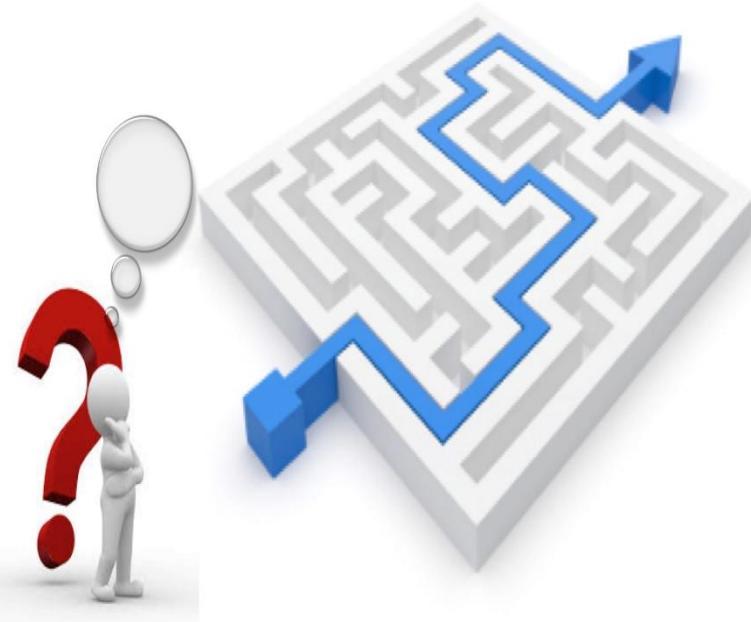
Note

- Enabling server-side encryption encrypts the log files but not the digest files with SSE-KMS. Digest files are encrypted with Amazon S3-managed encryption keys (SSE-S3).



AWS CloudTrail

Monitoring & Best Practices



AWS CloudTrail

Monitoring and Notifications

SNS notifications:

- If you want notifications about log file delivery and validation,
 - Create and subscribe to an Amazon SNS topic to receive notifications about log file delivery to your bucket.
 - Amazon SNS can notify you in multiple ways, including programmatically with Amazon Simple Queue Service.

Monitor events with CloudWatch Logs

- You can configure your trail to send events to CloudWatch Logs. You can then use CloudWatch Logs to monitor your account for specific API calls and events



AWS CloudTrail

Logging Best Practices

- Limit who can access your CloudTrail and CloudTrail logs
 - Limit access to the bucket policy to only those who need to view it
 - Enable Bucket MFA Delete protection
- Use notification in the event of a misconfiguration
- Log sizes can grow with time, enable S3 lifecycle policies to store logs
 - Compliance requires maintain the logs for extended periods of time



AWS CloudTrail

CloudTrail VPC Endpoint

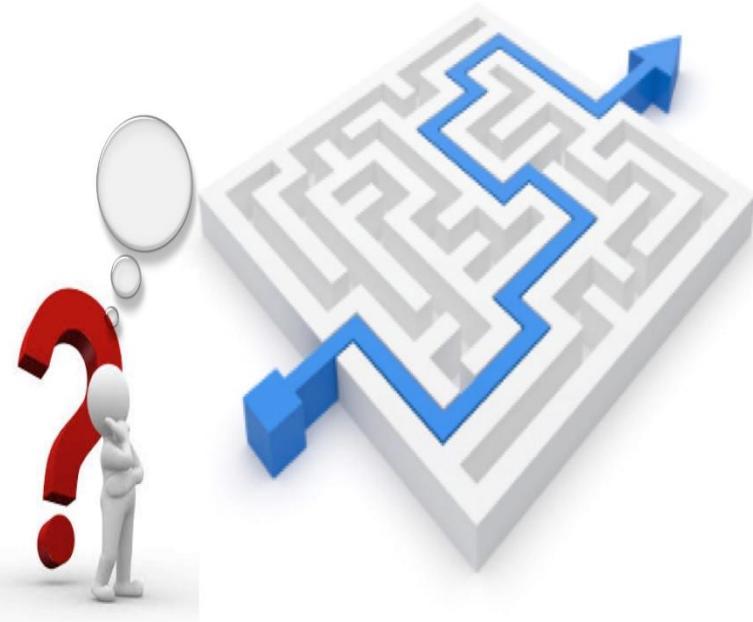
Using AWS CloudTrail with Interface VPC Endpoints

- If you use Amazon VPC to host your AWS resources, you can establish a private connection between your VPC and AWS CloudTrail.
 - You can use this connection to enable CloudTrail to communicate with your resources on your VPC without going through the public internet.
 - To connect your VPC to CloudTrail, you define an interface VPC endpoint for CloudTrail.
 - An interface endpoint is an elastic network interface with a private IP address that serves as an entry point for traffic destined to a supported AWS service.
 - The endpoint provides reliable, scalable connectivity to CloudTrail without requiring an internet gateway, network address translation (NAT) instance, or VPN connection.



AWS CloudTrail

Log File Integrity Validation



AWS CloudTrail Log File Integrity Validation – What is it

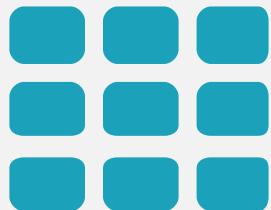
- Is the ability of Amazon CloudTrail to determine whether a log file was modified, deleted, or unchanged after CloudTrail trail has delivered it to your S3 bucket.
- This feature is built using industry standard algorithms: SHA-256 for hashing and SHA-256 with RSA for digital signing.
 - This makes it computationally infeasible to modify, delete or forge CloudTrail log files without detection.
- You can use the AWS CLI to validate the files in the location where CloudTrail delivered them.
- Benefits of this feature:
 - Validated log files are invaluable in security and forensic investigations.
 - A validated log file enables you to assert positively that the log file itself has not changed, or
 - Whether a particular user credentials performed specific API activity.
 - It lets you know if a log file has been deleted or changed, or
 - Assert positively that no log files were delivered to your account during a given period of time.



CloudTrail Log File Integrity Validation – How It Works

- When the validation feature is enabled
 - CloudTrail creates a hash for every log file that it delivers.
 - Every hour, CloudTrail creates and delivers a file that references the log files for the last hour and contains a hash of each.
 - This file is called a digest file.
 - CloudTrail signs each digest file using the private key of a public and private key pair.
 - After delivery, the public key can be used to validate the digest file.
 - CloudTrail uses different key pairs for each AWS region.
- The digest files are delivered to the same Amazon S3 bucket associated with the trail as your CloudTrail log files.
- Each digest file also contains the digital signature of the previous digest file if one exists.
- The signature for the current digest file is in the metadata properties of the digest file Amazon S3 object





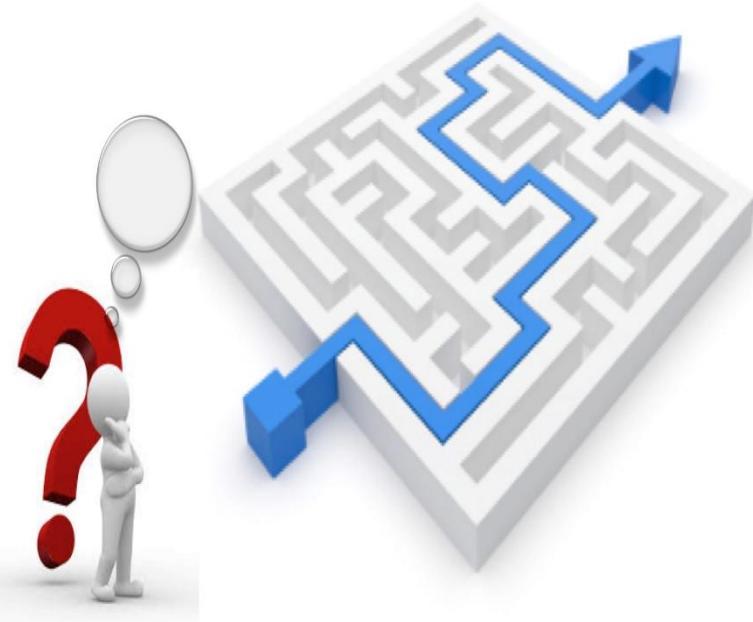
CLOUDWATCH

You Can Do It Too!



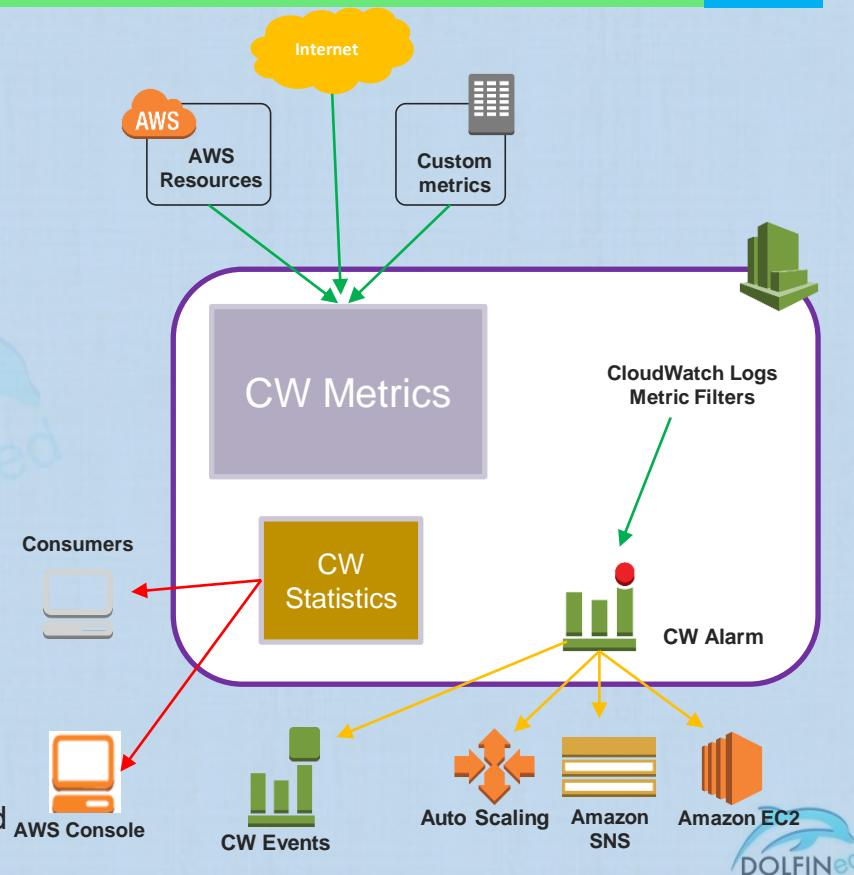
AWS Monitoring Services

CloudWatch



CloudWatch

- Amazon CloudWatch:
 - Provides system-wide visibility into resource utilization, application performance, and operational health.
 - Monitors AWS resources and any applications that are run on AWS in real time
 - Can be used to collect and track metrics
 - Can monitor the built-in metrics that come with AWS or your custom metrics.
- Amazon CloudWatch is a metrics repository
 - Input to this repository can be:
 - Out of the box metrics from the AWS services using CW
 - Custom metrics based on custom data that you can decide
- VPC resources can connect to CloudWatch by creating and CloudWatch VPC Interface Endpoint



CloudWatch Concepts - Metrics

- Metrics are the fundamental concept in CloudWatch.
 - A metric represents a **time-ordered set of data points** that are published to CloudWatch.
 - Think of a metric as a variable to monitor,
 - The data points represent the values of that variable over time.
 - CPU usage of an EC2 instance is one metric provided by Amazon EC2.
- Data points themselves can come from any application or business activity from which you collect data.
- AWS services send metrics to CloudWatch
 - You can send your own custom metrics to CloudWatch.
- Metrics exist only in the region in which they are created, in other words,
 - **Metrics are completely separate between regions.**
- Metrics are uniquely defined by a name, a namespace, and zero or more dimensions.

CloudWatch Concepts - Namespaces

- A namespace is a container for CloudWatch metrics.
- The AWS namespaces use the following naming convention: **AWS/service**
 - For instance, EC2 uses the AWS/EC2 namespace.
- You can specify a namespace name when you create a metric.
- There is no default namespace.
 - Specify a namespace for each data point (i.e metric value at a point in time - take CPU utilization at a specific time) that you publish to CloudWatch.



CloudWatch Concepts - Timestamps

- Each metric data point must be marked with a time stamp.

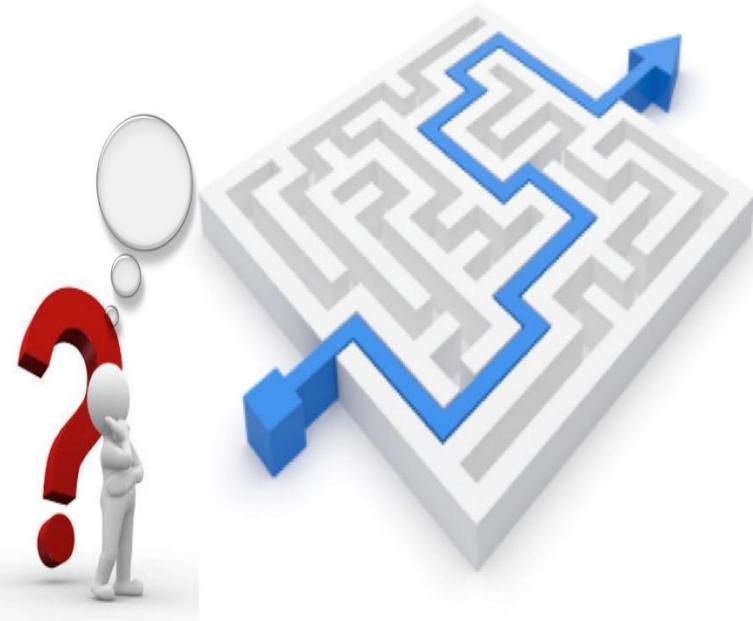


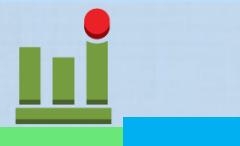
CloudWatch Concepts – Metric Retention

- Metrics cannot be deleted, but they automatically expire after 15 months if no new data is published to them.
- CloudWatch retains metric data as follows:
 - Data points with a period of less than 60 seconds are available for 3 hours. These data points are high-resolution custom metrics.
 - Data points with a period of 60 seconds (1 minute) are available for 15 days
 - Data points with a period of 300 seconds (5 minute) are available for 63 days
 - Data points with a period of 3600 seconds (1 hour) are available for 455 days (15 months)
 - Data points older than 15 months expire on a rolling basis; as new data points come in, data older than 15 months is dropped.
 - Data points that are initially published with a shorter period are aggregated together for long-term storage.

AWS CloudWatch

CloudWatch Alarms

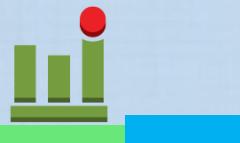




CloudWatch Alarms

- A CloudWatch alarm can be created to watch a **single CloudWatch metric or the result of a math expression** based on CloudWatch metrics.
- Alarms can be created or can be based on CloudWatch Logs metric filters
- The alarm can perform one or more actions based on the value of the metric or expression relative to a threshold over a number of time periods (the duration of time over which the alarm is evaluated).
- The possible alarm states:
 - **OK** – The metric or expression is within the defined threshold.
 - **ALARM** – The metric or expression is outside of the defined threshold (Alarm is triggered).
 - **INSUFFICIENT_DATA** – The alarm has just started, the metric is not available, or not enough data is available for the metric to determine the alarm state.
- Alarms can be added to CloudWatch dashboards and monitor them visually (They turn RED when in ALARM state).





CloudWatch Alarms - Evaluating an Alarm

- When creating an alarm you need to specify:
 - Period (seconds)
 - Evaluation Period
 - Datapoints to Alarm
- CloudWatch uses these settings to determine the Alarm state.





CloudWatch Alarms – Actions and Targets

- CloudWatch Alarms can do Auto Scaling, EC2, or SNS actions only.
- CloudWatch Alarms can NOT invoke Lambda functions directly
- The action can be an Amazon
 - EC2 action (Recover, Start, Reboot, or Terminate an EC2 instance),
 - Trigger an Amazon EC2 Auto Scaling action, or
 - Send a notification to an Amazon SNS topic.
- CloudWatch Alarms invoke actions for sustained state changes only, they do not invoke actions because they are in a particular state
 - The state must have changed and been maintained for a specified number of periods.
- CloudWatch doesn't test or validate the actions that you specify, nor does it detect any Amazon EC2 Auto Scaling or Amazon SNS errors resulting from an attempt to invoke nonexistent actions.

AWS CloudWatch

Custom Metrics



CloudWatch – Custom Metrics

- You can publish your own metrics to CloudWatch using **the AWS CLI or an API**.
- You can view statistical graphs of your published metrics with the AWS Management Console.
- CloudWatch stores data about a metric as a series of data points.
 - Each data point has an associated time stamp.
 - You can publish an aggregated set of data points called a statistic set

CloudWatch – High Resolution Metrics

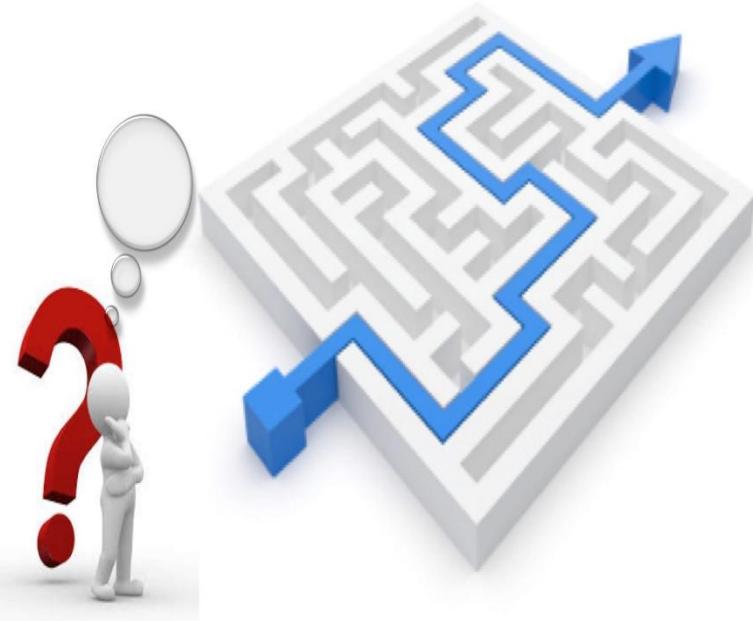
- Each metric is one of the following:
 - Standard resolution, with data having a one-minute granularity
 - High resolution, with data at a granularity of one second
- Metrics produced by AWS services are standard resolution by default.
- When you publish a custom metric, you can define it as either standard resolution or high resolution.
- When you publish a high-resolution metric, CloudWatch stores it with a resolution of 1 second, and you can read and retrieve it with a period of 1 second, 5 seconds, 10 seconds, 30 seconds, or any multiple of 60 seconds.

Controlling Access to CloudWatch

- Every AWS resource is owned by an AWS account, and permissions to create or access a resource are governed by permissions policies.
- You can have valid credentials to authenticate your requests, but unless you have permissions you cannot create or access CloudWatch resources. For example, you must have permissions to create CloudWatch dashboard widgets, view metrics, and so on.
- For example, you can't give a user access to CloudWatch data for only a specific set of EC2 instances or a specific load balancer.
- Permissions granted using IAM cover all the cloud resources you use or monitor with CloudWatch. In addition,

AWS Logging Service

CloudWatch Logs



CloudWatch Logs

- Amazon CloudWatch Logs service can be used to monitor, store, and access log files from many **AWS and Non-AWS sources**.
 - CloudWatch Logs **uses your existing log data for monitoring**; so, no code changes are required.
- CloudWatch Logs enables you to centralize the logs from all systems, applications, and AWS services that are in use, in a single, highly scalable logging service.
- This makes it easy to:
 - View the logs, regardless of the source as a single and consistent flow of time ordered events.
 - Sort and query/search the logs for specific error codes or patterns,
 - Filter them based on specific fields, or
 - Archive them securely for future analysis.
- You can also create custom/powerful queries (Using CloudWatch Insights) and visualize the log data in dashboards.



CloudWatch Logs – Log data sources

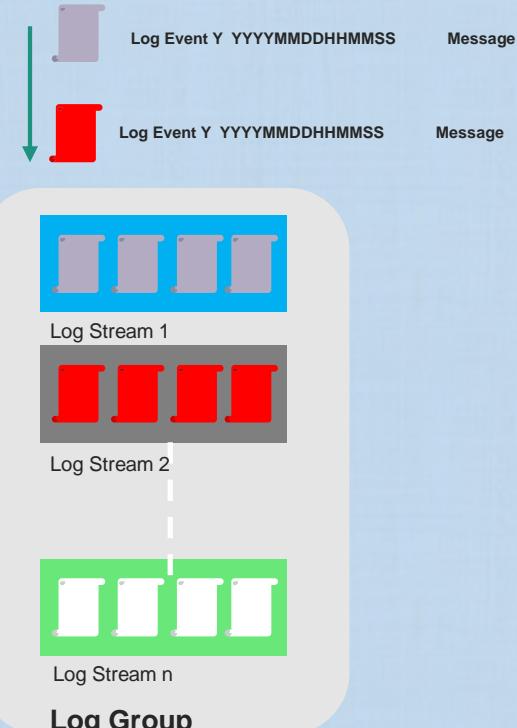
- You can use Amazon CloudWatch Logs to monitor, store, and access your log files from:
 - Amazon EC2 instances and containers on ECS instances (Requires a CloudWatch Agent)
 - AWS CloudTrail,
 - Route 53 DNS queries logs,
 - RDS Aurora, MySQL and MariaDB,
 - Amazon Neptune
 - VPC Flow Logs
 - Elastic Beanstalk for EC2 instances in the EB environment
 - API Gateway Execution logging
 - Lambda functions logs and other sources.
 - Logs from **On Premise Servers** (Requires CloudWatch agent)

CloudWatch Logs - Concepts

- **Log Events**
 - A log event is a record of some activity recorded **by the application or resource being monitored.**

- **Log Streams**
 - A log stream is a sequence of log events that **share the same source** (e.g. Apache access log on a host).
 - AWS deletes empty log streams that are more than 2 months old, or you can delete them manually

- **Log Groups**
 - Log groups define groups of log streams **that share the same retention, monitoring, and access control settings.**
 - Each *log stream has to belong to one log group.*
 - There is no limit on the number of log streams that can belong to one log group.



CloudWatch Logs - Concepts

- **Log Retention**
 - By default, logs are kept indefinitely and never expire (Logs data incurs charges)
 - The retention policy's retention settings can be adjusted for each log group (**at the Log group level**), by choosing retention periods between 10 years and one day.
 - Expired log events get deleted automatically.
 - The retention settings assigned to a log group gets applied to its log streams.
- **Metric Filters**
 - You can use filters to configure **custom** CloudWatch metrics from CloudWatch logs
 - Metric filters are used to extract metric observations from ingested Log events and transform them to data points in a CloudWatch metric.
 - Metric filters are **assigned to log groups**, and all of the filters assigned to a log group are applied to their log streams.
 - When the term the filter is configured to search for is found, **CloudWatch Logs** reports the data to the specified **CloudWatch metric**.
 - CloudWatch Logs sends metrics to CloudWatch every minute.



Encrypting Log data in CloudWatch Logs using KMS

Encryption in Transit

- CloudWatch Logs uses end-to-end encryption of data in transit.
 - CloudWatch Logs service manages the server-side encryption keys.

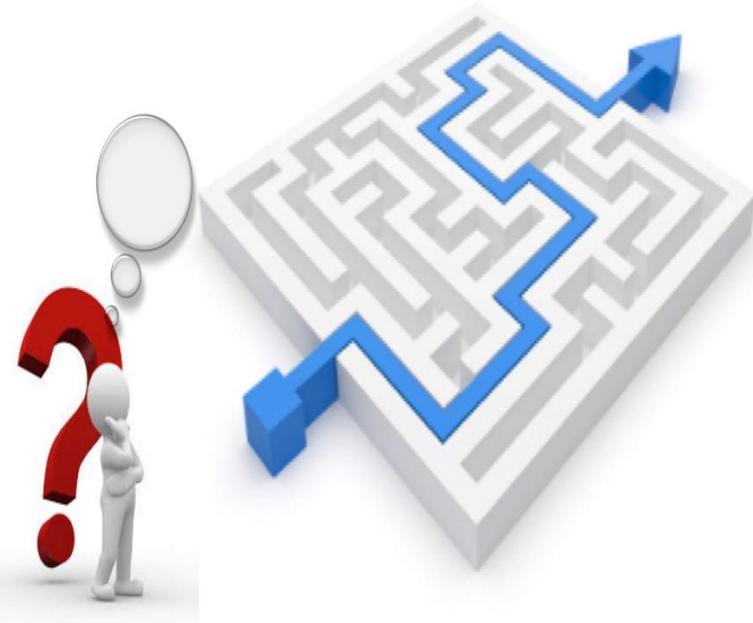
Encryption at Rest

- CloudWatch Logs protects data at rest using encryption.
 - CloudWatch Logs service manages the server-side encryption keys.
 - You can encrypt the log data in CloudWatch Logs using an AWS KMS customer master key (CMK).
 - Encryption is enabled **at the log group level**, by associating a CMK with a log group,
 - This can be done either when you create the log group or after it exists.
- After you associate a CMK with a log group
 - All **newly** ingested data for the log group is encrypted using the CMK.
 - This data is stored in encrypted format throughout its retention period.
 - CloudWatch Logs decrypts this data whenever it is requested.
 - CloudWatch Logs must have permissions for the CMK whenever encrypted data is requested.



AWS CloudWatch Logs

- CloudWatch Logs Log Insights
- Real time monitoring for EC2
- CloudWatch and CloudTrail
- Exporting CloudWatch logs data to S3/ElasticSearch



CloudWatch Logs Insights

- CloudWatch Logs Insights enables the interactive search and analysis of the log data in Amazon CloudWatch Logs.
 - Queries can be used to help more efficiently and effectively respond to operational issues.
- CloudWatch Logs Insights includes a purpose-built query language with a few simple but powerful commands.
- CloudWatch Logs Insights automatically discovers fields in logs from AWS services such as Amazon Route 53, AWS Lambda, AWS CloudTrail, and Amazon VPC, and any application or custom log that emits log events as JSON.
- CloudWatch Logs Insights can be used to search older log data that was sent to CloudWatch on or after Nov 5th 2018
- A single request can query up to 20 log groups.

CloudWatch Logs and Amazon EC2 Instances

- CloudWatch Logs can be used to monitor **Logs from Amazon EC2 Instances in Real-time**
 - This is how CloudWatch Logs monitors applications and systems on these EC2 instances using log data.



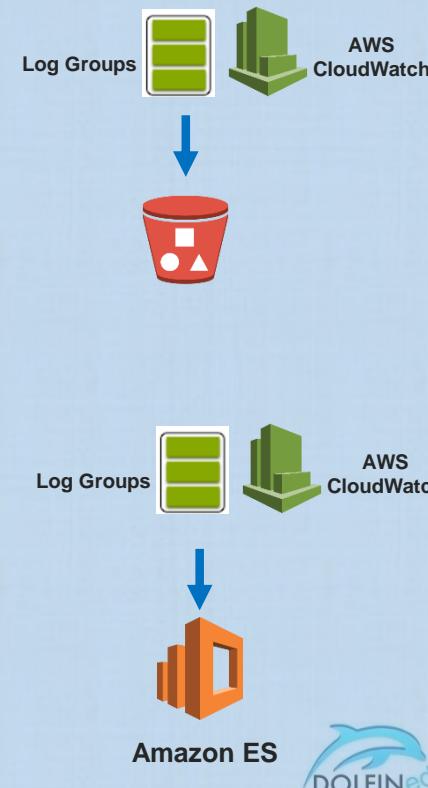
CloudWatch Logs - Monitoring AWS CloudTrail Logged Events

- You can create alarms in CloudWatch and receive notifications of particular API activity as captured by CloudTrail.
 - This notification can be used to perform troubleshooting.
 - The way it works is as follows
 - CloudTrail will send the Logs to CloudWatch logs,
 - CloudWatch Logs will have metric filters which will analyze based on criteria the Client sets,
 - Then CloudWatch Logs can send notifications, or a CloudWatch alarm action is triggered based on that.



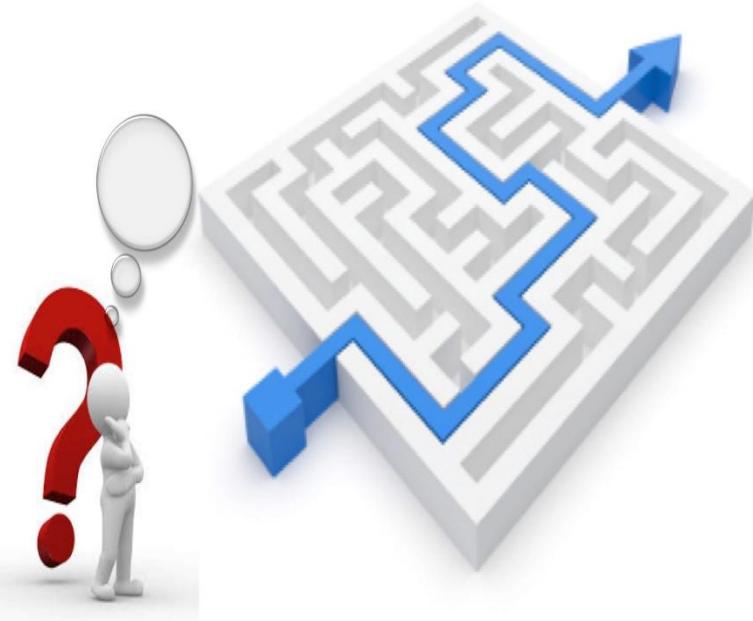
CloudWatch Logs Storage and Archival – Exporting data to S3/ES

- Log data from your log groups can be exported to an Amazon S3 bucket
 - This data can then be used in custom processing and analysis, or to load onto other systems.
 - The data can also be archived if not required or deleted using Amazon S3 Life cycle rules
 - Log data can take up to 12 hours to become available for export.
- For near real-time analysis of log data,
 - Consider using Amazon CloudWatch Logs insights, or
 - Real-time processing of Log data using subscriptions instead of exporting to S3.
- A CloudWatch Logs log group can be configured to stream data it receives to an **Amazon Elasticsearch Service (Amazon ES)** cluster in near real-time through a CloudWatch Logs subscription.



AWS CW Logs

- CloudWatch Agent
- CloudWatch Logs Subscriptions
- Sharing CloudWatch Logs log data



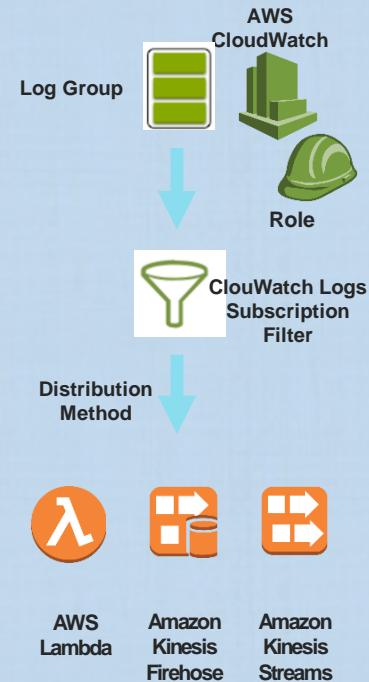
Unified CloudWatch Agent

- To collect logs from your Amazon EC2 instances and on-premises servers into CloudWatch Logs,
 - AWS offers both a new unified CloudWatch agent, and an older CloudWatch Logs agent.
 - AWS recommends the use of the unified CloudWatch agent.
 - The CloudWatch agent needs to be installed on the instance/server to collect Metrics and Logs from it
- The **CloudWatch Logs agent** makes it easy to quickly send both rotated and non-rotated log data off of a host and into the log service.
 - You can then access the raw log data when you need it.
- The new unified agent has the following advantages over the older agent.
 - You can collect both logs and advanced metrics with the installation and configuration of just one agent.
 - The unified agent enables the collection of logs from servers running Windows Server.
 - If you are using the agent to collect CloudWatch metrics, the unified agent also enables the collection of additional system metrics, for in-guest visibility.
 - The unified agent provides better performance.



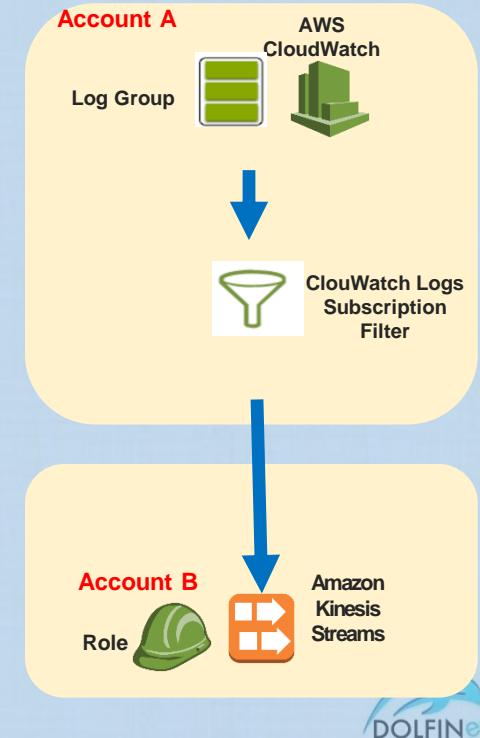
CloudWatch Logs - Real-time Processing of Log Data with Subscriptions

- Use subscriptions to get access to **a real-time feed of log events** from CloudWatch Logs and have it delivered to other services , for custom processing, analysis, or loading to other systems, such as:
 - An Amazon Kinesis stream,
 - Amazon Kinesis Data Firehose stream, or
 - AWS Lambda
- The subscription filter defines the filter pattern to use for filtering which log events get delivered to your AWS resource, as well as information about where to send matching log events to.
- CloudWatch Logs produces CloudWatch metrics about the forwarding of log events to subscriptions.



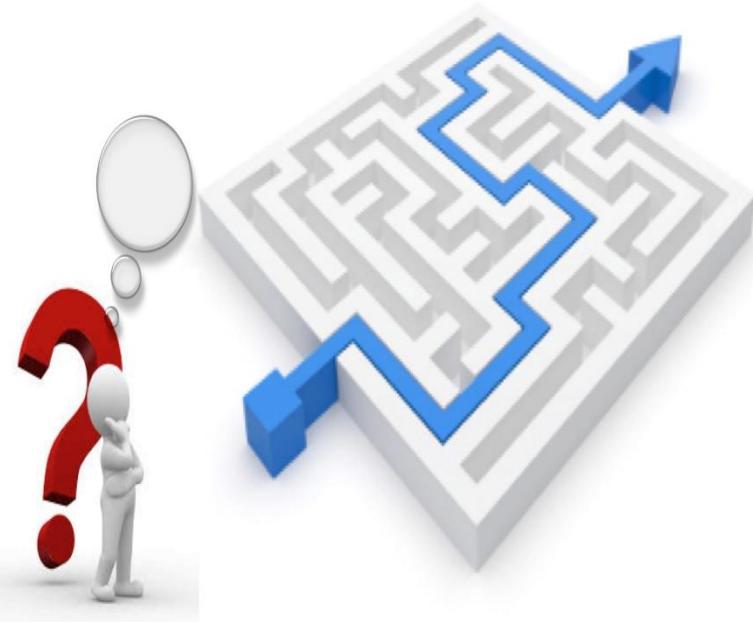
CloudWatch Logs - Cross-Account Log Data Sharing with Subscriptions

- You can collaborate with an owner of a different AWS account and receive their log events on your AWS resources,
 - Such as an Amazon Kinesis stream (this is known as cross-account data sharing).
- To achieve this, you need to define a log data sender and receiver accounts
 - Log sender then create subscription filters against their log streams with the destination being the one defined by the receiver.
 - Kinesis streams are currently the only resource supported as a destination for cross-account subscriptions.
- The log group and the destination must be in the same AWS region. However, the AWS resource that the destination points to can be located in a different region.



AWS CloudWatch

CloudWatch Events



CloudWatch Events

- Amazon CloudWatch Events delivers **a near real-time** stream of system events that describe changes in Amazon Web Services (AWS) resources.
- Using a simple rule, Events can be matched and routed to one or more target functions or streams.
- CloudWatch Events becomes aware of operational changes as they occur.
 - CloudWatch Events responds to these operational changes and takes corrective action as necessary,
 - Sending messages to respond to the environment,
 - Activating functions,
 - Making changes, and
 - Capturing state information.
- You can also use CloudWatch Events to schedule automated actions that self-trigger at certain times using cron or rate expressions.



CloudWatch Events Concepts – Events, Targets and Rules

Events

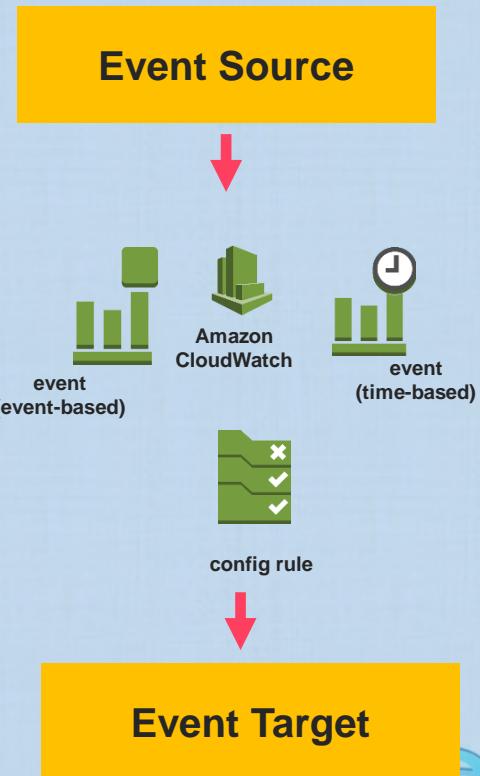
- An event indicates a change in your AWS environment (state change).
- Custom application-level events can also be created and published to CloudWatch Events.
- Scheduled events can be set up that are generated on a periodic basis (Example is creating EBS snapshots).

Targets

- A target receives events in JSON format and processes events.
- Targets include a long list of AWS services and **built-in targets**.

Rules

- A rule matches incoming events and routes them to targets for processing.
- **A single rule can route to multiple targets**, all of which are **processed in parallel**.
- Rules are not processed in a particular order.
 - This enables different parts of an organization to look for and process the events that are of interest to them.
- A rule can customize the JSON sent to the target, by passing only certain parts or by overwriting it with a constant.



CloudWatch Events – Possible Targets

For an updated list of AWS services that can be configured as Targets for CloudWatch events

<https://docs.aws.amazon.com/AmazonCloudWatch/latest/events/WhatIsCloudWatchEvents.html> :

Amazon EC2 instances

Streams in Amazon Kinesis Data Streams

Log groups in Amazon CloudWatch Logs

Systems Manager Run Command

AWS Batch jobs

Pipelines in CodePipeline

Amazon Inspector assessment templates

Amazon SQS queues

AWS Lambda functions

Delivery streams in Amazon Kinesis Data Firehose

Amazon ECS tasks

Systems Manager Automation

Step Functions state machines

CodeBuild projects

Amazon SNS topics

Built-in targets:

EC2 CreateSnapshot API call,
TerminateInstances API call.

EC2 RebootInstances API call,

EC2 StopInstances API call, and EC2

The default event bus of another AWS account



CloudWatch Events & Related AWS Services

The following services are used in conjunction with CloudWatch Events:

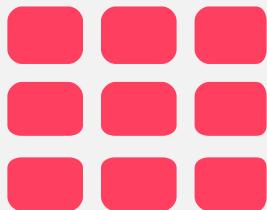
- **AWS CloudTrail**
 - When CloudTrail logging is turned on, CloudWatch Events writes log files to an S3 bucket.
 - Each log file contains one or more records, depending on how many actions are performed to satisfy a request.
- **AWS CloudFormation**
 - CloudWatch Events rules can be used in your AWS CloudFormation templates.
- **AWS Config**
 - AWS Config rules can be created to check whether your resources are compliant or noncompliant with your organization's policies.
- **AWS Identity and Access Management (IAM)**
 - Use it for authentication and access control to the CloudWatch events
- **Amazon Kinesis Data Streams**
- **AWS Lambda**
 - Use it to build applications that respond quickly to new information.

CloudWatch Events

Sharing CloudWatch Events between Accounts

- An AWS account can be set up to send events to another AWS account, or to receive events from another account.
 - This can be useful if the two accounts belong to the same organization, or belong to organizations that are partners or have a similar relationship.
- You can specify which AWS accounts it sends events to or receives events from.
- Both sender and receiver accounts must be in the same AWS region
- Events sent from one account to another are charged to the sending account as custom events.
 - The receiving account is not charged.





AMAZON SIMPLE STORAGE SERVICE (S3)

You Can Do It Too!



Amazon S3

Block vs Object Storage



Review Topic : Simple Storage Service (S3)

Object Storage

- Object storage stores the files as a whole and does not divide them
- In object storage, an object is:
 - The file/data itself
 - Its metadata (data created, modified, security attributes, content type...etc)
 - Object's global unique ID
- The Object Global Unique ID, Is a unique identifier for the object (can be the object name itself), and it must be unique such that it can be retrieved disregarding where its physical storage location is
- Examples for Objects,
 - Photos, Videos, Music, Static Web Content, Data backups (snapshots), Archival images
 - Any data that can be incrementally updated and will not have a lot of writes/updates (Like Snapshots of EBS, Storage Gateway, DB..etc)



Review Topic : Simple Storage Service (S3)

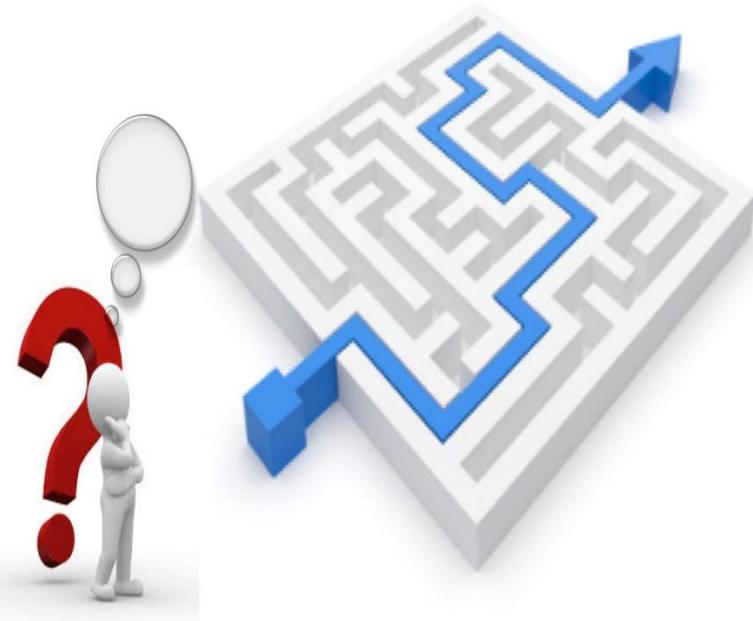
Object Storage

- Since object storage stores the object (file), its metadata, its global unique ID all together
 - It is ideally suited for Distributed storage architectures
 - Which also means, it can scale easily **using cheaper hardware (compared to Block storage)** by just adding additional storage units
- In object storage there is no limit on the type or amount of metadata in an object
- Object storage can guarantee high availability and durability
 - data copies are stored on multiple, geographically distributed locations
- Object storage can NOT be mounted as a drive, or directory, directly to an EC2 instance
- Object storage is a perfect solution for data growth storage problems



Amazon S3

S3 Introduction & Data Consistency Models



Review Topic : Simple Storage Service (S3)

S3

- S3 is a storage for the internet. It has a simple web services interface for simple storing & retrieval of any amount of data, anytime, from anywhere on the internet
- S3 is Object-based storage, and NOT a block storage
- S3 has a distributed data-store architecture, where objects are redundantly stored in multiple locations



Review Topic : Simple Storage Service (S3)

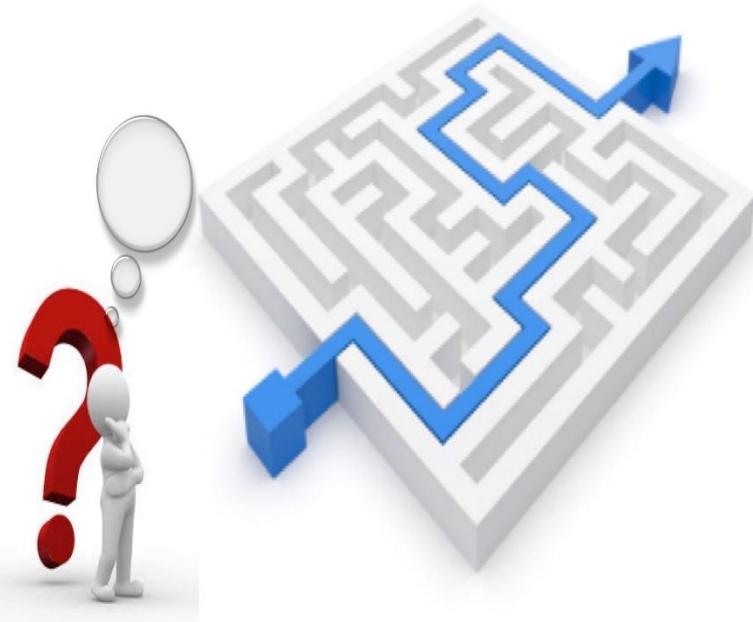
S3 Consistency Levels

- S3 Provides :
 - Read-after-write (Immediate or Strong) Consistency of PUTs of new objects (new object loads to S3)
 - A PUT is an HTTP request to store the enclosed data (with the request)
 - Eventual Consistency for overwrite PUTs and DELETEs (for Changes/updates to existing Objects in S3)
- Updates to an Object are atomic, i.e when you run a PUT for an object then you read (GET) that object, you will either get the updated object or the old one (before the update), you will never get partial or corrupt data



Amazon S3

Buckets



Review Topic : Simple Storage Service (S3)

S3 Buckets

- Data is stored in buckets
 - A bucket can be viewed as a container for objects
 - A bucket is a flat container of objects
 - It does not provide any hierarchical of objects (actual folders)
 - You can use object key (name) to mimic folders in a bucket when using the AWS console
- You can store unlimited objects in your bucket, but an **Object can NOT exceed 5 TB**
- You can create folders in your bucket (available through Console only)
- You can NOT create nested buckets (a bucket inside another)



Review Topic : Simple Storage Service (S3)

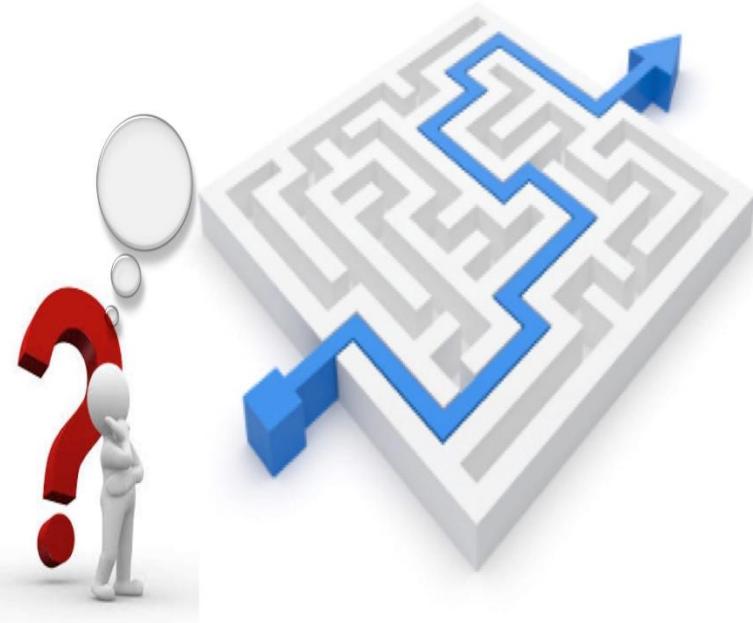
S3 Buckets

- Bucket ownership is not transferrable
- It is recommended to access S3 through SDKs or APIs (console internally uses APIs too)
- An S3 bucket is region specific
- You can have up to 100 Buckets (soft limit) per account
- An S3 bucket has properties including, among others:
 - Access Permissions
 - Versioning Status
 - Storage Class
- For better performance, lower latency, and to minimize costs, create the S3 bucket closer to your client DC (or source of data to be stored)



Amazon S3

Objects



Review Topic : Simple Storage Service (S3)

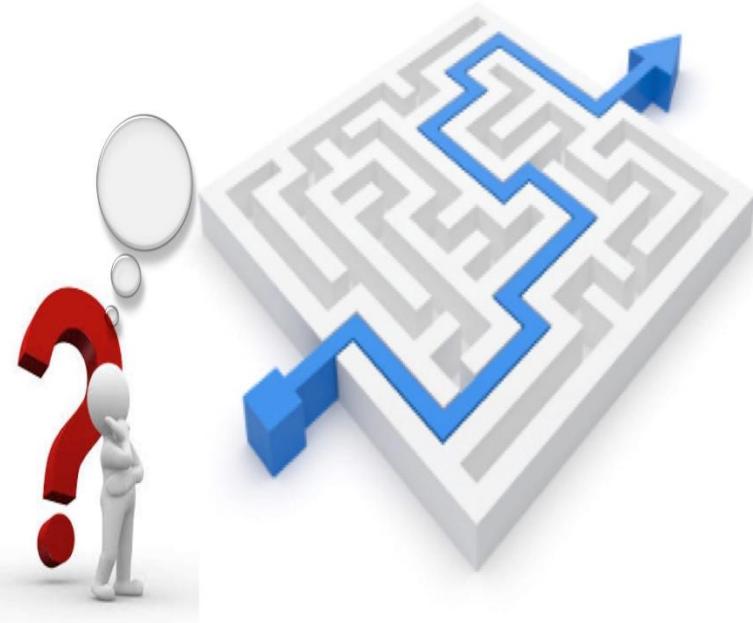
S3 Objects

- An object size stored in an S3 bucket can be **Zero bytes (0 bytes) up to 5 TB**
- Each object is stored and retrieved by a unique Key (ID or name)
- An Object in AWS S3 is uniquely identified and addressed through:
 - Service end point
 - Bucket name
 - Object Key (name)
 - Optionally, Object version
- Objects stored in a S3 bucket in a region **will NEVER leave that region** unless you specifically move them to another region, or enable Cross Region Replication
- S3 provides high data durability, Objects are redundantly stored on multiple devices across multiple facilities in an Amazon S3 region (where the bucket exists)



Amazon S3

Bucket Versioning



Review Topic : Simple Storage Service (S3)

S3 Bucket Versioning

- Bucket versioning is a S3 bucket sub-resource used to protect against accidental object/data deletion or overwrites
 - Versioning can also be used for data retention and archive
- You will be charged for all S3 storage costs for all Object versions stored
 - You can use versioning with S3 lifecycle policies to delete older versions, OR, you can move them to a cheaper S3 storage (or Glacier)
- Once you enable Bucket versioning on a bucket, it can not be disabled, however, it can be suspended
- When enabled, bucket versioning will protect existing and new objects, and maintains their versions as they are updated (edited, written to...)
 - Updating objects refers to PUT, POST, COPY, DELETE actions on objects
- Only S3 bucket owner can permanently delete objects once versioning is enabled



Review Topic : Simple Storage Service (S3)

S3 Bucket Versioning

- Bucket versioning states:
 - Enabled
 - Suspended
 - Un-versioned
- Versioning applies to all objects in a bucket and not partially applied
 - By default an HTTP GET retrieves the most recent version
 - Objects existing before enabling versioning will have a version ID of “NULL”
- If you have a bucket that is already versioned, then you suspend versioning, existing objects and their versions remain as is
 - However they will not be updated/versioned further with future updates



Review Topic : Simple Storage Service (S3)

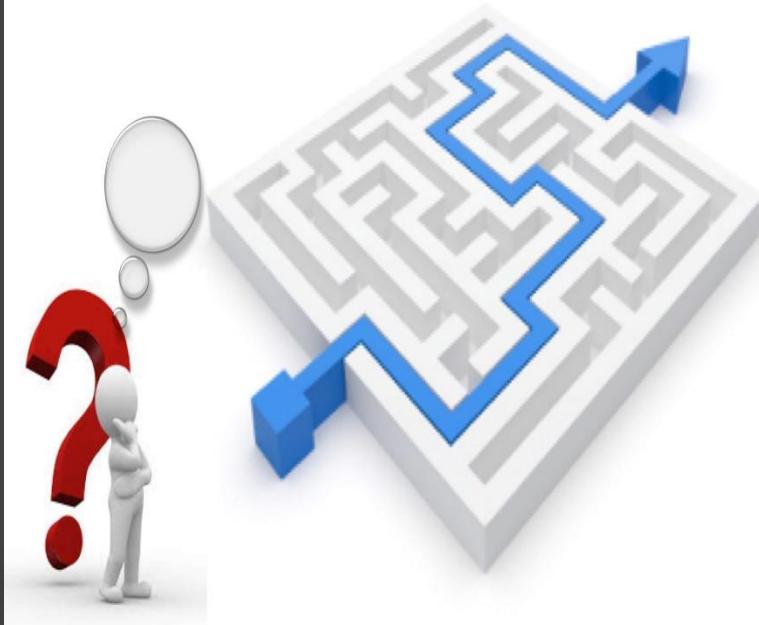
Bucket Versioning – MFA Delete

- Multi Factor Authentication (MFA) Delete is a versioning capability that adds another level of security in case your account is compromised
- This adds another layer of security for the following:
 - Changing your bucket's versioning state
 - Permanently deleting an object version
- MFA Delete requires:
 - Your security credentials
 - The code displayed on an approved physical or SW-based authentication device
- This provides maximum protection to your objects



Amazon S3

Copying / Uploading Objects &
Multipart Uploads



Review Topic : Simple Storage Service (S3)

Multipart Upload

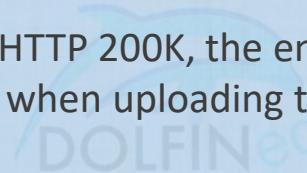
- Is used to upload an object (objects) in parts
 - Parts are uploaded independently and in parallel, in any order
- It is recommended for objects sizes of 100MB or larger
 - However, you can use for object sizes starting from 5MB up to 5TB
 - You must use it for Objects larger than 5GB
- This is done through the S3 Multipart upload API



Review Topic : Simple Storage Service (S3)

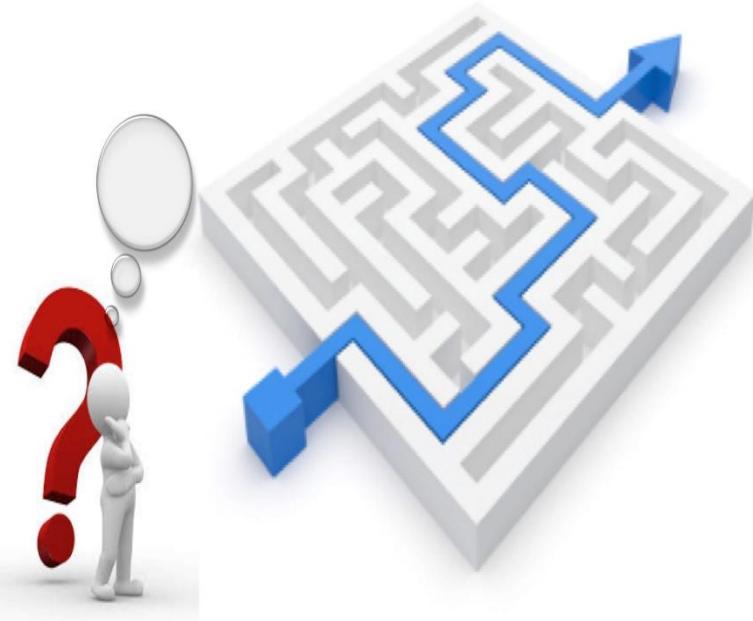
Successful Upload Acknowledgement

- When you successfully upload an object to S3, S3 returns a HTTP 200 OK message for a successful PUT Operation
- If uploading an Object and requesting SSE using Customer Provided Keys, and if the PUT operation is successful
 - Amazon S3 then returns the HTTP 200K, the encryption algorithm, and MD5 of the encryption key you specified when uploading the object.



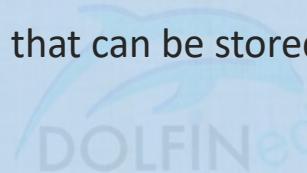
Amazon S3

Tiered Storage Classes



Storage Classes – Real time Frequent Access classes

- **S3 STANDARD**
 - performance-sensitive use cases (those that require millisecond access time) and frequently accessed data
- RRS – avoid not recommended
 - noncritical, reproducible data that can be stored with less redundancy



Storage Classes – Real time & Infrequent Access classes

- **STANDARD_IA**
 - long-lived and infrequently accessed data
 - Storing Backups
 - 128KB file size or bigger (you get charged for 128KB if you store smaller ones)
 - 30 days min charge/storage (you still can move it, delete it, but still will be charged for 30 days)
 - Primary or only copy of data that can't be recreated.
- **ONEZONE_IA**
 - Use if you can recreate the data if the Availability Zone fails, and for object replicas when setting cross-region replication (CRR).

Storage Classes – Intelligent Tiering

- It automatically moves data to the most cost-effective storage tier
- Is designed for storage cost optimization
- Use it to optimize storage costs automatically for long-lived data when access patterns are unknown.
- It does not cause performance impact or operational overhead.
- Is suitable for 128KB objects or larger
- For smaller objects, auto-tiering will not function.
- Min storage is 30 days
- It operates at a granular object level between and moves it between two access tiers, a frequent access tier and a lower-cost infrequent access tier (when the object is not accessed for 30 days), when access patterns change.
- A small monthly monitoring and automation fee applies per object.
- No retrieval fees
- No fees when objects are moved between the access tiers



Storage Classes – Archiving Classes – Non Real time access

- **Glacier** (Not for real time access of data)
 - Use for archives where portions of the data might need to be retrieved in minutes.
 - Minimum storage duration of 90 days
 - Can be accessed in as little as 1-5 minutes
 - You are charged for 90 days if you move the data before **the 90 days min storage duration**
- **DEEP_ARCHIVE** (Not for real time access of data)
 - Use for archiving data that rarely needs to be accessed.
 - Data stored in DEEP_ARCHIVE storage class has a **minimum storage duration period of 180 days**
 - Default retrieval time of 12 hours. If you have deleted, overwritten, or transitioned to a different storage class an object before the 180-day minimum, you are charged for 180 days.
 - DEEP_ARCHIVE is the lowest cost storage option in AWS.
 - Storage costs for DEEP_ARCHIVE are less expensive than using the GLACIER storage class.
 - You can reduce DEEP_ARCHIVE retrieval costs by using bulk retrieval, which returns data within 48 hours.



Glacier

- Is designed to sustain data loss in two facilities
- Glacier automatically encrypts data at rest using AES-256-bit symmetric keys and supports secure transfer of your data (In-transit) over Secure Sockets Layer (SSL)
- Glacier archives are visible and available only through AWS S3 not through AWS Glacier
- Glacier Retrieval Options:

Storage Class	Expedited	Standard	Bulk
GLACIER	1–5 minutes	3–5 hours	5–12 hours
DEEP_ARCHIVE	Not available	Within 12 hours	Within 48 hours



Storage Classes Comparison

<https://docs.aws.amazon.com/AmazonS3/latest/dev/storage-class-intro.html>

Storage Class	Designed for	Durability (designed for)	Availability (designed for)	Availability Zones	Min storage duration	Min billable object size	Other Considerations
STANDARD	Frequently accessed data	99.999999999%	99.99%	>= 3	None	None	None
STANDARD_IA	Long-lived, infrequently accessed data	99.999999999%	99.9%	>= 3	30 days	128 KB	Per GB retrieval fees apply.
INTELLIGENT_TIERING	Long-lived data with changing or unknown access patterns	99.999999999%	99.9%	>= 3	30 days	None	Monitoring and automation fees per object apply. No retrieval fees.
ONEZONE_IA	Long-lived, infrequently accessed, non-critical data	99.999999999%	99.5%	1	30 days	128 KB	Per GB retrieval fees apply. Not resilient to the loss of the Availability Zone.
GLACIER	Long-term data archiving with retrieval times ranging from minutes to hours	99.999999999%	99.99% (after you restore objects)	>= 3	90 days	None	Per GB retrieval fees apply. You must first restore archived objects before you can access them. For more information, see Restoring Archived Objects .
DEEP_ARCHIVE	Archiving rarely accessed data with a default retrieval time of 12 hours	99.999999999%	99.99% (after you restore objects)	>= 3	180 days	None	Per GB retrieval fees apply. You must first restore archived objects before you can access them. For more information, see Restoring Archived Objects .
RRS (Not recommended)	Frequently accessed, non-critical data	99.99%	99.99%	>= 3	None	None	None

Review Topic : Simple Storage Service (S3)

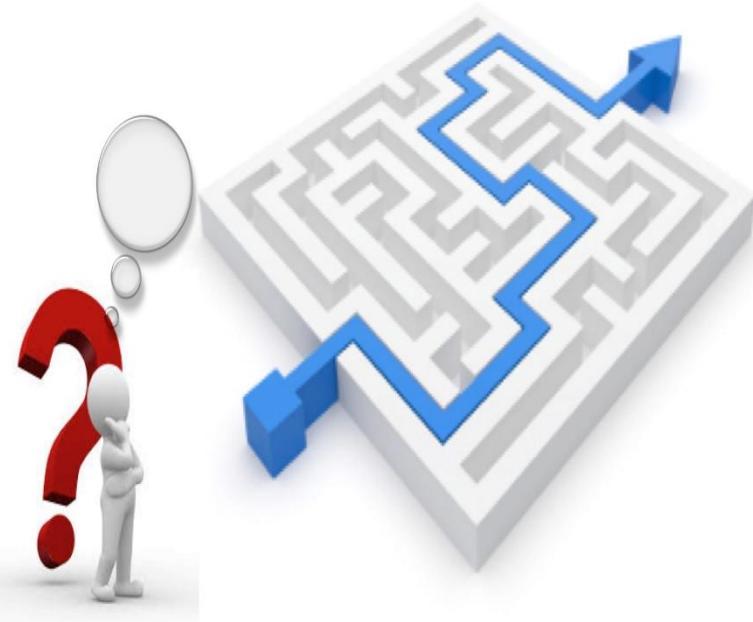
Glacier – Archiving Storage

- Glacier redundantly stores your data to multiple facilities, on multiple devices within each facility synchronously – You can receive SNS notification when upload is completed
 - Downloading archives is Asynchronous though
- You can upload an archive to Glacier of sizes from 1Byte to 40TB
 - Use multipart upload to load archives more than 5GB in size.
- Glacier awaits until the multiple facility upload is completed successfully before confirming a successful upload to the user
- Uploading archives is Synchronous while Downloading archives is Asynchronous
 - The contents of an archive, once uploaded, can not be updated
- Glacier does NOT archive object metadata, you need to maintain a client-side database to maintain this information about the objects (Basically which archives hold which objects)



Amazon S3

Life Cycle Policies



Review Topic : Simple Storage Service (S3)

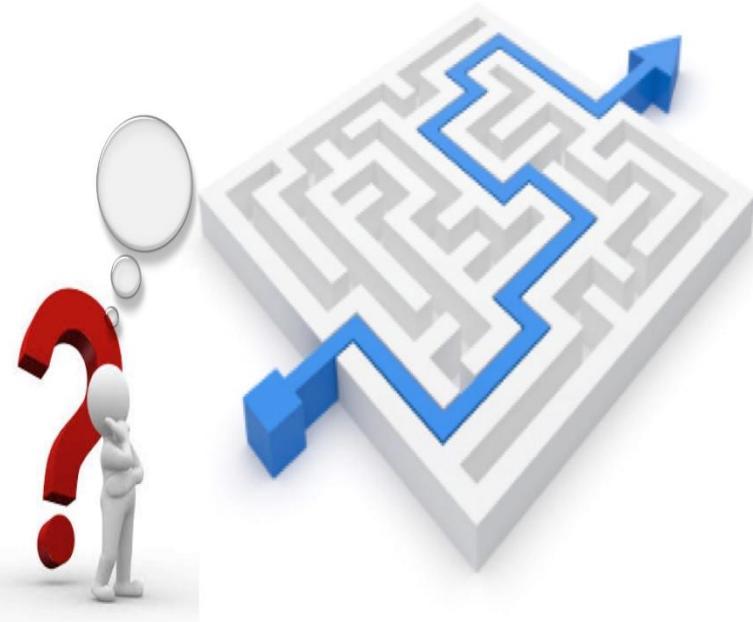
S3 Life Cycle Policy

- It can be applied to certain objects in a bucket folder, objects with a specific tag, or objects with a specific prefix
- The purpose is to primarily perform desired actions on contents (or some) of the bucket
- You can configure two actions:
 - Transition actions: to another S3 storage tier after a configured period
 - Expiration actions: Where an expiration duration for object(s) is set, S3 will then delete the expired objects on your behalf



Amazon S3

Server-Side Encryption (SSE)



Review Topic : Simple Storage Service (S3)

S3 Server-Side Encryption (SSE)

- This is primarily about encrypting data at rest on S3 buckets
- There are two main ways to encrypt data stored on S3 buckets:
 - Client side encryption:
 - Where the client (user or application) encrypts data on the Client side , then transfer the data encrypted to S3 buckets (hence data is encrypted In-transit and at Rest)
 - Server Side Encryption (SSE):
 - Data is encrypted by the S3 service before it is saved to S3 storage disks
 - Data is decrypted when you download it
- User access to S3 bucket objects is the same whether the data is encrypted or not on S3 buckets (as long as the user has permission to access the data)



Review Topic : Simple Storage Service (S3)

S3 Server-Side Encryption using AWS S3 Managed Keys (SSE-S3)

- Each object is encrypted by a unique key, then the encryption key itself is encrypted using a master key
- S3 regularly rotates the master key
- Uses AES-256 bits



Review Topic : Simple Storage Service (S3)

S3 Server-Side Encryption using AWS KMS Managed Keys (SSE-KMS)

- KMS uses Customer Master Keys (CMKs) stored in KMS to encrypt your S3 objects
 - You can continue to use the automatically created default CMK key for encryption
 - OR, You can select a CMK that you created separately using AWS Key Management Service.
 - Creating your own CMK will allow you to create, rotate, disable, and define access controls,
 - KMS provides an **audit trail** for when the key was used and by whom
 - Also allows you to audit the encryption keys used to protect your data.
 - The CMK used and the bucket must be in the same region
- Separate permissions for an envelope key that protects/encrypts your object encryption keys
- This service is chargeable



Review Topic : Simple Storage Service (S3)

S3 Server-Side Encryption using Customer Provided Keys (SSE-C)

- Server-Side Encryption using Client provided keys
- Client manages the keys, S3 service manages encryption
- AWS does not store the client provided encryption key(s), so if the client loses the key(s), they can't access the object, basically they lose the data.
 - Instead, S3 stores a randomly slated HMAC (hash) of the key to validate future requests
 - Client must provide the key with every request
 - S3 validates the key using the HMAC, and if it matches, will decrypt the data using the provided key, then return the object to the Client



Amazon S3

Static WebSite Hosting & Redirection



Review Topic : Simple Storage Service (S3)

S3 Static Website Hosting

- S3 buckets can be used to host static content (not dynamic) websites
 - Static content refers to content that does not need to run server-side scripts such as PHP, JSP, or ASP.NET.
 - Content can be HTMP pages, images, videos, Client-side scripts such as Javascript
- AWS S3 Hosted websites scale automatically to meet demand
- You can use your own domain with S3 hosted static websites (Route53 CNAME/Alias)
- There is no additional charge for hosting static websites on S3
- S3 hosted static websites can enable redirection for the whole domain, pages within a domain, or specific objects.



Review Topic : Simple Storage Service (S3)

S3 Static Website Hosting

- The website S3 endpoint URL is :
`http://<S3bucketname>.S3-website-<AWS-Region>.amazonaws.com OR`
`tp://<S3bucketname>.S3-website.<AWS-Region>.amazonaws.com`
- Amazon website endpoints DO NOT support HTTPS
- You need to allow public read access to all the bucket used for hosting the static website.
 - Use bucket policy or ACLs for this
- S3 Requester Pays bucket do not work with website endpoint
 - Such a request (requester pays) to an S3 website endpoint will return
 - HTTP 403 Access Denied



Review Topic : Simple Storage Service (S3)

Key difference between S3 REST API and S3 Website Endpoints

Key Difference	REST API Endpoint	Website Endpoint
Access control	Supports both public and private content.	Supports only publicly readable content.
Error message handling	Returns an XML-formatted error response.	Returns an HTML document.
Redirection support	Not applicable	Supports both object-level and bucket-level redirects.
Requests supported	Supports all bucket and object operations	Supports only GET and HEAD requests on objects.
Responses to GET and HEAD requests at the root of a bucket	Returns a list of the object keys in the bucket.	Returns the index document that is specified in the website configuration.
Secure Sockets Layer (SSL) support	Supports SSL connections.	Does not support SSL connections.



Review Topic : Simple Storage Service (S3)

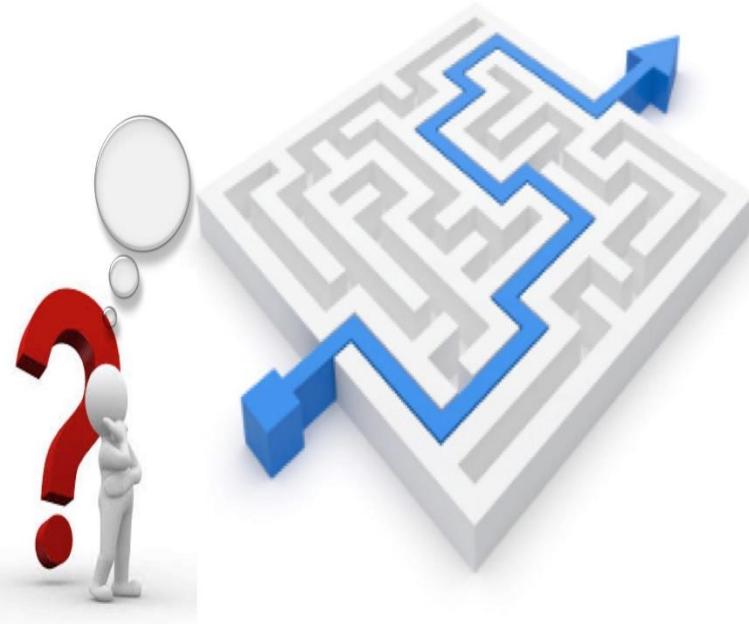
S3 Static Website Hosting – Redirection

- If your bucket is configured for website hosting, you can redirect requests for an object to another object in the same bucket or to an external URL/Domain
- Redirection is a bucket level operation
- You can redirect (re-route) all requests (at the bucket level) to another website
- You can do conditional redirection based on object or prefixes in a request
- You can also redirect requests that would return an error



Amazon S3

Using Pre-Signed URLs
to share objects



Review Topic : Simple Storage Service (S3)

Sharing S3 Objects via Pre-Signed URLs

- By default, all objects are private and only the object owner has permission to access them.
- Pre-signed URLs can be used to provide temporary access to a specific object (sharing the object), to those who do not have AWS credentials,
 - Example is customers who bought website subscription, or product subscription online
- Generally speaking, to generate a pre-signed URL to upload (or download) an object, this can only be done if the creator (IAM user, Root Account, EC2 instance/App, some with STS tokens) of the pre-signed URL has the necessary permissions to upload (or download) that object.
- The pre-signed URL will grant access to a specific object, in a specific bucket, for a specific HTTP method (GET, PUT...etc), for a limited time.
- This will work even if the bucket and object are both private



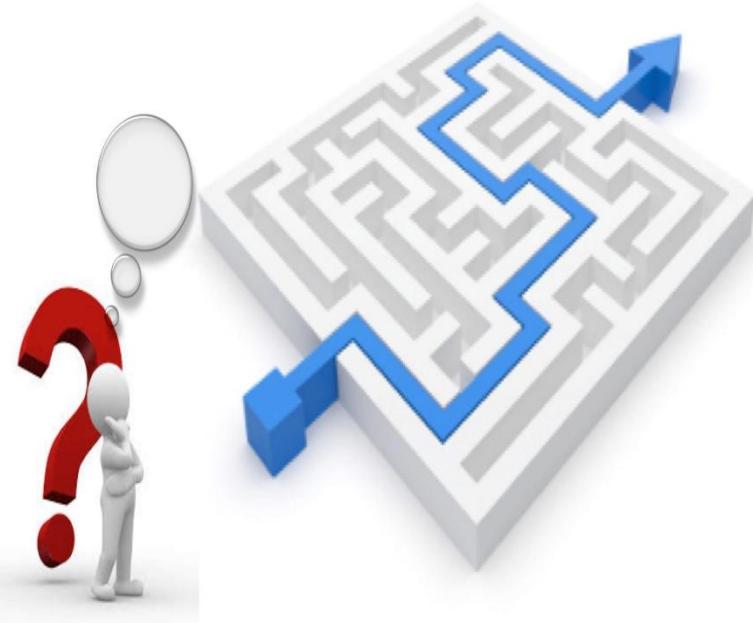
Review Topic : Simple Storage Service (S3)

Sharing S3 Objects via Pre-Signed URLs

- One way to generate these Signed URLs can be through using Lambda function behind an API Gateway
- The API gateway will pass the request after authentication to the Lambda function
- The Lambda function with the right permissions on the objects/bucket will request and return the Signed URL through the S3 API.

AWS S3

Object Locking



S3 Object Lock

- With Object Lock you can store objects in S3 buckets following the Write Once Read Many (WORM) model.
 - Hence it does help S3 users/customers meet any regulatory requirements that require WORM storage.
 - It also provides another layer of protection for objects
- It is used to protect an object(s) from being deleted or overwritten for a period of time or indefinitely
- Object lock can be configured at the bucket level (as default configuration for the bucket) or at the object version level
- Object Lock requires versioning to be enabled on the bucket
 - It will be enabled automatically when enabling Object lock (if versioning was not enabled before hand)
- S3 buckets with Object lock enabled can NOT be used as destination buckets for S3 server access logging



S3 Object Lock – Two ways to Object Lock

The following two ways can be used at the Object level to Lock objects

- **Retention Period:**
 - Specifies a period of time during which the object remains WORM protected (locked)
 - Applies at the object version level. Different object versions can have different retention periods
 - Retention period (Retain Until date) is placed in the object version's metadata
 - You can extend the retention period of a locked object by submitting a new lock request
 - This can be configured at the Object level or to all Objects in a bucket through the Bucket default Retention settings.
- **Legal Hold:**
 - It does the same function as retention period, but has no expiration.
 - It will remain in place until it is removed
 - It is completely independent from Retention period
 - Can be placed or removed by any user with the s3:PUTObjectLegalHold permission
- An object version can have both, either one, or none of the above ways applied.



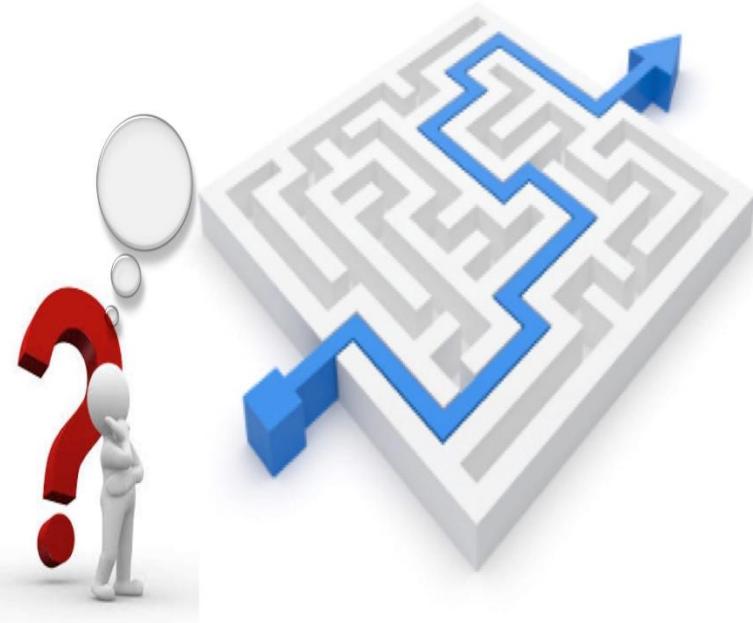
S3 Object Lock – Retention Modes

- **Governance Mode:**
 - Users can not overwrite or delete an object version or alter its lock settings unless they have special permissions to do so.
 - Protects the objects from being deleted or overwritten to by most users, but not all (those who do not have the S3:BypassGovernanceRetention permission)
 - Users with the right permissions can change Retention settings or even delete the object if required
 - Users with the above permission must include the **x-amz-bypass-governance-retention:true** as a request header to override governance mode
 - It can be used to test Retention before going into Compliance mode
- **Compliance mode:**
 - A protected object can not be deleted or overwritten by any user, not even the root user
 - The retention mode can not be changed, and the retention period can not be shortened
 - It ensures that the object is not deleted or overwritten for the entire retention period duration
- An object can have either mode, but not both.



Amazon S3

Same and Cross Region Replication
(SRR and CRR)



Review Topic : Simple Storage Service (S3)

S3 Replication (SRR & CRR)

- Is a bucket level replication which enables
 - Automatic , Asynchronous copying of Objects across buckets.
 - The source and destination buckets can belong to the same or different AWS accounts
 - They source and destination buckets can be in the same or different AWS regions
- Types of Object Replication:
 - Same Region Replication (SRR) & Cross Region Replication (CRR)
- You can configure it from AWS Console, CLI, SDKs, APIs
- You can request to replicate all or a subset of objects with specific key name prefixes.
- You can configure AWS S3 Replication with S3 Lifecycle management rules
- Requires versioning to be enabled on both source and destination buckets



Review Topic : Simple Storage Service (S3)

S3 Replication

- The replicas will:
 - Be exact replicas of the source bucket objects.
 - Share the same key names and metadata (Creation time, version ID, ACL, Storage Class, User-defined metadata)
- Optional:
 - You can specify a different storage class for object replicas while creating the replication configuration
 - If you did not specify it, the same storage class as the source object class will be used for the replica in the destination bucket
- AWS S3 will encrypt data in-transit across regions using SSL



Review Topic : Simple Storage Service (S3)

S3 Replication – Use cases

- SRR Can be used for:
 - Aggregating log data into a single bucket (from same or different account buckets)
 - You can replicate between different environments that work on the same data (ex. Production and testing)
 - Replicate critical data within the same region for data sovereignty laws
- CRR can be used for:
 - Meeting compliance requirements, where you need to store copy of your data a distance away
 - Minimize latency to your users/customers by availing the data in other AWS regions closer to them
 - Availing the data for processing by any sort of clusters in two different regions



Review Topic : Simple Storage Service (S3)

S3 Replication Requirements

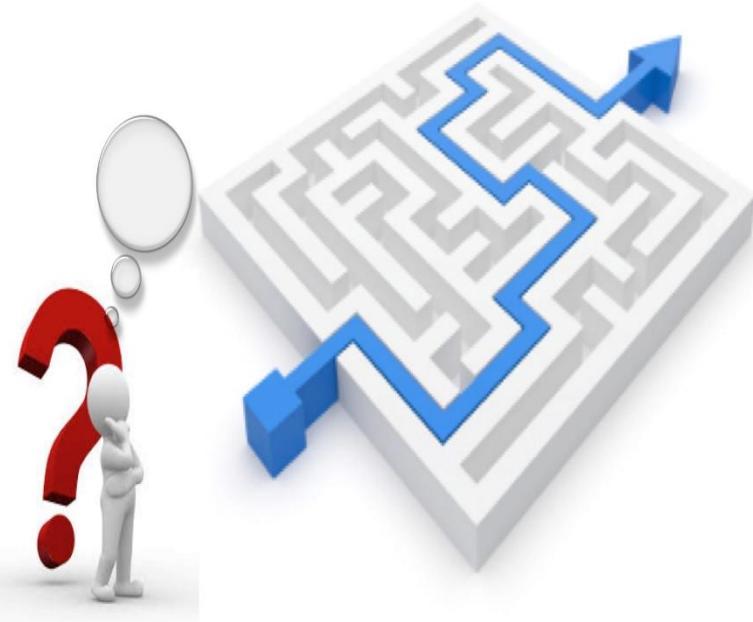
It requires:

- Both source and destination buckets MUST have Versioning enabled
- Replication can happen to only one destination bucket
- If the source has S3 object lock enabled, the destination bucket must have that enabled too.
- AWS S3 must have permission (IAM Role) to replicate objects from the source bucket to the destination bucket on your behalf.



Amazon S3

Same and Cross Region Replication (SRR and CRR) – Part 2



Review Topic : Simple Storage Service (S3)

S3 Replication – What is Replicated

- Any new objects created after you add a replication configuration, and changes to existing objects
- Object metadata
- Objects encrypted using SSE-S3, and optionally those encrypted with SSE-KMS
- Unencrypted objects
- Amazon S3 replicates only objects in the source bucket for which the bucket owner has permission to read objects and read ACLs.
- Any object ACL updates are replicated
- Object lock retention information if there is any
- S3 replicates object tags, if any.



Review Topic : Simple Storage Service (S3)

S3 Replication – What is Replicated (DELETE Operation)

If you delete an object from the source bucket, the cross-region replication behavior is as follows:

- If a DELETE request is made without specifying an object version ID,
 - AWS S3 adds a delete marker, which cross-region replication replicates to the destination bucket.
- If a DELETE request specifies a particular object version ID to delete,
 - AWS S3 deletes that object version in the source bucket, but it does not replicate the deletion in the destination bucket
 - It does not delete the same object version from the destination bucket
 - » This behavior protects data from malicious deletions.



Review Topic : Simple Storage Service (S3)

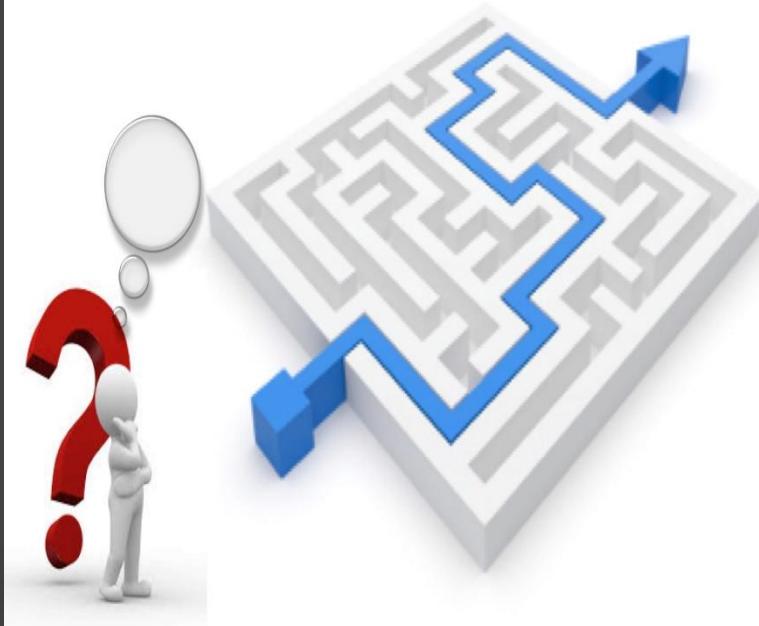
S3 Replication – What is NOT replicated

- AWS S3 will not replicate objects that existed before you added replication configuration
 - You can use Copy API to copy existing bucket data
- Objects created with SSE-C or SSE-KMS (if not enabled) are not replicated
- If the object owner is different from the source bucket owner, and bucket owner does not have permissions to the object, AWS S3 will not replicate these objects
- Actions performed by life cycle configuration
- Objects in the source bucket that were replicated from another bucket into the source
- Updates to bucket-level sub-resources are not replicated (life cycle rules, notifications)
 - This allows you to have different bucket configurations on the source and destination buckets.



Amazon S3

Cross Origin Resource Sharing (CORS)



Simple Storage Service (S3)

S3 Cross-Origin Resource Sharing (CORS)

CORS is a security feature of modern web browsers

- It is a way by which a client web application that are loaded in one domain can interact (using HTTP methods) with resources from another domain
- CORS can be used to allow web applications to access resources on your S3 buckets/resources
 - An example: You can host web fonts on your S3 bucket, then configure your bucket to allow CORS requests for web fonts.
 - Other domains (web pages) will issue CORS requests to load web fonts from your S3 bucket
 - Another example, CORS can be used to allow clients of a website hosted on S3 (<http://bucket.s3-website.us-east-1.amazonaws.com>) to use JavaScript on these pages to make authenticated GET and PUT requests from the S3 bucket API endpoint (`bucket.s3.us-east-1.amazonaws.com`).
 - Without CORS instructing the client web browsers to allow this, it would normally get blocked by the web browser.



Simple Storage Service (S3)

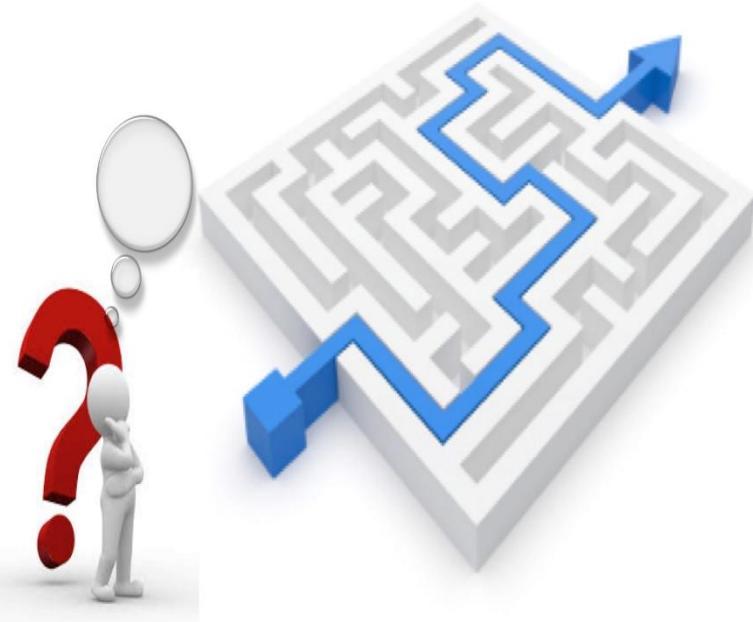
S3 Cross-Origin Resource Sharing (CORS)

- CORS allows to easily build web applications that use JavaScript and HTML5 to interact with resources in Amazon S3,
 - This enables:
 - The implementation of HTML5 drag and drop uploads to Amazon S3,
 - Show upload progress, or
 - Update content.
- CORS configuration on an S3 bucket, is an XML document with CORSrules that identify:
 - The origins (domains) allowed to access the bucket
 - The HTTP methods/operations permissible (GET, PUT, POST, DELETE...)



Amazon S3

Transfer Acceleration



Review Topic : Simple Storage Service (S3)

S3 Transfer Acceleration

- Is used to accelerate object uploads/downloads to S3 buckets from users over long distances
- Typical use case is when uploading objects to your S3 bucket happens from users across the world, over the internet
- Transfer acceleration is as secure as direct upload to S3 over the internet
- It utilizes AWS Cloudfront's edge location nearest to the upload source (user), once data arrives at the edge location, it gets routed to the destination S3 bucket over an optimized network path
- You can use transfer accelerated buckets for all S3 Operations except GET Service (list buckets), PUT Bucket (create bucket), DELETE Bucket
 - It also does not support cross region Copy



Review Topic : Simple Storage Service (S3)

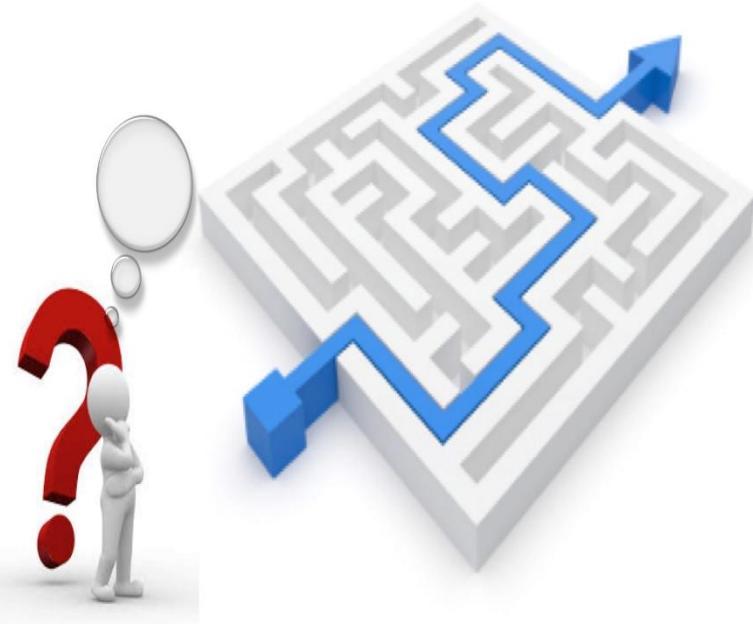
S3 Transfer Acceleration

- Using Transfer acceleration incurs a charge
 - AWS checks for speed enhancements, and if no enhancement is provided, client does not get charged for using Transfer Acceleration
- No data is saved at Cloudfront edge locations (not cached)
- You can use multipart uploads with Transfer Acceleration
- Transfer Acceleration is HIPAA Compliant



AWS S3

Performance Considerations



S3 Performance Considerations

- Amazon S3 automatically scales to high request rates.
- Applications can achieve
 - 3,500 PUT/COPY/POST/DELETE and
 - 5,500 GET/HEAD requests per second per prefix in a bucket.
- There are no limits to the number of prefixes in a bucket.
 - You can increase your read or write performance by parallelizing reads.
 - For example, if you create 10 prefixes in an Amazon S3 bucket to parallelize reads, you could scale your read performance to 55,000 read requests per second.
- S3 maintains an index of object key names in each region
 - Object keys (names) are stored in an order across multiple partitions of this index
 - The object key name dictates which partition the key is stored in
- Sequentially named objects are more likely to be saved in the same partition



Amazon Glacier Select

- At high request rates to access sequentially named objects, S3 will place a higher load on the available I/O of the partition hosting these keys (names) which will impact performance
- Some data lake applications on Amazon S3 scan millions or billions of objects for queries that run over petabytes of data.
 - These data lake applications achieve single-instance transfer rates that maximize the network interface use for their Amazon EC2 instance, which can be up to 100 Gb/s on a single instance.
 - These applications then aggregate throughput across multiple instances to get multiple terabits per second.

The logo consists of the word "DOLFIN" in a bold, sans-serif font, with "ed" in a smaller, lighter font to the right. A stylized blue dolphin is positioned to the left of the text, with its body forming the letter "D".

S3 Performance Considerations – Best Practices

- Using Caching for Frequently Accessed Content
- Timeouts and Retries for Latency-Sensitive Applications
- Horizontal Scaling and Request Parallelization for High Throughput
 - For high-throughput transfers, Amazon S3 advises using applications that use multiple connections to GET or PUT data in parallel.
- Using Amazon S3 Transfer Acceleration to Accelerate Geographically Disparate Data Transfers
- Use Byte-Range Fetches

S3 Performance Considerations – KMS Transaction Limitation

- After configuring CRR, and when many new objects with AWS KMS encryption are added, throttling (HTTP 503 Slow Down errors) can be experienced.
- Throttling occurs when the number of AWS KMS transactions per second exceeds the current limit.
- If the Amazon S3 workload uses server-side encryption with AWS KMS (SSE-KMS),
 - AWS KMS service has limits for the number of encrypt and decrypt requests (or generating keys)
 - It can cause throttling/performance issues for higher request rates.

S3 Performance Considerations - Caching

- To achieve higher transfer rates over a single HTTP connection or single-digit millisecond latencies,
 - Use Amazon **CloudFront** or Amazon **ElastiCache** for caching with Amazon S3.
- Amazon CloudFront transparently caches data from Amazon S3 in a large set of geographically distributed points of presence (PoPs). This can result in high performance delivery of popular Amazon S3 content.
- Amazon ElastiCache is a managed, in-memory cache.
 - With ElastiCache, Amazon EC2 instances can be provisioned to cache objects in memory.
 - This caching results in orders of magnitude reduction in GET latency and substantial increases in download throughput.
- Elemental MediaStore is a caching and content distribution system built for **video workflows** and media delivery from Amazon S3.
 - It provides end-to-end storage APIs for video,
 - It is recommended for performance-sensitive video workloads.



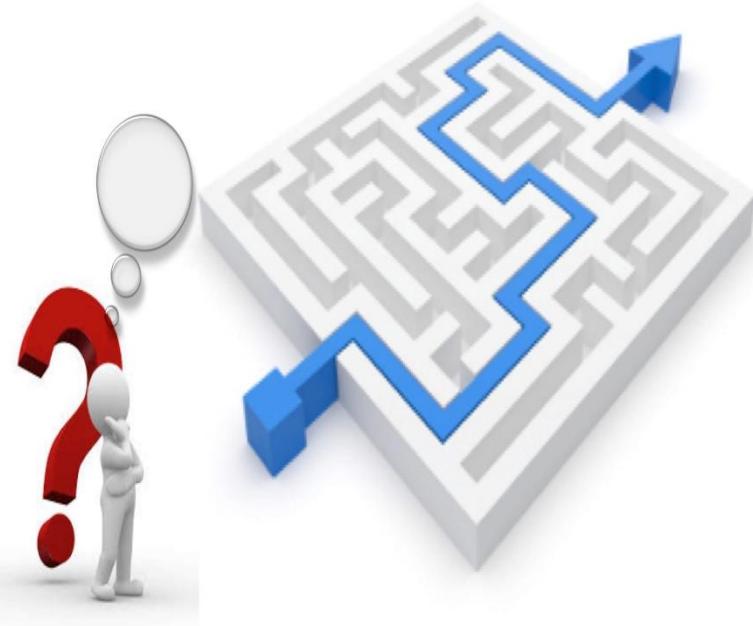
S3 Performance Considerations – Byte Range fetches

- Relies on the Range HTTP header in a GET Object request to fetch a byte-range from an object, transferring only the specified portion.
- If concurrent connections to Amazon S3 are used to fetch different byte ranges from within the same object.
 - This will help achieve higher aggregate throughput compared to a single whole-object request.
- In addition, fetching smaller ranges of a large object allows the application to improve retry times when requests are interrupted.



AWS S3

S3 and Glacier SELECT



Amazon S3 SELECT

- With Amazon S3 SELECT, simple structured query language (SQL) statements can be used to filter the contents of Amazon S3 objects and retrieve just the required subset of data.
- By using Amazon S3 SELECT, the amount of data that Amazon S3 transfers can be reduced, which reduces the cost and latency to retrieve this data.
 - Amazon S3 SELECT works on objects stored in CSV, JSON, or Apache Parquet format.
 - It also works with objects that are compressed with GZIP or BZIP2 (for CSV and JSON objects only), and server-side encrypted objects.
- You can specify the format of the results as either CSV or JSON, and you can determine how the records in the result are delimited.
- You pass SQL expressions to Amazon S3 in the request. Amazon S3 SELECT supports a subset of SQL.
 - You can perform SQL queries using AWS SDKs, the SELECT Object Content REST API, the AWS Command Line Interface (AWS CLI), or the Amazon S3 console.



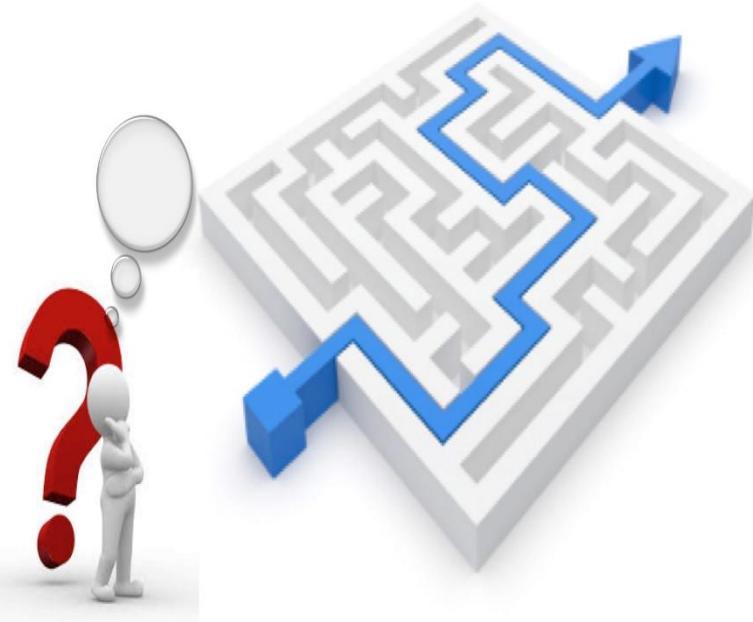
Amazon Glacier Select

- Using Amazon S3 SELECT and Glacier SELECT you can run queries and custom analytics on your data that is stored in Glacier, without having to restore your entire object to Amazon S3.
 - Amazon S3 SELECT and Glacier SELECT support only the SELECT SQL command.
- Querying Archived Objects can be performed (filtering operations) using simple SQL statements directly on your data that is archived by Amazon S3 to Glacier.
- When you provide an SQL query for an archived object, SELECT runs the query in place and writes the output results to an S3 bucket.
- When you perform SELECT queries, Glacier provides three data access tiers—expedited, standard, and bulk.
 - All of these tiers provide different data access times and costs, and you can choose any one of them depending on how quickly you want your data to be available.



Amazon S3

Monitoring and Event Notification



Review Topic : Simple Storage Service (S3)

S3 Monitoring via CloudWatch

- With AWS CloudWatch you can monitor multiple metrics.
- CloudWatch can take actions based on a single metric
- For S3, metrics that can be monitored by CloudWatch include:
 - S3 Requests, Bucket Storage, Bucket Size, All requests, HTTP 4XX, 5XX errors among others
- Daily CloudWatch, Bucket level, storage metrics is turned on by default at no additional cost
 - 1 minute CloudWatch metric can be configured both at the bucket and object level



Review Topic : Simple Storage Service (S3)

S3 API log monitoring via CloudTrail

- CloudTrail captures all API requests made to S3 API
- By default, AWS CloudTrail logs bucket level actions, however you can configure it to log the object level actions (such as DELETE, GET, PUT, POST object events)
- CloudTrail delivers these logs to a S3 bucket that you configure
- CloudTrail collected information includes:
 - Who made the request
 - When it was made
 - For what...etc



Review Topic : Simple Storage Service (S3)

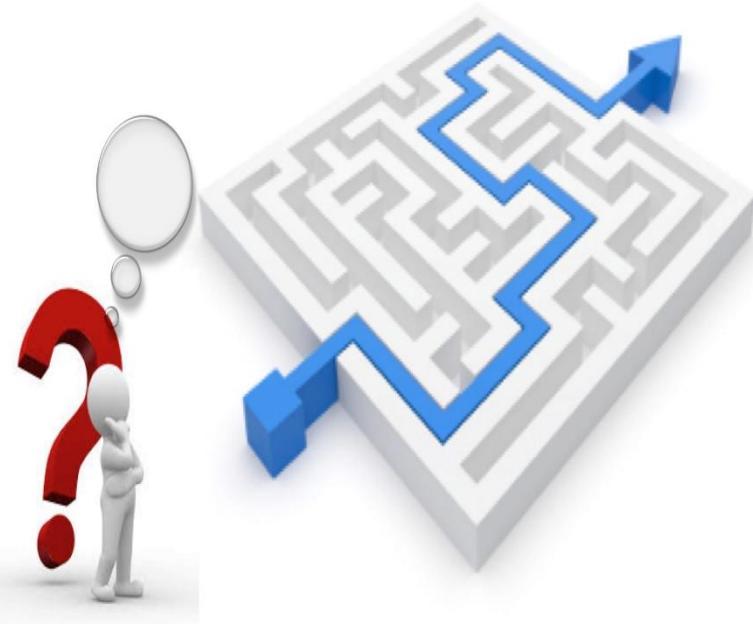
S3 Event Notification

- When certain bucket events occur, AWS S3 Can be configured to automatically send notifications to one of the following AWS Services:
 - SNS Topic(s)
 - SQS queue
 - or AWS Lambda function(s)
- This is a bucket level configuration, and you can configure multiple events as required
- No extra charge is incurred for enabling event notifications to your bucket, however, SNS, SQS, Lambda charges apply
- You can set notifications for, Create object, Object delete, Object Delete marker created.. and many other bucked related events



AWS S3

Batch Operations



Batch Operations

- Amazon S3 batch operations can be used to perform large-scale batch operations on Amazon S3 objects.
- Amazon S3 batch operations can execute a single operation on lists of Amazon S3 objects that you specify.
- A single job can perform the specified operation on billions of objects containing exabytes of data.
- Amazon S3 tracks progress, sends notifications, and stores a detailed completion report of all actions, providing a fully managed, auditable, serverless experience.
- You can use Amazon S3 batch operations through the AWS Management Console, AWS CLI, AWS SDKs, or REST API.



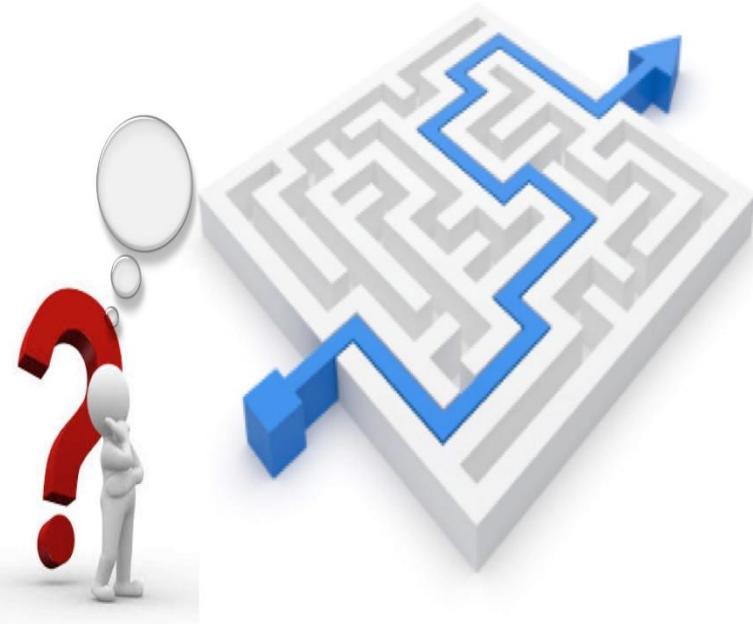
Batch Operations

- Use Amazon S3 batch operations to:
 - Copy objects
 - Set object tags
 - Access control lists (ACLs).
 - Initiate object restores from Amazon S3 Glacier
 - Invoke an AWS Lambda function to perform custom actions using your objects.
- These operations can be performed on a custom list of objects, or
- Amazon S3 inventory report can be used to make generating larger lists of objects easy.
- Amazon S3 batch operations use the same Amazon S3 APIs that you already use with Amazon S3, so you'll find the interface familiar.



AWS S3

Server Access Logging



Server Access Logging

- Server access logging provides detailed records for the requests that are made to a bucket.
- These logs can be useful for many applications for security purposes or access auditing, it can help analyze the S3 bill, and can help understand the customer base.
- It is disabled by default.
- Each access log record provides details about a single access request, and it provide information such as the requester, bucket name, request time, request action, response status, and an error code, if relevant.
- There is not extra charge for using the feature aside from storage costs
- You need to specify the source bucket, and the target bucket where the logs will be delivered.



Server Access Logging

To enable it

- Turn on the log delivery on the source bucket
- Grant the Amazon S3 Log Delivery group write permission on the target bucket

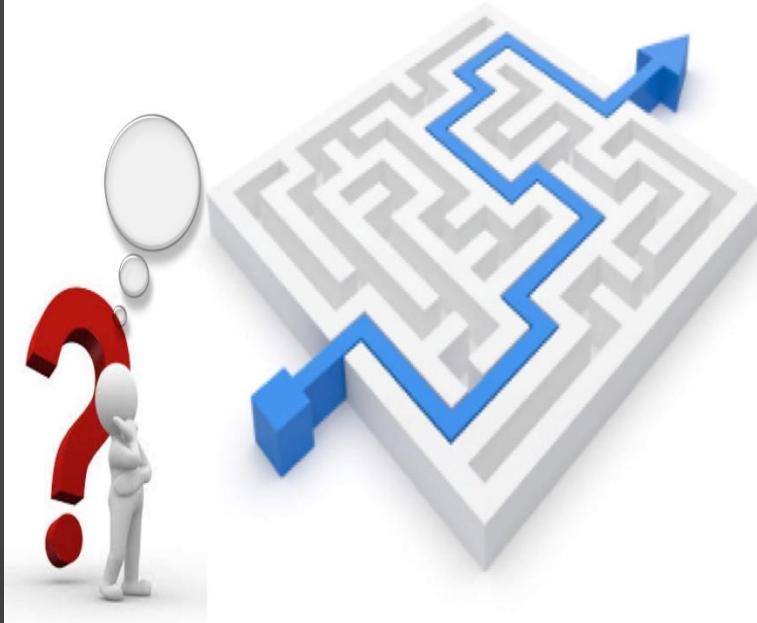
Notes

- Both the source and target S3 buckets must be owned by the same AWS account and must both be in the same AWS region
- Amazon S3 only supports granting permission to deliver access logs via bucket ACL, not via bucket policy.
- Adding deny conditions to a bucket policy may prevent Amazon S3 from delivering access logs.
- Default bucket encryption on the destination bucket may only be used if AES256 (SSE-S3) is selected.
 - SSE-KMS encryption is not supported currently



Amazon S3

Requester Pays



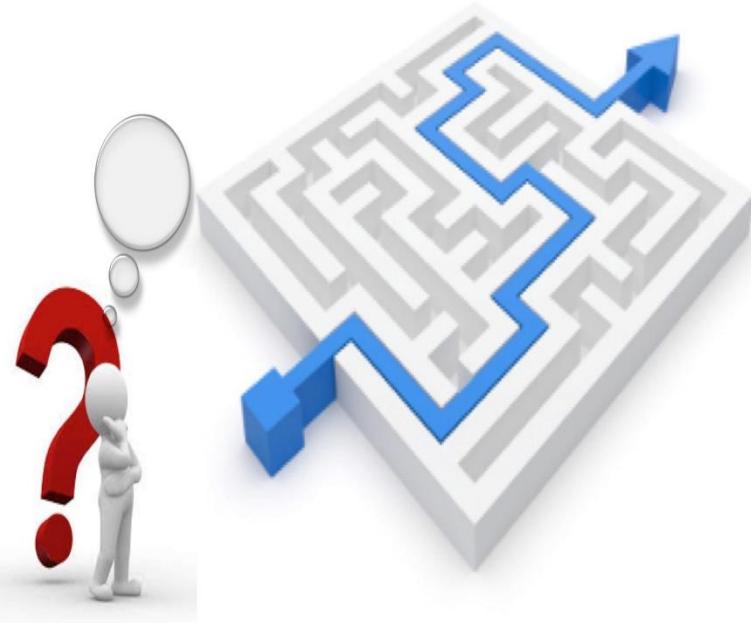
Requester Pays Buckets

- Assume that a bucket owner would like to share large data sets (zip code directories, reference data, geographical information, or web crawling data) from an S3 bucket, but does not want to incur charges for Requesting or Downloading the data
 - Requester pays buckets is the solution in this case.
 - The requester pays for data download charges and the request fees
 - The owner of the bucket pays for the data storage fees
- With requester pays buckets
 - Anonymous access to the bucket is not allowed, since the user must be authenticated (known) to pay for their usage
 - The requests made to the requester pay buckets must include the x-amz-request-payer header in their requests

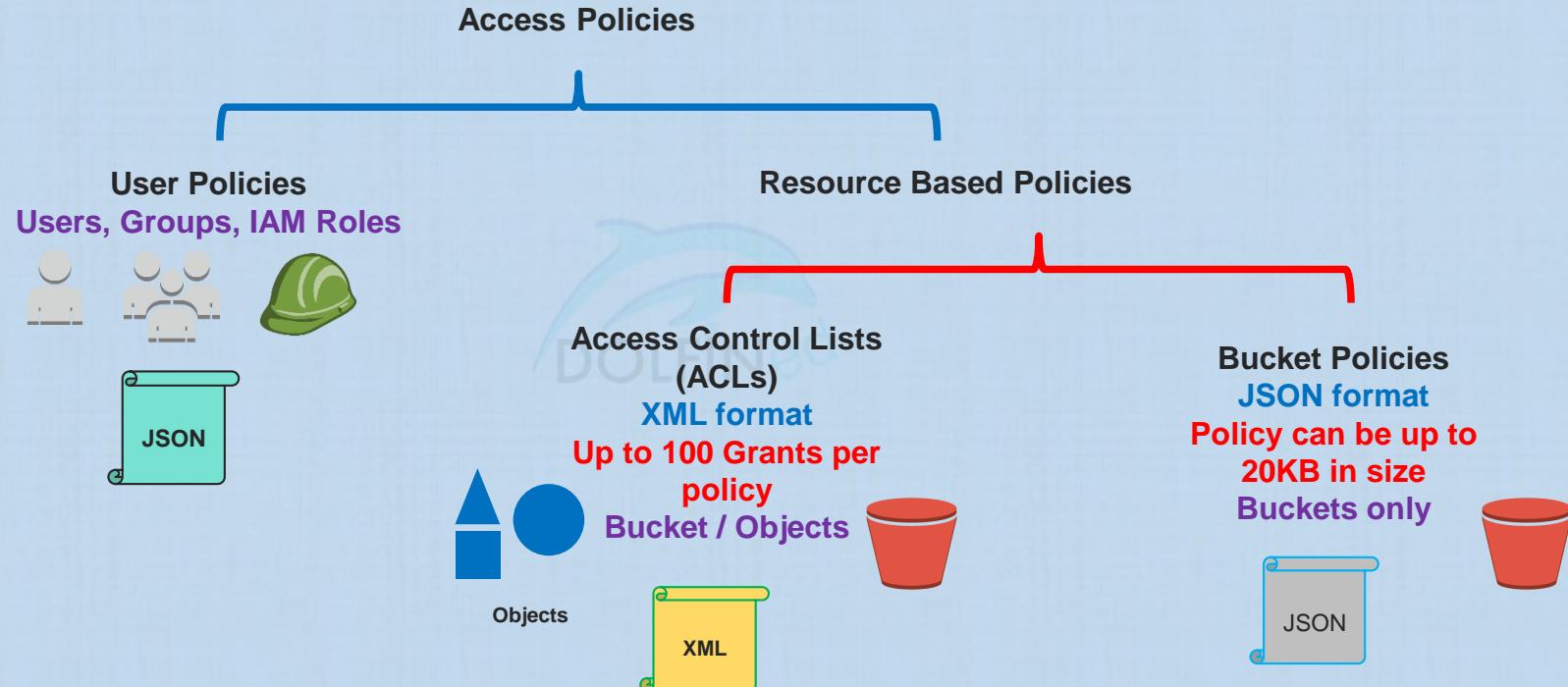


Amazon S3

Identity and Access Management in S3



Access Policy Categories – User policies or Resource-based policies



Depending on the required, you can use user policies only, Resource-based policies only or both together



Resource Ownership

- Objects and Buckets are private by default.
- Only the respective resource owner has FULL CONTROL access to the resource by default
- The resource owner is the account the created the resource (Bucket or object)
 - If an IAM user in an account creates a bucket or uploads an object
 - Still the account that user belongs to, is the resource owner



Access Control Lists (ACLs)

- ACLs are a list of grants
 - Each grant defines a grantee (to whom permission is allowed) and the allowed permissions
 - ACLs provide basic Read/Write operations on the resources
- ACLs can not do conditional permissions, Bucket policies can
- ACLs can define permissions for:
 - Accounts (Cross Account permissions) – They can not provide user level permissions directly
 - Pre-defined S3 groups:
 - Authenticated Users Groups (Users with AWS Credentials)
 - All users group (includes everyone – even those with AWS credentials)
 - Log Delivery Groups – Used for S3 Access Management to write Logs to an S3 bucket
- **Only use object ACLs to manage permissions for specific scenarios and only if ACLs meet your needs better than IAM and S3 bucket policies.**

Object Access Control Lists (ACLs)

- Are only used when it is required to provide permissions on Objects in a bucket, that are not owned by the bucket owner
 - This is the case of Cross Account permissions to upload objects in a bucket owned by other account
 - The ACLs need to be defined by the Object Owner (not the bucket owner)
- Limited to 100 grants per ACL
- The bucket owner can (although they do not own these objects)
 - Deny access to any object in their bucket – including these they do not own
 - Delete any object in their bucket – including these they do not own
 - Archive/Restore any objects in their bucket – including these they do not own



Bucket Access Control Lists (ACLs)

- Can provide permission to Accounts or pre-defined groups
- Can provide cross account permissions to other accounts (not to specific users within other accounts)
- Provide the WRITE permission over the bucket for Log Delivery Groups to write Access Logs into the bucket



Access Control Lists (ACLs) – Possible permissions

Permission	When granted on a bucket	When granted on an object
READ	Allows grantee to list the objects in the bucket	Allows grantee to read the object data and its metadata
WRITE	Allows grantee to create, overwrite, and delete any object in the bucket	Not applicable
READ_ACP	Allows grantee to read the bucket ACL	Allows grantee to read the object ACL
WRITE_ACP	Allows grantee to write the ACL for the applicable bucket	Allows grantee to write the ACL for the applicable object
FULL_CONTROL	Allows grantee the READ, WRITE, READ_ACP, and WRITE_ACP permissions on the bucket	Allows grantee the READ, READ_ACP, and WRITE_ACP permissions on the object



ACL limitations

- Through S3 ACLs, only the following permissions sets: READ, WRITE, READ_ACP, WRITE_ACP, and FULL_CONTROL can be defined.
- Only an AWS account or one of the predefined Amazon S3 groups can be defined as a grantee for the Amazon S3 ACL.
- When specifying email address or the canonical user ID for an AWS account, the ACL applies to all entities in the grantee AWS account (not specific users, Roles or Groups).
- You can't use an ACL to restrict access to individual IAM users, IAM Roles, or apply ACLs to different objects that share the same prefixes.
- An ACL doesn't support having conditions for the S3 operation that the ACL authorizes. Therefore, the bucket owner might not have full control over the objects uploaded by the ACL grantee.



Bucket Policies

- Can grant access to users or accounts (Same account or Cross Account) to bucket or objects owned by the account owner
- They apply only to resources (bucket or objects) owned by the bucket owner
- If the bucket owner allowed other accounts to upload objects in its bucket that it does not own, can the bucket owner provide permissions on these objects to its own account users or cross account?
 - Yes, They can provide in-account or cross-account access to these objects, Provided that the object owner account(s) grant Full-Control permission to the bucket owner, through Object ACLs.
- Are limited to 20KB in size, hence, they can't scale to be granular at the object level for every object in the bucket
- Can do conditional permissions
- Can do the full set of S3 operations



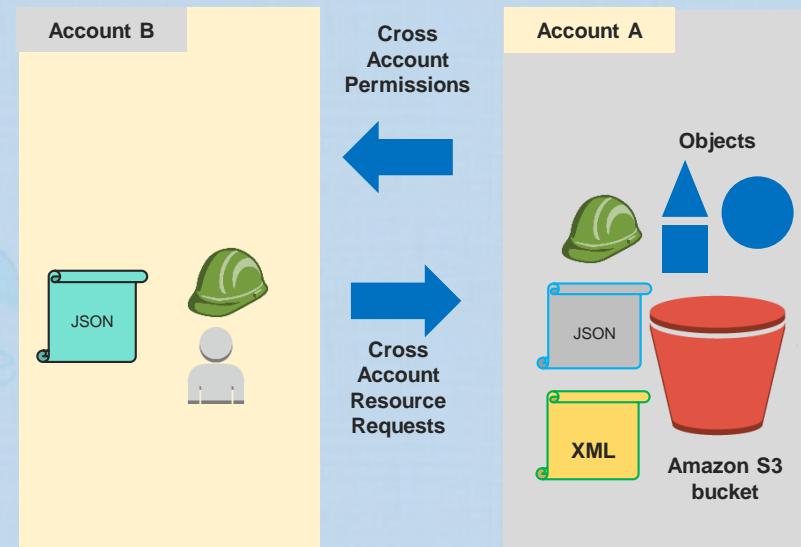
User policies

- Attaches to a user, group, or IAM Role (basically who uses them is authenticated)
- You can NOT grant anonymous access through a user access policy
- IAM User or IAM Role policies are inline-policies
- IAM Group policies are standalone policies



Cross-Account permissions

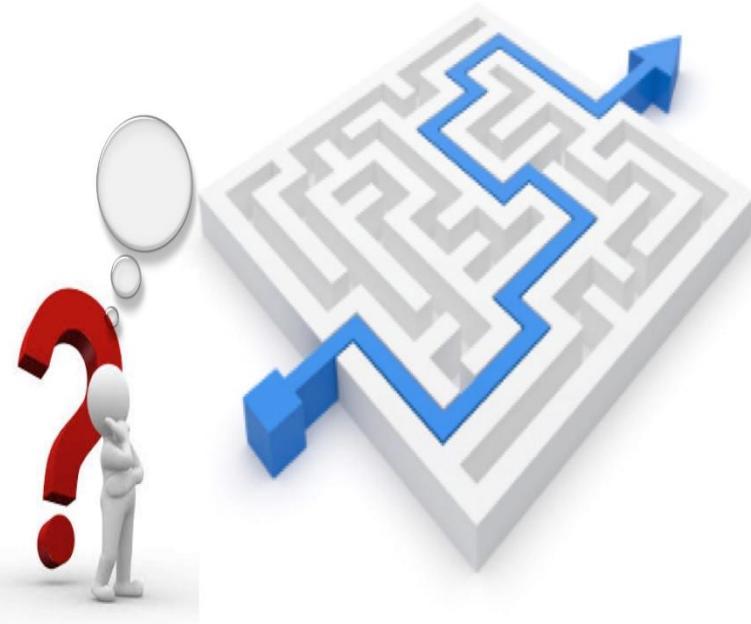
- A bucket owner can grant cross-account permissions to other accounts or to users in other accounts to upload objects
- When the other accounts (or users of those accounts) upload objects into the bucket, they become the resource owner(s) of these objects



- Inline User Policy
- Inline IAM Role Policy

Amazon S3

Billing



Review Topic : Simple Storage Service (S3)

Billing Charges in S3

- No charge for data transferred from/to EC2 to/from S3 in the same region
 - This includes data transferred via Copy command
- Data transfer into S3 is free of charge
- Data transfer out of S3 to EC2 instance in the same region is free
 - Data transferred out of S3 bucket region is charged
- Data transfer via Copy to other regions is charged at the Internet data transfer rates
- Data transferred from S3 to Cloudfront is free
- Upload requests (\$0.05/1000 requests)
 - Both PUT and GET requests are charged
- PUT/GET/DELETE request fee
- Retrieval fee for S3 IA and OneZone IA

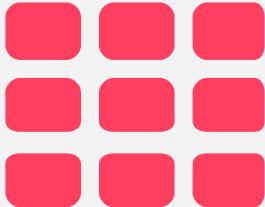


Review Topic : Simple Storage Service (S3)

Requester Pays

- If Requester pays is enabled on a bucket:
 - The bucket owner will only pay for Object Storage fees
 - The requester will pay for:
 - Requests to S3 to upload/download objects
 - Data transfer charges
- Buckets with Requester Pays enabled do not allow anonymous access & Do not support BitTorrent
- Requester pays buckets can not be the target for user logging
- You can NOT specify requester pays at the Object level
- Can be enabled from the bucket properties under the AWS Console





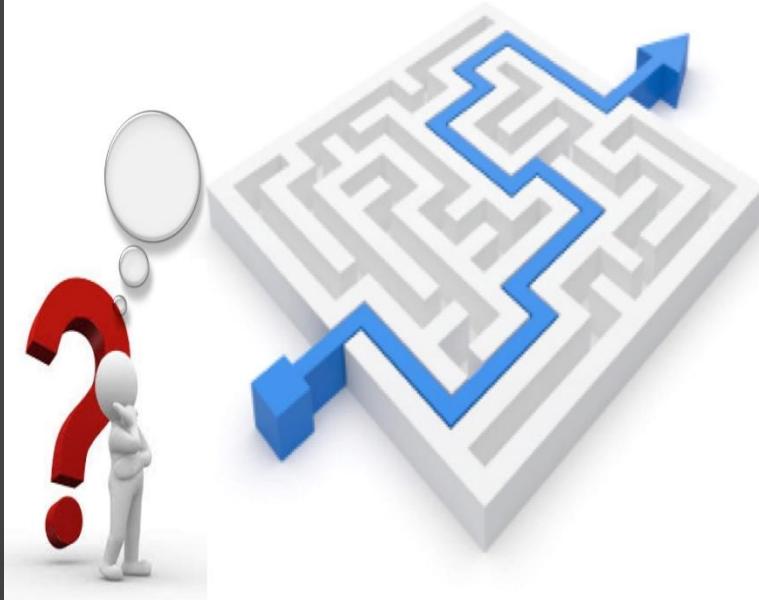
FILE SYSTEM OPTIONS AMAZON ELASTIC FILE SYSTEM (EFS)

You Can Do It Too!



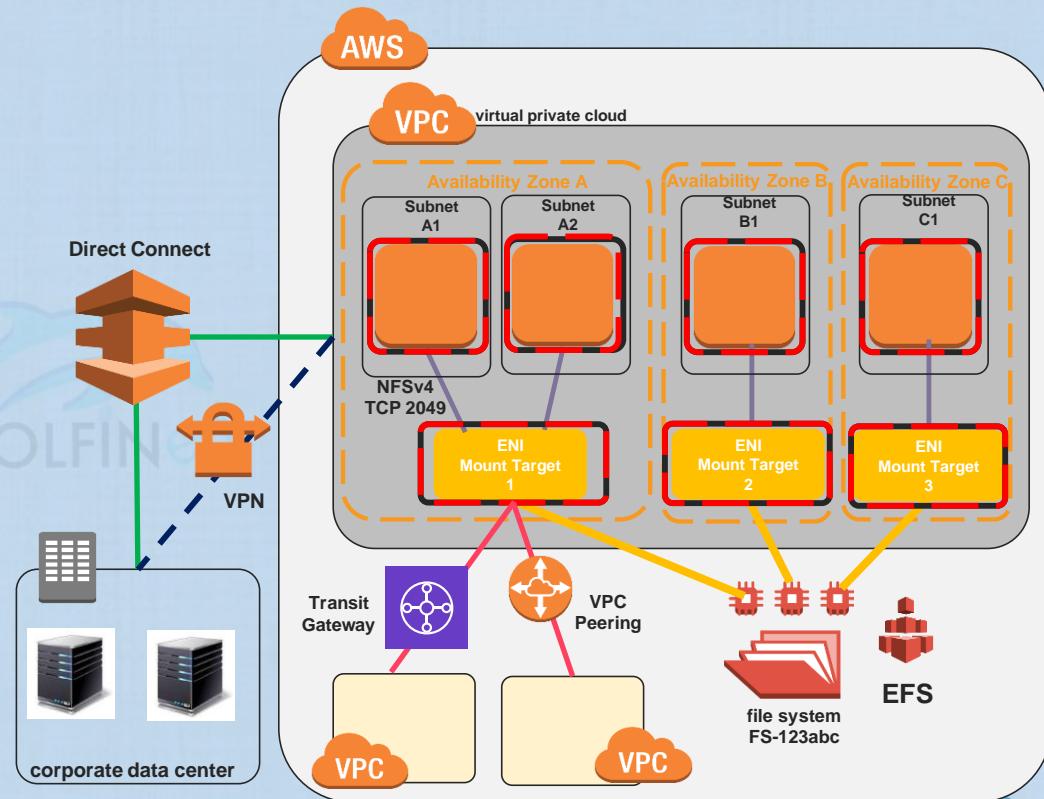
Amazon Elastic File System (EFS)

Introduction to EFS



Amazon Elastic File System (EFS) – What is it?

- Amazon Elastic File System (Amazon EFS) provides simple, highly available, highly durable, and scalable file storage in the cloud for use with Amazon EC2 or on-premise servers.
- Amazon EFS file systems store **data and metadata** across multiple Availability Zones in an AWS Region.
- Amazon EFS file systems can be mounted on Amazon EC2 instances, on-premises servers through an AWS Direct Connect or VPN connection, through a VPC Peering Connection or a transit gateway (same or different Account VPCs).



Amazon Elastic File System (EFS) – What is it?

- EFS service manages all the file storage infrastructure
- EFS file systems (EFS storage capacity) is elastic, growing and shrinking automatically as you files get added and removed.
 - EFS file systems can grow to petabyte scale, drive high levels of throughput, and allow access to the stored data from a large number of EC2 instances in parallel
- Amazon EFS provides file system access semantics, such as strong data consistency and file locking.
- Portable Operating System Interface (POSIX) Compliant. **It is limited to Linux Instances**
- For some AMIs, you'll need to install an NFS client to mount your file system on your Amazon EC2 instance.
- You can control access to your file system and data through IAM and POSIX file permissions



Amazon Elastic File System (EFS) – How it works

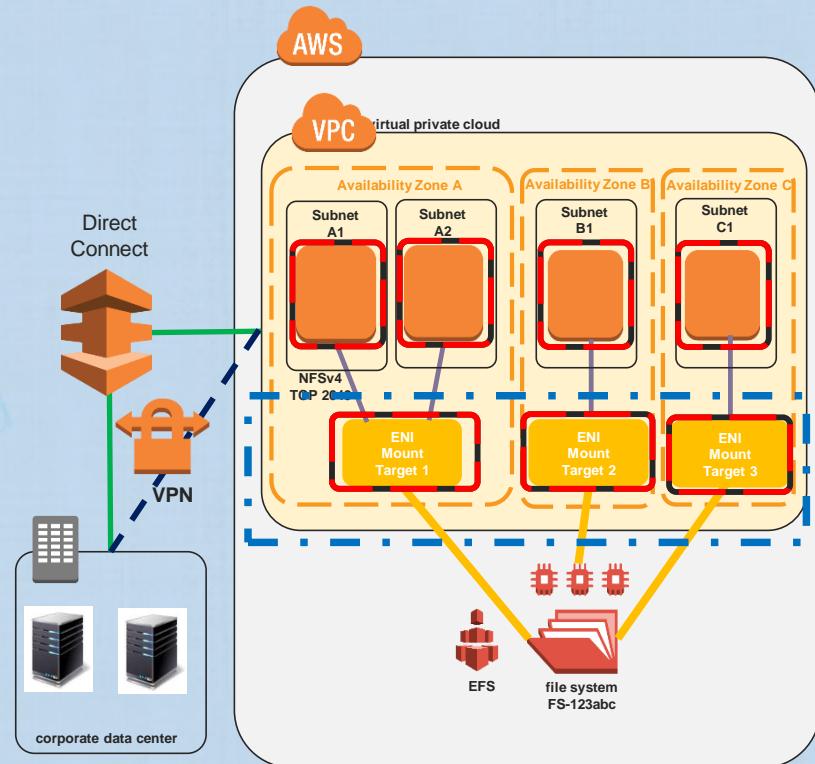
- Amazon EFS supports the Network File System version 4 (NFSv4.1 and NFSv4.0) protocol.
 - The applications and tools that you use today work seamlessly with Amazon EFS.
- EFS file system has mount targets, through which EC2 instances can mount the file system
- Multiple Amazon EC2 instances in the same region, same VPC and in different availability zones, can access an Amazon EFS file system at the same time
 - This provides a common data source for workloads and applications running on more than one instance or server.



Amazon Elastic File System (EFS) – Mount Targets

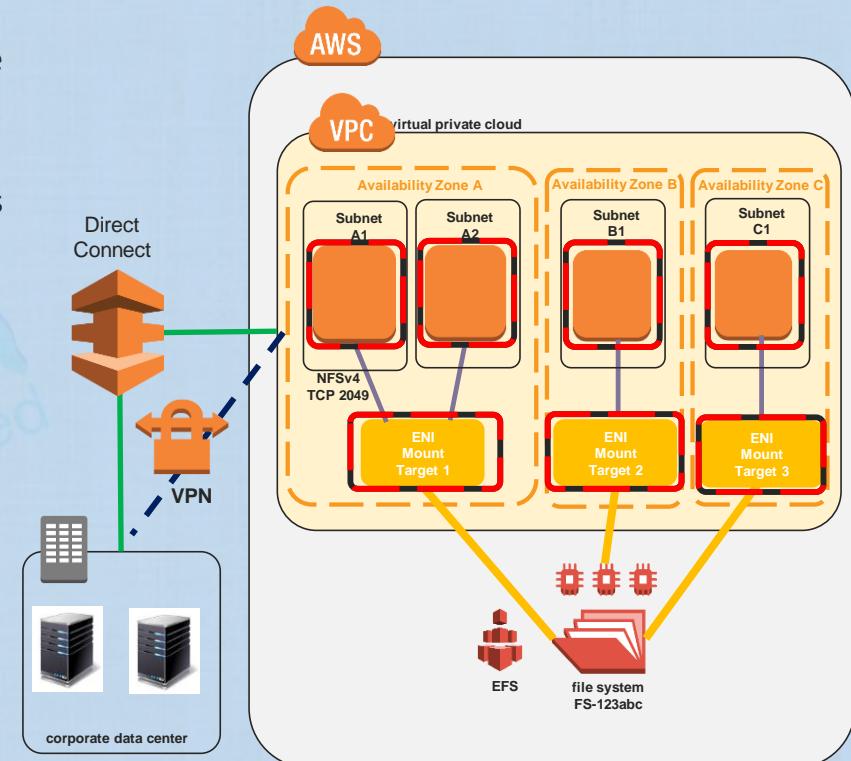
To access your Amazon EFS file system in a VPC, you create one or more mount targets in the VPC.

- You can create only one mount target in each Availability Zone in an AWS Region.
- The IP addresses and DNS for your mount targets in each AZ are static.
- If there are multiple subnets in an Availability Zone in your VPC, you create a mount target in one of the subnets which will be used by all EC2 instances in the different subnets in that AZ.
- A mount target provides an IP address for an NFSv4 endpoint at which you can mount an Amazon EFS file system.
- AWS recommends creating mount targets in all AZ's in the region



Amazon Elastic File System (EFS) – Mount Targets

- Mounts targets themselves are highly available.
- A File System can be used through mount Targets in a single VPC at a time.
- Mount targets need to have associated security groups.
- These security groups act as a virtual firewall that controls the traffic between them.



Amazon Elastic File System (EFS) – Use Cases

- Amazon EFS is designed to meet the performance needs of the following use cases.
 - **Big Data and Analytics**
 - **Media Processing Workflows**
 - **Content Management and Web Serving**
 - **Home Directories**



Amazon Elastic File System (EFS) – Use Cases

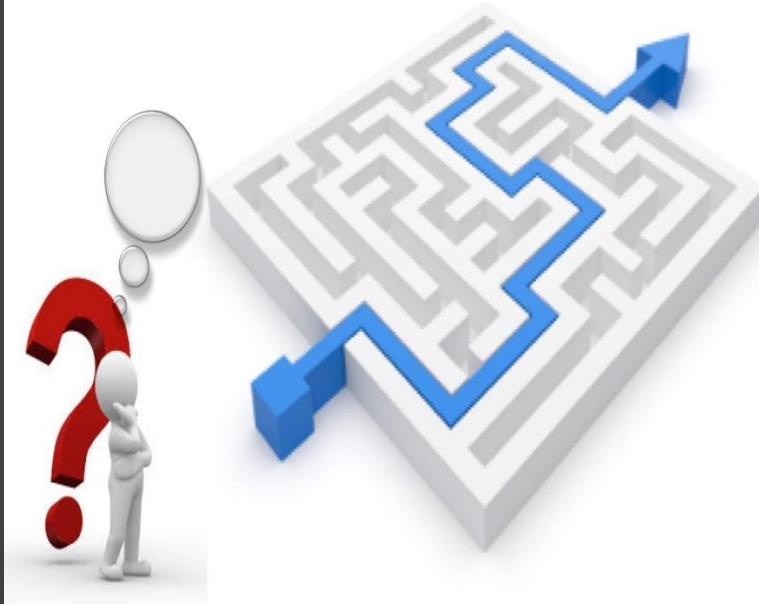
Amazon EFS is designed to meet the performance needs of the following use cases.

- **Big Data and Analytics**
 - Amazon EFS provides the **scale and performance** required for big data applications that require **high throughput** to compute nodes coupled with **read-after-write consistency and low-latency file operations**.
- **Media Processing Workflows**
 - Media workflows like video editing, studio production, broadcast processing, sound design, and rendering often depend on **shared storage to manipulate large files**.
 - A **strong data consistency model with high throughput and shared file access can** cut the time it takes to perform these jobs and **consolidate multiple local file repositories into a single location** for all users.
- **Content Management and Web Serving**
 - Amazon EFS provides a **durable, high throughput file system for content management systems** that store and serve information for a range of applications like websites, online publications, and archives.
- **Home Directories**
 - Amazon EFS can provide **storage for organizations that have many users that need to access and share common data sets**.
 - An administrator can use Amazon EFS to create a file system accessible to people across an organization and establish permissions for users and groups at the file or directory level.

Amazon Elastic File System (EFS)

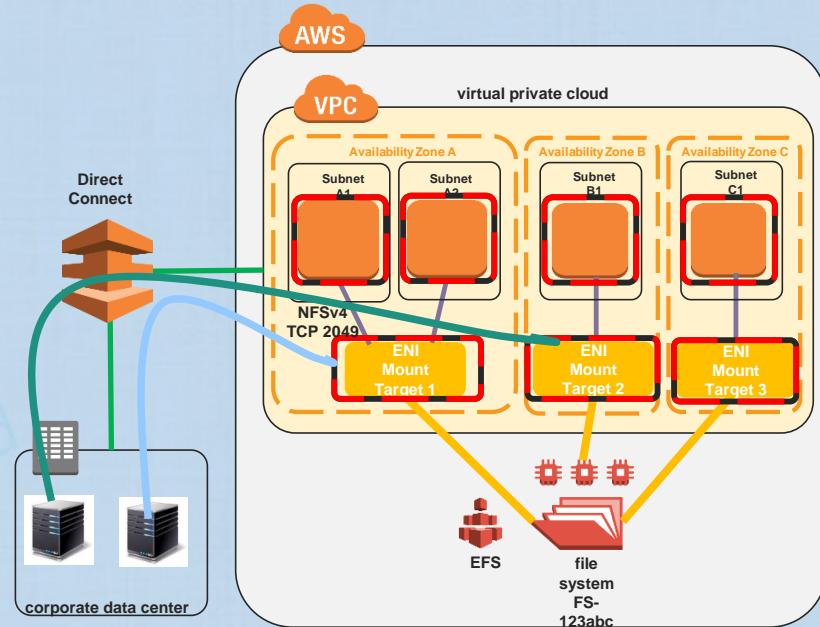
EFS:

- EFS and On-Premise Compute
- EFS Storage Classes



Amazon Elastic File System (EFS) – Mounting to On premise Servers

- You can mount your Amazon EFS file systems on your on-premises data center servers when connected to your Amazon VPC with AWS Direct Connect or VPN.
 - AWS recommends using the IP address of the mount targets not the DNS name
 - You can mount your EFS file systems on on-premises servers to migrate data sets to EFS, which will:
 - Enable cloud bursting scenarios, or
 - Backup on-premises data to EFS.
- AWS recommends configuring two AWS DX connections for high availability.
- AWS recommends that your application be designed to recover from potential connection interruptions.



The on-premises server must have a Linux-based operating system.



EFS and On-Premise servers – Cloud Bursting use case

- Using an Amazon EFS file system mounted on an on-premises server, you can migrate on-premises data into the AWS Cloud hosted in an Amazon EFS file system. You can also take advantage of bursting.
- A Customer can move data from their on-premises servers into Amazon EFS and analyze it on a fleet of Amazon EC2 instances in their Amazon VPC.
 - Then, results can be stored in EFS permanently or be moved back to the on-premises server(s).



Amazon Elastic File System (EFS) Storage Classes

- Amazon EFS file systems have two storage classes available:
 - **Standard**
 - **Infrequent Access**
- EFS IA storage is compatible with all EFS features, and is available in all AWS Regions where Amazon EFS is available.



EFS Lifecycle Management

- EFS Lifecycle Management
 - EFS Lifecycle Management automatically manages cost-effective file storage.
 - All writes to files in IA storage are first written to Standard storage, then transitioned to IA storage.
 - When enabled, lifecycle management automatically migrates files that have not been accessed for 30 days to the EFS IA storage class.
 - When enabled, lifecycle management automates moving files from Standard storage to IA storage.
- Notes:
 - File metadata, such as file names, ownership information, and file system directory structure, is always stored in Standard storage to ensure consistent metadata performance.
 - Files smaller than 128 KB are not eligible for lifecycle management and are always stored in the standard class.

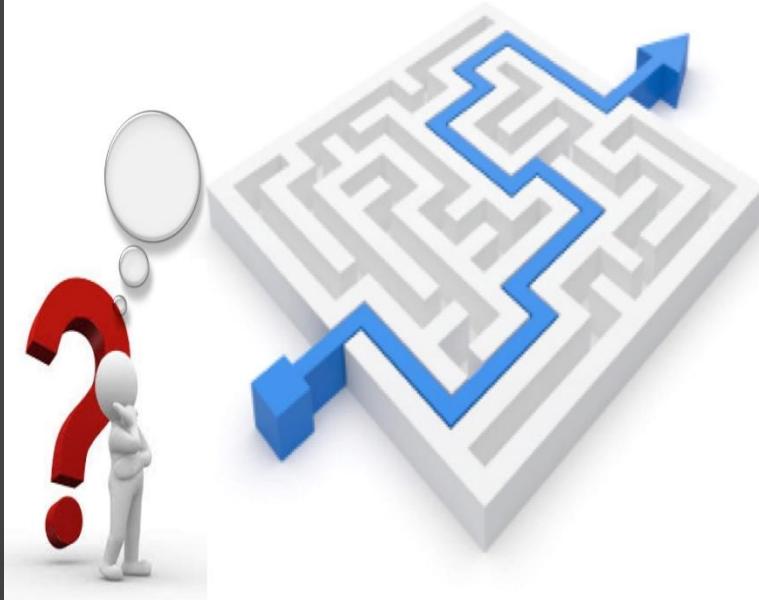
Amazon Elastic File System (EFS) – Pricing

- Using EFS, you only pay for the storage used by the file system(s) you create/use
 - No minimum fee or setup cost.
- EFS IA billing is based on the amount of data stored in IA and the data I/O used to access the data stored in IA.
 - Access charges are incurred when files in IA storage are read, and when files are transitioned to IA storage from Standard storage.
- For On Premises servers:
 - Customers will be charged for the AWS Direct Connect connection to their Amazon VPCs.

Amazon Elastic File System (EFS)

EFS:

- Data Encryption
- Backup



Amazon Elastic File System (EFS) – Data Encryption

- Using Amazon EFS, you can create encrypted file systems.
- Amazon EFS supports two forms of encryption for file systems, encryption in transit and encryption at rest.
 - Amazon EFS automatically manages the keys for encryption in transit, client to EFS encryption.
 - Any key management customer can do is only related to encryption at rest.
- Amazon EFS uses AWS Key Management Service (AWS KMS) for key management.
- If you create a file system that uses encryption at rest, data and metadata will be encrypted at rest.
 - When you create a file system using encryption at rest, you specify a Customer Master Key (CMK).
 - The CMK can be:
 - The AWS-managed CMK for Amazon EFS (aws/ elasticfilesystem), or
 - It can be a CMK that you manage.
- **File Data :**
 - Will be encrypted at rest using the CMK that you specified when you created your file system.
- **Meta Data :**
 - The AWS-managed CMK for your file system is used as the master key for the metadata in your file system,
 - This includes file names, directory names, and directory contents.

Amazon Elastic File System (EFS) – Backing up EFS

- There are two options available for backing up your EFS file systems.
 - The EFS-to-EFS backup solution
 - AWS Backup service
- **The EFS-to-EFS backup** solution is suitable for all Amazon EFS file systems in all AWS Regions.
 - This solution follows AWS best practices for security and availability.
 - The solution includes a Cloud Watch Event that invokes an Orchestrator AWS Lambda
 - More details in the link below for further reading

➤ <https://aws.amazon.com/solutions/efs-to-efs-backup-solution/>

Amazon Elastic File System (EFS) – Backing up EFS Using AWS Backup

- AWS Backup is a backup service designed to simplify the creation, migration, restoration, and deletion of backups, while providing improved reporting and auditing.
- Amazon EFS is integrated with AWS Backup.
 - You can use AWS Backup to set backup plans where you specify your backup frequency, when to back up, how long to retain backups, and a lifecycle policy for backups.
 - Then assign the EFS file system to this backup plan
- AWS backup does incremental backups of EFS file system
- You can restore from a backup to a new EFS file system or to the same EFS File system



Amazon EFS & AWS DataSync – Data Migration to EFS

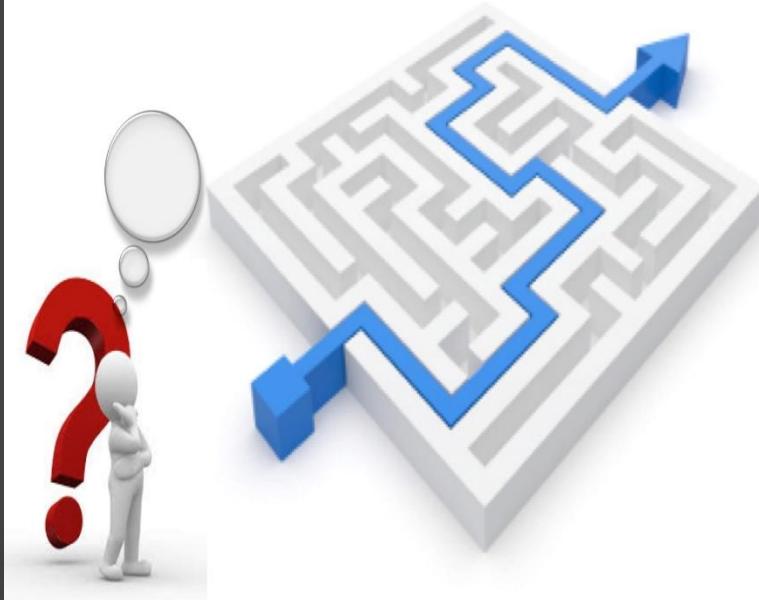
- AWS DataSync is a data transfer service that simplifies, automates, and accelerates moving and replicating data between on-premises storage systems and AWS storage services **over the internet or AWS Direct Connect.**
- AWS DataSync can transfer your file data and file system metadata such as ownership, time stamps, and access permissions.
- AWS recommends using AWS DataSync to transfer data into Amazon EFS.
- You can use AWS DataSync to:
 - Transfer files from an existing file system to Amazon EFS.
 - Transfer files between two EFS file systems, this includes file system in different AWS Regions and file systems owned by different AWS accounts.
- Using DataSync to copy data between EFS file systems, you can perform:
 - One-time data migrations,
 - Periodic data ingestion for distributed workloads, and
 - Automate replication for data protection and recovery.

Amazon Elastic File System (EFS) - Monitoring

- You can use the following automated monitoring tools to watch Amazon EFS and report when something is wrong:
 - **Amazon CloudWatch Alarms**
 - Watch a single metric over a time period that you specify and perform one or more actions based on the value of the metric relative to a given threshold over several time periods.
 - The action is a notification sent to an Amazon SNS topic or Amazon EC2 Auto Scaling policy.
 - CloudWatch alarms do not invoke actions simply because they are in a particular state; the state must have changed and been maintained for a specified number of periods.
 - **Amazon CloudWatch Logs**
 - Monitor, store, and access your log files from AWS CloudTrail or other sources.
 - **Amazon CloudWatch Events**
 - Match events and route them to one or more target functions or streams to make changes, capture state information, and take corrective action.
 - **AWS CloudTrail Log Monitoring**
 - Share log files between accounts, monitor CloudTrail log files in real time by sending them to CloudWatch Logs, write log processing applications in Java, and validate that your log files have not changed after delivery by CloudTrail.

Amazon Elastic File System (EFS)

**EFS Data Consistency
, Monitoring &
Mounting File Systems from other
accounts**

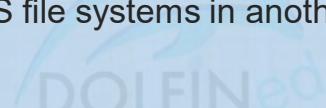


Amazon EFS - Data Consistency in Amazon EFS

- Amazon EFS provides the **Close-to-open (Open-after-Close) Consistency** semantics that applications expect from NFS.
 - NFS provides no guarantees that all clients see exactly the same data at all times. (Remember eventual consistency)
- In Amazon EFS, write operations are durably stored across Availability Zones in these situations:
 - An application performs a synchronous write operation, or
 - An application closes a file.
- Depending on the access pattern, Amazon EFS can provide **Stronger Data Consistency** guarantees than open-after-close semantics.
 - Applications that perform synchronous data access and perform non-appending writes have **read-after-write (Strong) consistency** for data access.

Amazon EFS - Mounting EFS File Systems from Another Account or VPC

- You can mount your Amazon EFS file system from Amazon EC2 instances that are in a different account or virtual private cloud (VPC).
- **Mounting from Another Account in the same (shared) VPC**
- **Mounting from another VPC in the same or different account**
 - When you use a VPC peering connection or transit gateway to connect VPCs, Amazon EC2 instances that are in one VPC can access EFS file systems in another VPC, even if the VPCs belong to different accounts.

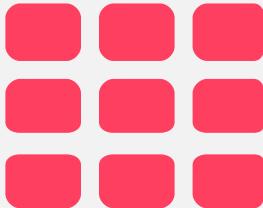


Amazon Elastic File System (EFS) – Performance Modes

Amazon EFS offers **two performance modes**, that allow EFS to support a wide variety of cloud storage use cases, these are:

- General Purpose is:
 - Ideal for latency-sensitive use cases, like web serving environments, content management systems, home directories, and general file serving.
 - Amazon EFS selects the General-Purpose mode for you by default, but you have the option to change it during creation.
 - AWS recommends the General-Purpose performance mode for majority of your Amazon EFS file systems.
- Max I/O Performance Mode
 - File systems in the Max I/O mode can scale to higher levels of aggregate throughput and operations per second with a tradeoff of slightly higher latencies for file operations.
 - Use it for highly parallelized applications and workloads, such as big data analysis, media processing, and genomics analysis.
 - Your EFS file system bills are not impacted by which performance mode you choose.
 - You can not change the file system's performance mode after it gets created.





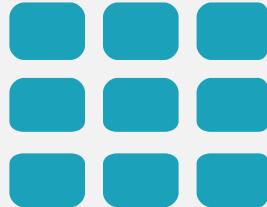
FILE SYSTEM OPTIONS

AMAZON FSX

FSx

You Can Do It Too!





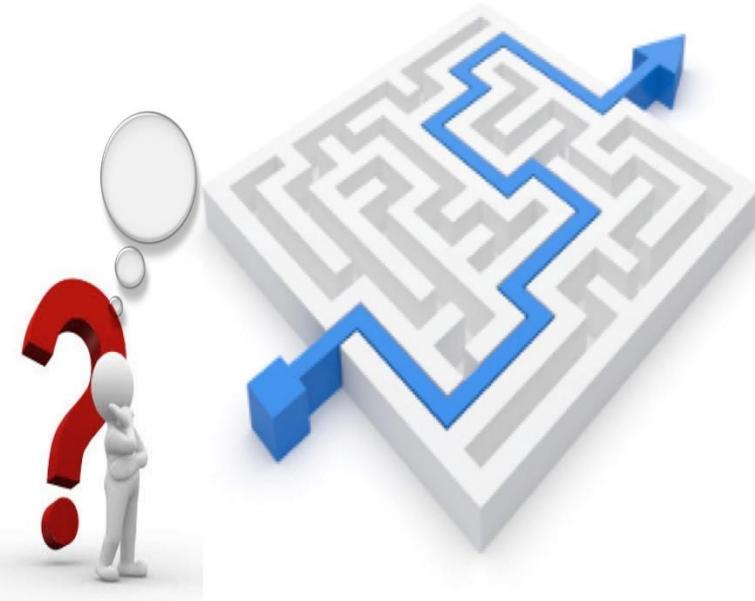
AMAZON FSX FOR WINDOWS FILE SERVER

You Can Do It Too!



Amazon FSx

- Introduction to Amazon FSx
- Single vs Multi-AZ Deployments
- Data Encryption



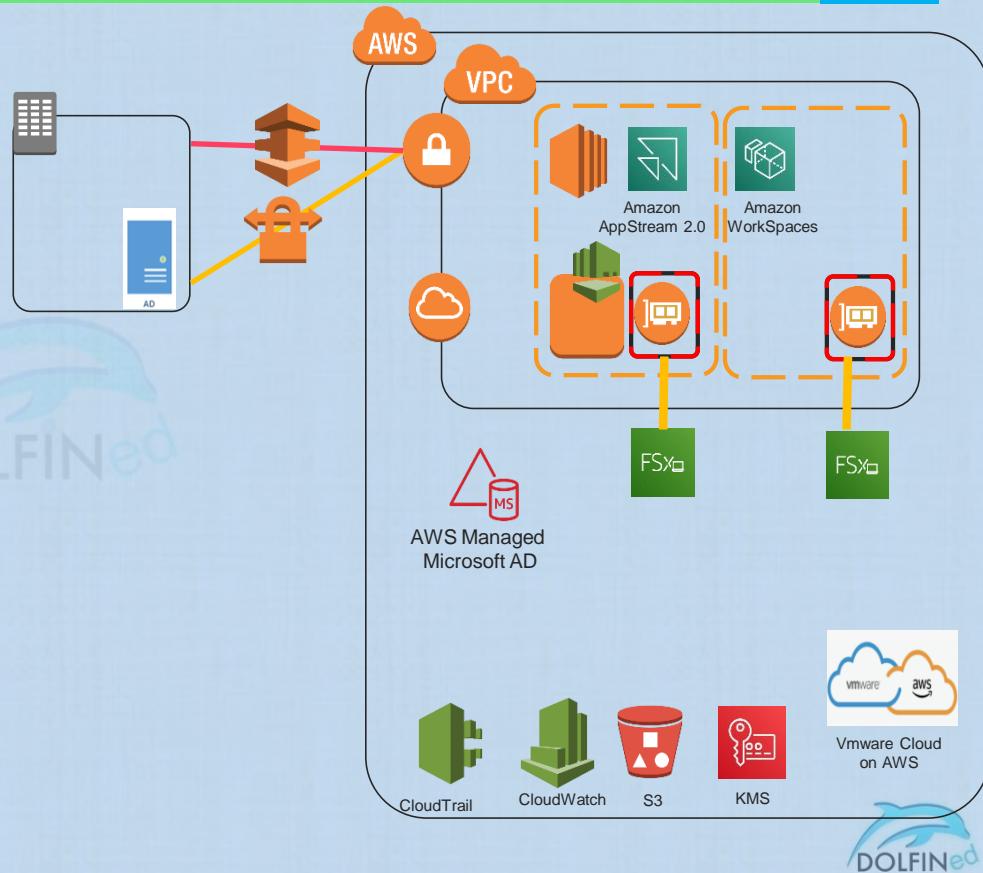
Amazon FSx for Windows File Server – What is it?

- Amazon FSx for Windows File Server provides a fully managed native Microsoft Windows file system.
 - This is useful for Windows-based apps that need file storage in AWS.
 - Amazon FSx provides shared file storage with full support for the SMB protocol, Windows NTFS, and Active Directory (AD) integration.
 - Amazon FSx uses SSD storage to provide the fast performance with high levels of throughput, IOPS, and consistent sub-millisecond latencies.
- With Amazon FSx, highly durable and available Windows file systems that can be created and accessed from up to thousands of clients using the SMB protocol.
- Amazon FSx needs to work with an AWS Directory Service for Microsoft Active Directory (AWS Managed MS AD in the same VPC, different VPC in the same account using VPC peering, or a completely different account using directory sharing), or an AD that is self-managed by the client in AWS or On-premise.



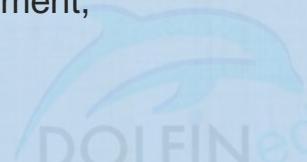
Amazon FSx for Windows File Server – What is it

- Integrates with many other AWS services including IAM, CloudWatch, CloudTrail, KMS, S3, AD services
- The **file system** is the primary resource in Amazon FSx. It is the server hosting the files and folders.
 - The file system is configured with a storage amount and a throughput capacity.
 - It also has a DNS name for accessing it.
- A **file share** is a specific folder (and its subfolders) within your file system that is made accessible to the clients and compute instances



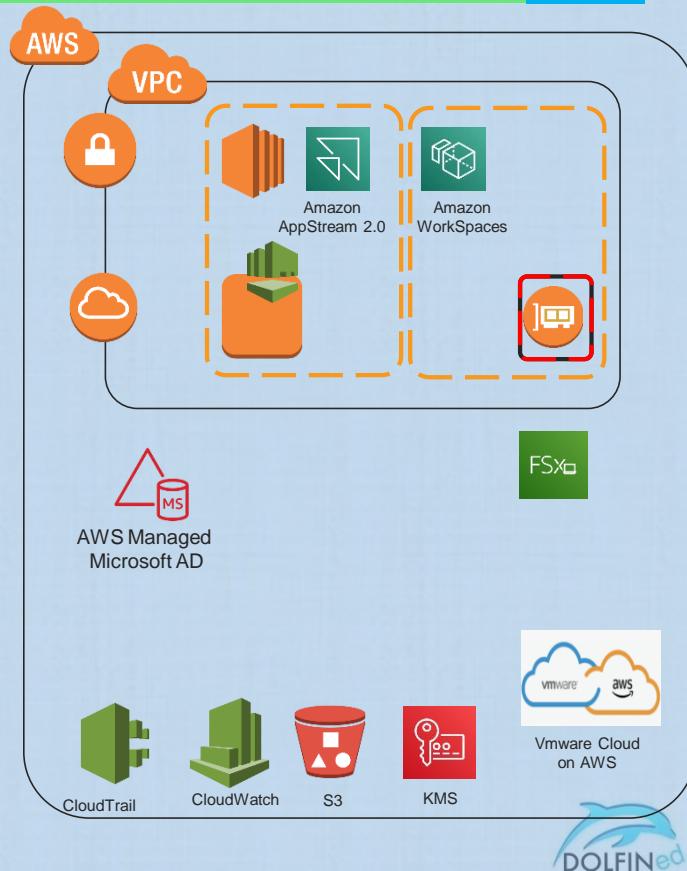
Amazon FSx for Windows File Server – Use cases

- Amazon FSx is suitable for the following use cases where a Windows shared file storage is required:
 - CRM, ERP, custom or .NET applications,
 - Home directories,
 - Data analytics,
 - Media and entertainment workflows,
 - Web serving and content management,
 - Software build environments, and
 - Microsoft SQL Server.



Amazon FSx for Windows File Server – Availability and Durability

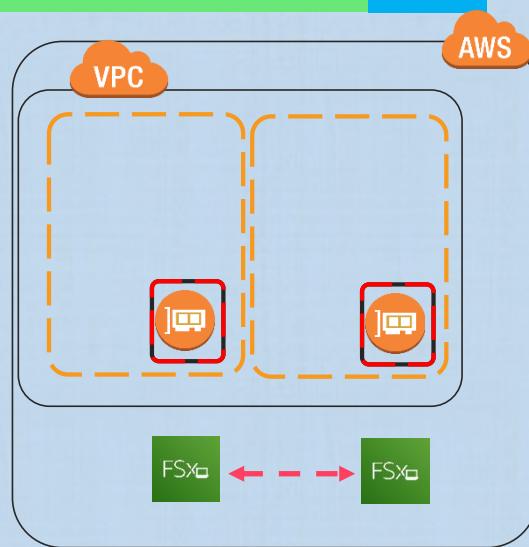
- Amazon FSx can be configured in Single-AZ or Multi AZ file systems
- For Single AZ file systems**
 - They are designed to be highly available and durable within an AZ
 - FSx replicates data within an AZ to protect it from component level failures,
 - FSx automatically replaces infrastructure components in the event of a failure.
 - Amazon FSx takes highly durable backups (stored in S3) of the file system daily using Windows's Volume Shadow Copy Service, and also allows for taking additional backups at any point.



Amazon FSx for Windows File Server – Availability and Durability

- **For Multi-AZ file systems**

- In addition to supporting all single AZ features, Multi AZ file system provides protection against instance failure and AZ disruption/failure
- Amazon FSx automatically provisions and maintains a **standby file server** in a different Availability Zone.
- Any changes written to disk in the file system **are synchronously replicated** across AZs to the standby.
- Amazon FSx automatically fails over to the secondary file server, to continue accessing data without manual intervention.
 - It fails back to the then active file server when it is brought up (takes 30 seconds to fail over or fail back)

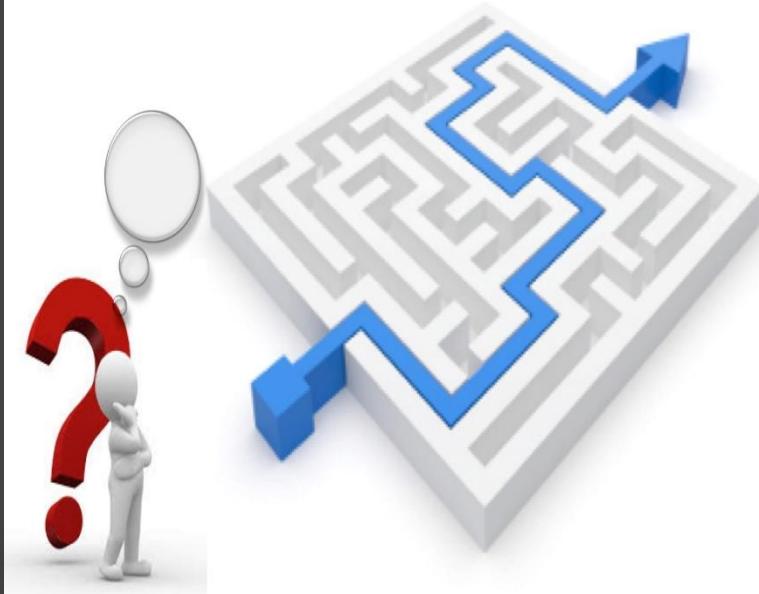


Amazon FSx for Windows File Server – Encryption

- Amazon FSx for Windows File Server always **encrypts file system data and backups at-rest** using keys you manage through AWS KMS.
 - Data is automatically encrypted before being written to the file system, and automatically decrypted as it is read.
 - These processes are handled transparently by Amazon FSx.
 - Amazon FSx uses an industry-standard AES-256 encryption algorithm to encrypt Amazon FSx data and metadata at rest.
 - The encryption key can be an AWS-managed CMK or Customer-managed CMK
- Amazon FSx **encrypts data-in-transit** using SMB Kerberos session keys, from clients that support SMB 3.0 (and higher).

Amazon FSx

- Data Protection, Backup and Restore
- Scaling Out Storage
- Migration from On Premise to FSx
- Access from On Premise and other VPCs/Accounts/Regions
- Securing the FSx file system using Security Groups



Amazon FSx for Windows File Server – Data Protection, Backup, Restore

- Amazon FSx automatically replicates the file system's data to ensure high availability.
- In addition to that, Amazon FSx provides two options to further protect the data stored on file systems:
 - Windows shadow copies (MS Windows Volume Shadow copy Service (VSS)) –
 - Needs to be enabled on the file system
 - MS shadow copies are point in time snapshots of the file system stored in S3
 - Shadow copies are included in the backups taken of your file systems
 - Shadow copies data is stored with the file system's data and therefore they incur storage costs
 - Shadow copies are incremental snapshots
 - Backup

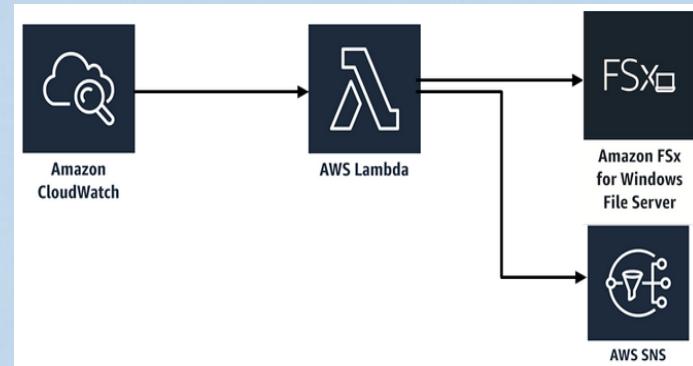
Amazon FSx for Windows File Server – Backup & Restore

Automatic backups:

- Amazon FSx automatically takes a daily, incremental, backups of the file systems during a daily, configurable, 30 min backup window.
- It supports backup retention and compliance needs.
- By default these automatic backups are retained for 7 days (configurable 0 – 35 days)
- Automatic backup will be deleted when the file system is deleted, the manual backups will remain
- Backups are stored in S3
- Available backup can be used to create a new file system by restoring a point-in-time snapshot of another file system.

User-initiated (manual) backups are also supported with FSx.

- Manual backups can also be automated using a custom schedule using CloudWatch events with Cron/Scheduled events to trigger a lambda function with the proper IAM roles to initiate an FSx for Windows File Server backup of the file system(s).

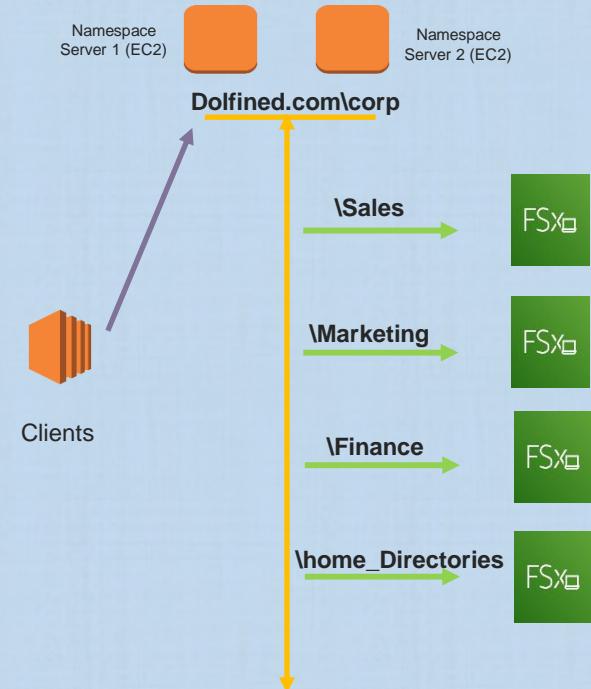


<https://docs.aws.amazon.com/fsx/latest/WindowsGuide/custom-backup-schedule.html>



Amazon FSx for Windows File Server – Scaling Out Storage and Throughput

- Amazon FSx file system can scale up to 64 TBs.
- If you need to scale beyond this to hundreds of Petabytes you can use Windows MS Distributed File System (DFS) namespaces which are supported by Amazon FSx for windows file server.

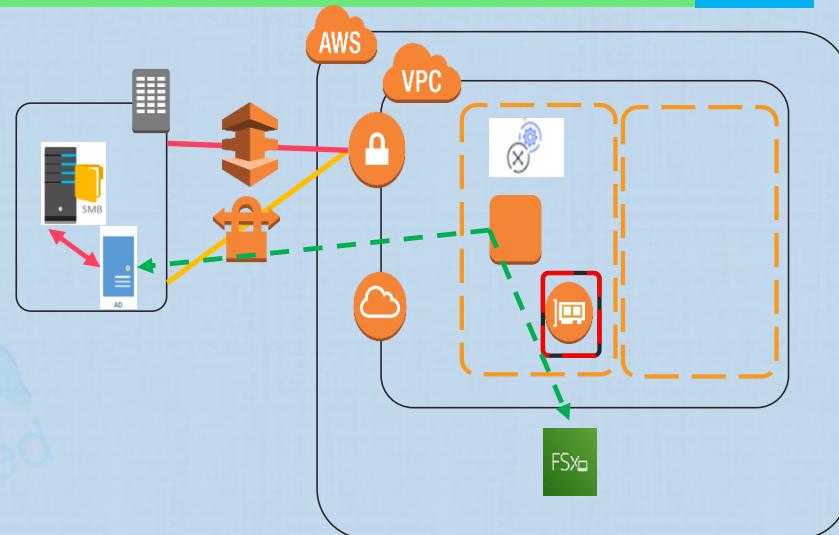


source: aws.amazon.com



Amazon FSx for Windows File Server – Migrating on Premise to AWS

- FSx for Windows file servers can be accessed from on-premise through a DX or VPN connections.
- Windows's Robust File Copy (RoboCopy) can be used to copy existing data files (both the data and the full set of metadata like ownership and Access Control Lists) directly to Amazon FSx.



MS DFS is also required to replicate between multiple Amazon FSx file systems across AWS Regions for data migration, cloud-bursting, data backup, and disaster recovery workflows.



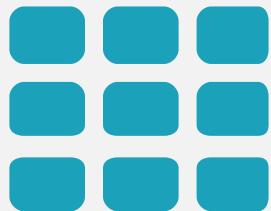
Access to Amazon FSx File Systems/Shares from other VPC/Account/Region

- The Amazon FSx file system can be accessed **both from On-Premise and from other VPCs/Accounts/Regions.**
- Using Amazon Direct connect or VPN to access your file system in a VPC.
 - With on-premises access, Amazon FSx can be used for:
 - Hosting user shares accessible by on-premises end-users,
 - Cloud bursting workloads to offload your application processing to the cloud,
 - Backup and disaster recovery solutions.
- Using VPC Peering or Transit Gateway, other VPCs and Accounts in the same or different Regions can access the Amazon FSx file systems.



Amazon FSx for Windows File Server – Security Group for the File System

- Clients access the FSx file system through an ENI in the VPC configured during the creation process.
 - Clients use the DNS to access the Private IPv4 address of the ENI
- Ensure that the following is allowed through the security group rules:
 - Inbound and outbound rules to allow the following ports:
 - TCP/UDP 445 (SMB)
 - TCP 135 (RPC)
 - TCP/UDP 1024–65535 (Ephemeral ports for RPC)
 - From and to IP addresses or security group IDs associated with:
 - Client compute instances from which you want to access the file system.
 - Other file servers that you expect this file system to participate with in DFS Replication groups.
 - Outbound rules to allow all traffic to the security group ID associated with the AWS Managed Microsoft AD directory to which the file system needs to join.



AMAZON FSX FOR LUSTRE

You Can Do It Too!

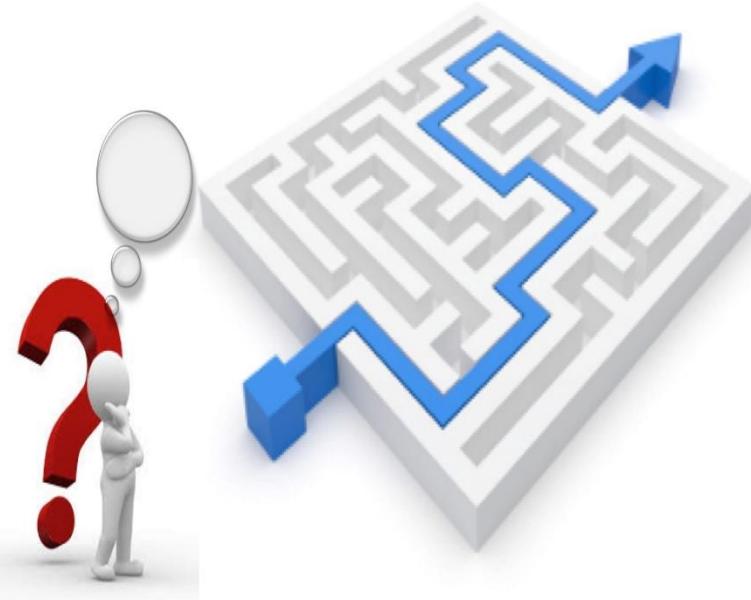


Amazon FSx for Lustre



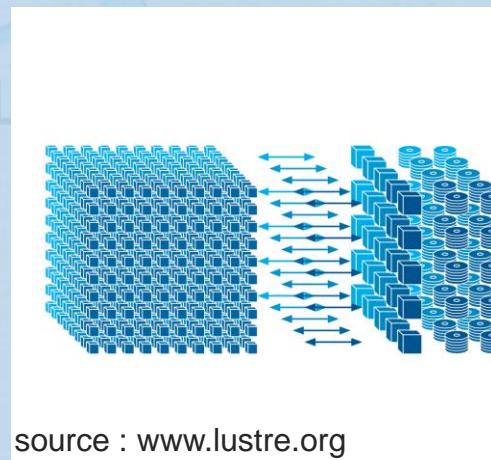
Introduction to Amazon FSx for Lustre

- What is it
- How it works
- Access to FSx for Lustre file systems
- Security and Encryption



What is Lustre

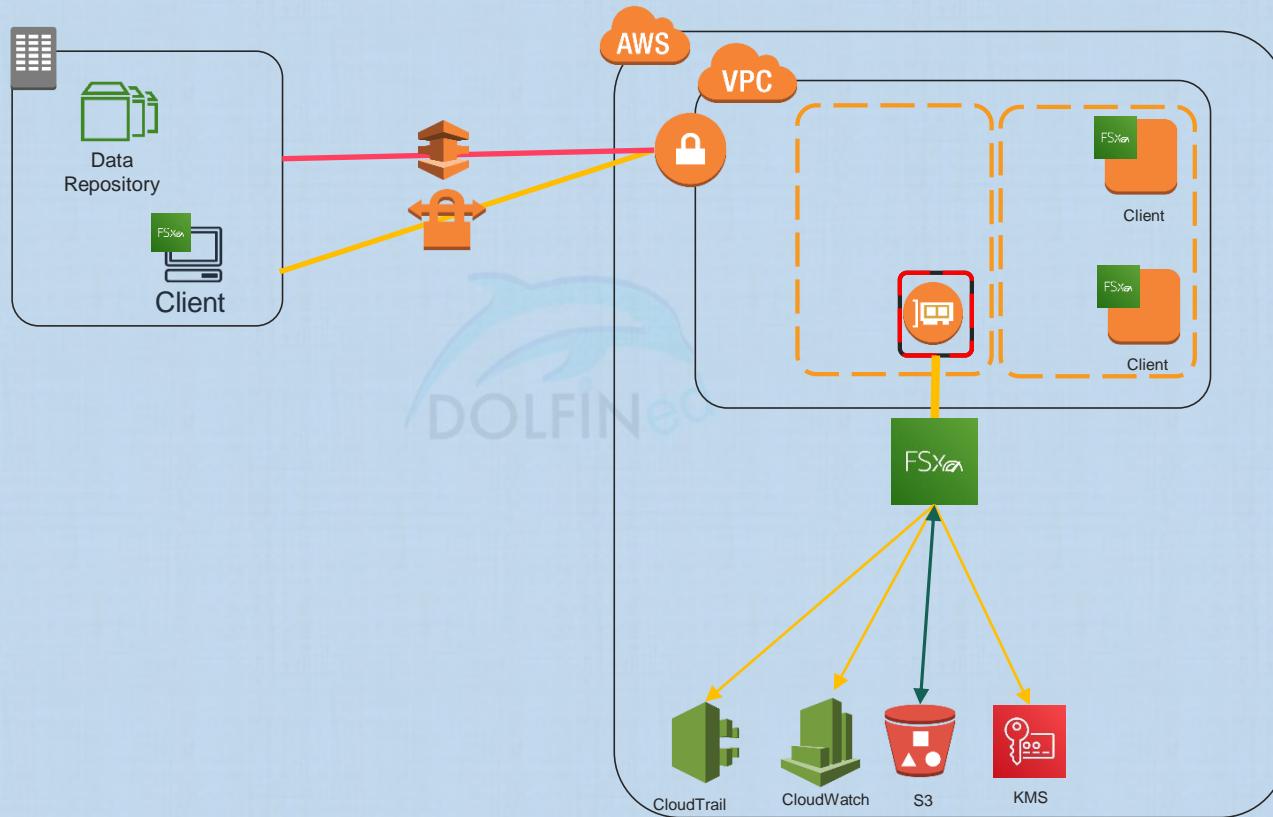
- Lustre is an open-source, **distributed, parallel data storage** platform designed for **massive scalability, high-performance, and high-availability**. It is very popular within the HPC community,
- Lustre file systems can scale from very small platforms of a few hundred terabytes up to large scale platforms with hundreds of petabytes, in a single, **POSIX-compliant, name space**.
- Lustre runs on **Linux-based operating systems** and employs a client-server network architecture.
- Lustre servers a single file system instance and can, in aggregate, present up to tens of petabytes of storage to thousands of compute clients simultaneously, and deliver more than a terabyte-per-second of combined throughput.



Lustre* is an open-source, object-based, distributed, parallel, clustered file system

- Designed for maximum performance at massive scale
- Capable of Exascale capacities
- Highest IO performance available for the world's largest supercomputers
- POSIX compliant
- Efficient and cost effective

Amazon FSx for Lustre – What is it?



Amazon FSx for Lustre – What is it?

- Amazon FSx for Lustre provides **fully managed Lustre file systems** that are optimized for compute-intensive workloads requiring high performance and low latencies of scale-out, parallel file systems.
- Use cases include:
 - High-performance computing (HPC),
 - Machine learning, Electronic Design Automation (EDA)
 - Video Processing and Financial modeling
- With Amazon FSx for Lustre, in minutes, a Lustre file system can be launched that can process massive datasets at:
 - Up to **hundreds of gigabytes per second of throughput, millions of IOPS, and sub-millisecond latencies.**
- Amazon FSx for Lustre provides **high-performance storage at low cost, because it is nonreplicated, on-demand storage for short-term, compute-intensive processing of datasets.**



Amazon FSx for Lustre – Access to file systems

- An Amazon FSx for Lustre file system is accessed through an ENI created inside the VPC where you associated the file system
- Lustre clients (Linux machines only) are required to mount the FSx for Lustre file system to use them
 - This requires the installation of the Lustre client software on the client machines
- FSx for Lustre file system can be mounted to a Lustre Client in AWS (same VPC or in a VPC peered with the File system's VPC) , or to a Lustre client On-premise.
- For on-premise case, copy the data into the Lustre file system and run the compute-intensive workloads on in-cloud instances (not the on-premise instances).
- You can configure your EC2 instances to remount the FSx file system automatically after an instance reboot
- Amazon FSx for Lustre keeps the underlying software powering the file systems up-to-date
 - It also has a rich integration with other AWS services like Amazon S3, CloudWatch, and CloudTrail.



Amazon FSx for Lustre – Using Data from Repositories

- Amazon FSx for Lustre is designed for **short-term, compute-intensive workloads** where the long-term data is stored in a durable data repository, such as Amazon S3 or an on-premises data store.
 - An Amazon FSx for Lustre repository **isn't meant to store durable, long-term data.**
 - The file system is generally brought up only for the duration of intended compute job (typically several hours or days).
- Amazon FSx for Lustre is deeply integrated with Amazon S3.
- When the Amazon FSx for Lustre file system is being created, there is an option to link it to an Amazon S3 data repository
 - This means, FSx for Lustre file systems can automatically copy Amazon S3 data into the file system to run analysis for hours to days, then
 - Write results back to S3, and then delete the file system.
- Also, applications mounting the Amazon FSx for Lustre file system can seamlessly access the objects stored in the Amazon S3 buckets



Amazon FSx for Lustre – Dealing with data in On-Premise Repositories

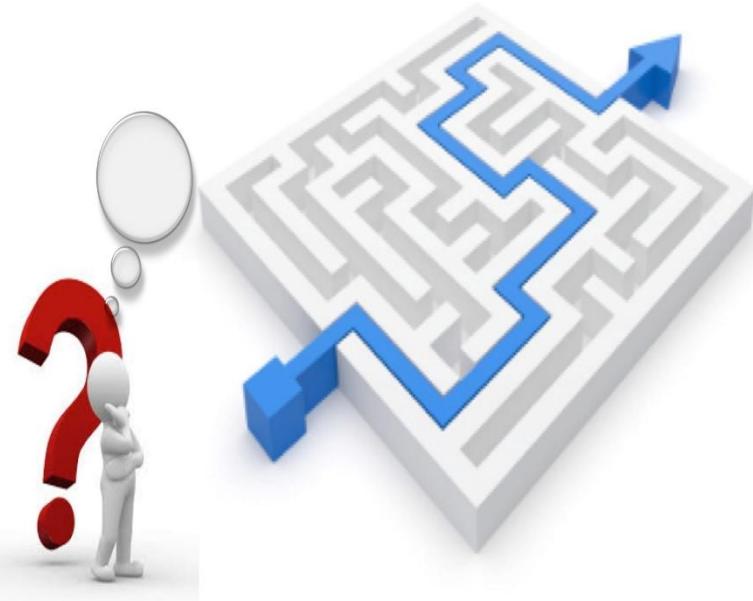
- Amazon FSx for Lustre can be used to process data stored in on-premises data repository with in-cloud compute instances.
 - Amazon FSx for Lustre supports cloud bursting workloads with on-premises data repositories
- This can be achieved by **enabling the copying of data from on-premises Lustre clients** using AWS Direct Connect or VPN.
- Compute-intensive workloads can be run on Amazon **EC2 instances in the AWS Cloud** and then results are copied back to the on-premise data repository when the workload processing in AWS is done.
 - Processing is not done in on-premise servers

Amazon FSx for Lustre – Security and Encryption

- Like FSx for Windows File Server, an ENI is created in the VPC to associate with the File System
- Security groups are used to protect the ENI
 - Specific ports needs to be allowed Inbound and Outbound to ensure successful access to the FSx for Lustre
- **Encryption**
 - FSx service automatically encrypts data before it is written to the file system, and decrypts it as it is being read
 - FSx service manages the KMS keys
 - Amazon FSx for Lustre does not support using Amazon S3 buckets encrypted using SSE-KMS and SSE-C for data repositories.

EFS vs FSx

EFS vs Amazon FSx for Windows File
Server vs Amazon FSx for Lustre



EFS vs FSx for Windows File Server vs. FSx for Lustre

EFS

- Amazon EFS is a cloud-native fully managed file system that provides simple, scalable, elastic file storage accessible from Linux instances via the **NFS protocol**.
- Can only be used with **Linux clients**/applications.

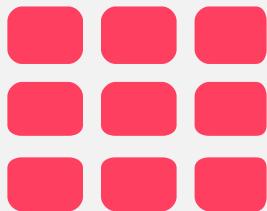
FSx for Windows File Server:

- Use it for Windows-based applications,
- Amazon FSx provides **fully managed Windows file servers** with features and performance optimized for "lift-and-shift" business-critical application workloads including home directories (user shares), media workflows, and ERP applications.
- It is accessible from **Windows and Linux** instances via the **SMB protocol**.

FSx for Lustre:

- For compute-intensive and fast processing workloads such as HPC, Machine learning, EDA, and Media processing,
- Amazon FSx for Lustre, provides a file system that's optimized for performance, with input and output stored on Amazon S3.
- Only for **Linux clients**





AMAZON ROUTE 53

You Can Do It Too!



Amazon Route53

Domain Registration with Route 53



Amazon ROUTE 53

ROUTE53

Route53 performs three main functions:

- a) Register a domain
- b) As a DNS, it routes Internet traffic to the resources for your domain
- c) Check the health of your resources –
 - Route53 sends automated requests over the internet to a resource (can be a web server) to verify that the server is, reachable, functional, available.
 - Also, you can choose to receive notifications when a resource becomes unavailable and choose to route internet traffic away from unhealthy resources.
- You can use AWS Route53 for any combination of these functions.



Amazon ROUTE 53

ROUTE53

- When you register a domain with Route 53, the **service automatically makes itself the DNS service for the domain** by doing the following:
 - it creates a hosted zone that has the same name as your domain.
 - It assigns a set of four name servers to the hosted zone , unique to the account.,
 - When someone uses a browser to access your website, these name servers inform the browser where to find your resources, such as a web server or an Amazon S3 bucket.
 - It gets the name servers from the hosted zone and adds them to the domain.



Amazon ROUTE 53

Supported DNS Record Types

Supported DNS Record types by ROUTE 53

- **A Record** – Address Record – Maps domain name to IP address
 - www.dolfined.com IN A 2.2.2.2
- **AAAA Record** - IPv6 Address Record – Maps domain name to an IPv6 address
 - www.dolfined.com IN AAAA 2001:d8b1::1
- **CNAME Record** – Maps an alias to a hostname
 - web IN CNAME www.dolfined.com
- **NS Record** – Name server record – Used for delegating zone to a nameserver
 - dolfined.com IN NS ns1.dolfined.com
- **SOA Record** – Start of Authority Record -
- **MX Record** – Mail Exchanger – Defines where to deliver mail for user @ a domain name
 - dolfined.com IN MX 10 mail01.dolfined.com
 - IN MX 20 mail02.dolfined.com
- CAA, PTR, NAPTR, SPF, SRV, TXT Records



Amazon ROUTE 53

Record Types

- NS records defines which Name Server is authoritative to a particular Zone or domain name and point you to other DNS servers
- A/AAAA are called Host Records, Like business cards
- CNAME is an alternative record or an alias for another record
 - Helpful in redirection or if you want to hide details about your actual servers from the users

Amazon ROUTE 53

CNAME Record

CNAME Record Type

A CNAME Value element is the same format as a domain name.

- The DNS protocol does not allow you to create a CNAME record for the top node of a DNS namespace, also known as the zone apex (or Root Domain, or Naked Domain).
 - For example, if you register the DNS name dolfined.com, the zone apex is dolfined.com,
 - You cannot create a CNAME record for dolfined.com,
 - However, you can create CNAME records for www.dolfined.com, support.dolfined.com, and so on.
- In addition, if you create a CNAME record for a subdomain, you cannot create any other records for that subdomain.
 - For example, if you create a CNAME for www.dolfined.com,
 - You cannot create any other records for which the value of the Name field is www.dolfined.com.



Amazon ROUTE 53

Alias Record

- Specific to Route 53 and not seen outside.
- You can use it to create DNS Route53 Records and route queries to AWS services the IP address of which can change (CLB/ALB/NLB, CloudFront Distribution, S3 Bucket configured as a static website, ElasticBeanStalk environment, API Gatewaty, VPC interface endpoint, Global Accelerator accelerator, and another route53 record in the same hosted zone.
 - When you point an Alias to one of these AWS services, Route53 will fetch the IP address of that service's resource(s) in real time to respond to DNS queries
- You CAN'T create a CNAME for the apex/naked/root domain name,
 - Alias Record CAN do that
- You CAN NOT Alias to a record or resource outside of AWS Route 53 or AWS Services



AWS ROUTE 53

Working with Records

- The name of each record in a hosted zone must end with the name of the hosted zone. For example, the example.com hosted zone can contain records for www.example.com and accounting.tokyo.example.com subdomains, but cannot contain records for a www.example.ca subdomain.

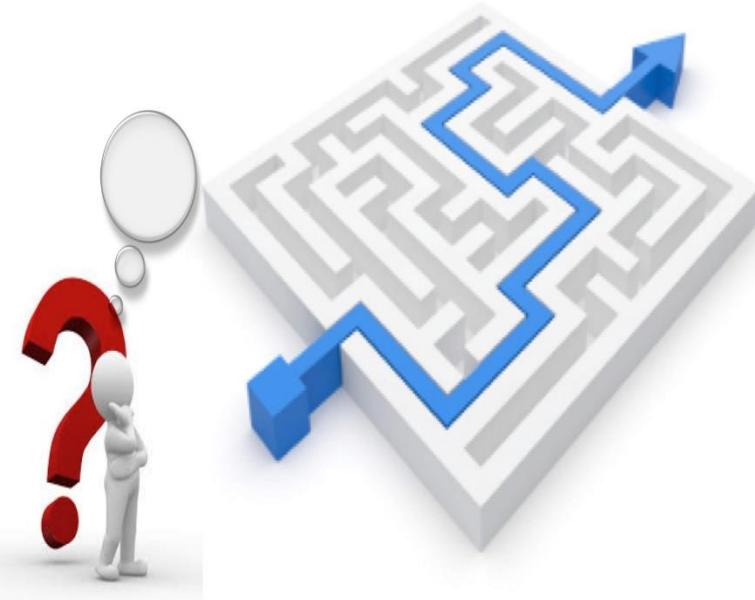
Note

- To create records for complex routing configurations, you can also use the **traffic flow visual editor** and save the configuration as a traffic policy. You can then associate the traffic policy with one or more domain names (such as example.com) or subdomain names (such as www.example.com), in the same hosted zone or in multiple hosted zones. In addition, you can roll back the updates if the new configuration isn't performing as you expected it to.



Amazon Route53

ALIAS vs CNAME Records



Amazon ROUTE 53

CNAME vs ALIAS Records

Alias records are similar to CNAME records, but there are some important differences:

CNAME Records	Alias Records
Route 53 charges for CNAME queries.	Route 53 doesn't charge for alias queries to CloudFront distributions, Elastic Beanstalk environments, ELB load balancers, or Amazon S3 buckets. For more information, see Amazon Route 53 Pricing .
You can't create a CNAME record at the top node of a DNS namespace, also known as the <i>zone apex</i> . For example, if you register the DNS name example.com, the zone apex is example.com.	You can create an alias record at the zone apex. Note If you create an alias record that routes traffic to another record in the same hosted zone, and if the record that you're routing traffic to has a type of CNAME, you can't create an alias record at the zone apex. This is because the alias record must have the same type as the record you're routing traffic to, and creating a CNAME record for the zone apex isn't supported even for an alias record.
A CNAME record redirects queries for a domain name regardless of record type.	Route 53 follows the pointer in an alias record only when the record type also matches.



Amazon ROUTE 53

CNAME vs ALIAS Records

CNAME Records	Alias Records
A CNAME record can point to any DNS record hosted anywhere, including to the record that Route 53 automatically creates when you create a policy record. For more information, see Using Traffic Flow to Route DNS Traffic .	An alias record can only point to a CloudFront distribution, an Elastic Beanstalk environment, an ELB load balancer, an Amazon S3 bucket that is configured as a static website, or another record in the same Route 53 hosted zone in which you're creating the alias record. However, you can't create an alias that points to the record that Route 53 creates when you create a policy record.
A CNAME record is visible in the answer section of a reply from a Route 53 DNS server.	An alias record is only visible in the Route 53 console or the Route 53 API.
A CNAME record is followed by a recursive resolver.	An alias record is only followed inside Route 53. This means that both the alias record and its target must exist in Route 53.



Amazon Route53

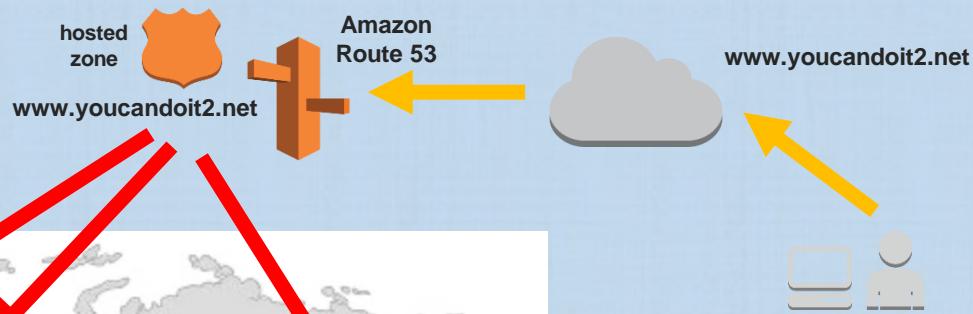
Health Checks



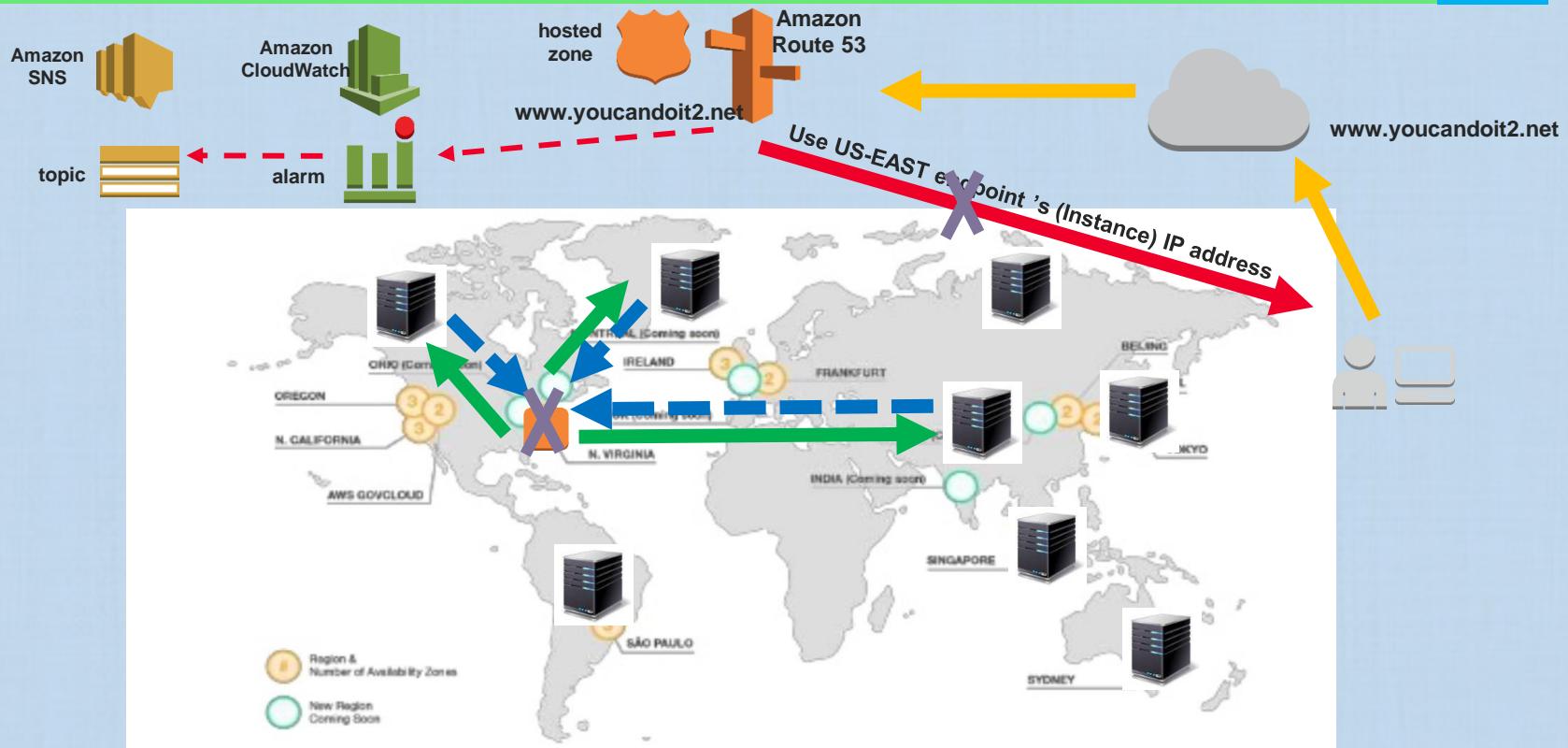
Route 53 – Health Checks

- **Route 53 supports** HTTP, HTTPS, TCP health checks
- You can define the IP address or the domain of the endpoint to be monitored by Route 53 Health checks
 - The endpoint can be in AWS or off AWS (charge is higher for off AWS endpoints)
 - Route 53 can't check the health of resources that have an IP address in local, private, non-routable, or multicast ranges.
- Route 53 begins to check the health of the endpoint that you specified in the health check when you associate a health check with the record.
- Optionally, if configured, Route 53 can notify CloudWatch of unhealthy instances, it sets a CloudWatch Alarm, Then CloudWatch will use SNS to send a notification about the unhealthy endpoint.
- If one or more records in a group of records do not have health checks associated with them,
 - Route 53 will treat these records as healthy, since it has no means to decide the health of the corresponding resource(s).

Route 53 - Without Health Checks



Route 53 – With health checks



Route 53 – Health Checks

- Create health checks for the resources that you can't create alias records for, this includes EC2 instances or servers in an on-premise data center.
 - **For Alias records** – Best is to specify **Yes** for **Evaluate Target Health** parameter
- Each health check created can monitor one of the following:
 - The health of the specified endpoint/resource, such as a web server
 - The status of other health checks themselves
 - Example, when it is required to be notified if 3 out of 6 available web servers (all are monitored by Route 53) are unhealthy.
 - The status of a configured CloudWatch alarm
- Firewalls, security groups, NACLs..etc, need to have rules configured to allow Route 53 to send regular requests to the endpoints specified in the configured health checks.



Amazon Route53

Routing Policies



AWS ROUTE 53

Choosing a Route 53 Routing Policy

- When you create a record, you choose a routing policy, which determines how Amazon Route 53 responds to queries. Possible values are:
- **Simple routing policy** – Default –
 - Use for a single resource that performs a given function for your domain,
 - Use case: a web server that serves content for the dolfined.com website.
- **Failover routing policy** –
 - Use when you want to configure active-passive failover.
- **Geolocation routing policy** –
 - Use when you want to route traffic based on the location of your users.



AWS ROUTE 53

Choosing a Route 53 Routing Policy

- **Latency routing policy –**
 - Use when you have resources in multiple locations and you want to route traffic to the resource that provides the best latency.
- **Weighted routing policy –**
 - Use to route traffic to multiple resources in proportions that you specify.
- **Geoproximity routing policy (Requires Route Flow) –**
 - Use when you want to route traffic based on the location of your resources and, optionally, shift traffic from resources in one location to resources in another.
- **Multivalue answer routing policy –**
 - Use when you want Route 53 to respond to DNS queries with up to eight healthy records selected at random.



AWS ROUTE 53

Failover Routing

Failover routing lets you route traffic to a resource when the resource is healthy

- If the main resource is not healthy, then route traffic to a different resource
- The primary and secondary records can route traffic to anything from an Amazon S3 bucket that is configured as a website to a complex tree of records.



Amazon ROUTE 53

Geolocation Routing

- Geolocation routing lets you choose the resources that serve your traffic based on the geographic location of your users, i.e the location that DNS queries originate from.
- For example, you may have presence in USA and Europe and want users in the US to be served in the US, and those in Europe to be served by servers in Europe.
- Use cases and benefits for using geolocation routing,
 - Localize your content and present some or all of your website in the language of your users.
 - Use geolocation routing to restrict distribution of content to only the locations in which you have distribution rights.
 - Balancing load across endpoints in a predictable, easy-to-manage way, so that each user location is consistently routed to the same endpoint.



Amazon ROUTE 53

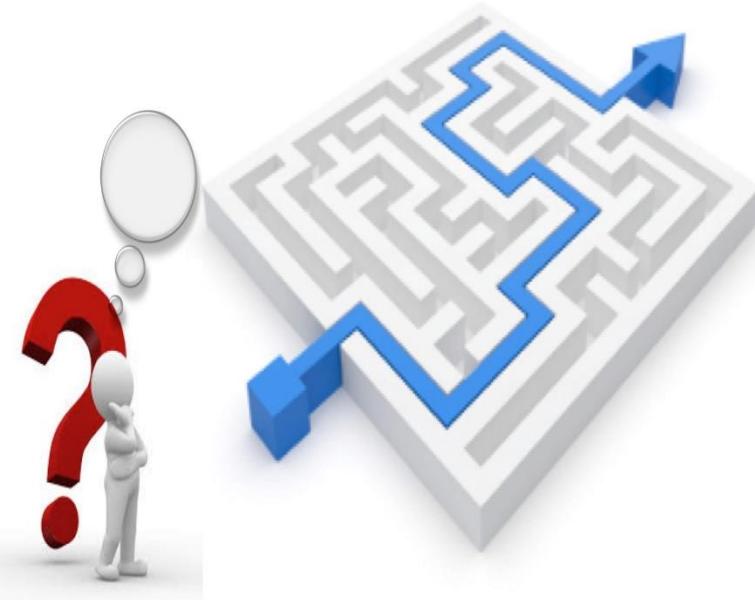
Geolocation Routing

- You can specify geographic locations by continent, by country, or by state in the United States.
- If you create separate records for overlapping geographic regions—for example, one record for North America and one for Canada—priority goes to the smallest geographic region.
 - This allows you to route some queries for a continent to one resource and to route queries for selected countries on that continent to a different resource.
- Geolocation works by mapping IP addresses to locations.
- However, some IP addresses aren't mapped to geographic locations,
 - For those IP addresses, even if you create geolocation records that cover all seven continents, Amazon Route 53 will receive some DNS queries from locations that it can't identify.
 - You can create a default record that handles both queries from IP addresses that aren't mapped to any location and queries that come from locations that you haven't created geolocation records for.
 - If you don't create a default record, Route 53 returns a "no answer" response for queries from those locations.



Amazon Route53

Routing Policies (cont.)



Amazon ROUTE 53

Latency Based Routing

- If an application is hosted in multiple regions, performance for your users can be improved by serving their requests from the Amazon region that provides the lowest latency.
- To use latency-based routing, you create latency records for your resources in multiple Regions.
- When Amazon Route 53 receives a DNS query for your domain or subdomain,
 - It determines which Amazon regions you've created latency records for,
 - Determines which region gives the user the lowest latency,
 - Then selects a latency record for that region,
 - Route 53 responds with the value from the selected record, such as the IP address for a web server.



Amazon ROUTE 53

Latency Based Routing

Example, suppose you have ELB load balancers in the US West (Oregon) Region and in the Asia Pacific (Singapore) Region.

- You created a latency record for each load balancer.
- Here's what happens when a user in London enters the name of your domain in a browser:
 - DNS routes the request to a Route 53 name server.
 - Route 53 refers to its data on latency between London and the Singapore region and between London and the Oregon region.
 - If latency is lower between the London and Oregon regions, Route 53 responds to the query with the IP address for the Oregon load balancer.
 - If latency is lower between London and the Singapore region, Route 53 responds with the IP address for the Singapore load balancer Singapore.



Amazon ROUTE 53

Weighted Routing

- Weighted routing lets you associate multiple resources with a single domain name, or subdomain name, and choose how much traffic is routed to each resource.
- This can be useful for a variety of purposes, including load balancing and testing new versions of software.
- To configure weighted routing, you create records that have **the same name and type** for each of your resources.
- You assign each record a relative weight that corresponds with how much traffic you want to send to each resource.
- Amazon Route 53 sends traffic to a resource based on the weight that you assign to the record as a proportion of the total weight for all records in the group:

$$\frac{\text{Weight of the specified Record}}{\text{Sum of the weight of all records}}$$



Amazon ROUTE 53

Weighted Routing

Example,

If you want to send a tiny portion of your traffic to one resource and the rest to another resource, you might specify weights of 1 and 255.

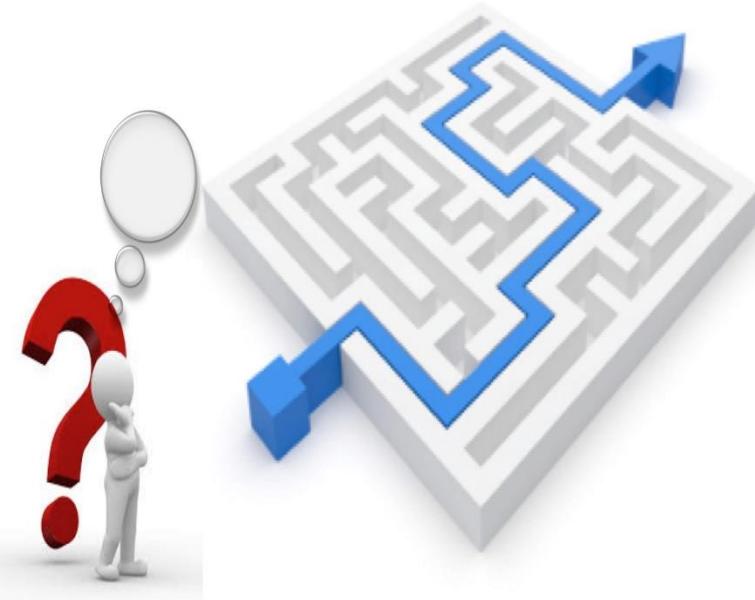
The resource with a weight of 1 gets 1/256th of the traffic ($1/1+255$),
The other resource gets 255/256ths ($255/1+255$).

You can gradually change the balance by changing the weights. If you want to stop sending traffic to a resource, you can change the weight for that record to 0.



Amazon Route53

Routing Policies (cont.)



Amazon ROUTE 53

Geo-proximity Routing

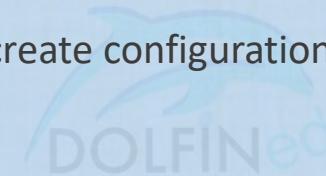
- Use Geoproximity routing to have Amazon Route 53 route traffic to your resources based on the geographic location of your users and your resources.
- You can also optionally choose to route more traffic or less to a given resource by specifying a value, known as a *bias*.
 - A bias expands or shrinks the size of the geographic region from which traffic is routed to a resource.
- Route 53 traffic flow is required to use Geoproximity routing.
- To create geoproximity rules for the resources, specify one of the following values for each rule:
 - If you're using AWS resources, the AWS Region that you created the resource in
 - If you're using non-AWS resources, the latitude and longitude of the resource



Amazon ROUTE 53

Traffic Flow

- Route 53 traffic flow provides a visual editor that help in creating complex trees easily.
- The created configuration (routing tree) can be saved as a *traffic policy*
- You can associate the traffic policy with one or more domain names (such as example.com) or subdomain names (such as www.example.com), in the same hosted zone or in multiple hosted zones.
- You can only use traffic flow to create configurations for public hosted zones.

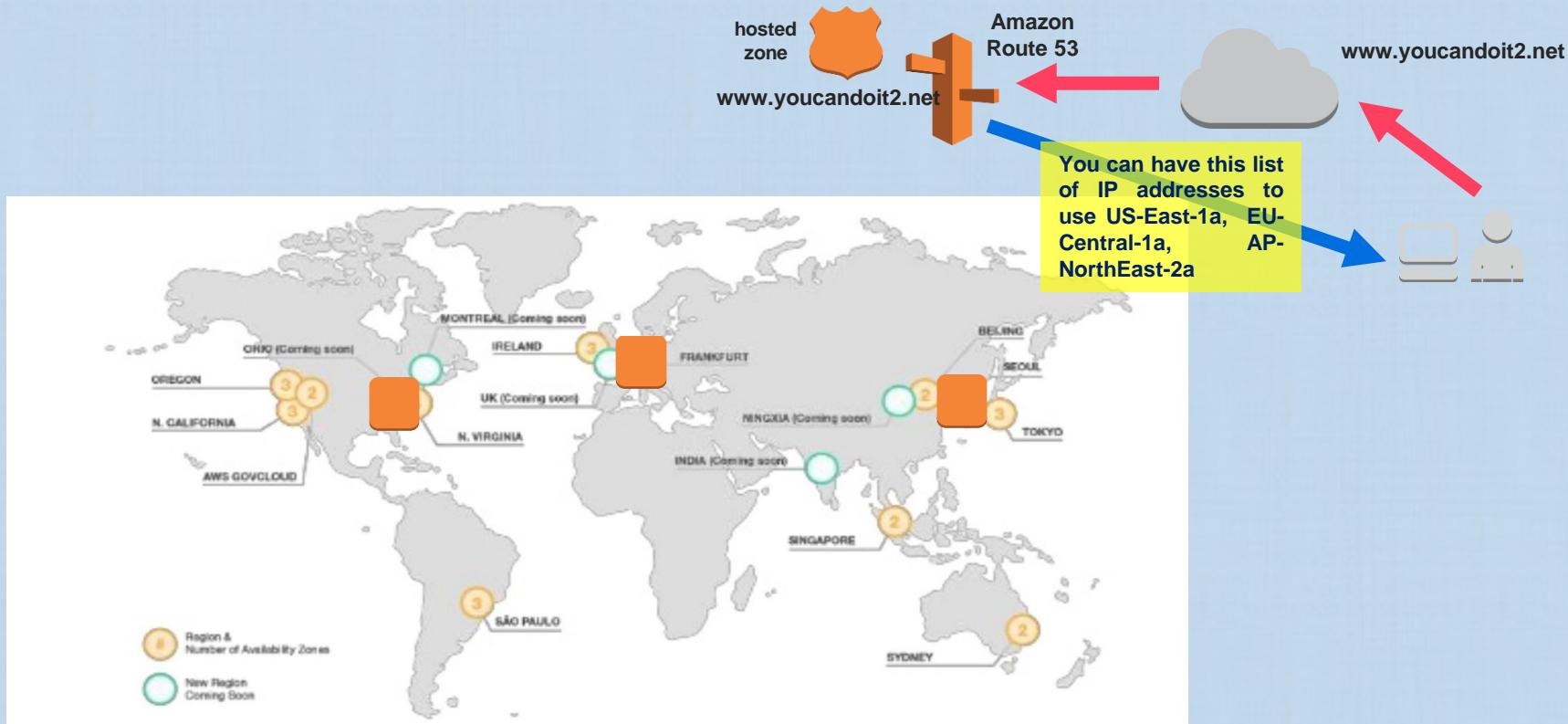


Route 53 – MultiValue Answer

- **Route 53 supports** configuring MultiValue answer routing policy, where more than one IP will be returned in the response to the DNS query
 - It's not a substitute for a load balancer, but the ability to return multiple health-checkable IP addresses is a way to use DNS to improve availability and load balancing.
- It support health checks on the different Route 53 records
 - Which means only healthy endpoint will be returned.
- Route 53 responds to DNS queries with up to eight healthy records and gives different answers to different DNS resolvers.
- If a web server becomes unavailable after a resolver caches a response, client software can try another IP address in the response.
- When all records are unhealthy, Route 53 responds to DNS queries with up to eight unhealthy records.



Route 53 – MultiValue Answer



Route 53 Resolver

Resolving DNS Queries Between VPCs
And the On-Premise Network



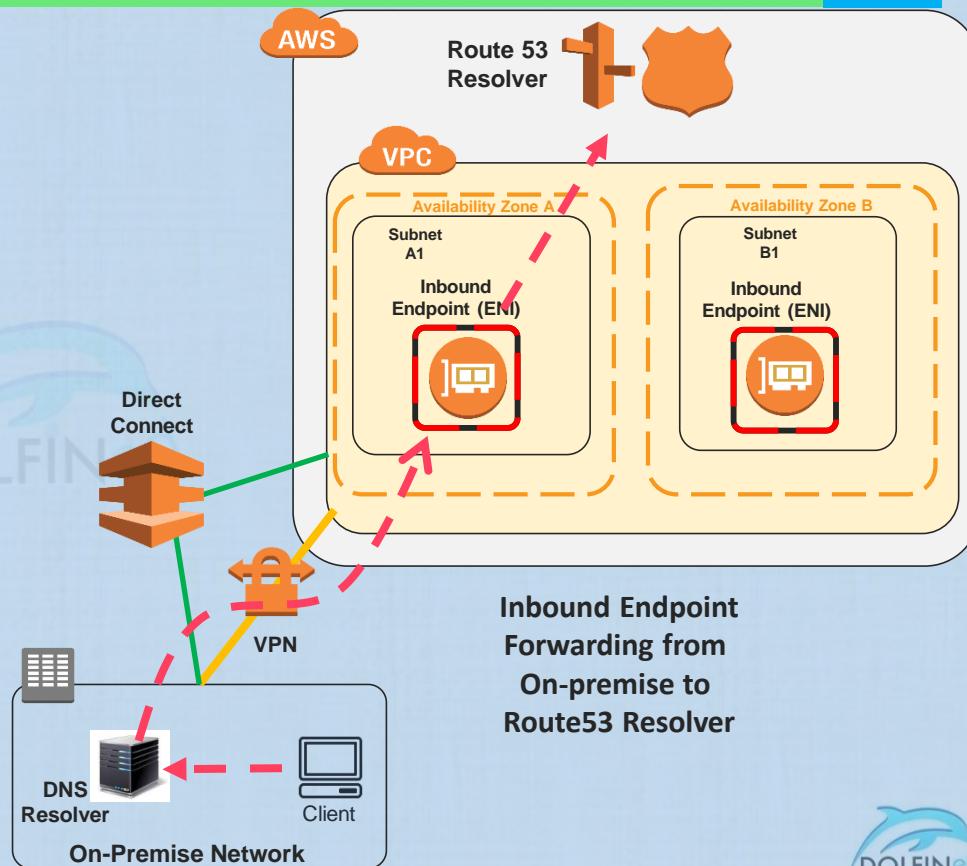
Route 53 Resolver

- A Route 53 DNS Resolver is there by default when you create a VPC in AWS
 - The default function is to resolve DNS queries within the VPC
 - It answers DNS queries for the VPC domain names (for ELBs, EC2 instances...etc within the VPC)
 - For all other Domain names (not within the VPC, such as Public Domain names on the internet), the Route 53 Resolver will do recursive lookups against public DNS resolvers
- You can additionally configure (manually) the Route 53 Resolver to:
 - Forward DNS queries from within the VPC to your DNS Resolvers on-premise (Outbound Queries), and/or
 - Answer DNS queries coming from your On-premise network clients (Inbound queries)
 - To allow on-premise clients to use the private hosted zones configured in your VPCs, and
 - Resolve domain names for AWS resources
- The above requires a Direct Connect or a VPN connection between your on-premise network and the AWS VPC(s) in question.
- Route 53 Resolvers are Region specific resources.



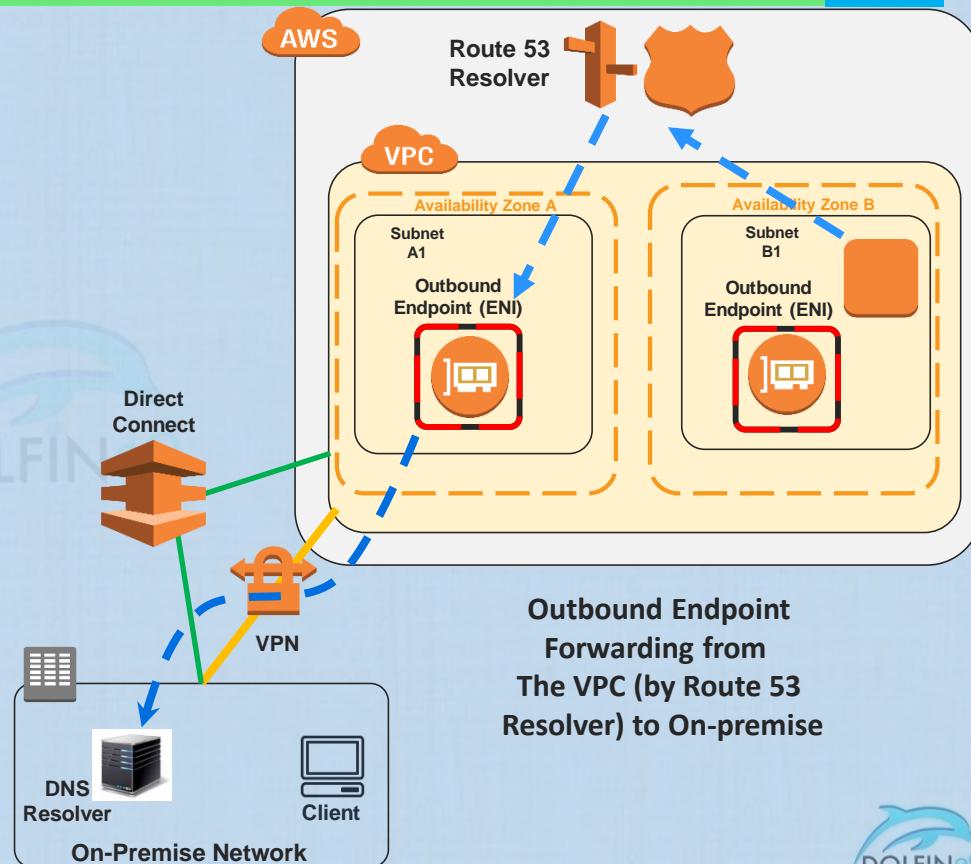
Inbound Query Forwarding – On-premise to Route53 Resolver

- The inbound endpoint is one or more ENIs created in your VPC with the IP address(es) [one ENI for each IP address] that you specified during the endpoint creation. One endpoint is enough to handle inbound direction queries.
- Each endpoint needs to be configured with two (min), or more, IP addresses in different subnets and in different availability zones.
 - All in the same AWS Region.
- You can create more than one endpoint, however, For increased load, AWS recommends adding more IP addresses to the same endpoint than creating new endpoints
- On Premise resolvers need to be configured to forward the DNS queries for the domains in AWS to the Inbound endpoints' IP address(es).
- Creating an inbound endpoint doesn't change the behavior of Resolver, it just provides a path from a location outside the AWS network to Resolver.



Outbound Query Forwarding – VPC to On-premise DNS Resolver

- To forward DNS queries from EC2 instances in one or more VPCs to the on-premise network, you need to configure outbound endpoints and one or more forwarding rules.
- The outbound endpoint is an ENI created in your VPC with an IP address and a subnet you specified
- The endpoint defines the VPC and IP address that the Resolver will forward DNS queries through (where the queries will originate from)
- You can use the same outbound endpoint for multiple VPCs in the AWS region.
 - Also you can create multiple endpoints
 - Two IP addresses, or more, need to be defined for the endpoint
- A forwarding rule can define a domain name for which the Resolver will forward DNS queries to the on-premise network
- Rules are associated with the VPC(s) for which queries will be forwarded



Route 53 Resolver - Rules

- Rules are required to help Resolver decide on which DNS queries to forward from VPC(s) to On-premise network (DNS)
- Rules are either Autodefined or Custom Rules
- Rule types:
 - Conditional Forwarding (or Forwarding) Rules
 - They define which DNS queries for which domain(s) to forward to On Premise DNS Resolver
 - System Rules:
 - These will have the Route 53 Resolver respond to queries and not send them to On-premise DNS
 - Recursive Rule:
 - It a rule that gets created automatically by the Route 53 Resolver, it is called Internet Resolver
 - This rules makes Route 53 Resolver act as a recursive resolver for any Domain names that do not have custom rules defined for, and Resolver did not have any auto-defined rules created for them
- Custom rules can be created for AWS domain names, Public domain names, or all domain names.
- Rules are region specific, to use a rule in more than one region, you must create the rule in each region

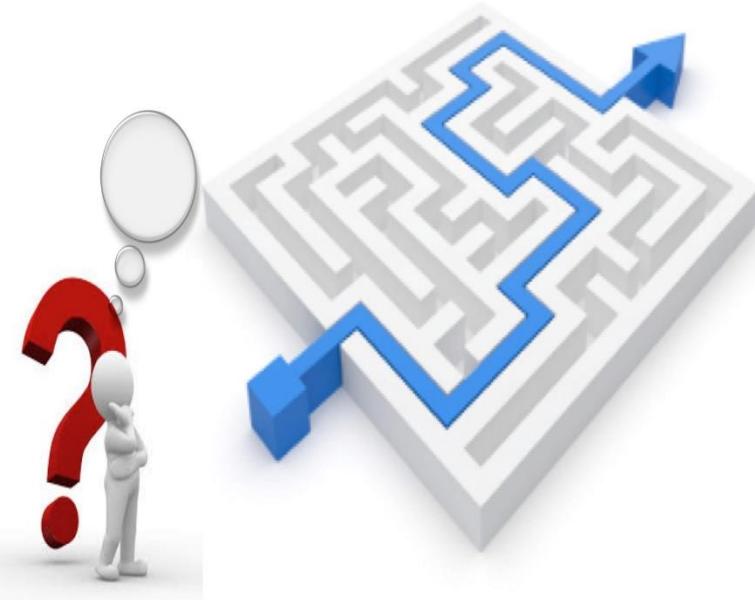
Sharing Rules between accounts

- You can share the forwarding rules that you created using one AWS account with other AWS accounts.
- To share rules, the Route 53 Resolver console integrates with AWS Resource Access Manager.
- When you create a rule, you specify the outbound endpoint that you want Resolver to use to forward DNS queries to your network.
 - If you share the rule with another AWS account, you also indirectly share the outbound endpoint that you specify in the rule.
- If you used more than one AWS account to create VPCs in an AWS Region, you can do the following:
 - Create one outbound endpoint in the Region.
 - Create rules using one AWS account.
 - Share the rules with all the AWS accounts that created VPCs in the Region.
- This allows you to use one outbound endpoint in a Region to forward DNS queries to your network from multiple VPCs even if the VPCs were created using different AWS accounts.



Amazon Route53

Pricing



AWS ROUTE 53

Route 53 - Pricing

- Hosted Zones
 - \$0.50 per hosted zone / month for the first 25 hosted zones
\$0.10 per hosted zone / month for additional hosted zones
 - The monthly hosted zone prices listed above are not prorated for partial months.
 - A hosted zone is charged upon set-up and on the first day of each subsequent month.
 - To allow testing, a hosted zone that is deleted within 12 hours of creation is not charged;
 - However, any queries on that zone will be charged at the rates below.
- Query charges
- Health Check charges
- Traffic flow policy records
- Route 53 Resolver and Recursive DNS Queries

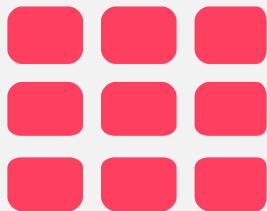


AWS ROUTE 53

ROUTE53

- Queries to Alias records that are mapped to Elastic Load Balancers, Amazon CloudFront distributions, AWS Elastic Beanstalk environments, and Amazon S3 website buckets are free.
- Alias records can be created for all query types: standard queries, latency-based routing queries, weighted and geo queries.
- Amazon Route 53 doesn't charge for the records that you add to a hosted zone.





GLOBAL ACCELERATOR

You Can Do It Too!



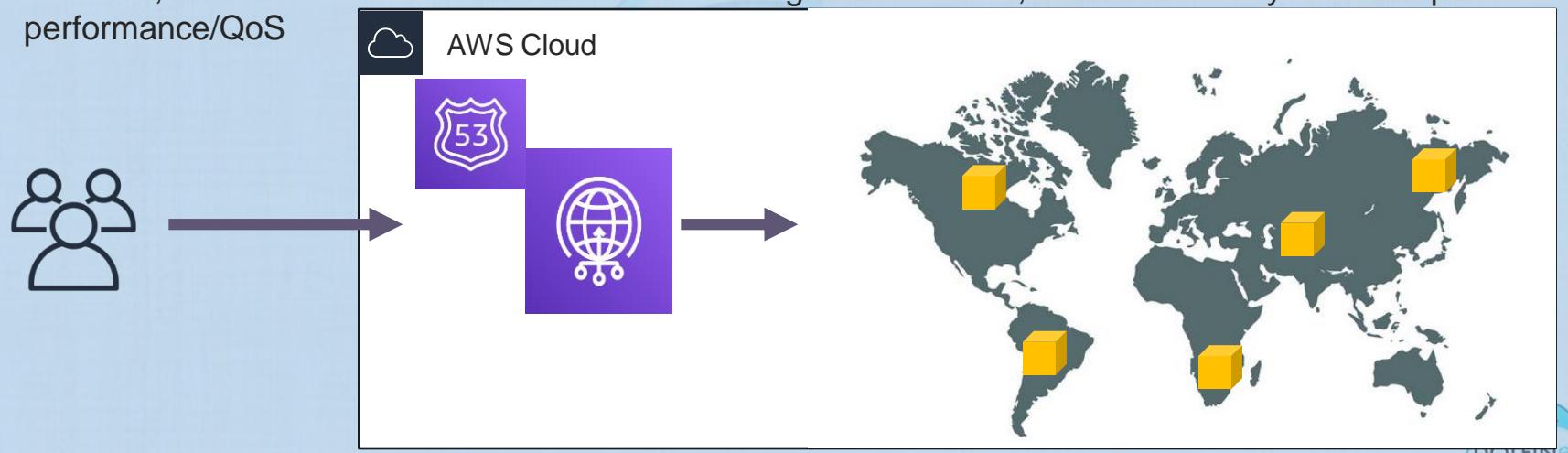
Amazon Global Accelerator

Global Accelerator



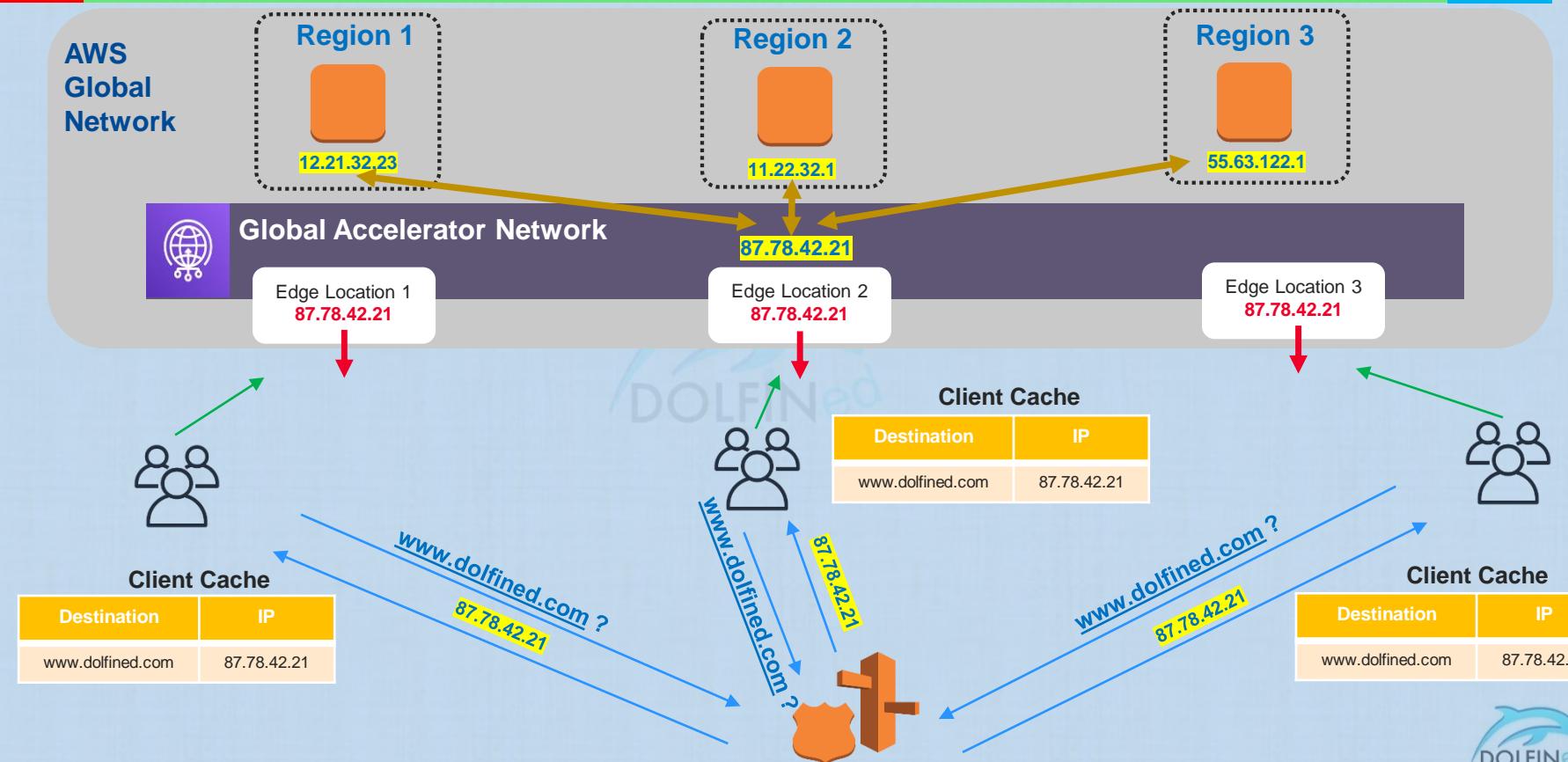
Without Global Accelerator – What is it?

- Global Accelerator is a network layer service that is horizontally scaled and highly available.
- It can be deployed in front of your Internet facing applications
- Incoming user traffic gets distributed intelligently across multiple endpoints in one or more AWS regions
- This will improve applications' availability and performance for the global user base
- With Global Accelerator Traffic from users to applications enters the AWS network at the nearest edge locations, then it is carried over the AWS global network, better security and improved performance/QoS



Global Accelerator – The Solution

Not for copy, modification or Redistribution –
Please report any breach to info@dolfined.com



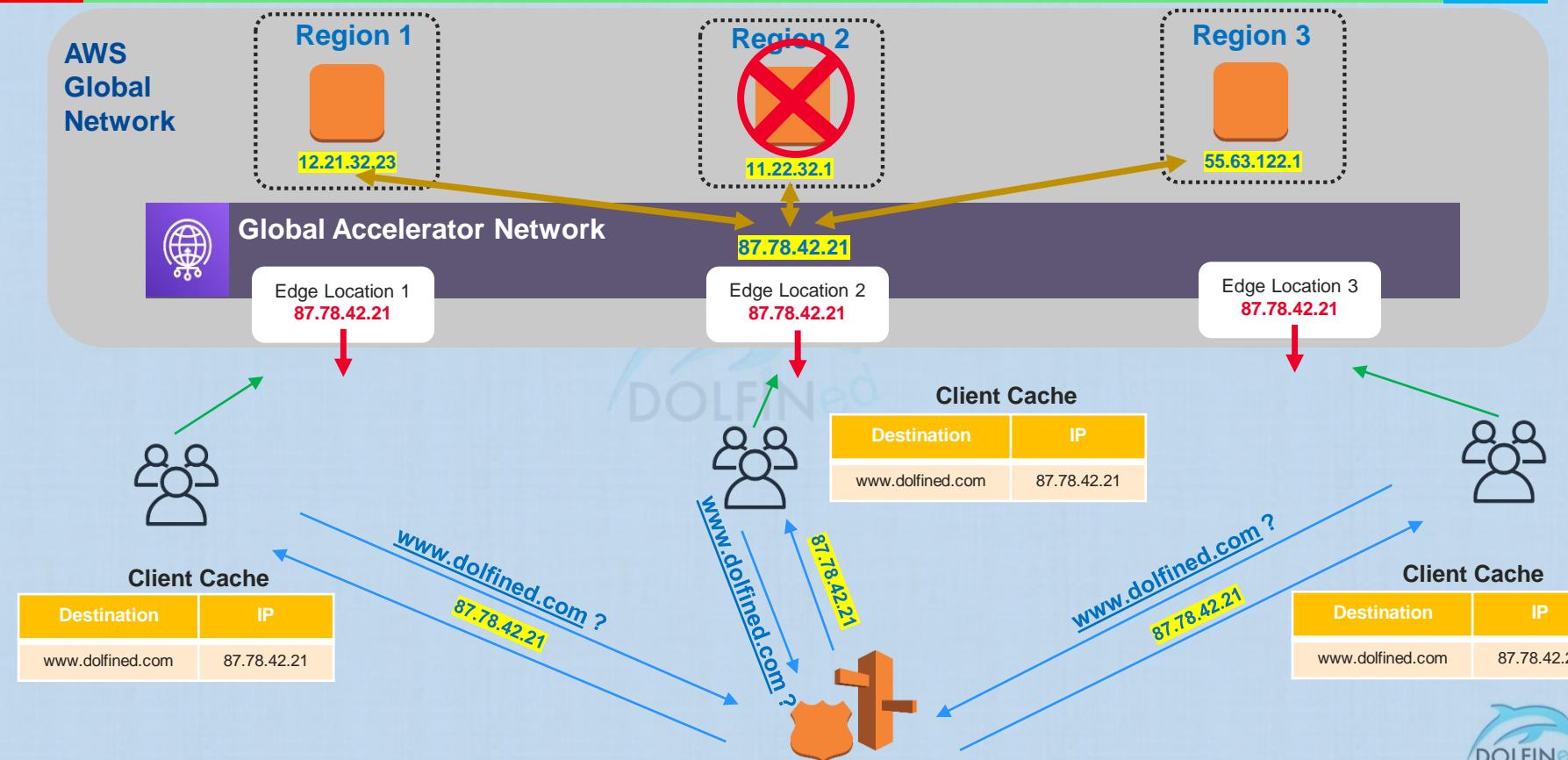
Global Accelerator

- All Users point to the same static IPs. Anycast IPs advertised from all Global Accelerator locations
- Since all Global Accelerator Locations can serve the content
 - Clients are directed to the closest location then the traffic is carried over the AWS global network
- Endpoints can be EC2 Instances, Application & Network Load Balancers, and Elastic IP addresses
- Global Accelerator maps the Anycast IP(s) to Endpoint IP address.



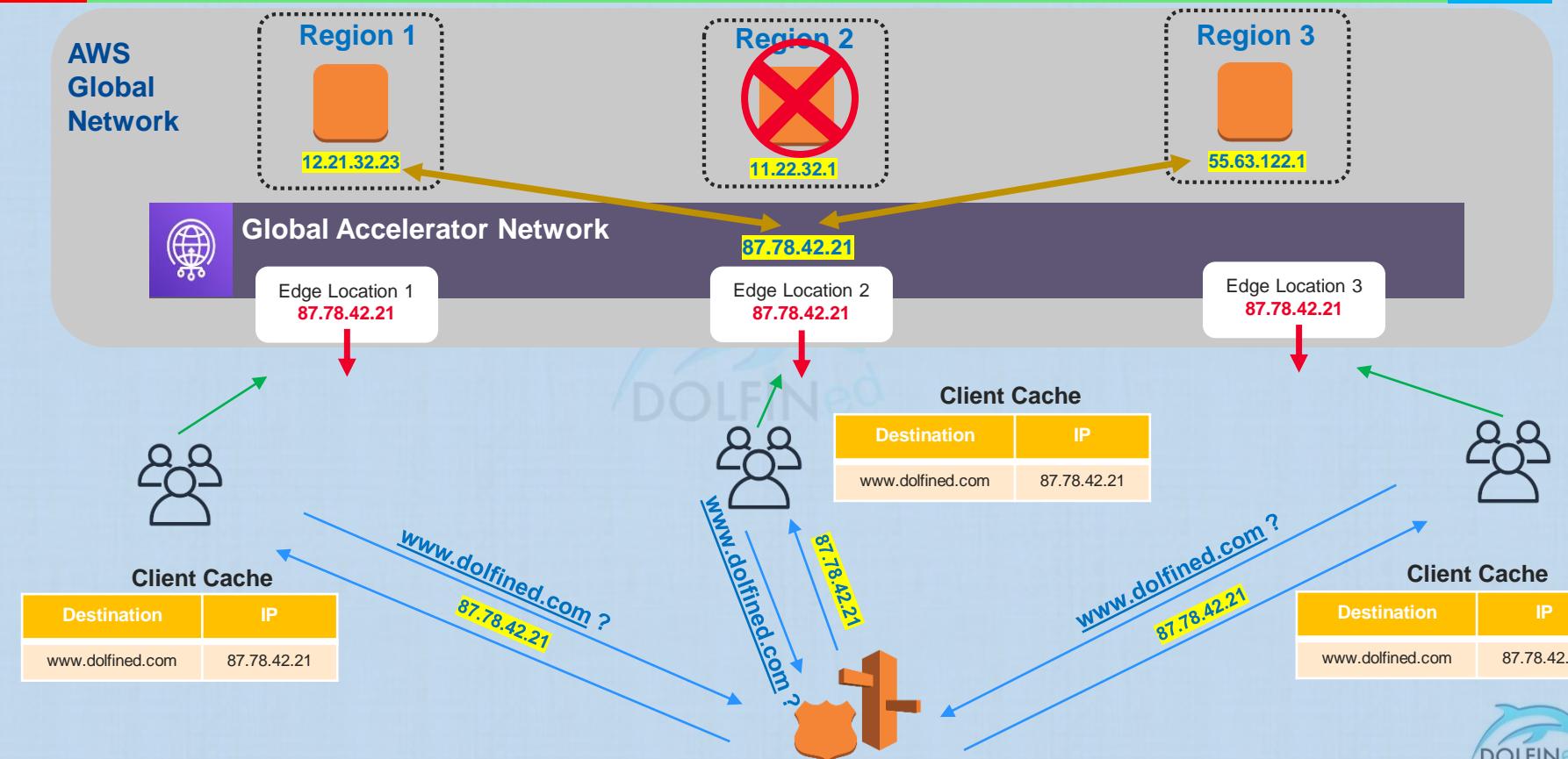
Global Accelerator – The Solution

Not for copy, modification or Redistribution –
Please report any breach to info@dolfined.com



With Global Accelerator – The Solution

Not for copy, modification or Redistribution –
Please report any breach to info@dolfined.com



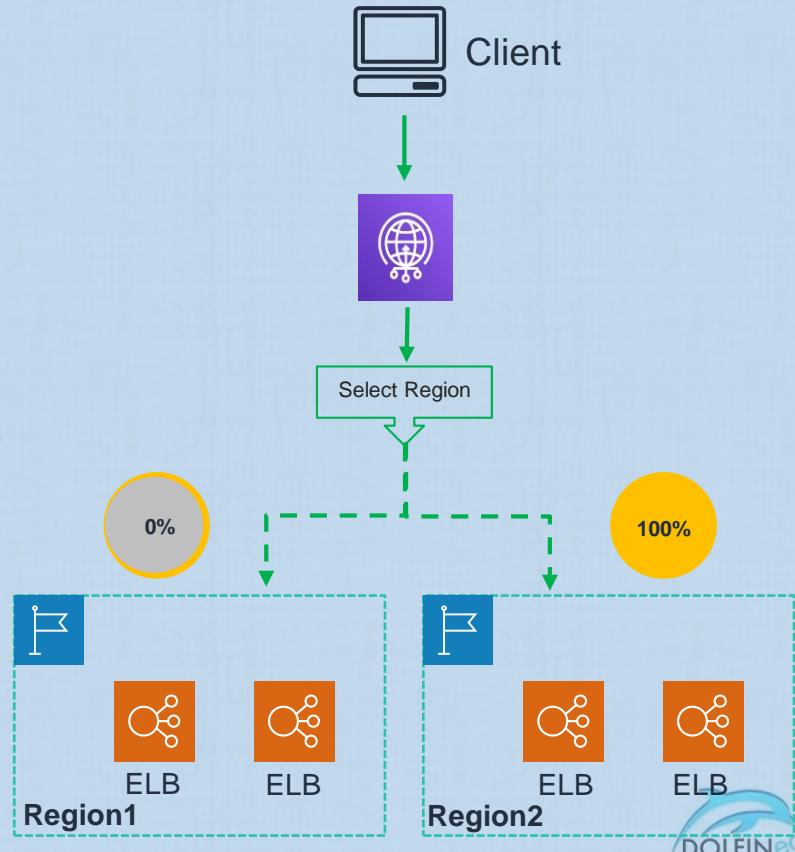
Global Accelerator – Key Benefits

- Two Static Anycast IP addresses are provided for each accelerator created
 - They are assigned to the accelerator as long as it exists, even if it is disabled and not routing traffic
 - They are taken back only when the accelerator is deleted
- Intelligent traffic distribution
- Enhanced fault tolerance to customer applications
- Supports both TCP and UDP protocols
- Endpoint health check monitoring and instant failover to another endpoint when the active endpoint becomes unhealthy
- Traffic rides over AWS's Global Network – Much better performance
- It integrates with AWS Shield to provide DDoS protection to your applications
- Fixed IP address benefits:
 - Application scaling to new AWS Regions or AZs
 - Migration between endpoint types
 - Whitelisting of IP addresses in Security applications
 - Stack upgrades and performance testing



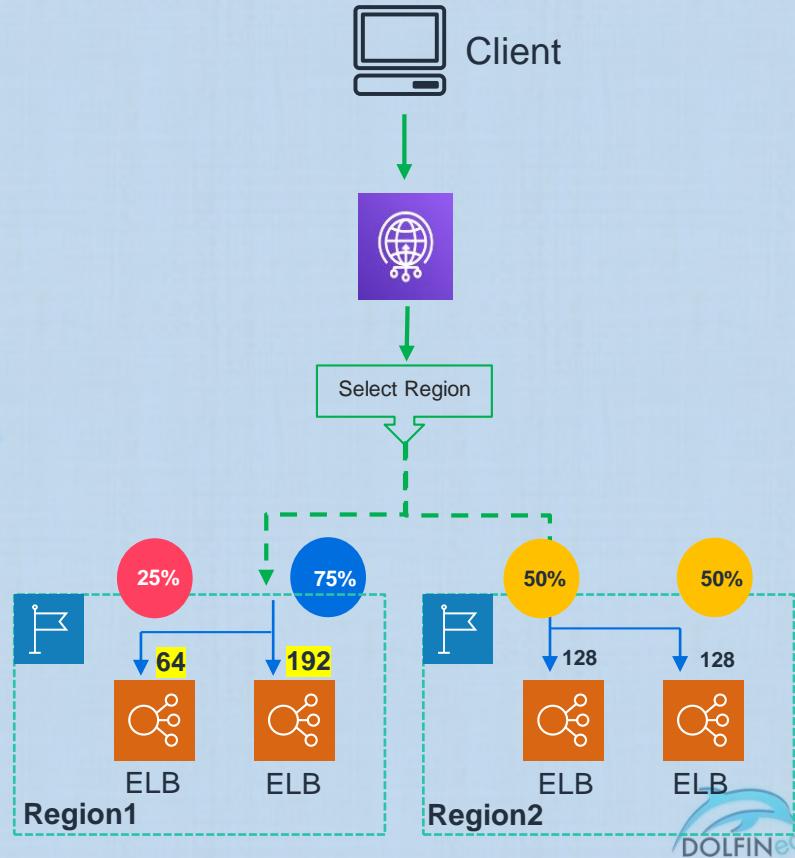
Regional Traffic Dial

- **Regional Traffic Dial** can be configured at the endpoint group level to increase (dial up) or reduce (dial down) the percentage of traffic that will be accepted (directed) to an endpoint group
 - Useful for performance testing or blue/green deployments



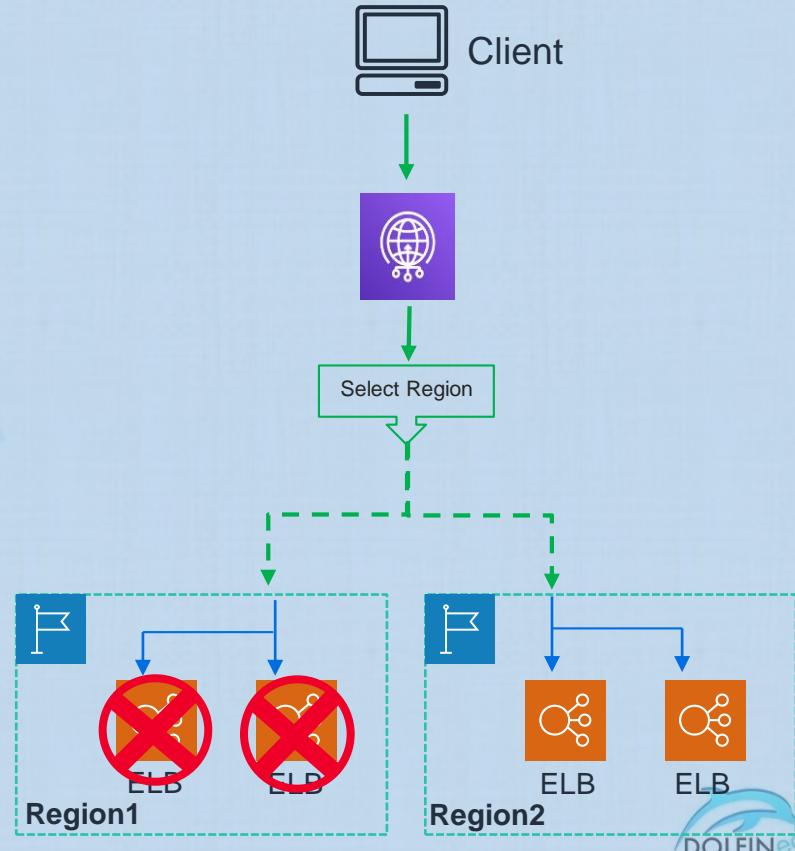
Endpoint Weight

- Weight of an endpoint is a number that you can configure to specify the proportion of traffic to route to the endpoint within an Endpoint group
- Default is 128
- Min is 0
- Max is 256



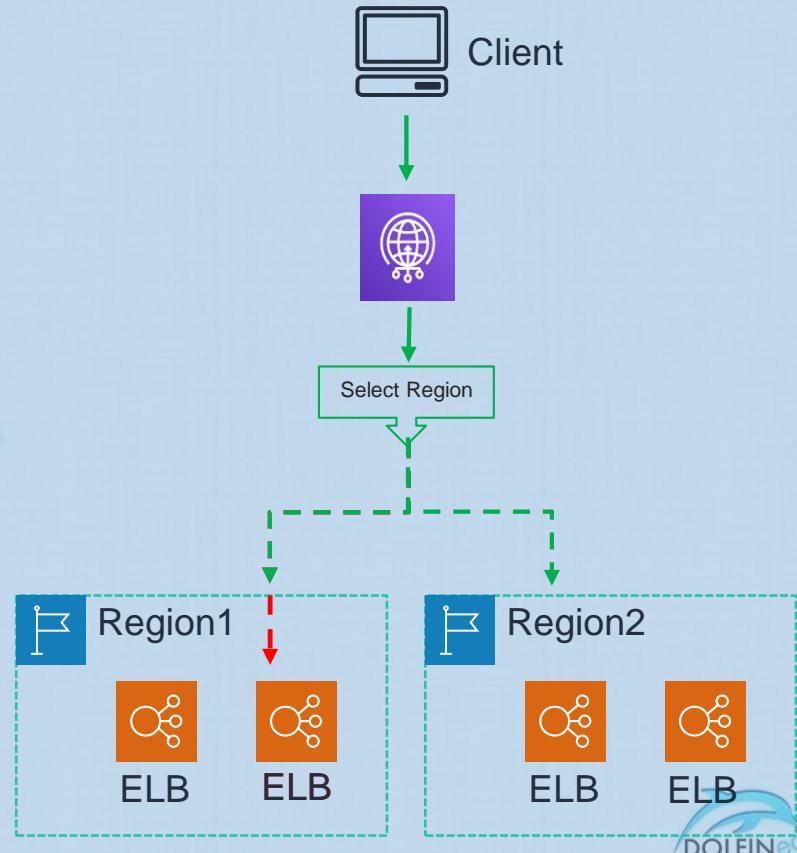
Global Accelerator – Endpoint Health Checks

- AWS Accelerator continuously checks the health of the endpoints associated with the accelerator's static IPs
- Traffic is directed to healthy endpoints only
- If there are no healthy endpoints the accelerator fails open
- You can configure the health checks at the endpoint group level
 - For ALB/NLB, their configured ELB health checks are used
 - For EC2 and Elastic IP addresses, the configured endpoint group settings are used (TCP/HTTP(s))
 - For UDP listeners, currently Global Accelerator only supports TCP health checks, hence, endpoints must have an active TCP health check process running



Global Accelerator – Client Affinity

- If you have stateful applications, you can choose to have Global Accelerator direct all requests from a user at a specific source (client) IP address to the same endpoint resource, to maintain client affinity.
- When configured, Optimal/Target region is selected, then
 - Traffic is directed to an endpoint which is selected based on a 2-tuple hash that is based on source (client) and endpoint IP address.
 - All traffic from that source IP to the destination is routed to the same endpoint for that flow.



Global Accelerator – Client IP address preservation

- With this feature, you preserve the source IP address of the original client for packets that arrive at the endpoint.
- You can use this feature with Application Load Balancer and EC2 instance endpoints
- When you use an internet-facing Application Load Balancer as an endpoint with Global Accelerator, you can choose to preserve the source IP address of the original client for packets that arrive at the load balancer by enabling client IP address preservation.



Global Accelerator – Use cases

- Application that require whitelisting of a small number of IPs
 - Autonomous vehicles
 - Payment/retail transactions
 - Healthcare
 - IoT
- Multi-region applications:
 - Financial Services
 - DR/Failover scenarios
- UDP traffic applications
 - Gaming
 - Voice over IP
 - DNS
- Live Video Ingestion for media applications
 - Latency sensitive applications



Global Accelerator – Visibility and Monitoring

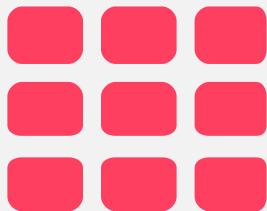
- **Flow Logs** are used to capture information about IP traffic going to and from ENIs in your configured accelerators.
 - Flow log data is published to S3 to a bucket that you can specify.
 - Log data captured includes among other information: Client IP address/port, Endpoint region, Endpoint IP address/port, statistics about packets and bytes.
- **CloudWatch:**
 - Global Accelerators publish metrics to AWS CloudWatch, every 60 seconds
 - Only when requests are flowing through the accelerator
- **CloudTrail:**
 - It logs all API calls made to global accelerator APIs



Global Accelerator - Pricing

- Fixed Fee for every hour or partial hour a configured accelerator is run
- Data Transfer fee for data transferred over the AWS global accelerator network (in GB)
 - This is in addition to the normal data transfer out fees





AMAZON CLOUDFRONT

You Can Do It Too!



Amazon CloudFront

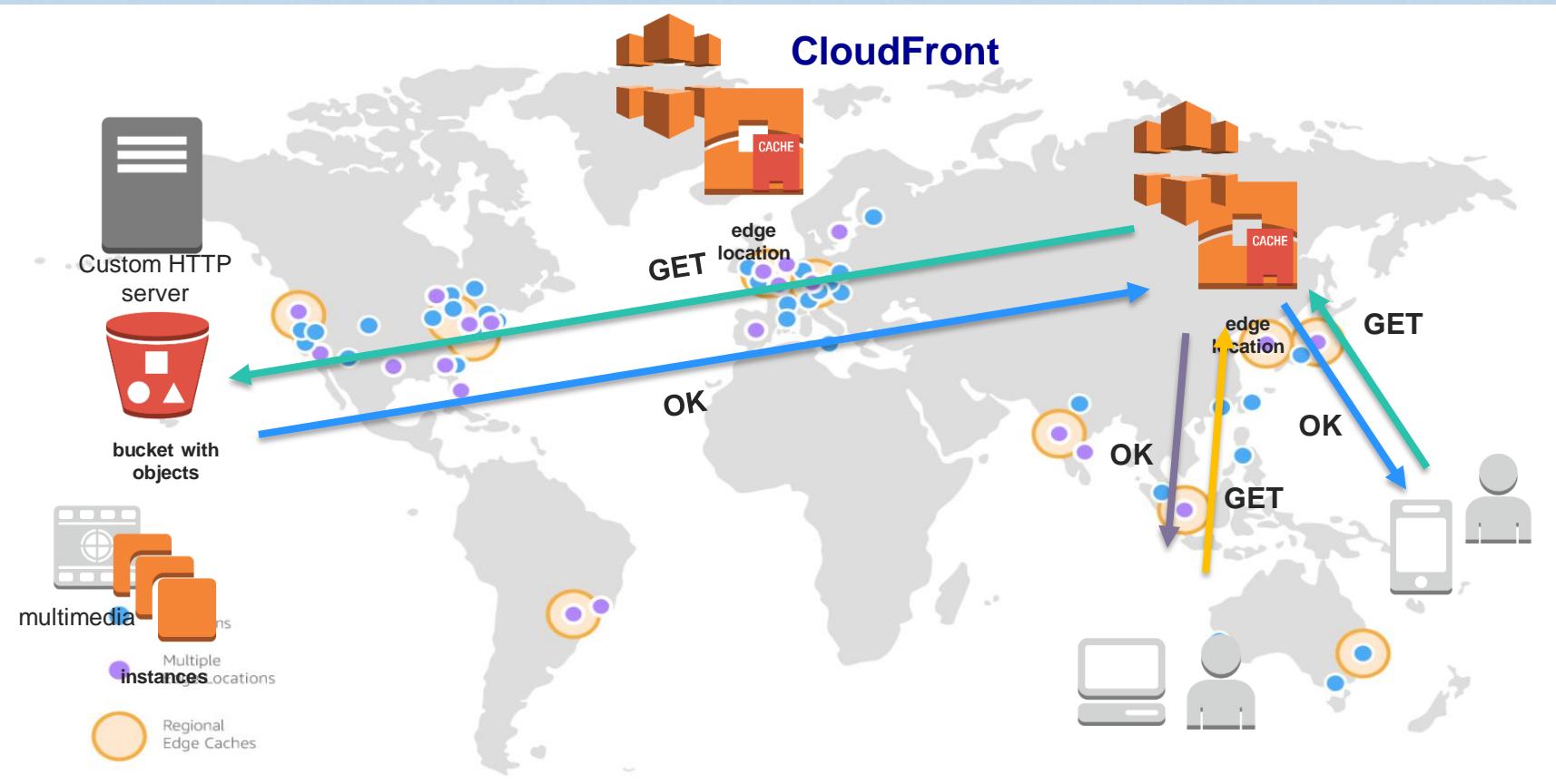
Introduction



AWS Cloud Front

DOLFINED ©

Not for copy, modification or Redistribution –
Please report any breach to info@dolfined.com



Review Topic : CloudFront

Cloudfront

- Cloudfront is a global (not regional) service.
 - It is used Ingress (injection proxy) to upload objects
 - and egress to distribute content
- Amazon Cloudfront is a web service that speeds up distribution of your static and dynamic web content, such as .html, .css, .js, and image files, to your users.
- Cloudfront delivers your content through a worldwide network of data centers called edge locations.
- When a user requests content that you're serving with Cloudfront, the user is routed (via DNS resolution) to the edge location that provides the lowest latency, so that content is delivered with the best possible performance.



Review Topic : CloudFront

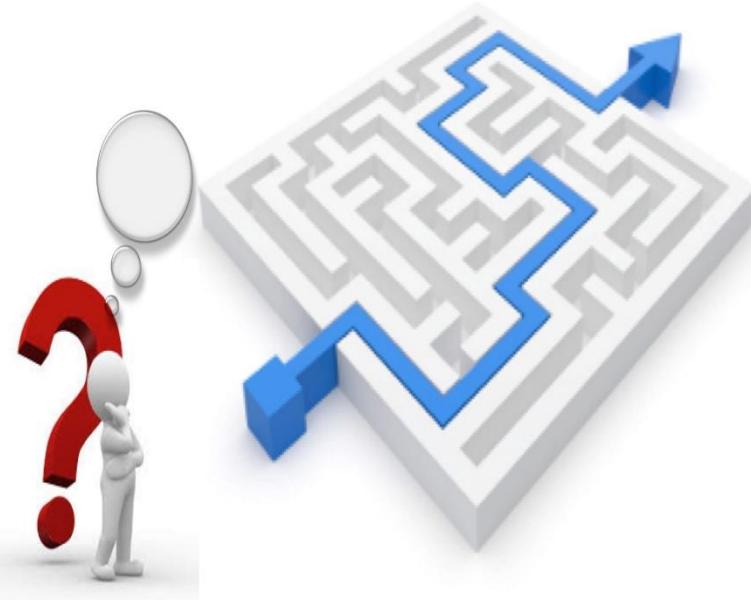
Cloudfront – Faster Response to Client requests

- If the content is already in the edge location with the lowest latency, CloudFront delivers it immediately.
 - This dramatically reduces the number of networks that your users' requests must pass through, which improves performance.
- If not, CloudFront retrieves it from an Amazon S3 bucket or an HTTP/web server that you have identified as the source for the definitive version of your content (Origin Server).
- CloudFront also keeps persistent connections with origin servers so files are fetched from the origins as quickly as possible.



Amazon CloudFront

Cache and Regional Edge Cache



Review Topic : CloudFront

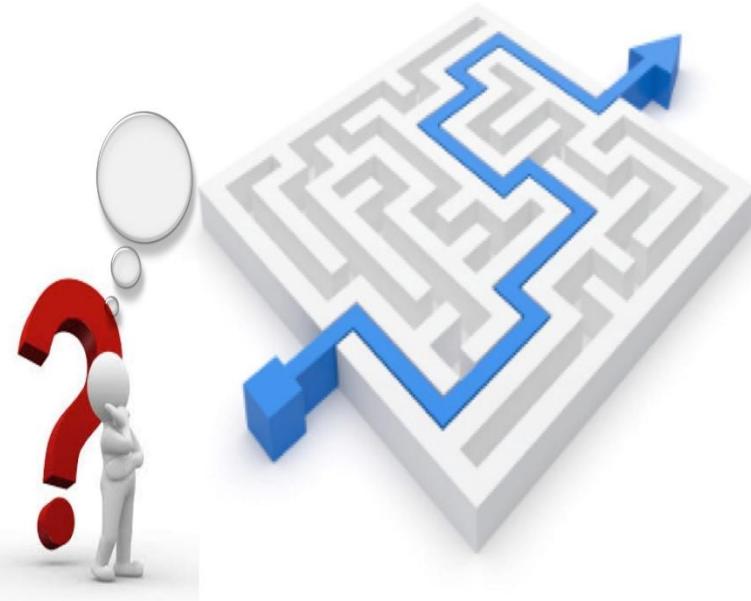
CloudFront- Regional Edge Cache

- Amazon CloudFront has added several **regional edge cache** locations globally, at close proximity to your viewers.
 - They are located between your origin webserver and the global edge locations that serve content directly to your viewers.
 - As objects become less popular, individual edge locations may remove those objects to make room for more popular content.
 - Regional Edge Caches have a larger cache width than any individual edge location, so objects remain in the cache longer at the nearest regional edge caches.
- Regional edge cache locations are currently used only for requests that need to go back to a custom origin; i.e. requests to S3 origins will skip regional edge cache locations.
- Serving less popular content from Regional Edge caches is enabled by default for all new and existing CloudFront distributions. There are no additional charges to use this feature.



Amazon CloudFront

Distributions



Review Topic : CloudFront

CloudFront – Configuration Settings

When you want to use CloudFront to distribute your content, you create a distribution and specify configuration settings such as:

- Your origin, which is the Amazon S3 bucket or HTTP server from which CloudFront gets the files that it distributes.
- You can specify any combination of up to 25 Amazon S3 buckets and/or HTTP servers as your origins.



Review Topic : CloudFront

CloudFront Distributions – Web (or Progressive Download) Distribution

- You can use **web distributions** to serve the following content over HTTP or HTTPS:
 - Static and dynamic download content, for example, .html, .css, .js, and image files, using HTTP or HTTPS.
 - Multimedia content on demand using progressive download and Apple HTTP Live Streaming (HLS).
- You can't serve Adobe Flash multimedia content over HTTP or HTTPS, but you can serve it using a CloudFront **RTMP distribution**.
- For web distributions, your origin can be either an Amazon S3 bucket or an HTTP server
- A live event, such as a meeting, conference, or concert, in real time.
 - For live streaming, you create the distribution automatically by using an AWS CloudFormation stack.



Review Topic : CloudFront

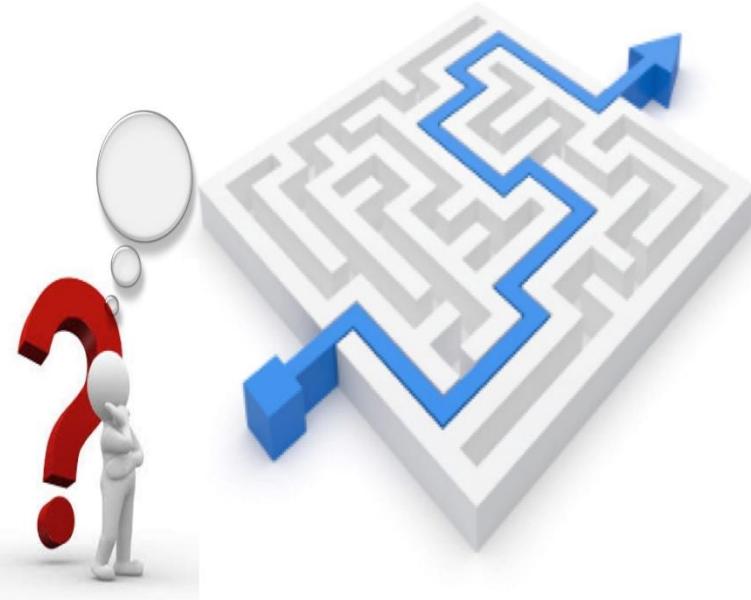
CloudFront Distribution – RTMP (or Streaming) Distribution

- RTMP distributions stream media files using Adobe Media Server and the Adobe Real-Time Messaging Protocol (RTMP).
- An RTMP distribution must use an Amazon S3 bucket as the origin.
- CloudFront lets you create a total of up to 200 web distributions and 100 RTMP distributions for an AWS account.



Amazon CloudFront

Origin and Custom Origin Servers



Review Topic : CloudFront

Origin Servers

- You specify *origin servers*, like an Amazon S3 bucket or your own HTTP server, from which CloudFront gets your files.
- An origin is the location where you store the original definitive version of your web content, which you want to distribute via Cloudfront.
 - If you're serving content over HTTP, your origin server is either an Amazon S3 bucket or an HTTP server, such as a web server.
 - Your HTTP server can run on an Amazon Elastic Compute Cloud (Amazon EC2) instance or on a server that you manage; these servers are also known as *custom origins*.
 - If you use the Adobe Media Server RTMP protocol to distribute media files on demand, your origin server is always an Amazon S3 bucket.
- Only S3 buckets are considered as Origin
 - S3 buckets configured as static website hosting are Custom Origins, same for EC2 instances and ELBs as origins.



Review Topic : CloudFront

Using EC2 Instance as a CloudFront Custom Origin Server

- A custom origin is an HTTP server, the HTTP server can be an Amazon EC2 instance, an ELB, or an HTTP server that you manage privately.
- When you use a custom origin that is your own HTTP server, you specify the DNS name of the server, along with the HTTP and HTTPS ports and the protocol that you want CloudFront to use when fetching objects from your origin.
- Most CloudFront features are supported when you use a custom origin with the following exceptions:
 - **RTMP distributions**—Not supported (the origin must be an S3 Bucket for Media files).
 - **Private content**—Although you can use a signed URL to distribute content from a custom origin, for CloudFront to access the custom origin, the origin must remain publicly accessible.



Review Topic : CloudFront

Using EC2 Webserver as a CloudFront Custom Origin Server

If you use Amazon Elastic Compute Cloud for your custom origins, AWS recommends the following:

- Use an Amazon Machine Image that automatically installs the software for a web server.
- Use an Elastic Load Balancing load balancer to handle traffic across multiple Amazon EC2 instances and to isolate your application from changes to Amazon EC2 instances.
- When you create your CloudFront distribution, specify the URL of the load balancer for the domain name of your origin server.



Review Topic : CloudFront

Using S3 bucket' static website as CloudFront Custom Origin Server

- You can set up an Amazon S3 bucket that is configured as a website endpoint **as custom origin with CloudFront.**
- When you specify the bucket name in this format as your origin, you can use Amazon S3 redirects and Amazon S3 custom error documents.



Amazon CloudFront

Cache Behavior



Review Topic : CloudFront

CloudFront Cache Behavior

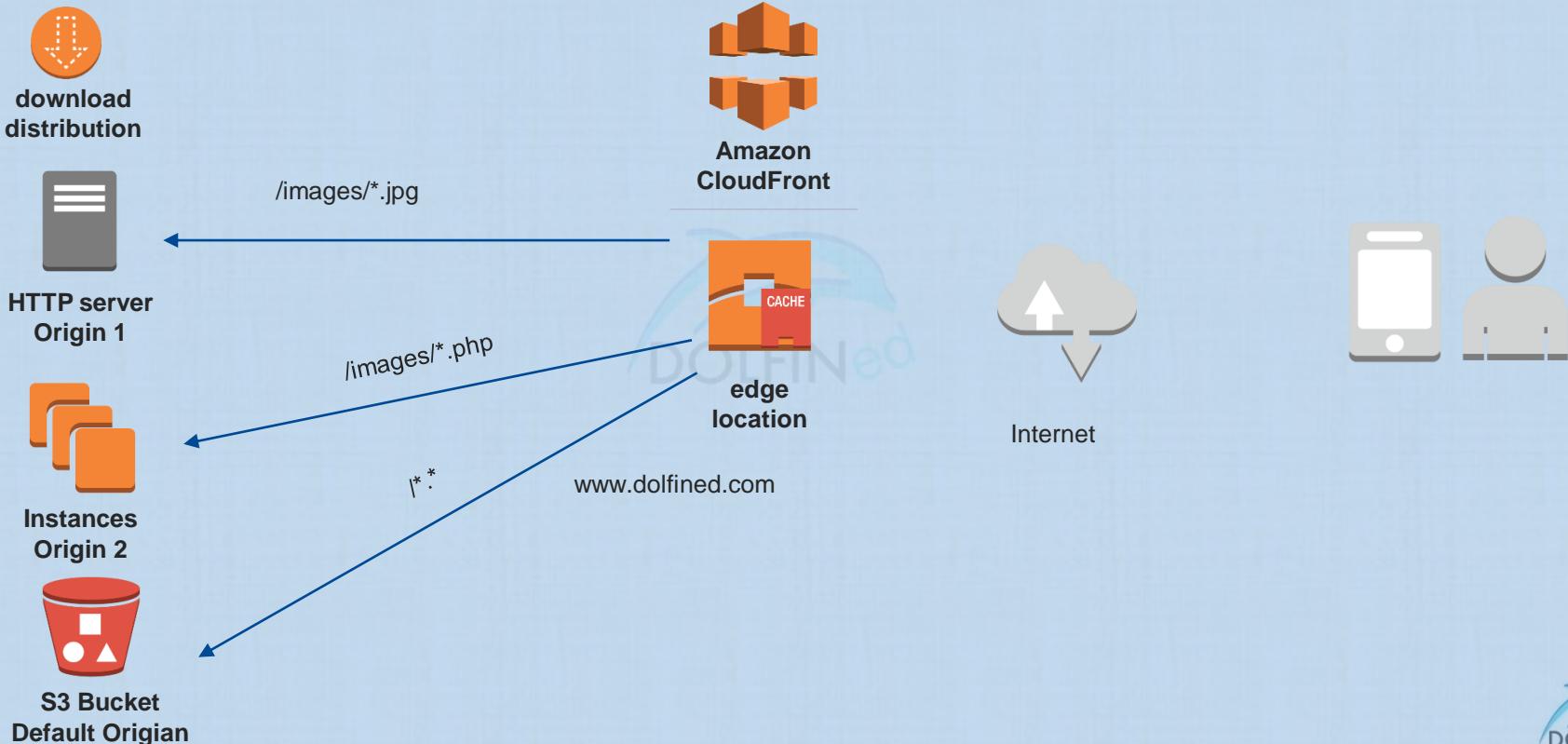
Cache Behavior

- A cache behavior lets you configure a variety of CloudFront functionality for a given URL path pattern for files on your website.
- For simplicity, the cache behaviors are seen as routing requests to the correct origins.
 - For instance, a cache behavior might apply to all .gif files in images directory on a web server that you're using as the origin server in CloudFront.
- List the cache behaviors in the order that you want CloudFront to evaluate them in, default will always be the last to be processed



AWS Cloud Front

CloudFront Cache Behavior – Path Pattern & Origin Selection



Review Topic : CloudFront

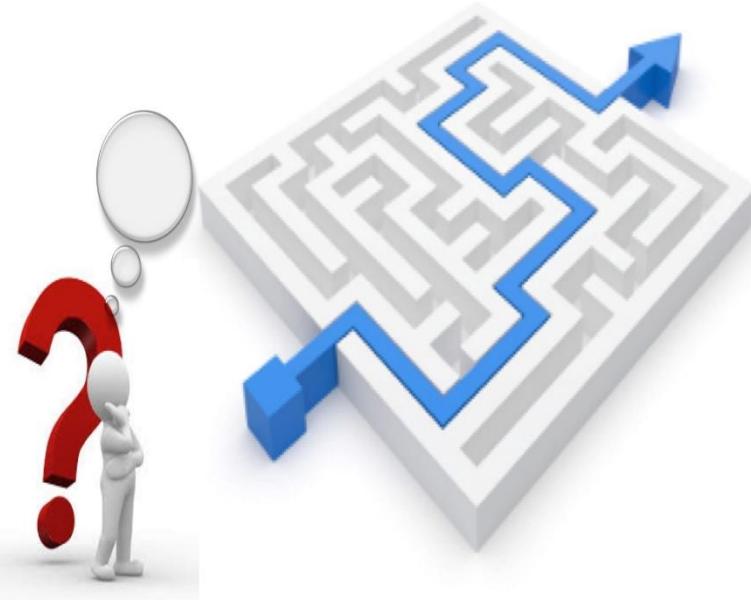
CloudFront – Path Pattern

Path Pattern

- A path pattern (for example, images/*.jpg) specifies which requests you want this cache behavior to apply to.
- When CloudFront receives an end-user request, the requested path is compared with path patterns in the order in which cache behaviors are listed in the distribution.
 - The first match determines which cache behavior is applied to that request.
 - Example, suppose you have three cache behaviors with the following three path patterns, in this order: images/*.jpg , images/* , *.gif
 - A request for the file images/sample.gif doesn't satisfy the first path pattern, so the associated cache behaviors are not be applied to the request.
 - The file does satisfy the 2nd path pattern, so the cache behaviors associated with the second path pattern are applied even though the request also matches the 3rd path pattern.

Amazon CloudFront

- Time To Live (TTL)
- Allowed HTTP Methods



Review Topic : CloudFront

CloudFront – Other configuration settings

TTL

- The amount of time, in seconds, that you want objects to stay in CloudFront caches before CloudFront forwards another request to your origin to determine whether the object has been updated.
 - The default value for **Minimum TTL** is 0 seconds, it means never cache any object.
 - The default value for **Maximum TTL** is 31536000 seconds (one year).
 - The default value for **Default TTL** is 86400 seconds (one day).

Review Topic : CloudFront

CloudFront Cache Behavior – HTTP Methods

Allowed HTTP Methods

Specify the HTTP methods that you want CloudFront to process and forward to your origin:

- **GET, HEAD (Cached):** You can use CloudFront only to get objects from your origin or to get object headers.
 - Responses to both methods are cached by default
- **GET, HEAD, OPTIONS:**
 - You can use CloudFront only to get objects from your origin, get object headers, or retrieve a list of the options that your origin server supports.
 - Responses to OPTIONS can be optionally cached
- **GET, HEAD, OPTIONS, PUT, POST, PATCH, DELETE:**
 - You can use CloudFront to get, add, update, and delete objects, and to get object headers. In addition, you can perform other POST operations such as submitting data from a web form.
 - Responses to PUT, POST, PATCH, and DELETE can not be cached



Amazon CloudFront

Servings Private Content



AWS Cloud Front

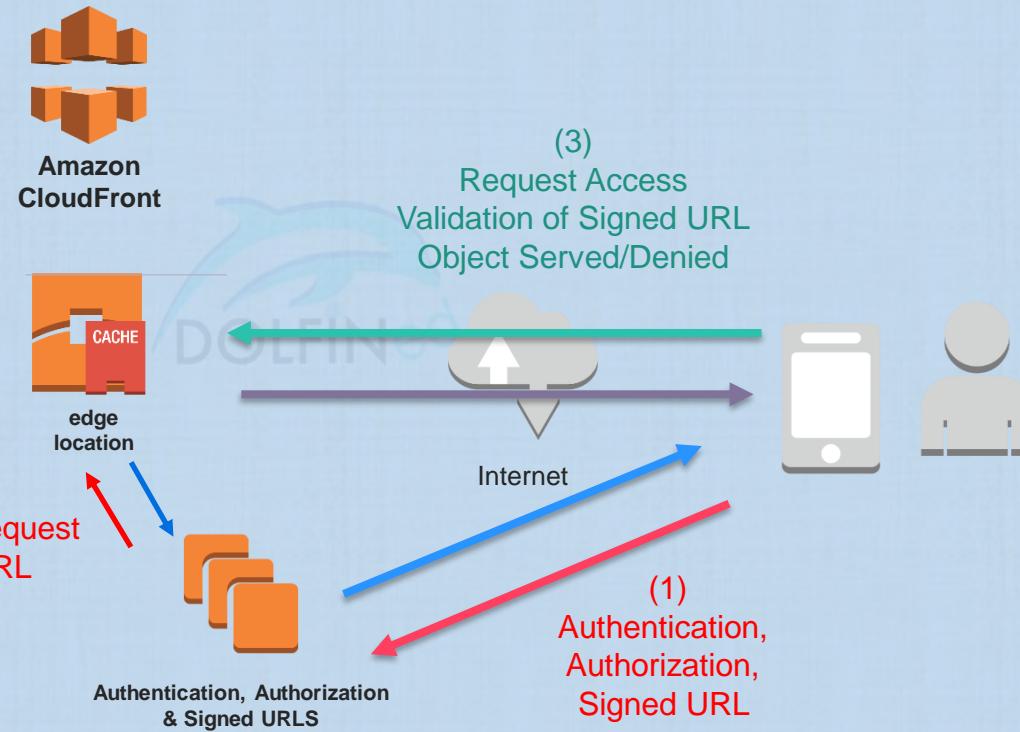
Serving Private Content



download
distribution



Origin
Server



Review Topic : CloudFront

Service Private Content via CloudFront

- Companies that distribute content via the internet may need to restrict access to private documents, sensitive business data, internal/subscribed-to media streams, or specific content that is intended for selected users/subscribers,
- An example, users who have paid a fee to subscribe to come content, to securely serve this private content using CloudFront, you can do the following:
 - Require that these users access the private content by using special CloudFront signed URLs or signed cookies.
 - Require that these users access the Amazon S3 content using CloudFront URLs, not Amazon S3 URLs.
 - Restricting access to CloudFront URLs (and S3 URLs) isn't required, but we recommend it to prevent users from bypassing the restrictions that you specify in signed URLs or signed cookies.

Review Topic : CloudFront

CloudFront – Controlling Access to your content

- You can control user access to your private content in two ways, as shown in the following illustration:
 - Restrict access to objects in CloudFront edge caches
 - Restrict access to objects in your Amazon S3 bucket



Review Topic : CloudFront

Restricting Access to Objects in CloudFront Edge Caches

- You can configure CloudFront to require that users access your objects using either **signed URLs** or **signed cookies**.
- Then develop your application either to create and distribute signed URLs to authenticated users OR to send Set-Cookie headers that set signed cookies on the viewers for authenticated users.
 - To give a few users long-term access to a limited number of objects, you can also create signed URLs manually.
- When you create signed URLs or signed cookies to control access to your objects, you can specify the following restrictions:
 - An ending date and time, after which the URL is no longer valid.
 - (Optional) The date and time that the URL becomes valid.
 - (Optional) The IP address or range of addresses of the computers that can be used to access your content.

Review Topic : CloudFront

CloudFront – Trusted Signers

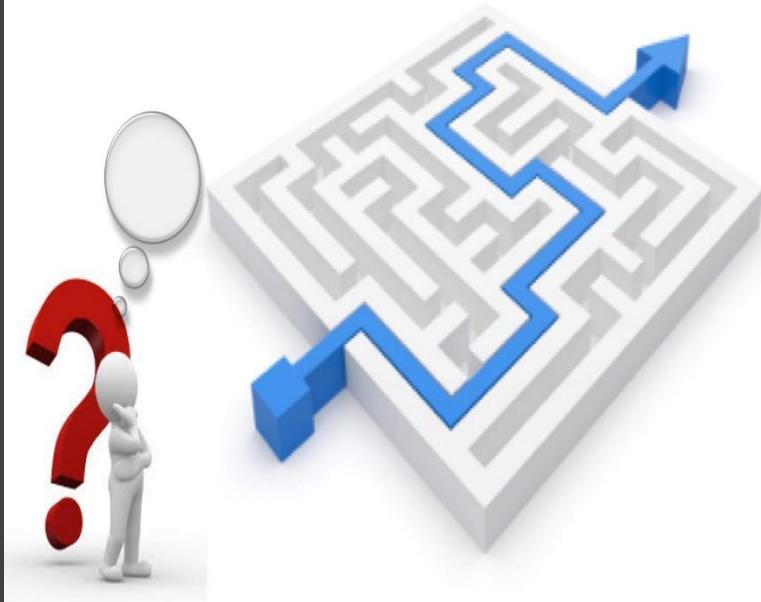
- To create signed URLs or signed cookies, you need at least one AWS account that has an active CloudFront key pair.
- Signed URLs have an expiry attached to them, great way to provide temporary access.
- **Web distributions (both signed URLs and signed Cookies) –**
 - Users can use signed URLs or signed Cookies
- **RTMP distributions (signed URLs only) –** You add trusted signers to a distribution.
 - After you add trusted signers to an RTMP distribution, users must use signed URLs to access any object associated with the distribution.



Amazon CloudFront

Restricting Access to your Origin Servers

Not for copy, modification or Redistribution –
Please report any breach to info@dolfined.com



Review Topic : CloudFront

Restricting Access to Objects in Amazon S3 Buckets

- Origin Access Identity (OAI), which is a special CloudFront user, and associate the origin access identity with your distribution.
- You can optionally secure the content in your Amazon S3 bucket so users can access it through CloudFront but cannot access it directly by using Amazon S3 URLs.
- To require that users access your content through CloudFront URLs, you perform the following tasks:
 - Create a special CloudFront user called an **origin access identity**.
 - Give the origin access identity S3 bucket policy permission to read the objects in your bucket.
 - Remove permission for anyone else to use Amazon S3 URLs to read the objects.
- This prevents anyone from bypassing CloudFront and using the Amazon S3 URL to get content that you want to restrict access to.
 - This step isn't required to use signed URLs, but AWS recommends it.



Review Topic : CloudFront

CloudFront – Origin Access Identity

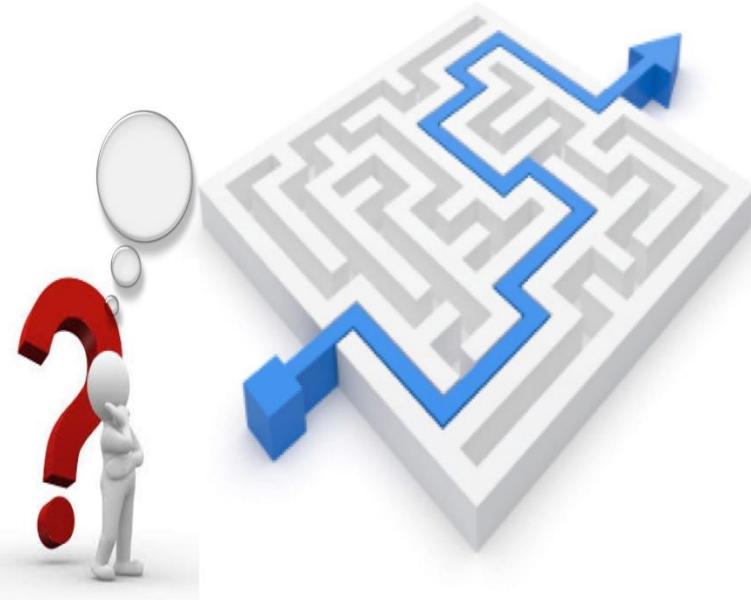
To ensure that users can access objects using only CloudFront URLs, regardless of whether the URLs are signed, perform the following tasks:

- Create an origin access identity, which is a special CloudFront user, and associate the origin access identity with your distribution.
 - For web distributions, associate the origin access identity with origins, to secure all or just some of your Amazon S3 content.
 - Also, it is possible to create an origin access identity and add it to your distribution when you create the distribution.
- Change the permissions either on your Amazon S3 bucket or on the objects in your bucket so only the origin access identity has read permission (or read and download permission).
- When users access the Amazon S3 objects through CloudFront, the CloudFront origin access identity gets the objects on behalf of your users.



Amazon CloudFront

Alternate Domain Names



Review Topic : CloudFront

CloudFront Domain Name vs CNAMEs

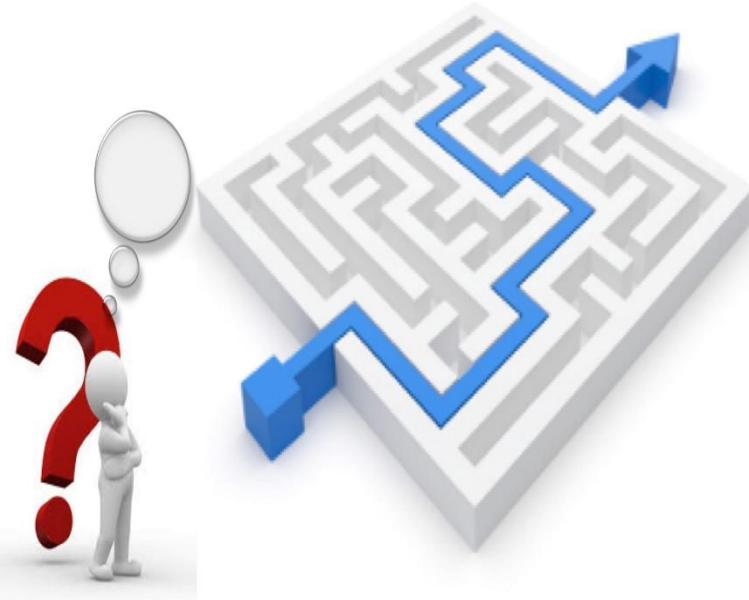
Adding and Moving Alternate Domain Names (CNAMEs)

- When you create a distribution, CloudFront returns a domain name for the distribution,
 - as an example: d112211abcdef8.cloudfront.net
- To use a different domain name, such as www.dolfined.com, instead of the cloudfront.net domain name that CloudFront had assigned to your distribution,
 - You can add an alternate domain name to your distribution for www.dolfined.com. You can then use the following URL for /images/tree1.jpg:
 - <http://www.dolfined.com/images/tree1.jpg>
 - Both web and RTMP distributions support alternate domain names.



Amazon CloudFront

Viewer and Origin Protocol Policies



Review Topic : CloudFront

CloudFront Cache Behavior – Viewer Protocol Policy

Viewer Protocol Policy

- Choose the protocol policy that you want viewers to use to access your content in CloudFront edge locations:
 - **HTTP and HTTPS:** Viewers can use both protocols.
 - **Redirect HTTP to HTTPS:** Viewers can use both protocols, but HTTP requests are automatically redirected to HTTPS requests.
 - **HTTPS Only:** Viewers can only access your content if they're using HTTPS.
- For web distributions, you can configure CloudFront to require that viewers use HTTPS to request your objects, so connections are encrypted when CloudFront communicates with viewers.



Review Topic : CloudFront

CloudFront – HTTP to HTTPS Redirects

- Viewers can use both protocols.
 - HTTP GET and HEAD requests are automatically redirected to HTTPS requests.
 - CloudFront returns HTTP status code 301 (Moved Permanently) along with the new HTTPS URL.
 - The viewer then resubmits the request to CloudFront using the HTTPS URL.



AWS Cloud Front

CloudFront – Origin Protocol Policy for Custom Origins

- The protocol policy that you want CloudFront to use when fetching objects from your origin server.
- You can choose one of the following values:
 - **HTTP Only:**
 - CloudFront uses only HTTP to access the origin.
 - **HTTPS Only:**
 - CloudFront uses only HTTPS to access the origin.
 - **Match Viewer:**
 - CloudFront communicates with your origin using HTTP or HTTPS, depending on the protocol of the viewer request.

Note:

- If your Amazon S3 bucket is configured as a website endpoint, you must specify **HTTP Only**
 - Amazon S3 doesn't support HTTPS connections in that configuration.



Amazon CloudFront

Working with Cached Objects – Invalidations



Review Topic : CloudFront

Invalidating Objects (Web Distributions Only)

If you need to remove an object from CloudFront edge caches before it expires, you can do one of the following:

- Invalidate the object from edge caches. The next time a viewer requests the object, CloudFront returns to the origin to fetch the latest version of the object.
- Use object versioning to serve a different version of the object that has a different name.
- You can't cancel an invalidation after you submit it.

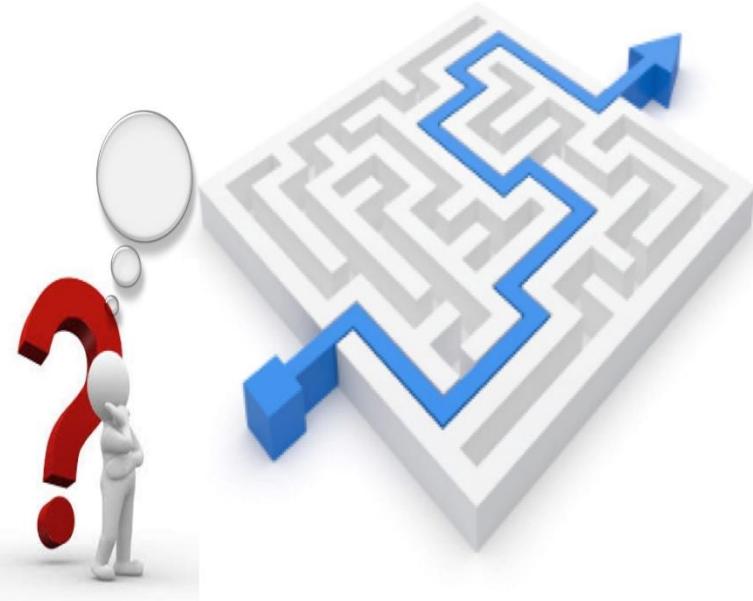
Important

- You can invalidate most types of objects that are served by a web distribution, However,
 - You cannot invalidate media files in the Microsoft Smooth Streaming format when you have enabled Smooth Streaming for the corresponding cache behavior.
 - In addition, you cannot invalidate objects that are served by an RTMP distribution.



Amazon CloudFront

Using Web Application Firewalls (WAF)



Review Topic : CloudFront

Using AWS WAF to Control Access to Your Content

- AWS WAF is a web application firewall that lets you monitor the HTTP and HTTPS requests that are forwarded to CloudFront, and lets you control access to your content.
- Based on conditions that you specify, such as the IP addresses that requests originate from or the values of query strings, CloudFront responds to requests either with the requested content or with an HTTP 403 status code (Forbidden).
- You can also configure CloudFront to return a custom error page when a request is blocked.
- After you create an AWS WAF web access control list (web ACL), you create or update a web distribution and associate the distribution with a web ACL. You can associate as many CloudFront distributions as you want with the same web ACL or with different web ACLs.

Amazon CloudFront

Geo Restrictions



Review Topic : CloudFront

Restricting the Geographic Distribution of Your Content

- You can use *geo restriction*, also known as *geoblocking*, to prevent users in specific geographic locations from accessing content that you're distributing through a CloudFront web distribution.
- To use geo restriction, you have two options:
 - Use the CloudFront geo restriction feature.
 - Use this option to restrict access to all of the files that are associated with a distribution and to restrict access at the country level.
 - Use a third-party geolocation service.
 - Use this option to restrict access to a subset of the files that are associated with a distribution or to restrict access at a finer granularity than the country level.



Review Topic : CloudFront

Using CloudFront Geo Restriction

- When a user requests your content, CloudFront typically serves the requested content regardless of where the user is located.
- If you need to prevent users in specific countries from accessing your content, you can use the CloudFront geo restriction feature to do one of the following:
 - Allow your users to access your content only if they're in one of the countries on a **whitelist** of approved countries.
 - Prevent your users from accessing your content if they're in one of the countries on a **blacklist** of banned countries.

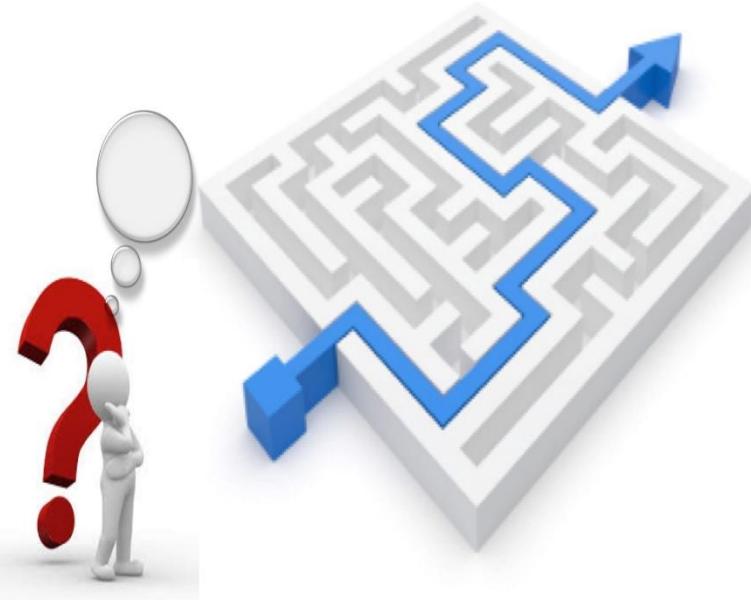
Note

- CloudFront determines the location of your users by using a third-party GeoIP database.
- The accuracy of the mapping between IP addresses and countries varies by region is 99.8%
- Be aware that if CloudFront can't determine a user's location, CloudFront will serve the content that the user has requested.



Amazon CloudFront

Video Streaming with CloudFront Web Distributions



Review Topic : CloudFront

Configuring On-Demand Streaming Web Distributions

- You can use CloudFront web distributions to serve **on-demand streaming media files** from any **HTTP origin (i.e Web Distribution)**. Below are several examples of working with different origins to serve streaming video content.
- Configuring On-demand with AWS Elemental Media-Store
- Configuring On-Demand Smooth Streaming
- Configuring On-Demand Progressive Downloads
- Configuring On-Demand Apple HTTP Live Streaming (HLS)
- The key message is, you do not have to use RTMP distributions to serve streaming video using Cloudfront, you can do so using Web distributions.
- **You can't** serve Adobe Flash multimedia content over HTTP or HTTPS, but you can serve it using a CloudFront RTMP distribution.



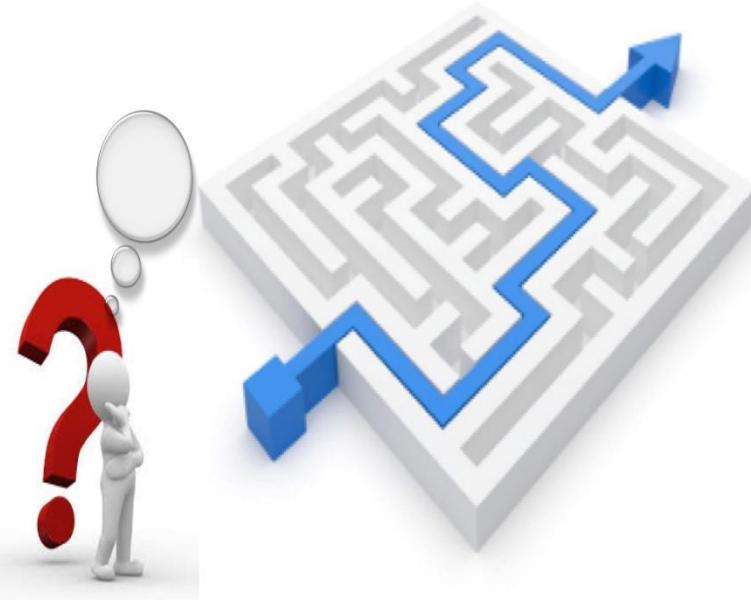
Review Topic : CloudFront

Serving Media Content by Using HTTP

- When you use HTTP to serve media content, AWS recommends that you use an HTTP-based (i.e web distribution not RTMP) dynamic streaming protocol such as
 - Apple HTTP Dynamic Streaming (Apple HDS),
 - Apple HTTP Live Streaming (Apple HLS), <<- Widely supported
 - Microsoft Smooth Streaming,
 - or MPEG-DASH.
- For dynamic-streaming protocols, a video is divided into a lot of small segments that are typically just a few seconds long each.
 - If the users commonly stop watching before the end of a video (for example, because they close their viewer during the credits),
 - CloudFront has still cached all of the small segments up to that point in the video.

Amazon CloudFront

Video Streaming with CloudFront RTMP Distributions



Review Topic : CloudFront

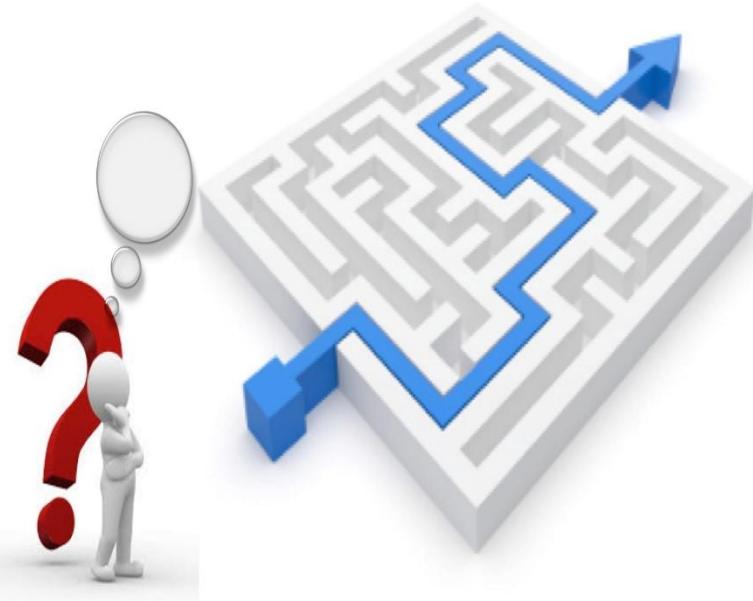
How RTMP Distributions Work

- To use CloudFront to serve both the media player and the media files, you need two types of distributions:
 - A web distribution for the media player, and
 - An RTMP distribution for the media files.
- **Web distributions serve files over HTTP, while RTMP distributions stream media files over RTMP (or a variant of RTMP).**
- Media files MUST be stored in an AWS S3 bucket, custom origins are not supported
- The media player can be in the same S3 bucket, a different S3 bucket, or in a custom HTTP origin server while served using Cloudfront
- For CloudFront to distribute media files, CloudFront uses **Adobe Flash Media Server** as the streaming server and streams media files **using Adobe's Real-Time Messaging Protocol (RTMP)**.



Amazon CloudFront

Access Logs and Reports



Review Topic : CloudFront

CloudFront - Access Logs

- You can configure CloudFront to create log files that contain detailed information about every user request that CloudFront receives.
- These access logs are available for both web and RTMP distributions.
- If you enable logging, you can also specify the Amazon S3 bucket that you want CloudFront to save files in.
- You can enable logging as an option that you specify when you're creating a distribution.
- One way to analyze your access logs is to use Amazon Athena. Athena is an interactive query service that can help you analyze data for AWS services, including CloudFront.
- AWS recommends that you use the logs to understand the nature of the requests for your content, not as a complete accounting of all requests. CloudFront delivers access logs on a best-effort basis.



Review Topic : CloudFront

Using AWS CloudTrail to Capture Requests Sent to the CloudFront API

- CloudFront is integrated with CloudTrail, an AWS service that captures information about every request that is sent to the CloudFront API by your AWS account, including your IAM users.
- CloudTrail periodically saves log files of these requests to an Amazon S3 bucket that you specify.

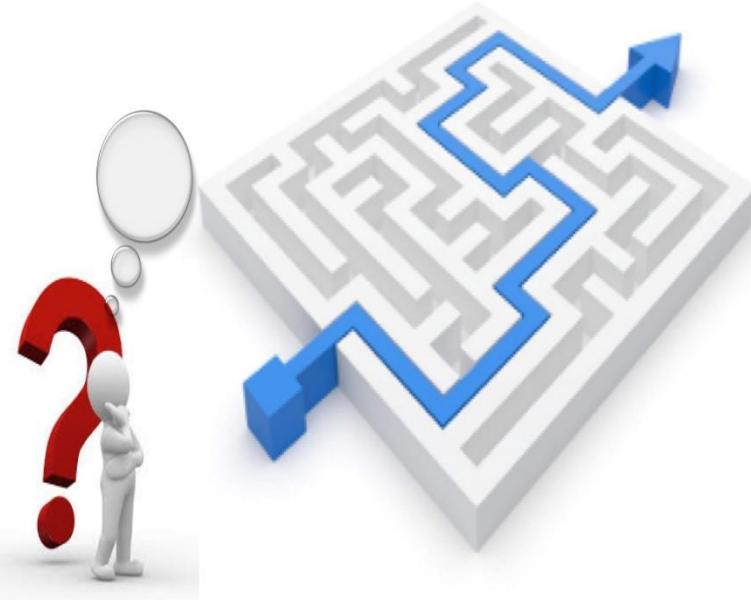
Note

- To view CloudFront requests in CloudTrail logs, you must update an existing trail to include global services.



Amazon CloudFront

Perfect Forward Secrecy
Used by ELB and CloudFront

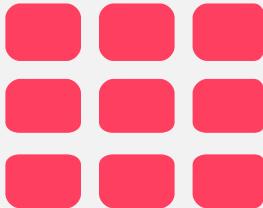


Review Topic : CloudFront

CloudFront Perfect Forward Secrecy

- Several AWS services offer more advanced cipher suites that use the Elliptic Curve Diffie-Hellman Ephemeral (ECDHE) protocol.
- ECDHE allows SSL/TLS clients to provide Perfect Forward Secrecy, which uses session keys that are ephemeral and not stored anywhere.
 - This helps prevent the decoding of captured data by unauthorized third parties, even if the secret long-term key itself is compromised.
- Clients using CloudFront APIs must support Transport Layer Security (TLS) 1.0 or later.
- Clients must also support cipher suites with **perfect forward secrecy (PFS)** such as Ephemeral Diffie-Hellman (DHE) or Elliptic Curve Ephemeral Diffie-Hellman (ECDHE).
- Most modern systems such as Java 7 and later support these modes.





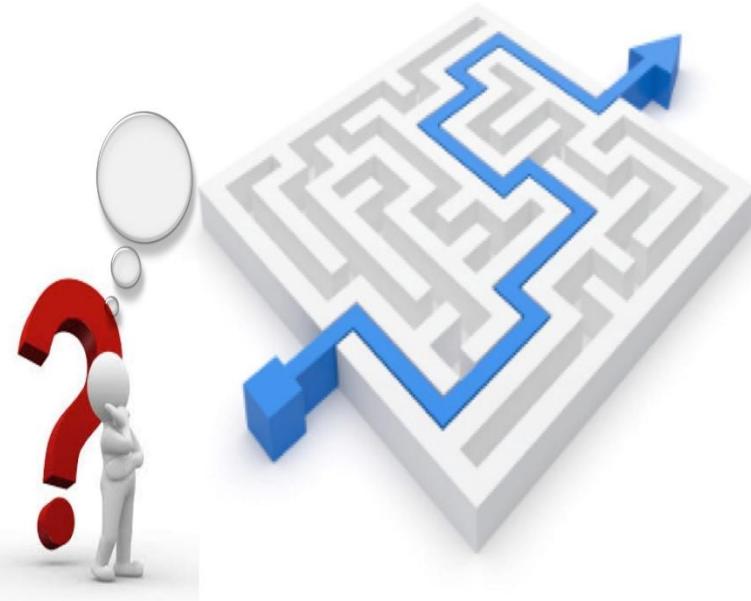
AWS SERVICES SIMPLE QUEUE SERVICE (SQS)

You Can Do It Too!



Amazon SQS

Introduction



Review Topic : Amazon SQS

AWS SQS

- SQS is a fast, reliable, durable, secure, and fully managed hosted message queue service
- Is a web service that gives you access to message queues that store messages waiting to be processed
- It offers a reliable, highly scalable, hosted queue for storing messages between distributed software systems and components.
- It allows the decoupling of application components such that a failure in one components does not cause a bigger problem to application functionality (like in coupled applications)
- Using SQS, you no longer need a highly available message cluster or the burden of building/running it
- SQS is a pay as you go service

source: aws.amazon.com/sqs/

Review Topic : Amazon SQS

AWS SQS – Queue Types

Standard Queue

- High (unlimited) throughput
- At least once delivery
- Duplicates are possible (can't guarantee no duplication)
- Best effort ordering

FIFO Queue (Not available in all regions)

- Limited throughput – 300 Transactions/sec/API (without batching) 3000 TPS/Sec/API with batching
- Exactly-once processing
- No duplicates guarantee
- Strict ordering - First-in-First-out

source: [aws.amazon.co](https://aws.amazon.com)

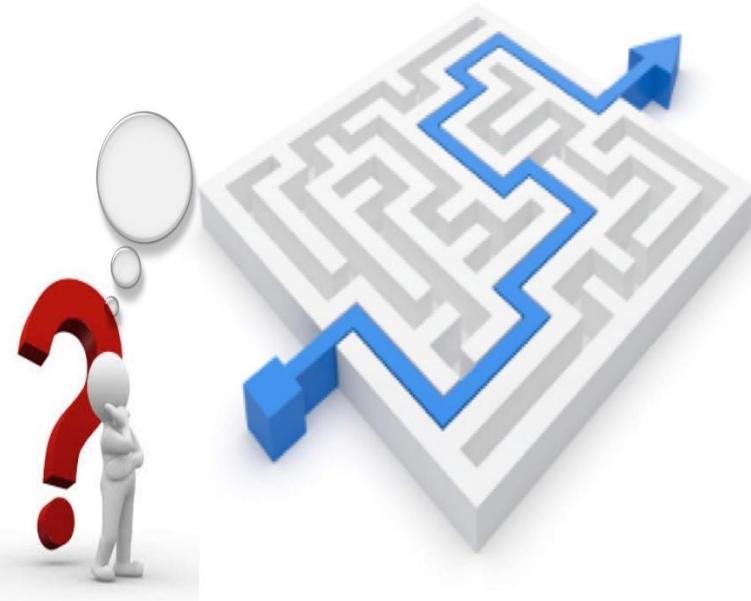
Review Topic : Amazon SQS

AWS SQS – Pricing

- Priced per Million requests
- A request is any SQS action
 - It can have 1-10 messages , up to a maximum request payload size of 256KB
 - SQS messages can be sent, received, deleted in batches up to 10 messages or 256 KB
 - Each 64 KB is a chunk, a chunk is one request
 - A SQS message size can be 1KB up to 256KB (can't be video or other long messages)
- If you use S3 to store queue messages, S3 charges will apply
- Data transferred between SQS and EC2 in the same region is free
 - While that transferred between EC2 and SQS in different regions is charged data transfer rates on both sides of the transfer

Amazon SQS

SQS Polling types and Timers



Review Topic : Amazon SQS

AWS SQS – Short/Regular Polling

- SQS is a polling based service (while SNS service is a push based service)
- Short Polling:
 - A request is returned immediately even if the queue is empty
 - It does not wait for messages to appear in the queue
 - It queries only a subset of the available servers for messages (based on weighted random distribution)
 - Default of SQS
 - ReceiveMessageWaitTime is set to 0
- More requests are used, which implies higher cost and empty reads

Review Topic : Amazon SQS

AWS SQS – Long Polling

- Is preferred to regular/short polling, it uses fewer requests and reduces cost by:
 - Eliminating false empty responses by querying all the servers
 - Reduces the number of empty responses, by allowing Amazon SQS to wait until a message is available in the queue before sending a response. Unless the connection times out, the response to the ReceiveMessage
- Request contains at least one of the available messages, up to the maximum number of messages specified in the ReceiveMessage action.
- Do not use if your application expects an immediate response to receive message calls
- ReceiveMessageWaitTime is set to a non zero value (up to 20 seconds)
- Same charge per million requests as the regular/short polling

Review Topic : Amazon SQS

SQS – Retention Period

- SQS messages can remain in the queue for up to 14 days (SQS retention period)
 - Range is 1 min to 14 days (default is 4 days)
 - Once the maximum retention period of a message is reached, it will be deleted automatically from the queue
- Messages can be sent to the queue and read from the queue simultaneously
- SQS can be used with :
 - Redshift, DynamoDB, EC2, ECS, RDS, S3, Lambda to make distributed/decoupled applications
- You can have multiple SQS queues with different priorities in case you want one SQS queue messages to be handled with higher priority over other SQS queue messages
- You can scale up the send/receive messages by creating more queues for different processes/actions



Review Topic : Amazon SQS

SQS Visibility Timeout

- Is the duration of time (length) a message is locked for read by other consumers (after it has been read by a consumer to process it) so they can not be read again (by another consumer)
 - Max is 12 hours, default is 30 seconds , range 0 to 12 hours
 - Consumer is an application processing the SQS queue messages
- A consumer that reads a message to process it, can change the message visibility timeout if it needs more time to process the message
- After a message is read, there are the following possibilities:
 - An ACK is received from the consumer that a message is processed, so it must be deleted from the queue to avoid duplicates
 - If a FAIL is received or the visibility timeout expires, the message will then be unlocked for read , such that it can be read and processed by another consumer

source: aws.amazon.com



Review Topic : Amazon SQS

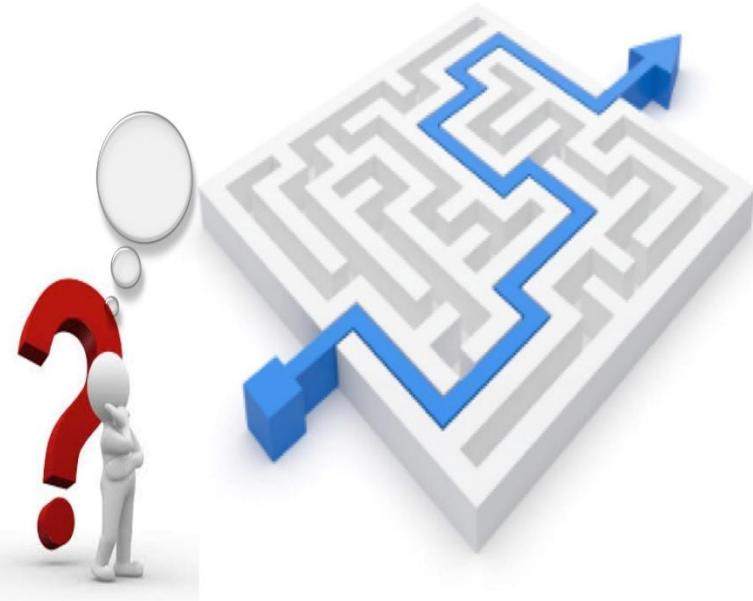
AWS SQS Delay Queue – Message Delay

- Delay queues allows for delaying the delivery of new messages to a queue for a number of seconds.
- In a delay queue, any messages sent to the queue remain invisible to consumers for the duration of the delay period.
- Default delay is 0 seconds
- If you want to space the messages in the queue in time,
 - You can configure individual message delay of up to 15 minutes
 - This helps when need to schedule jobs with a delay
- For standard queues, the per-queue delay setting is not retroactive (does not affect messages already in the queue)
- For FIFO queues, the per-queue delay setting is retroactive



Amazon SQS

Reliability, Security, and Encryption



Review Topic : Amazon SQS

AWS SQS - Reliability

- Amazon SQS stores all message queues and messages within a single, highly-available AWS region with multiple redundant Availability Zones (AZs),
 - No single computer, network, or AZ failure can make messages inaccessible.



Review Topic : Amazon SQS

AWS SQS - Security

- You can use IAM policies to control who can read/write messages from/to an SQS queue
- Authentication mechanisms ensure that messages stored in Amazon SQS message queues are secured against unauthorized access.
 - You can control who can send messages to a message queue and who can receive messages from a message queue.
 - For additional security, you can build your application to encrypt messages before they are placed in a message queue.
- SQS supports HTTPS & supports TLS versions, 1.0, 1.1, 1.2 in all regions
- SQS is PCI DSS (Payment Card Industry Data Security Standard) level 1 compliant
- SQS is HIPAA (US Health Insurance Portability and Accountability Act) Eligible

Review Topic : Amazon SQS

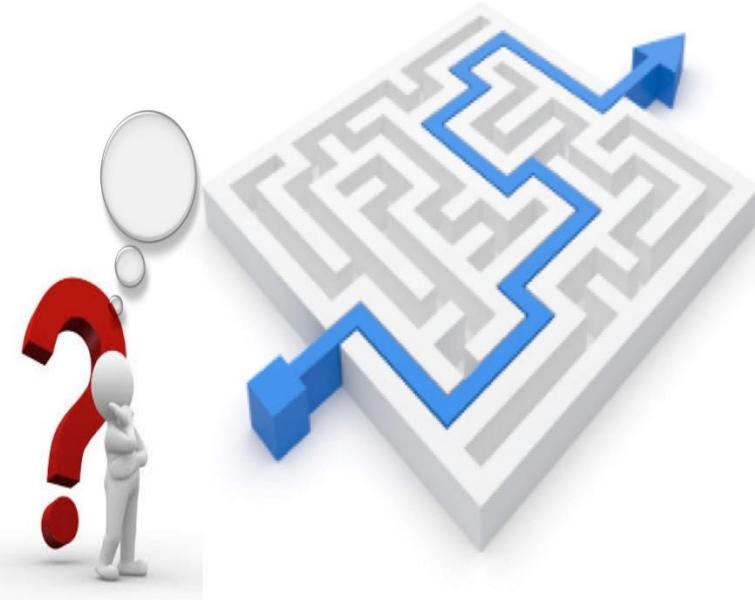
AWS SQS – SSE Encryption

- Server-side encryption (SSE) lets you transmit sensitive data in encrypted queues.
 - SSE protects the contents of messages in Amazon SQS queues using KMS managed keys
 - SSE encrypts messages as soon as Amazon SQS receives them.
 - The messages are stored in encrypted form and Amazon SQS decrypts messages only when they are sent to an authorized consumer
 - It uses AES-256 bits encryption
- AWS KMS combines secure, highly available hardware and software to provide a key management system scaled for the cloud.
- Both standard and FIFO queues support SSE.



Amazon SQS

SQS Service quotas & Monitoring



Review Topic : Amazon SQS

AWS SQS – Limits

- Unlimited number of messages can be received by an SQS Queue
- Maximum message size is 256 KB
- To send messages larger than 256KB, you can use Amazon SQS Extended Client Library for Java where a reference to the message is sent to the queue, while the message payload is stored in S3
 - Max message payload is 2GB



Review Topic : Amazon SQS

AWS SQS - Monitoring

- AWS SQS and CloudWatch are integrated and you can view and monitor SQS queues' metrics using CloudWatch
 - This is supported for both Standard and FIFO based SQS Queues
- CloudWatch metrics for your Amazon SQS queues are automatically collected and pushed to CloudWatch every five minutes.
 - These metrics are gathered on all queues that meet the CloudWatch guidelines for being active
 - CloudWatch considers a queue to be active for up to six hours if it contains any messages or if any API action accesses it.
- There is no charge for the Amazon SQS metrics reported in CloudWatch. They're provided as part of the Amazon SQS service.
- Detailed monitoring (or one-minute metrics) is currently unavailable for Amazon SQS.

Review Topic : Amazon SQS

AWS SQS – Cloud Trail logging

- Amazon SQS is integrated with CloudTrail, a service that captures API calls made by or on behalf of Amazon SQS in your AWS account and delivers the log files to the specified Amazon S3 bucket.
- CloudTrail captures API calls made from the Amazon SQS console or from the Amazon SQS API.
- You can use the information collected by CloudTrail to determine which requests are made to Amazon SQS, the source IP address from which the request is made, who made the request, when it is made



source: [aws.amazon.co](https://aws.amazon.com)

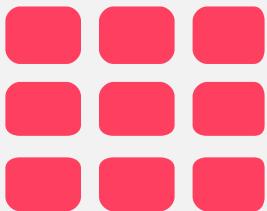
Review Topic : Amazon SQS

SQS Message Groups and Message De-Duplication

- Amazon SQS is integrated with CloudTrail, a service that captures API calls made by or on behalf of Amazon SQS in your AWS account and delivers the log files to the specified Amazon S3 bucket.
- CloudTrail captures API calls made from the Amazon SQS console or from the Amazon SQS API.
- You can use the information collected by CloudTrail to determine which requests are made to Amazon SQS, the source IP address from which the request is made, who made the request, when it is made



source: [aws.amazon.co](https://aws.amazon.com)



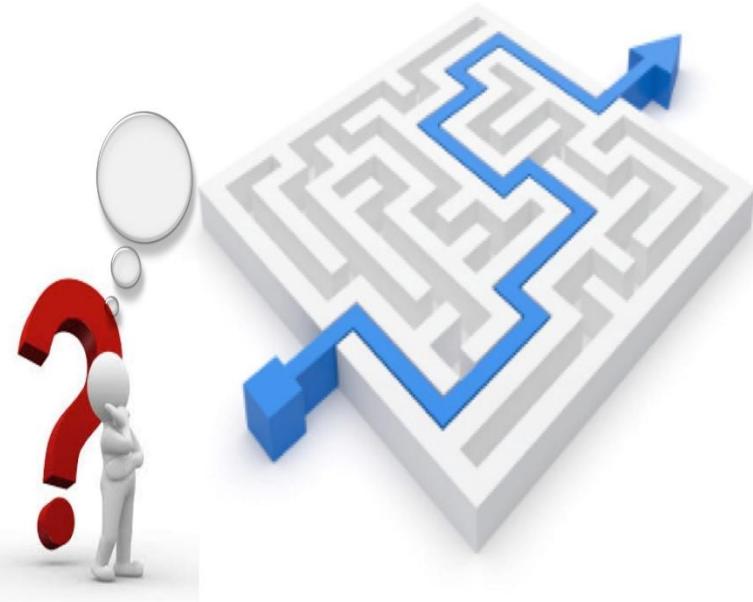
SERVERLESS AWS LAMBDA

You Can Do It Too!



AWS Lambda

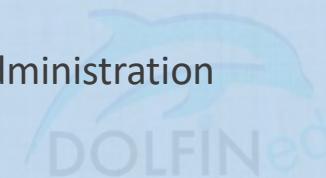
Introduction



AWS Services

AWS Lambda

- AWS Lambda is a compute service that lets you run code without provisioning or managing servers.
- With AWS Lambda, you can run code for virtually any type of application or backend service - **all with zero administration**.
- AWS Lambda manages all the administration



AWS Services

AWS Lambda – How it works

- AWS Lambda runs your code **on a high-availability compute infrastructure**
- AWS Lambda executes your code only when needed and **scales automatically**, from a few requests per day to thousands per second.
- You **pay only for the compute time you consume** – No charge when your code is not running.
- All you need to do is supply your code in the form of one or more Lambda functions to AWS Lambda, in one of the languages that AWS Lambda supports (currently Node.js, Ruby, Java, C#, GO, PowerShell, and Python), and the service can run the code on your behalf.
- AWS Lambda takes care of provisioning and managing the servers to run the code upon invocation.



source: aws.amazon.com

AWS Services

AWS Lambda – Triggers

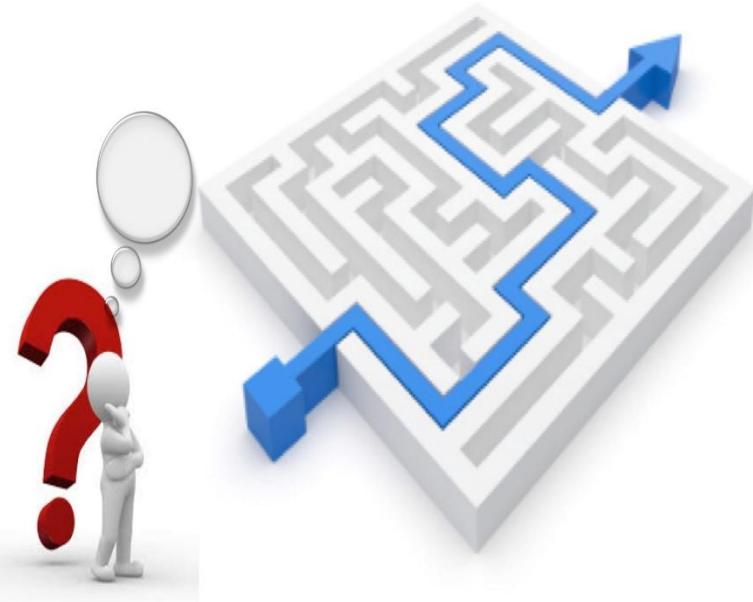
- You can use AWS Lambda to run your code in response to:
 - Events, such as changes to data in an Amazon S3 bucket or an Amazon DynamoDB table,
 - To run your code in response to HTTP requests using Amazon API Gateway, or
 - Invoke your code using API calls made using AWS SDKs.
- With these capabilities, you can use Lambda to easily build data processing triggers for AWS services like Amazon S3 and Amazon DynamoDB, process streaming data stored in Kinesis, or create your own back end that operates at AWS scale, performance, and security.



source: aws.amazon.co

AWS Lambda

- When to use
- Serverless Apps building Blocks



AWS Lambda – When to use?

- AWS Lambda is an ideal compute platform for many application scenarios, provided you can :
 - Write your application code (You are responsible for your own code) in languages supported by AWS Lambda (that is, Node.js, Java, C# and Python),
 - and run within the AWS Lambda standard runtime environment and resources provided by Lambda.



AWS Services

AWS Lambda Based Applications (Serverless Applications)

- A typical serverless application consists of one or more Lambda functions triggered by events such as object uploads to Amazon S3, Amazon SNS notifications, and API actions.
- Those functions can stand alone or leverage other resources such as DynamoDB tables or Amazon S3 buckets.
- The most basic serverless application is simply a function.



AWS Services

AWS Lambda – Building Blocks of a Lambda-Based Apps

Lambda function:

The foundation, it is comprised of your custom code and any dependent libraries.

Event source:

An AWS service, such as Amazon SNS, or a custom service, that triggers your function and executes its logic.

Downstream resources:

An AWS service, such DynamoDB tables or Amazon S3 buckets, that your Lambda function calls once it is triggered.

AWS Lambda – Building Blocks of a Lambda-Based Apps

Log streams:

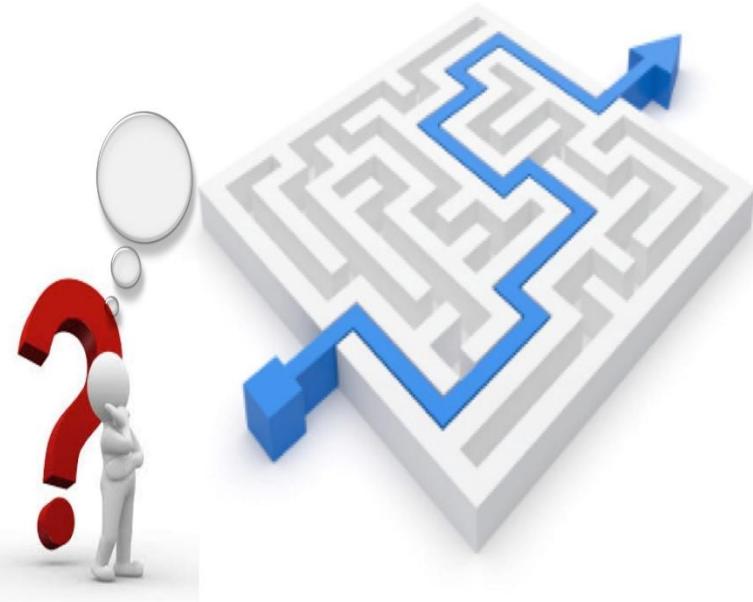
While Lambda automatically monitors your function invocations and reports metrics to CloudWatch, you can annotate your function code with custom logging statements that allow you to analyze the execution flow and performance of your Lambda function to ensure it's working properly.

AWS Serverless Application Model (AWS SAM)

A model to define serverless applications. AWS SAM is natively supported by AWS CloudFormation and defines simplified syntax for expressing serverless resources.

AWS Lambda

- Configuration
- Invocation



AWS Services

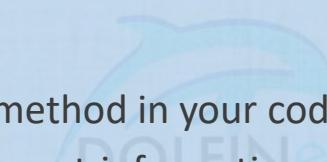
AWS Lambda Functions Configuration

Lambda function configuration information includes the following key elements:

- **Compute resources that you need**
 - You only specify the amount of memory you want to allocate for your Lambda function.
 - AWS Lambda allocates CPU power proportional to the memory
 - You can update the configuration and request additional memory in 64 MB increments from 128MB to 3008 MB.
 - Functions larger than 1536MB are allocated multiple CPU threads, and multi-threaded or multi-process code is needed to take advantage of the additional performance.
- **Maximum execution time (timeout)**
 - You pay for the AWS resources that are used to run your Lambda function.
 - To prevent your Lambda function from running indefinitely, you specify a timeout.
 - When the specified timeout is reached, AWS Lambda terminates your Lambda function.
 - Default is 3 seconds, maximum is 900 seconds (15 minutes)

AWS Lambda Functions Configuration

- **IAM role (execution role)**
 - This is the role that AWS Lambda assumes when it executes the Lambda function on your behalf.
- **Handler name**
 - The handler refers to the method in your code where AWS Lambda begins execution.
 - AWS Lambda passes any event information, which triggered the invocation, as a parameter to the handler method.



AWS Services

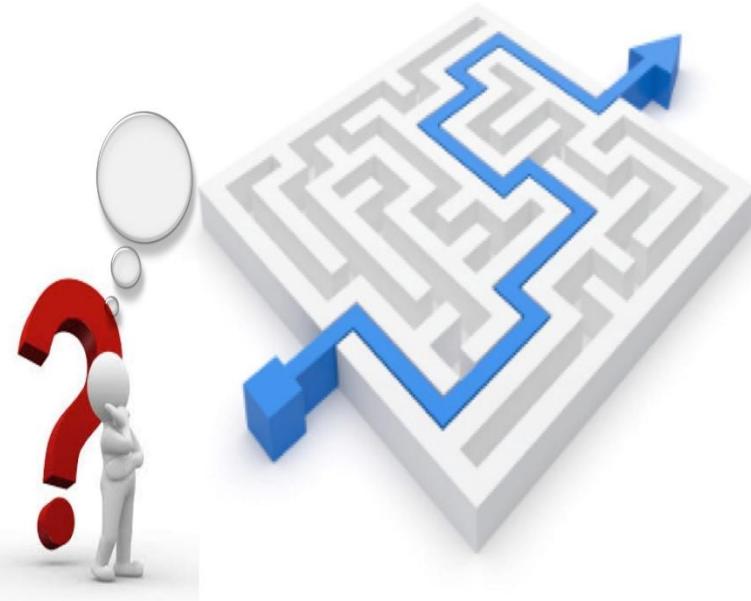
AWS Lambda Function – Services it can access

- Lambda functions can access :
 - AWS Services or non-AWS services
 - AWS Services running in AWS VPCs (Ex. Redshift , ElastiCache, RDS instances)
 - Non-AWS Services running on EC2 instances in an AWS VPC
 - Additional configuration will be required for VPC access (Security group and subnet IDs)
- AWS Lambda runs your function code securely within an internal AWS VPC (not your VPC) by default.
 - This VPC has connectivity to AWS services and the internet.
- You can configure a lambda function to connect to private subnets (DBs, Cache instances, or Internal services) in your VPC in your accounts
- To enable your Lambda function to access resources inside your VPC:
 - Provide additional VPC-specific configuration information that includes VPC subnet IDs and security group IDs
 - Lambda will then create an ENI for each subnet and security group attached to the Lambda function
 - Running Lambda functions this way is slower.

source: aws.amazon.co

AWS Lambda

Supported Triggers



AWS Lambda Functions – Invocation types

Invoking Lambda Functions

- When building applications on AWS Lambda, including serverless applications, the core components are Lambda functions and event sources.
- An event source is the AWS service or custom application that publishes events,
- A Lambda function is the custom code that processes the events.

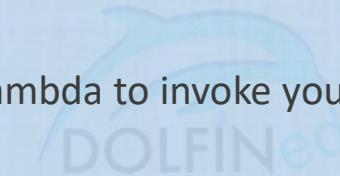
The use cases for AWS Lambda can be grouped into the following categories:

- **Using AWS Lambda with AWS services as event sources**
 - Event sources publish events that cause the Lambda function to be invoked.
- **On-demand Lambda function invocation over HTTPS (Amazon API Gateway)**
 - You can also invoke your Lambda function over HTTPS.
 - You can do this by defining a custom REST API and endpoint using API Gateway.

AWS Services

AWS Lambda Functions – Invocation types

- **On-demand Lambda function invocation** (build your own event sources using custom apps)
 - User applications such as client, mobile, or web applications can publish events and invoke Lambda functions using the AWS SDKs or AWS Mobile SDKs, such as the AWS Mobile SDK for Android.
- **Scheduled events**
 - You can also set up AWS Lambda to invoke your code on a regular, scheduled basis using the AWS Lambda console.
 - You can specify a fixed rate (number of hours, days, or weeks) or you can specify a cron expression.



AWS Lambda Functions – Event Source Mapping

- In AWS Lambda, Lambda functions and event sources are the core components in AWS Lambda.
- An event source is the entity that publishes events, and a Lambda function is the custom code that processes the events.
- Supported event sources refer to those AWS services that can be preconfigured to work with AWS Lambda.
 - The configuration is referred to as **event source mapping** which maps an event source to a Lambda function.



source: aws.amazon.co

AWS Lambda – Supported AWS event sources

- AWS Lambda supports many AWS services as event sources.
- When you configure these event sources to trigger a Lambda function, the Lambda function is invoked automatically when events occur. You define event source mapping, which is how you identify what events to track and which Lambda function to invoke.
- **Event sources** maintain the event source mapping, **except for the stream-based services (*Amazon Kinesis Streams and Amazon DynamoDB Streams*) and SQS**.
 - **For the stream-based services, AWS Lambda** maintains the event source mapping.
 - Lambda function performs polling

AWS Services

AWS Lambda – Supported AWS event sources for Lambda Functions

- Amazon S3
- Amazon DynamoDB
- Amazon Kinesis Streams
- Amazon Simple Notification Service
- Amazon Simple Email Service
- Amazon Cognito
- AWS CloudFormation
- Amazon CloudWatch Logs
- Amazon CloudWatch Events
- AWS CodeCommit
- Scheduled Events (powered by Amazon CloudWatch Events)
- AWS Config
- Amazon Alexa
- Amazon Lex
- Amazon API Gateway
- AWS IoT Button
- Amazon CloudFront
- Amazon Kinesis Firehose
- Amazon SQS
- Other Event Sources: Invoking a Lambda Function On Demand



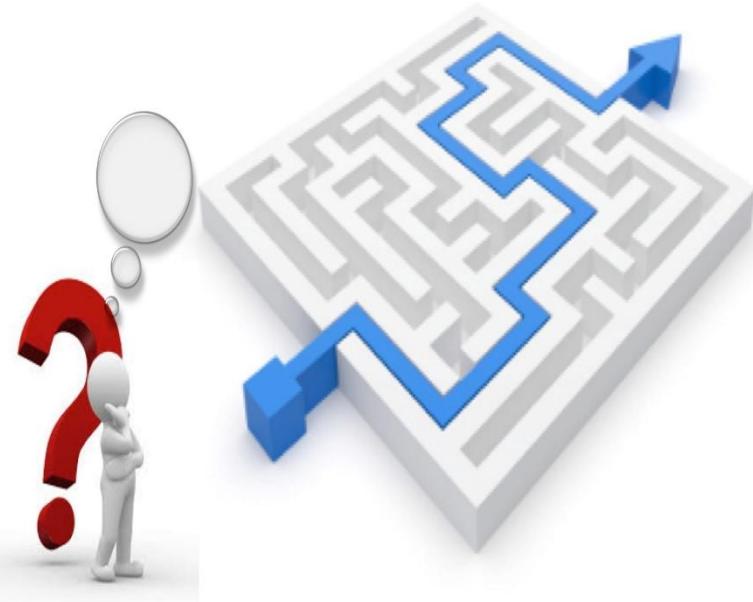
AWS Lambda – Invoking Lambda Functions On-demand

Invoking a Lambda Function On Demand

- In addition to invoking Lambda functions using event sources, you can also invoke your Lambda function on demand.
- You don't need to preconfigure any event source mapping in this case.
- However, make sure that the custom application has the necessary permissions to invoke your Lambda function.
 - For example, user applications can also generate events (build your own custom event sources)
 - User applications such as client, mobile, or web applications can publish events and invoke Lambda functions using the AWS SDKs or AWS Mobile SDKs such as the AWS Mobile SDK for Android

AWS Lambda

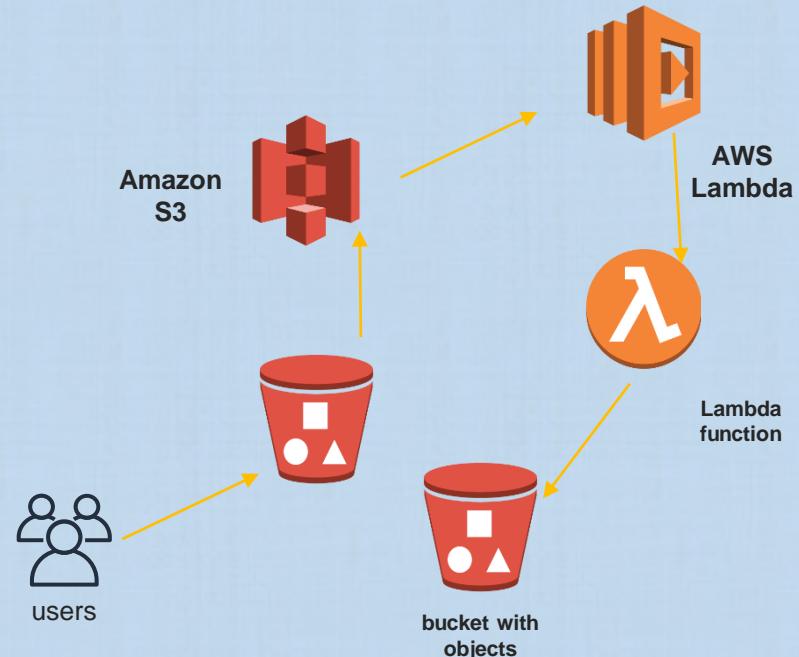
Use Cases and Examples



AWS Services

AWS Lambda – File processing use case

- You have a photo sharing application. People use your application to upload photos, and the application stores these user photos in an Amazon S3 bucket.
- Then, your application creates a thumbnail version of each user's photos and displays them on the user's profile page. A Lambda function is created which will create a thumbnail automatically.
- Amazon S3 is one of the supported AWS event sources that can publish object-created events and invoke your Lambda function.
- Your Lambda function code can read the photo object from the S3 bucket, create a thumbnail version, and then save it in another S3 bucket.

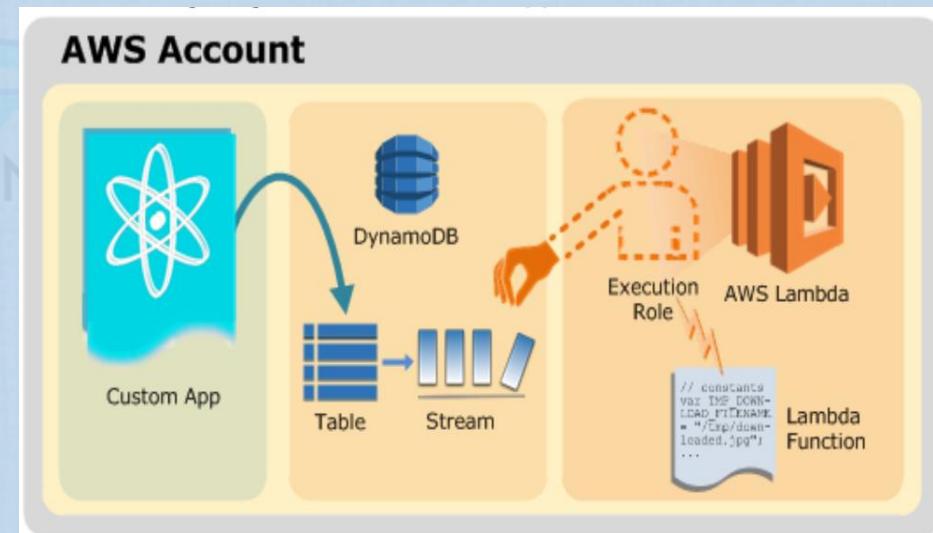


source: aws.amazon.com

AWS Services

AWS Lambda – Data and Analytics use case

- You are building an analytics application and Storing raw data in a DynamoDB table.
- Custom app updates the DynamoDB table.
- Amazon DynamoDB publishes item updates to the stream.
- AWS Lambda polls the stream and invokes your Lambda function when it detects new records in the stream.
- AWS Lambda executes the Lambda function by assuming the execution role you specified at the time you created the Lambda function.



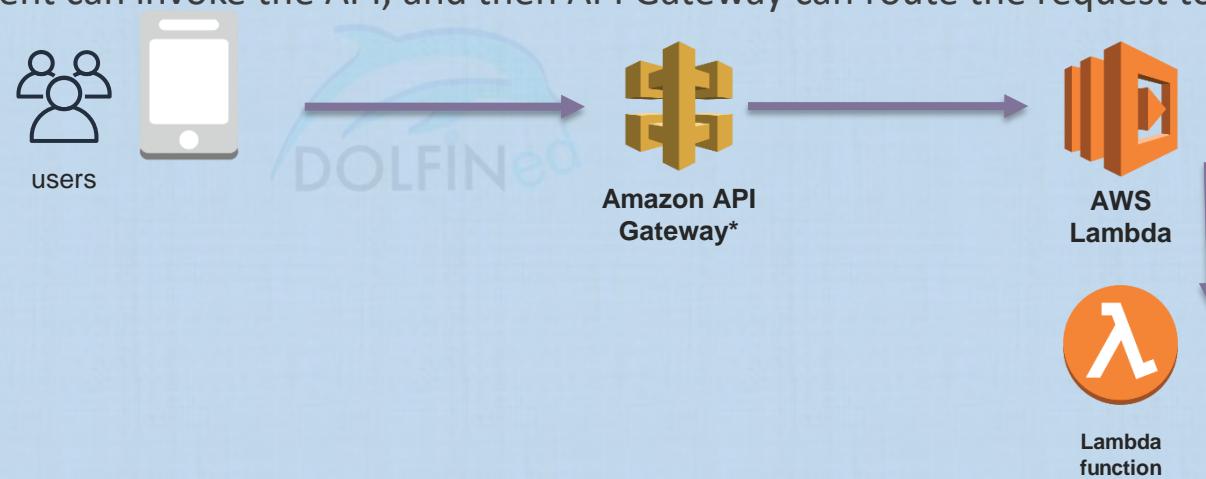
source: [aws.amazon.co](http://aws.amazon.com)

AWS Services

AWS Lambda – Website use case

Websites

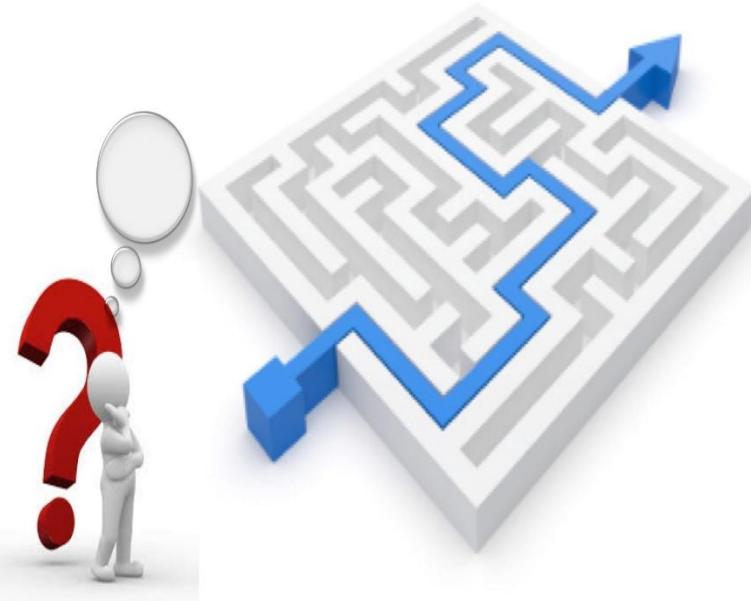
- Suppose you are creating a website and you want to host the backend logic on Lambda.
- You can invoke your Lambda function over HTTP using Amazon API Gateway as the HTTP endpoint.
- Now, your web client can invoke the API, and then API Gateway can route the request to Lambda.



source: aws.amazon.com

AWS Lambda

Scaling & Limits



AWS Lambda - Scaling

- AWS Lambda will dynamically scale capacity in response to increased traffic, subject to your account's Account Level Concurrent Execution Limit
- To handle any burst in traffic, Lambda will immediately increase your concurrently executing functions by a predetermined amount, dependent on which region it's executed
- Lambda depends on Amazon EC2 to provide Elastic Network Interfaces for VPC-enabled Lambda functions, these functions are also subject to Amazon EC2's rate limits as they scale.



source: aws.amazon.co

AWS Lambda – Understanding Scaling Behavior

- Concurrent executions refers to the number of executions of your function code that are happening at any given time.
- **Event sources that aren't stream-based**
 - If you create a Lambda function to process events from event sources that aren't stream-based
 - Each published event is a unit of work, in parallel, up to your account limits.
 - **This means one Lambda function invocation per event**
 - Therefore, the number of events (or requests) these event sources publish influences the concurrency.
- **Stream-based event sources**
 - For Lambda functions that process Kinesis or DynamoDB streams the number of shards is the unit of concurrency.
 - If your stream has 100 active shards, there will be at most 100 Lambda function invocations running concurrently.

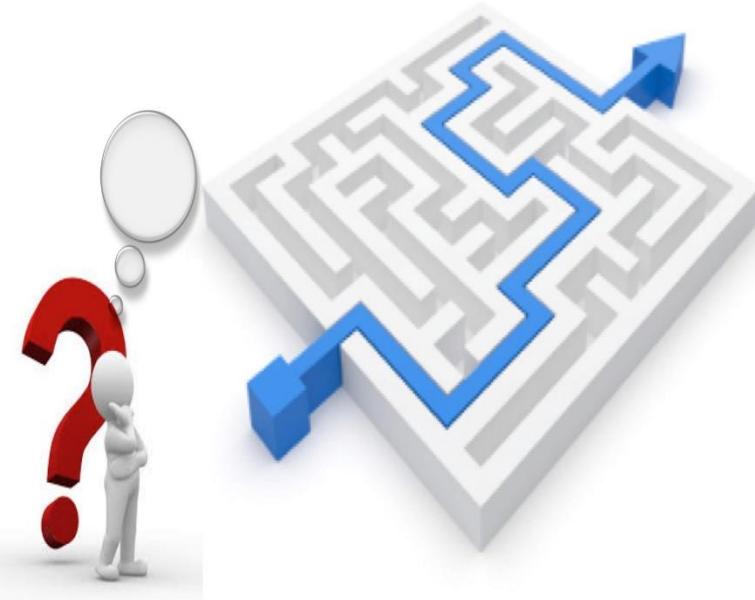
AWS Lambda – Resource Limits per Invocation

Resource	Limits
Memory allocation range	Minimum = 128 MB / Maximum = 3008 MB (with 64 MB increments). If the maximum memory use is exceeded, function invocation will be terminated.
Ephemeral disk capacity ("tmp" space)	512 MB
Number of file descriptors	1,024
Number of processes and threads (combined total)	1,024
Maximum execution duration per request	300 seconds

Also there is a 1000 concurrent executions limit per account (soft limit)

AWS Lambda

Monitoring & Pricing



AWS Lambda Monitoring

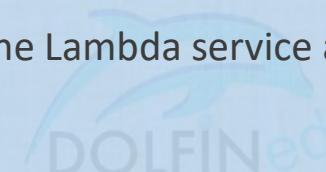
Using Amazon CloudWatch

- AWS Lambda automatically monitors Lambda functions on your behalf, reporting metrics through Amazon CloudWatch.
- To help you monitor your code as it executes, Lambda automatically tracks and publishes the associated CloudWatch metrics.
- You can leverage these metrics to set CloudWatch custom alarms.
- You can view request rates and error rates for each of your Lambda functions by **using the AWS Lambda console, the CloudWatch console, and other Amazon Web Services (AWS) resources.**

AWS Lambda X-Ray

Tracing Lambda-Based Applications with AWS X-Ray

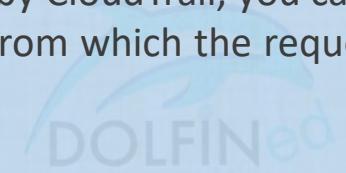
- AWS X-Ray is an AWS service that allows you to detect, analyze, and optimize performance issues with your AWS Lambda applications.
- X-Ray collects metadata from the Lambda service and any upstream or downstream services that make up your application.
- X-Ray uses this metadata to generate a detailed service graph that illustrates performance bottlenecks, latency spikes, and other issues that impact the performance of your Lambda application.



AWS Lambda and AWS CloudTrail

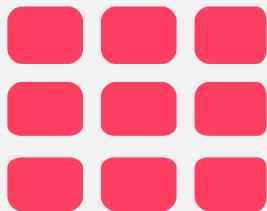
Logging AWS Lambda API Calls By Using AWS CloudTrail

- AWS Lambda is integrated with AWS CloudTrail, CloudTrail captures API calls made from the AWS Lambda console or from the AWS Lambda API.
- Using the information collected by CloudTrail, you can determine what request was made to AWS Lambda, the source IP address from which the request was made, who made the request, when it was made, and so on.



AWS Lambda Pricing

- With AWS Lambda, you pay only for what you use.
- You are charged based on the number of **requests** for your functions and the **duration**, the time it takes for your code to execute.
- Lambda counts a **request** each time it starts executing in response to an event notification or invoke call, including test invokes from the console.
- **Duration** is calculated from the time your code begins executing until it returns or otherwise terminates, rounded up to the nearest 100ms. The price depends on the amount of memory you allocate to your function.
- <https://aws.amazon.com/lambda/pricing/>

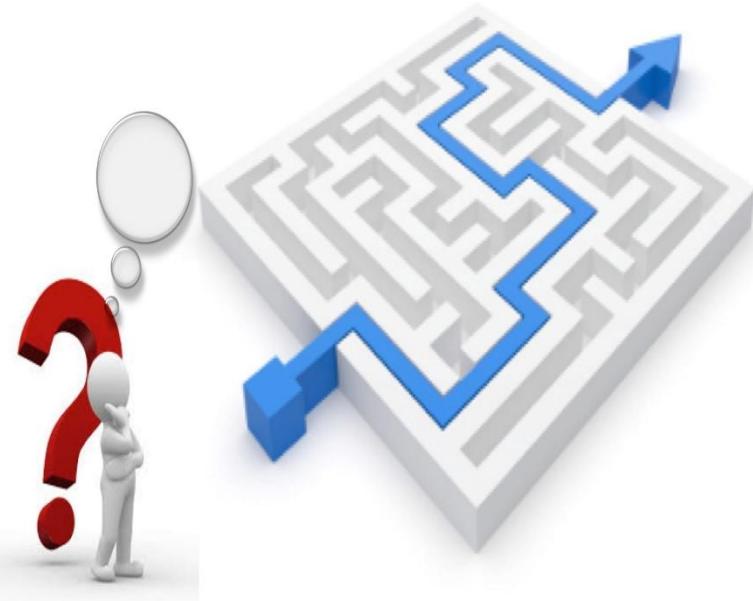


SERVERLESS LAMBDA @ EDGE

You Can Do It Too!



AWS Lambda @ Edge



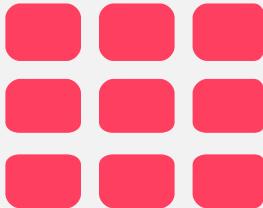
Review Topic : CloudFront

CloudFront and Lambda Edge

Using CloudFront with Lambda@Edge

- Lambda@Edge is an extension of AWS Lambda, a compute service that lets you execute functions that customize the content that CloudFront delivers.
- You can author functions in one region and execute them in AWS locations globally that are closer to the viewer, without provisioning or managing servers.
- Lambda@Edge scales automatically, from a few requests per day to thousands per second.
- Processing requests at AWS locations closer to the viewer instead of on origin servers significantly reduces latency and improves the user experience.
- When you associate a CloudFront distribution with a Lambda@Edge function, CloudFront intercepts requests and responses at CloudFront edge locations.
- You can execute Lambda functions when the following CloudFront events occur:
 - When CloudFront receives a request from a viewer (viewer request)
 - Before CloudFront forwards a request to the origin (origin request)
 - When CloudFront receives a response from the origin (origin response)
 - Before CloudFront returns the response to the viewer (viewer response)





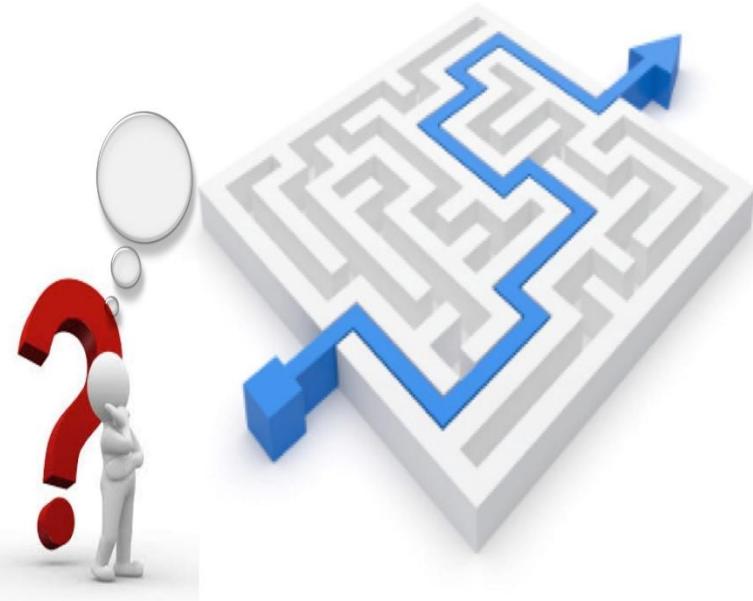
SERVERLESS AMAZON API GATEWAY

You Can Do It Too!



Amazon API Gateway

Introduction



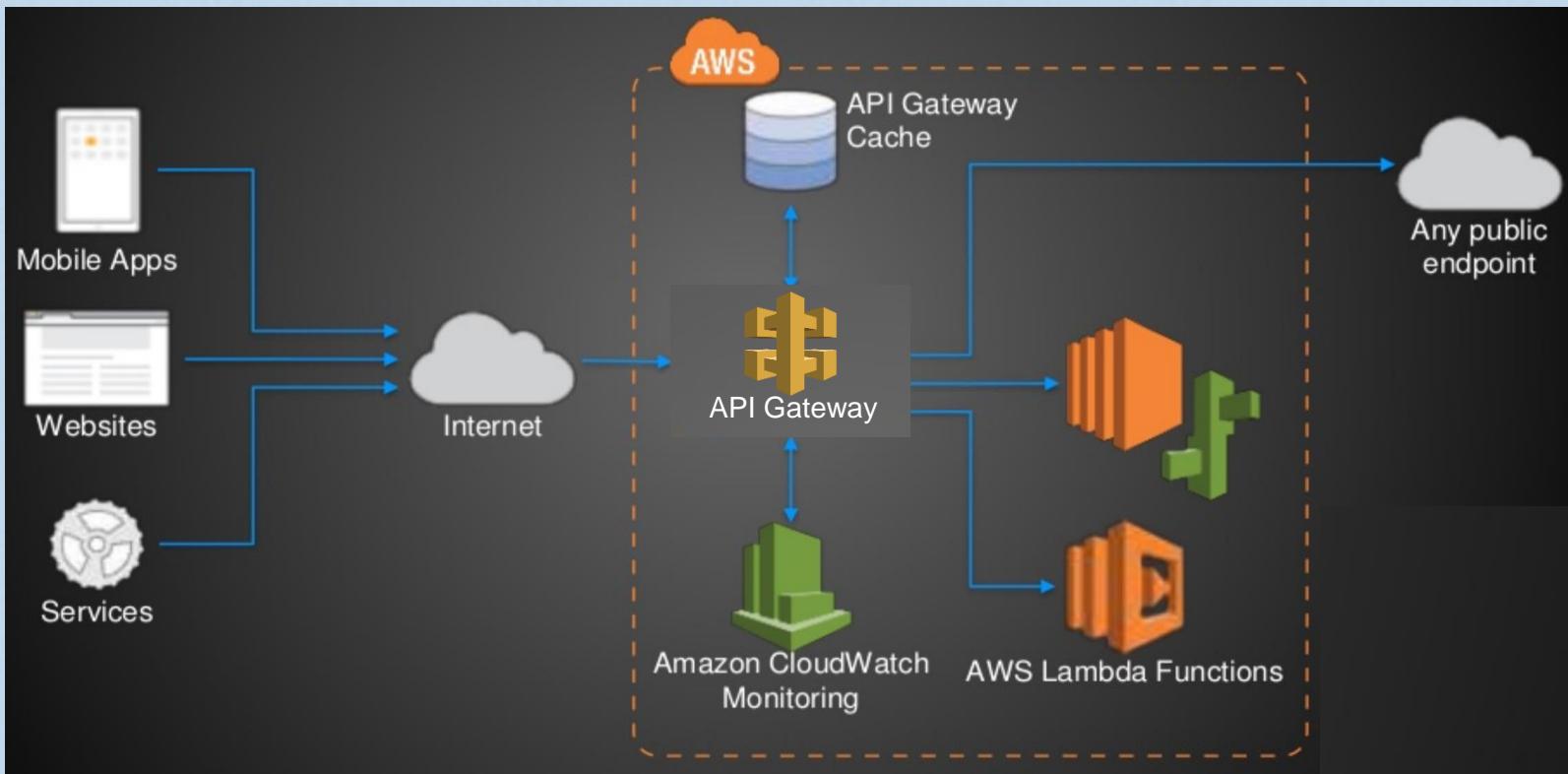
AWS Services

API Gateway

- Amazon API Gateway is a fully managed service that makes it easy for developers to publish, maintain, monitor, and secure APIs **at any scale**.
 - Together with AWS Lambda, API Gateway forms the app-facing part of the AWS serverless infrastructure.
- Amazon API Gateway handles all the tasks involved in accepting and processing **up to hundreds of thousands of concurrent API calls**, including traffic management, authorization and access control, monitoring, and API version management.
- With a few clicks in the AWS Management Console, you can create an API that acts as a “**front door**” for applications to access data, business logic, or functionality from your back-end services
 - Such as applications running on Amazon Elastic Compute Cloud (Amazon EC2), code running on AWS Lambda, or any web application (public or private).

AWS Services : API Gateway

API Gateway



source: aws.amazon.com



API Gateway - Pricing

- Amazon API Gateway has no minimum fees or startup costs.
- You pay only for the API calls received and the amount of data transferred out.



API Gateway - API

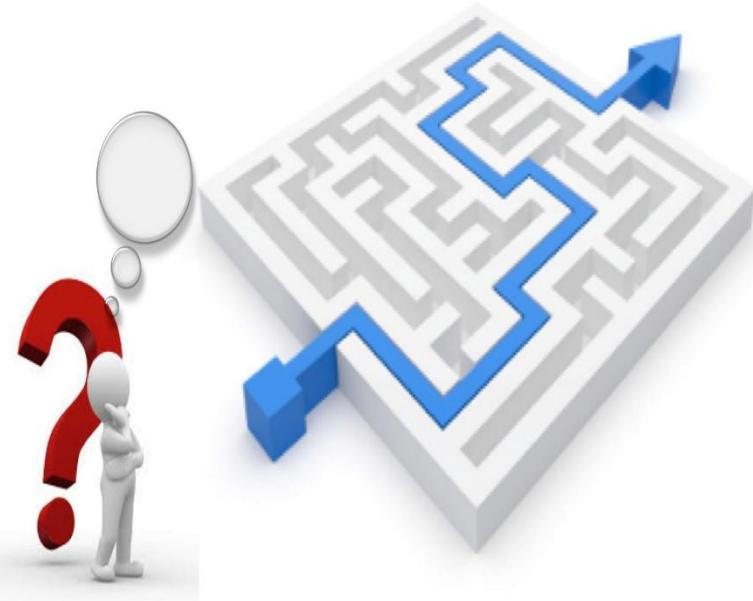
- An API gateway API is a collection of resources and methods that are integrated with backend HTTP endpoints, Lambda functions, or other AWS services. The collection can be deployed in one or more stages.
- Permissions to invoke a method are granted using IAM roles and policies or API Gateway custom authorizers. An API can present a certificate to be authenticated by the backend.
- Typically, API resources are organized in a resource tree according to the application logic.
 - Each API resource can expose one or more API methods that must have unique HTTP verbs supported by API Gateway.



source: aws.amazon.com

Amazon API Gateway

Backend, Methods, and Integration



AWS Services

API Gateway

- With Amazon API Gateway, you can provide your clients with a consistent and scalable programming interface to access three types of endpoints in the backend:
 - Invoking AWS Lambda functions,
 - Calling other AWS services, and
 - Accessing an HTTP website or webpage.
- To do this, you create an API Gateway API to integrate each API method with a backend endpoint.
 - Each backend endpoint is associated with an API Gateway integration type.

API Gateway Methods

- Each resource within a REST API can support one or more of the standard HTTP methods.
- You define which verbs should be supported for each resource (GET, POST, PUT, PATCH, DELETE, HEAD, OPTIONS) and their implementation.
 - For example, a GET to the cars resource should return a list of cars.



API Gateway Methods

HTTPS endpoints

- All of the APIs created with Amazon API Gateway expose (to the Clients) HTTPS endpoints only.
 - Amazon API Gateway does not support unencrypted (HTTP) endpoints with the clients.
- By default, Amazon API Gateway assigns an internal domain to the API that automatically uses the Amazon API Gateway certificate.
 - When configuring your APIs to run under a custom domain name, you can provide your own certificate for the domain.

AWS Services

API Gateway – API endpoints (Hostnames of Deployed APIs)

API Gateway supports the following types of API endpoints,

Edge-optimized API endpoint:

- Relies on Amazon **Cloudfront** distributions, **is the default endpoint types**
- An edge-optimized API endpoint enables clients to access an API through an Amazon CloudFront distribution.
- API requests are routed to the nearest CloudFront Point of Presence (POP) which typically improves connection time for geographically diverse clients.

Regional API endpoint:

- It is intended to serve clients, such as EC2 instances, in the same AWS region where the API is deployed.
- Together with Route53 latency-based routing, regional endpoints enable an API developer to deploy an API to multiple regions using the same regional API endpoint configuration, setting the same custom domain name for each deployed API

Private API endpoint:

- Runs inside a VPC. Example backend services are EC2, ELB services and Lambda.

AWS Services

API Gateway

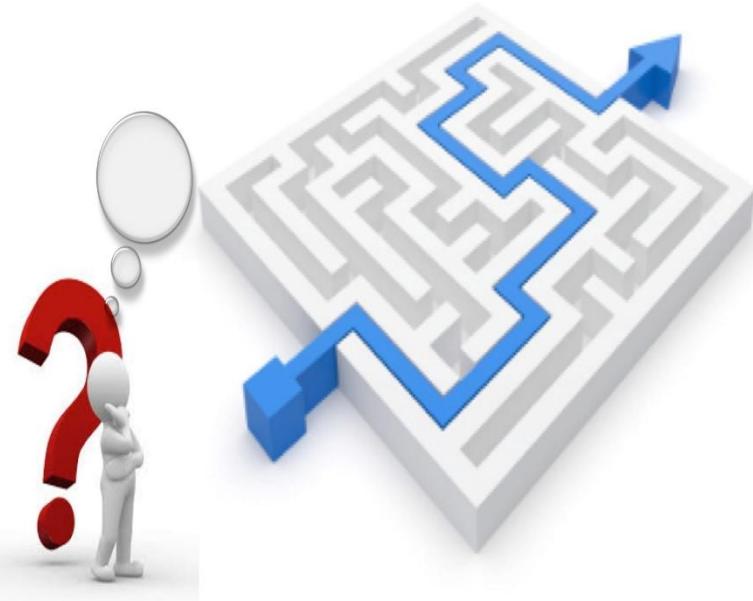
Backend Services

- Amazon API Gateway can execute AWS Lambda functions in your account,
- It can start AWS Step Functions state machines, or
- Call HTTP endpoints hosted on AWS Elastic Beanstalk, Amazon EC2,
- And also non-AWS hosted HTTP based operations that are accessible via the public Internet.
- API Gateway also allows you to specify a mapping template to generate static content to be returned, helping you mock your APIs before the backend is ready (response from within the API itself – Mock Integration)
- You can also integrate API Gateway with other AWS services directly –
 - For example, you could expose an API method in API Gateway that sends data directly to Amazon Kinesis.



Amazon API Gateway

Features and Benefits



AWS Services – API Gateway Features

API Gateway Benefits/Features

- Robust, secure, and scalable access to backend APIs and Hosts **multiple versions** and **release stages** of your APIs
- Create and distribute **API Keys** to developers
- Use of **AWS Sig-v4** to authorize access to APIs
- **Throttle** and Monitor requests to protect your backend
- Integrates with X-Ray, WAF, CloudWatch and CloudTrail for Protection, Troubleshooting, monitoring and logging



source: aws.amazon.com

AWS Services – API Gateway Features

API Gateway Benefits/Features

- Manage **Cache** to store API responses
- **SDK Generation** for iOS, Android, and JavaScript
- Reduced Latency and Distributed Denial of Service protection through the use of CloudFront
- Request/Response data transformation and API mocking
- Swagger support
- Open APIs, API Keys, Usage Plans for 3rd party API developers



source: aws.amazon.com

API Gateway Features – Throttling/Caching/Scaling

Resiliency

- **Through Throttling rules,** Amazon API Gateway helps you manage traffic to your back-end systems via throttling rules that are based on the number of requests per second, for each HTTP method (GET, PUT..) in your APIs.

Caching:

- You can set up a cache with customizable keys and **time-to-live (TTL)** in seconds for your API data to avoid hitting your back-end services for each request.
 - Enhanced response times and reduces load on backend services

Scaling:

- Amazon API Gateway handles any level of traffic received by an API, so you are free to focus on your business logic and services rather than maintaining infrastructure.

API Gateway Features – API Versions

API Lifecycle Management -

Multiple Versions of the same REST API

- Amazon API Gateway lets you run multiple versions of the same API simultaneously so that applications can continue to call previous API versions even after the latest versions are published.
- Amazon API Gateway gives you the **ability to clone an existing API to create a new version**
 - When you are ready to start working on the next major version of your API, you will be able to keep working on your version 1 and version 2 APIs simultaneously.
- You can determine which version of the API is being accessed/used.



source: [aws.amazon.co](https://aws.amazon.com)

API Gateway – API Multiple Release Stages

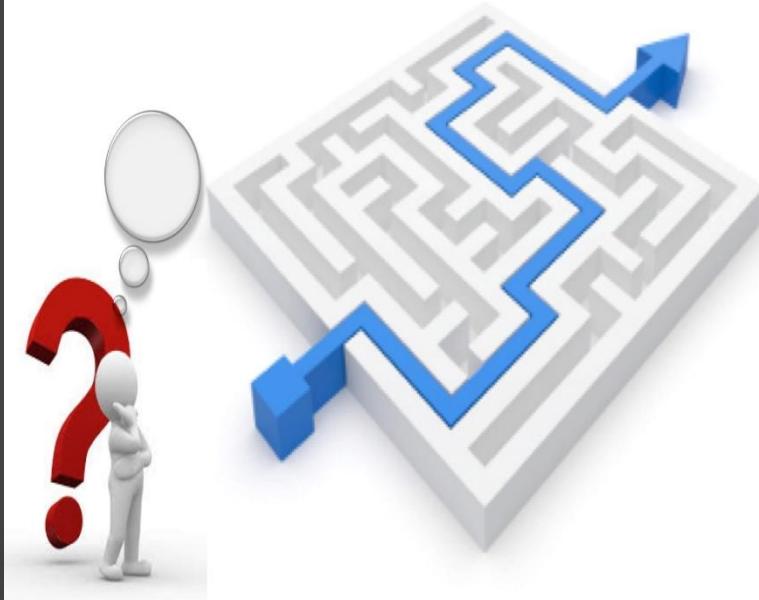
API Lifecycle Management

Multiple Release Stages:

- Amazon API Gateway also helps you manage multiple release stages for each API version, such as alpha, beta, and production.
 - Each API stage can be configured to interact with different backend endpoints based on your API setup.
 - Specific stages and versions of an API can be associated with a custom domain name and managed through Amazon API Gateway.
 - Stage and version management allow you to easily test new API versions that enhance or add new functionality to earlier API releases,
 - This ensures backward-compatibility as your user communities transition to adopt the latest release.

Amazon API Gateway

Security, Authorization, and Monitoring



API Gateway

HTTPS endpoints

- You can create HTTPS endpoints.
 - All of the APIs created with Amazon API Gateway expose HTTPS endpoints only.
 - Amazon API Gateway does not support unencrypted (HTTP) endpoints.
 - By default, Amazon API Gateway assigns an internal domain to the API that automatically uses the Amazon API Gateway certificate.
 - When configuring your APIs to run under a custom domain name, you can provide your own certificate for the domain.

API Gateway – AWS Authorization

With Amazon API Gateway, you can optionally set your API methods to require authorization.

- **IAM:**
 - To authorize and verify API requests to AWS services, API Gateway can leverage signature version 4
 - Using signature version 4 authentication, you can use Identity and Access Management (IAM) and access policies to authorize access to your APIs and all your other AWS resources.
- **Lambda Authorizers:**
 - You can also use AWS Lambda functions to verify and authorize bearer tokens such as JWT tokens or SAML assertions.
- **Amazon Cognito User Pools (No authorization – only authentication)**
 - You can retrieve temporary credentials associated with a role in your AWS account using Amazon Cognito.

API Gateway – Cross Account Access to APIs

When AWS identity and access management is enabled on a specific resource,

- IAM users from different AWS accounts cannot access that resource unless the caller is allowed to assume the resource owner's role,
 - i.e API Gateway does not currently support cross-account authentication.



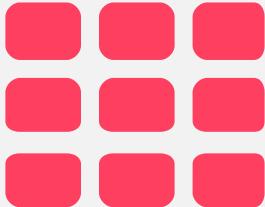
API Gateway – API Operations and Monitoring

Monitoring through API Gateway dashboard:

- After an API is deployed and in use, Amazon API Gateway provides you with a REST API dashboard to visually monitor calls to the services.
- API Gateway also meters utilization by third-party developers, the data is available in the API Gateway console and through the APIs.

Monitoring through CloudWatch:

- The Amazon API Gateway logs (near real time) back-end performance metrics such as API calls, Latency, and error rates to AWS CloudWatch in your account
 - This allow for the set up of custom CloudWatch alarms on Amazon API Gateway APIs



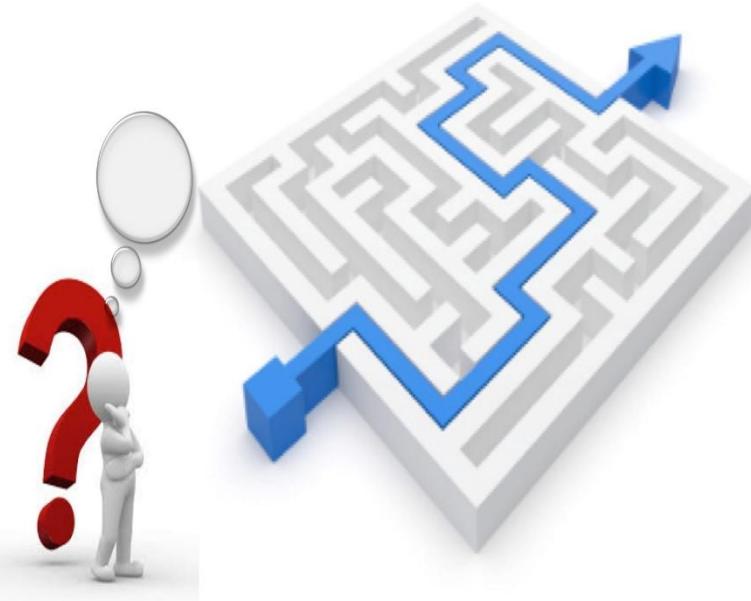
SERVERLESS NON-SQL DATABASES AMAZON DYNAMODB

You Can Do It Too!



Amazon DynamoDB

Features



Review Topic : AWS Database

DynamoDB

DynamoDB is a fully managed NoSQL database that supports both document & Key-Value store

- Is extremely fast and delivers predictable performance with seamless scalability
- Use for applications that need **consistent, single-digit millisecond latency** at any scale.
- Is a web service that uses HTTP over SSL (HTTPS) as a transport and JSON as a message serialization format
- Use cases:
 - Mobile Apps
 - Web Apps
 - Gaming Apps
 - Ad-tech Apps
 - Internet of things (IoT)



Review Topic : AWS Database

DynamoDB – Tables

- DynamoDB tables are schemaless
 - Which means that neither the attributes nor their data types need to be defined beforehand
 - Each item can have its own distinct attributes
- DynamoDB does not support:
 - Complex relations DB querying or Joins
 - Does not support complex transactions



Review Topic : AWS Database

DynamoDB – Durability & Performance

- DynamoDB automatically replicates data across three facilities (Data Centers [Not AZs])in an AWS region for H.A and data durability
 - It also partitions your DB over sufficient number of servers according to your read/write capacity
 - Performs automatic failover in case of any failure
- DynamoDB runs exclusively on SSD volumes which provides:
 - Low latency
 - Predictable performance
 - High I/Os

Review Topic : AWS Database

DynamoDB – Read Consistency

DynamoDB supports both Eventual Consistency (default) and Strong Consistency models

- **Eventually Consistent reads (Default):**

- When you read data from a DynamoDB table, the response might not reflect the results of a recently completed write operation.
- Best read throughput
- Consistency across all copies is reached in 1 second

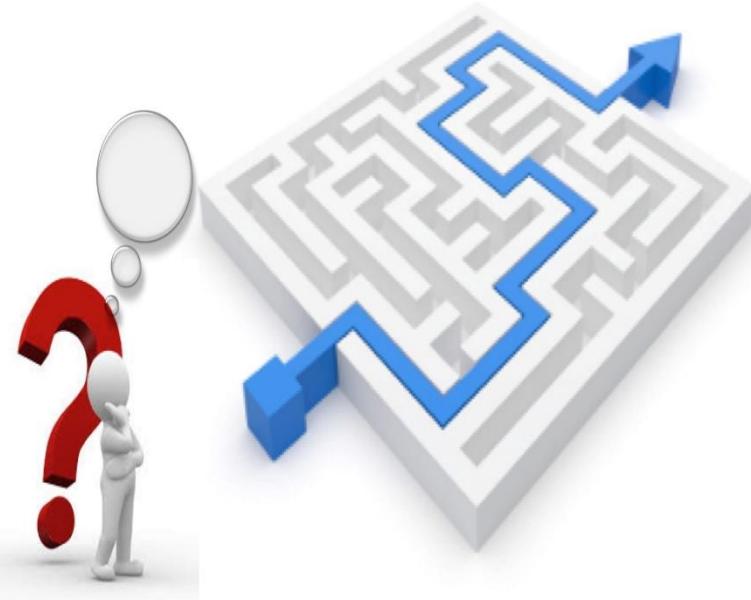
- **Strongly Consistent reads:**

- A read returns a result that reflects all writes that received a successful response prior to the read
- Users/Applications reading from DynamoDB tables can specify in their requests if they want strong consistency, otherwise it will be eventually consistent reads (default)
 - So the application will dictate what is required , Strong, Eventual, or both



Amazon DynamoDB

Table Components



Review Topic : AWS Database

DynamoDB - Tables

DynamoDB's basic components:

- **Tables:**
 - Like all other DBs , DynamoDB stores data in tables
 - A table is a collection of data items
 - Each table can have an infinite number of data items
- **Items:**
 - Each table contains multiple items.
 - An item, is a group of attributes that is uniquely identifiable among all of the other items. (PersonID in the example table)
 - An item consists of a primary or composite key and a flexible number of attributes
 - Items in DynamoDB are similar in to rows, records in other DBs
 - There is no limit to the number of items you can store in a table

```
{  
    "PersonID": 101,  
    "LastName": "Smith",  
    "FirstName": "Fred",  
    "Phone": "555-4321"  
}  
  
{  
    "PersonID": 102,  
    "LastName": "Jones",  
    "FirstName": "Mary",  
    "Address": {  
        "Street": "123 Main",  
        "City": "Anytown",  
        "State": "OH",  
        "ZIPCode": 12345  
    }  
}  
  
{  
    "PersonID": 103,  
    "LastName": "Stephens",  
    "FirstName": "Howard",  
    "Address": {  
        "Street": "123 Main",  
        "City": "London",  
        "PostalCode": "ER3 5K8"  
    },  
    "FavoriteColor": "Blue"  
}
```

Source:aws.amazon.co

Review Topic : AWS Database

DynamoDB - Tables

DynamoDB's basic components:

- **Attributes:**
 - Each item is composed of one or more attributes.
 - An attribute consists of the attribute name and a value or a set of values
 - An attribute is a fundamental data element, something that does not need to be broken down any further
 - Attributes in DynamoDB are similar in to fields or columns in other database systems.
- Note:
 - Aggregate size of an item **can NOT exceed 400KB including key and all attributes**

```
{  
    "PersonID": 101,  
    "LastName": "Smith",  
    "FirstName": "Fred",  
    "Phone": "555-4321"  
}  
  
{  
    "PersonID": 102,  
    "LastName": "Jones",  
    "FirstName": "Mary",  
    "Address": {  
        "Street": "123 Main",  
        "City": "Anytown",  
        "State": "OH",  
        "ZIPCode": 12345  
    }  
}  
  
{  
    "PersonID": 103,  
    "LastName": "Stephens",  
    "FirstName": "Howard",  
    "Address": {  
        "Street": "123 Main",  
        "City": "London",  
        "PostalCode": "ER3 5K8"  
    },  
    "FavoriteColor": "Blue"  
}
```

Source.aws.amazon.co

Review Topic : AWS Database

DynamoDB

- DynamoDB allows low latency read/write access to items ranging from 1 byte to 400KBytes
- DynamoDB can be used to store pointers to S3 stored objects, or items of sizes larger than 400KB, too if needed
- DynamoDB stores data indexed **by a primary key**
 - You specify the primary key when you create the table
- Each item in the table has a unique identifier, or primary key, that distinguishes the item from all of the others in the table.
- The primary key is the only required attribute for items in a table
- DynamoDB supports GET/PUT operations using a user defined Primary Key



Source:aws.amazon.co

Amazon DynamoDB

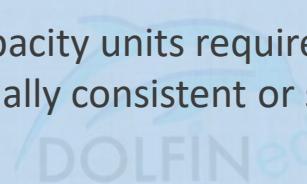
Read/Write Capacity Units & Pricing



Review Topic : AWS Database

DynamoDB – Read Capacity Units

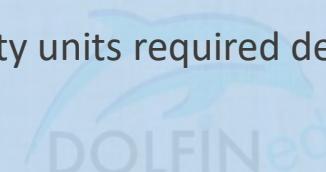
- One read capacity unit represents one strongly consistent read per second, or two eventually consistent reads per second for an item up to 4KB in size
- If you need to read an item that is larger than 4 KB, DynamoDB will need to consume additional read capacity units.
 - The total number of read capacity units required depends on the item size, and whether you want an eventually consistent or strongly consistent read.



Review Topic : AWS Database

DynamoDB – Write Capacity Units

- One write capacity unit represents one write per second for an item up to 1 KB in size.
- If you need to write an item that is larger than 1 KB, DynamoDB will need to consume additional write capacity units.
- The total number of write capacity units required depends on the item size.



Review Topic : AWS Database

DynamoDB - Pricing

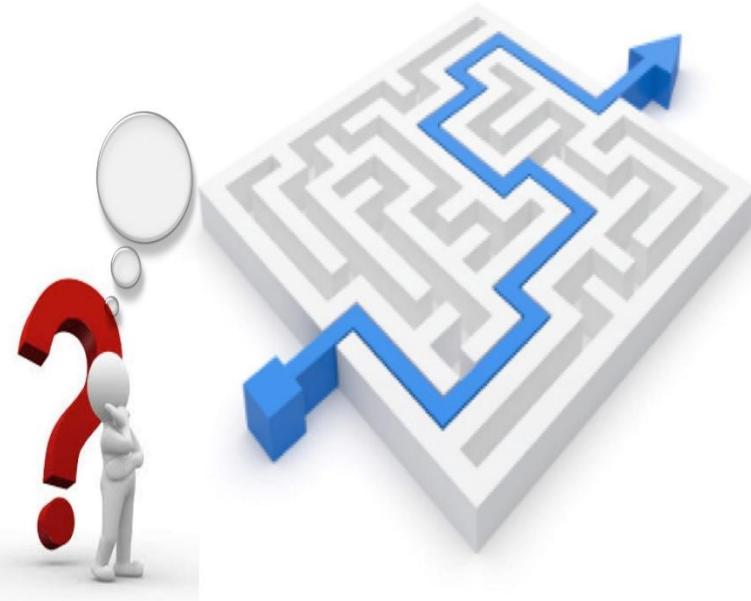
As a rule of thumb, reads are cheaper than writes when using DynamoDb

You pay for :

- Each table's provisioned read/wrote throughput (Hourly rates)
 - You are charged for the provisioned throughput regardless whether you use it or not
- Indexed Data Storage
- Internet Data Transfer (if crosses a region)
- Free tier per account (across all tables) of 25 read capacity units, and 25 write capacity units per month

Amazon DynamoDB

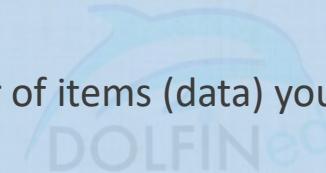
Scalability



Review Topic : AWS Database

DynamoDB –Scalability

- It provides for a push button scaling on AWS where you can increase the read/write throughput and AWS will go ahead and scale it for you (up or down) without downtime or performance degradations
- You can scale the provisioned capacity of your DynamoDB table anytime you want
- There is no limit to the number of items (data) you can store in a DynamoDB table
- There is no limit on how much data you can store per DynamoDB table



Review Topic : AWS Database

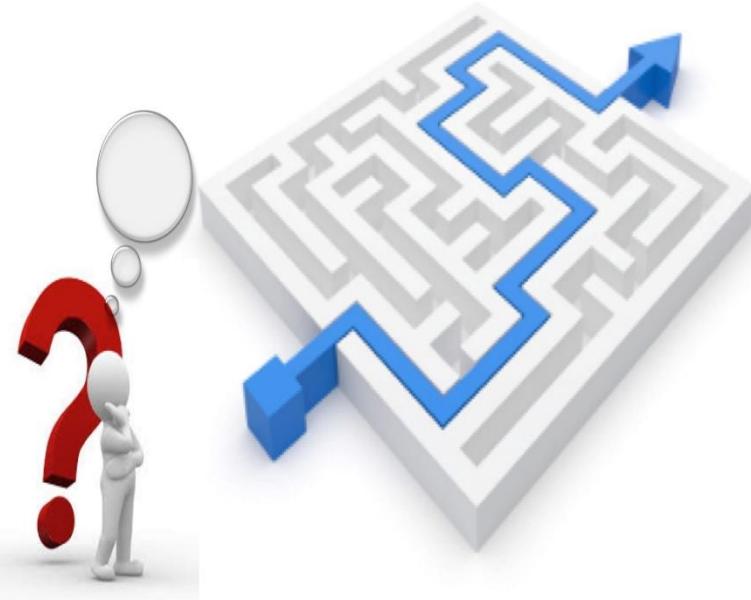
DynamoDB - Throttling

- If your read or write requests exceed the throughput settings for a table, DynamoDB can throttle that request.
- DynamoDB can also throttle read requests exceeds for an index.
 - Throttling prevents your application from consuming too many capacity units.
 - When a request is throttled, it fails with an HTTP 400 code (Bad Request) and a ProvisionedThroughputExceededException
- The AWS SDKs have built-in support for retrying throttled requests



Amazon DynamoDB

- **DynamoDB Local**
- **Point in Time Recovery**
- **Backup**
- **Time To Live (TTL)**
- **DynamoDB Accelerator (DAX)**
- **DynamoDB Streams**
- **DynamoDB Transactions**



DynamoDB Local

- You can download a version of DynamoDB that is self contained on a local computer
- This enables you to write and/or test application without having to access the AWS DynamoDB service
- With minor code changes, this can then be deployed to the DynamoDB web service when ready

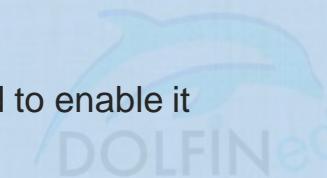


On-Demand Backup and Restore for DynamoDB

- Amazon DynamoDB provides on-demand backup capability.
 - This allows you to create full backups of your tables for long-term retention and archival for regulatory compliance needs.
 - Backup and restore actions execute with zero impact on table performance or availability.
- Backup can be completed in seconds regardless of the table size.
- All backups persist until manually deleted
- The backup and restore functionality works in the same Region as the source table.
- They do not charge any extra cost beyond the storage costs
- All backups in DynamoDB work without consuming any provisioned throughput on the table
- Global and Local secondary indexes, DynamoDB Streams, and Provisioned Read/Write capacity are included

DynamoDB Point In Time Recovery (PITR)

- On-demand backups and point in time recovery can be enabled for your Amazon DynamoDB tables
- PITR provides continuous backups of your DynamoDB table data.
 - This will help protect against accidental table or item deletions.
 - With this feature enabled, there is no need to worry about creating, maintaining, and scheduling on-demand backups
- It is disabled by default, you need to enable it
- Global and Local Secondary indexes, Provisioned Read/Write capacity and Encryption settings are also restored to the new table
 - Recovery happens to a new table, not the same base table.



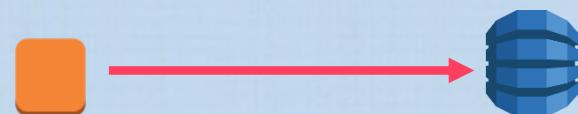
DynamoDB TTL

- Amazon DynamoDB Time to Live (TTL) defines when items in a table expire so that they can be automatically deleted from the database.
 - This is particularly helpful in reducing storage usage by deleting the irrelevant data without consuming from the provision throughput
- TTL does not cost extra.
- With TTL enabled on a table, you need to create an attribute in each item (expiry column) which defines a timestamp for deletion on a per item basis.
- Examples of data that can benefit from TTL, session state data, event logs, usage patterns, and other temporary data.
- If you have sensitive data that must be retained only for a certain amount of time according to contractual or regulatory obligations, TTL helps you ensure that it is removed promptly and as scheduled.



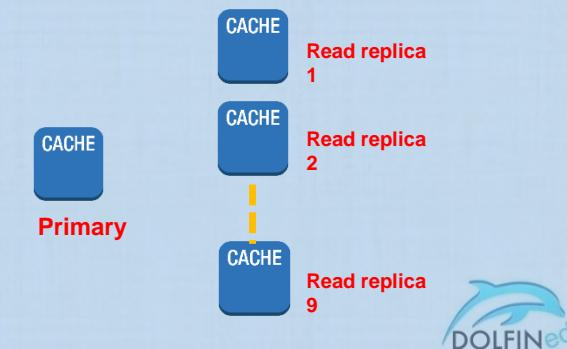
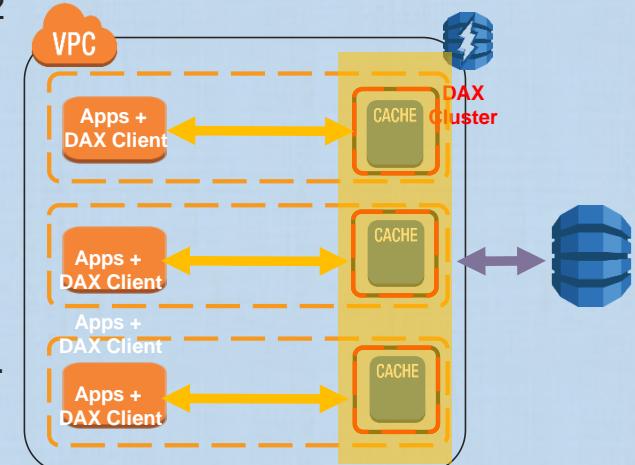
In-Memory Acceleration with DynamoDB Accelerator (DAX)

- DAX is a DynamoDB-compatible caching service that enables you to benefit from fast in-memory performance for demanding applications.
 - It is designed for DynamoDB only and NOT for other AWS services.
- DynamoDB is designed to provide a consistent milliseconds response time
- Using DynamoDB DAX provides access to microsecond latency data reads from DynamoDB tables.
 - DAX scales where a Multi-AZ DAX cluster can serve millions of requests per second.
- DAX is for eventually **consistent reads only, not for strongly consistent reads**
- DAX is deployed as a cluster in a VPC (default VPC by default)



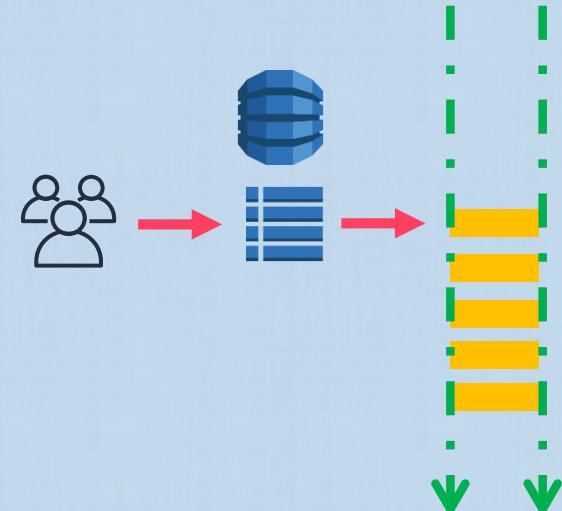
DynamoDB Accelerator (DAX) – Clusters and Fault tolerance

- Applications that use DAX will be deployed on Amazon EC2 instances, in your VPC, with the **DAX Client**.
- DAX cluster consists of one or more nodes (up to 10).
- Each node runs its own instance of the DAX caching software.
- One of the nodes serves as the primary node for the cluster. Additional nodes (if present) serve as read replicas
- For a fault tolerant cluster, AWS recommends deploying a minimum of 3 nodes in 3 different AZs
- DAX cluster in an AWS Region can only interact with DynamoDB tables that are in the same Region.
 - It is also designed for use with DynamoDB only, not other AWS services



DynamoDB Streams

- Having the ability to capture changes to the items in a DynamoDB table, at the time when the change occurs, can be useful in many applications.
- A DynamoDB stream is an ordered flow (carries sequence numbers) of data/information about changes to items in a DynamoDB table.
- DynamoDB Streams captures a time-ordered (with sequence numbers and time stamps) of item-level modifications in any DynamoDB table
 - This data is stored for 24 hours, after which it gets removed from the stream automatically.
- Applications can access this log and view the data items as they appeared before and after they were modified, in near-real time.
 - DynamoDB streams operate asynchronously with no performance hit on the table.
- Encryption at rest encrypts the data in DynamoDB streams.



DynamoDB Transactions

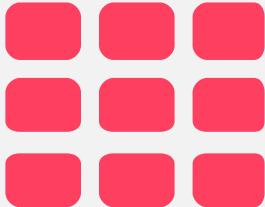
- DynamoDB **transactional read and write** APIs can be used to manage complex business workflows that require adding, updating, or deleting **multiple items as a single, all-or-nothing operation**.
- Amazon DynamoDB transactions simplify the development side of making **coordinated, all-or-nothing changes to multiple items both within and across tables**.
- For example, a video game developer can ensure that players' profiles are updated correctly when they exchange items in a game or make in-game purchases.
- Transactions provide atomicity, consistency, isolation, and durability (ACID) in DynamoDB, helping to maintain applications data correctness.
- With the transaction write API, it is possible to group multiple Put, Update, Delete, and Condition Check actions.



DynamoDB Transactions

- With the transaction write API, it is possible to group multiple Put, Update, Delete, and Condition Check actions.
- The actions can then be submitted as a single **TransactWriteItems** operation that either succeeds or fails as a unit.
- The same is true for multiple Get actions, which can be grouped and submitted as a single **TransactGetItems** operation.
- For every item involved in a DynamoDB transaction, DynamoDB performs two underlying reads (2 RCUs) or two writes (2 WCUs):
 - One to prepare the transaction and one to commit the transaction.
 - These two-underlying read/write operations are visible in your Amazon CloudWatch metrics.
- There is no additional cost to enable transactions for DynamoDB tables.
 - Charges are only for the reads or writes (WCUs and RCUs) that are part of the transaction.





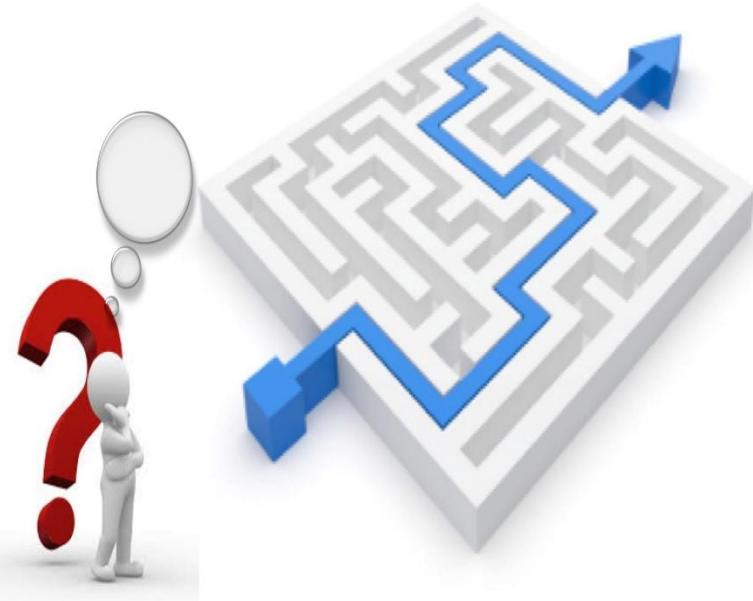
AWS SERVICES ELASTIC MAP REDUCE (EMR)

You Can Do It Too!



Amazon Elastic Map Reduce

Introduction



AWS Elastic Map Reduce (EMR)

Introduction

- Amazon EMR is a web service that enables businesses, researchers, data analysts, and developers to easily and cost-effectively process vast amounts of data.
- Amazon EMR lets you focus on crunching or analyzing your data without having to worry about time-consuming set-up, management or tuning of Hadoop clusters or the compute capacity upon which they sit.
- It utilizes a hosted Hadoop framework running on the web-scale infrastructure of Amazon Elastic Compute Cloud (Amazon EC2) and Amazon Simple Storage Service (Amazon S3).
- Amazon EMR is ideal for problems that necessitate the fast and efficient processing of large amounts of data.
- You can use the simple web interface of the AWS Management Console to launch your clusters and monitor processing-intensive computation on clusters of Amazon EC2 instances.



AWS Elastic Map Reduce (EMR)

EMR

- Using Amazon EMR, you can instantly provision as much or as little capacity as you like to perform data-intensive tasks for applications such as:
 - Web indexing,
 - Data mining,
 - Log file analysis,
 - Machine learning,
 - Financial analysis,
 - Scientific simulation, and
 - Bioinformatics research.
- You can also use EMR to transform and move large amounts of data into and out of other AWS data stores and databases, such as S3 and DynamoDB.



AWS Elastic Map Reduce (EMR)

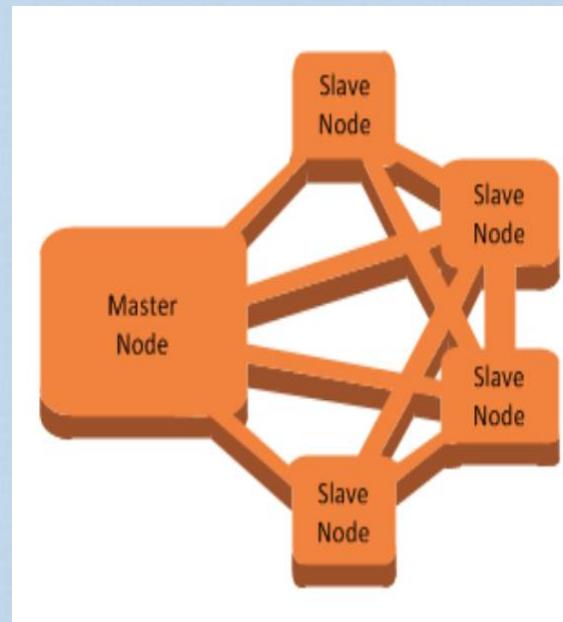
EMR and Apache Hadoop

- EMR leverages Apache Hadoop as its distributed data processing engine.
- Hadoop is an open source, Java software framework that supports data-intensive distributed applications running on large clusters of commodity hardware.
- Hadoop implements a processing/programming model named “MapReduce,” which is designed to process large data sets quickly. Used primarily in Analytics and Business intelligence purposes.

AWS Elastic Map Reduce (EMR)

EMR Clusters

- **Master node:**
 - A node that manages the cluster by running software components to coordinate the distribution of data and tasks among other nodes for processing.
 - The master node tracks the status of tasks and monitors the health of the cluster. Every cluster has a master node, and it's possible to create a single-node cluster with only the master node.
- **Core node:**
 - A node with software components that run tasks and store data in the Hadoop Distributed File System (HDFS) on your cluster.
 - Multi-node clusters have at least one core node.
- **Task node:**
 - A node with software components that only runs tasks and does not store data in HDFS. Task nodes are optional.



AWS Elastic Map Reduce (EMR)

EMR Integration with other AWS Services

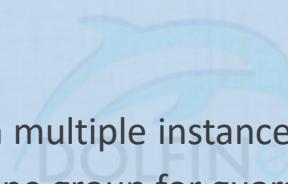
- Amazon EMR integrates with other AWS services to provide capabilities and functionality related to networking, storage, security, and so on, for your cluster. The following list provides several examples of this integration:
 - Amazon EC2 for the instances that comprise the nodes in the cluster
 - Amazon Virtual Private Cloud (Amazon VPC) to configure the virtual network in which you launch your instances
 - Amazon S3 to store input and output data
 - Amazon CloudWatch to monitor cluster performance and configure alarms
 - AWS Identity and Access Management (IAM) to configure permissions
 - AWS CloudTrail to audit requests made to the service
 - AWS Data Pipeline to schedule and start your clusters
 - EC2 Key pairs to allow for SSH access to EMR Cluster Nodes



AWS Elastic Map Reduce (EMR)

EMR Cluster Nodes

- You have root access to the EC2 instances in the EMR Cluster.
- EMR provides flexibility to scale your cluster up or down as your computing needs change.
- You can resize your cluster to add instances for peak workloads and remove instances to control costs when peak workloads subside.
- EMR also provides the option to run multiple instance groups so that you can:
 - Use On-Demand Instances in one group for guaranteed processing power
 - Use Spot Instances in another group to have your jobs completed faster and for lower costs.
 - You can also mix different instance types to take advantage of better pricing for one Spot Instance type over another.
- Additionally, Amazon EMR provides the flexibility to use several file systems for your input, output, and intermediate data, it supports HDFS (Hadoop Distributed File System), and EMRFS



AWS Elastic Map Reduce (EMR)

EMR and AWS Availability Zones

- Amazon EMR launches all nodes for a given cluster in the same Amazon EC2 Availability Zone.
- Running a cluster in the same zone improves performance of the jobs flows because it provides a higher data access rate.
- By default, Amazon EMR chooses the Availability Zone with the most available resources in which to run your cluster.
 - However, you can specify another Availability Zone if required.

AWS Elastic Map Reduce (EMR)

EMR and Data to be processed

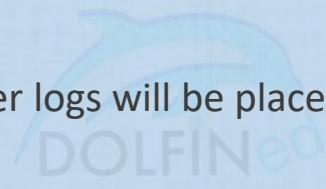
- EMR is not about real time fast, large data ingestion
- Load data to be processed by EMR to S3
- Customers upload their input data into Amazon S3.
 - Amazon EMR then launches a number of Amazon EC2 instances as specified by the customer.
 - EMR pulls the data from Amazon S3 into the launched Amazon EC2 instances.
 - Once the cluster is finished, Amazon EMR transfers the output data to Amazon S3, where customers can then retrieve it or use as input in another cluster.
- The Hadoop application can also load the data from anywhere on the internet or from other AWS services.

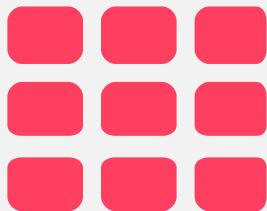


AWS Elastic Map Reduce (EMR)

EMR Encryption and Logs

- Amazon EMR supports optional Amazon S3 server-side and client-side encryption with EMRFS to help protect the data that you store in Amazon S3. With server-side encryption, Amazon S3 encrypts your data after you upload it.
- EMR always uses HTTPS to move data between S3 and EMR cluster's EC2 instances
- Hadoop system logs as well as user logs will be placed in the Amazon S3 bucket which you specify when creating a cluster.
- Also, EMR integrates with CloudTrail and can have all API calls logged to an S3 bucket of your choice





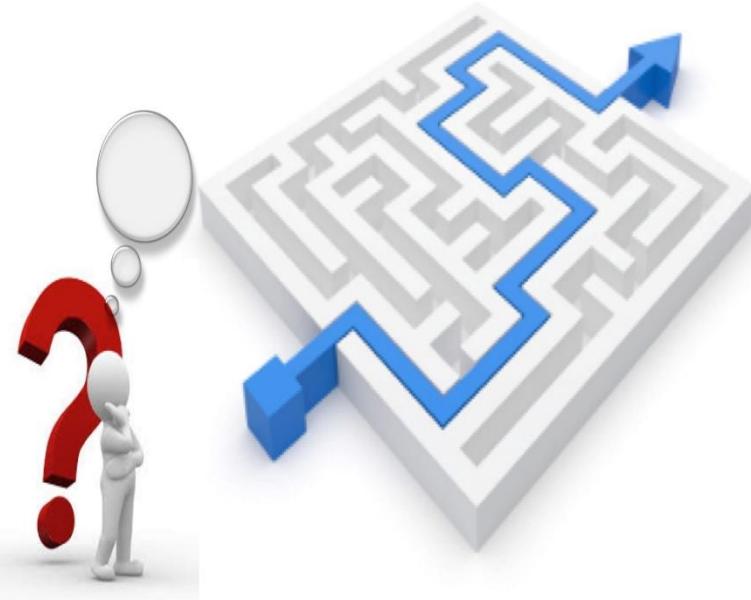
AWS SERVICES ELASTICACHE

You Can Do It Too!



Amazon ElastiCache

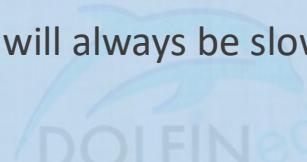
Introduction



Review Topic : AWS Services

ElastiCache

- The primary purpose of an in-memory key-value store is to provide ultra-fast (sub-millisecond latency) and inexpensive access to copies of data.
- Most data stores have areas of data that are frequently accessed but seldom updated.
- Additionally, querying a database will always be slower and more expensive than locating a key in a key-value pair cache.



Review Topic : AWS Services

ElastiCache

- Is an AWS fully managed web service
- It is an in-memory key value data store engine in the cloud
 - It improves the performance of web applications by allowing for the retrieval of information from a fast, managed, in-memory system (instead of reading from the DB itself)
 - Improves response times for user transactions and queries
 - Can enhance response time for read-intensive And/Or compute-intensive workloads
 - Examples: social networking, gaming, media sharing, Q&A portals
- It offloads the read workload from the main DB instances (less I/O load on the DB)
- It does this by storing the results of frequently accessed pieces of data (or computationally intensive calculations) in-memory
- Integrates with Cloudwatch
- Deployed using EC2 instances



Review Topic : AWS Services

ElastiCache

- ElastiCache EC2 nodes deployed can not be accessed from the internet, nor can they be accessed by EC2 instances in other VPCs
- Can be on-demand or Reserved Instances too (NOT Spot instances)
- Access to ElastiCache nodes is controlled by VPC security groups and Subnet groups
- You need to configure VPC Subnet groups for ElastiCache (VPC that hosts EC2 instances and the ElastiCache cluster)
 - Changing the subnet group of an existing ElastiCache cluster is not currently supported



Review Topic : AWS Services

ElastiCache

- If an ElastiCache node fails it is automatically replaced by AWS ElastiCache (fully managed service)
- ElastiCache nodes are launched in clusters, and can span more than one subnet of the same subnet group which was associated with the cluster when creating it
- Your application connects to your cluster using endpoints.
 - An endpoint is a node or cluster's unique address (use these endpoints rather than the IP addresses in your application)
- A cluster can have one or more nodes included within



Review Topic : AWS Services

ElastiCache

- Supports two caching engines:
 - Memcached (is not a Data store [DB], only a cache)
 - Redis – is the fastest NoSQL – can be used as a DB (data store)
- A cluster is a collection of one or more nodes using the same caching engine (Memcached or Redis)
- A cluster can be as small as one node



Amazon ElastiCache

Memcached engine



Review Topic : AWS Services

ElastiCache - Memcached

- Is not persistent
 - Can not be used as a data store
 - If the node fails, the cached data (in the node) is lost
- Ideal front-end for data stores (RDS, DynamoDB...etc)
- Use cases:
 - Cache contents of a DB
 - Cache data from dynamically generated webpages
 - Transient session data
 - High frequency counters for admission control in high volume web Apps



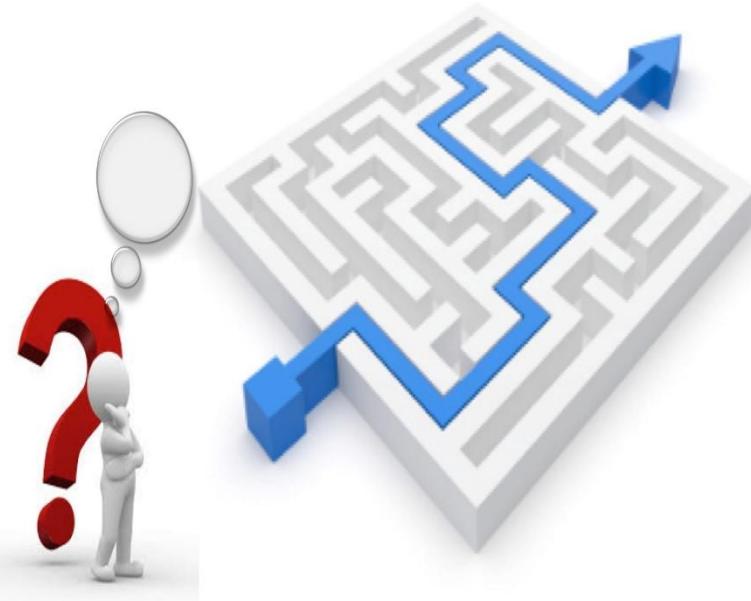
Review Topic : AWS Services

ElastiCache - Memcached

- Does not support Multi-AZ failover, replication, NOR does it support Snapshots for backup/resore
 - Node failure means data loss
- You can, however, place your Memcached nodes in different AZs to minimize the impact of an AZ failure and to contain the data loss in such an incident
 - You can horizontally partition your data across those nodes

Amazon ElastiCache

Redis engine



Review Topic : AWS Services

ElastiCache - Redis

- Is persistent, using the snapshotting feature.
 - At any time, you can restore your data by creating a new Redis cluster and populating it with data from a backup.
- Use cases: Web, Mobile Apps, Healthcare Apps, Financial Apps, Gaming, Ad-Tech, and IoT
- Supports Redis master/slave replication
- Supports snapshots (automatic and manual) to S3
 - The backup can be used to restore a cluster or to seed a new cluster
 - The backup includes cluster metadata and all data in the cluster



Review Topic : AWS Services

ElastiCache - Redis

- You can copy your snapshots to other AWS regions (indirectly though)
 - You do this by:
 - First exporting the snapshot from ElastiCache to an S3 bucket in the same region
 - Then you copy the exported copy of the snapshot to the destination region
 - This can be handy if you want to seed a new cluster in the other region, rather than waiting to populate a new cluster from the other region's database

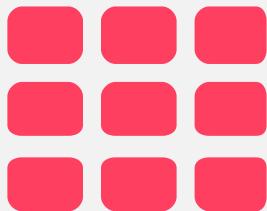


Review Topic : AWS Services

ElastiCache – Redis Multi-AZ support

- Multi-AZ is done by creating read replica(s) in another AZ in the same region
- **Clustering mode disabled :** Your Redis cluster can have only one shard
 - One shard can have one read/write primary node and 0-to-5 read only replicas
 - You can distribute the replicas over multiple AZs in the same region
 - Replication from the primary node to the read replica is asynchronous
 - Applications can read from any node in the cluster, but can write to the primary node only
- **Clustering mode enabled:** Your Redis cluster can have up to 15 shards,
 - With the data partitioned across the shards
 - Each shard has one primary node and –5 read only replicas
- Snapshots can slow down your nodes, better take snapshots from the read replicas





AMAZON KINESIS

You Can Do It Too!



AWS Services : Kinesis

Streaming of Data

Streaming of data:

- Is the data that is generated and sent continuously from a large number (1000s or hundreds of 1000s) of data sources, where data is sent in small sizes (usually in Kbytes or MBytes)
- Kinesis, is a platform for streaming data on AWS (used for IoT and Bigdata Analytics)
 - It offers powerful services to make it easy to load and analyze streaming data
 - It facilitates the way to build custom streaming data Apps for specialized needs
 - Kinesis is an AWS managed streaming data service(s)
- Kinesis can continuously capture and store Terabytes of data per hour from 100s of thousands of sources
- IT Offers three **managed services**, there are:
 - Kinesis Streams, Kinesis Firehose, Kinesis Analytics



AWS Kinesis

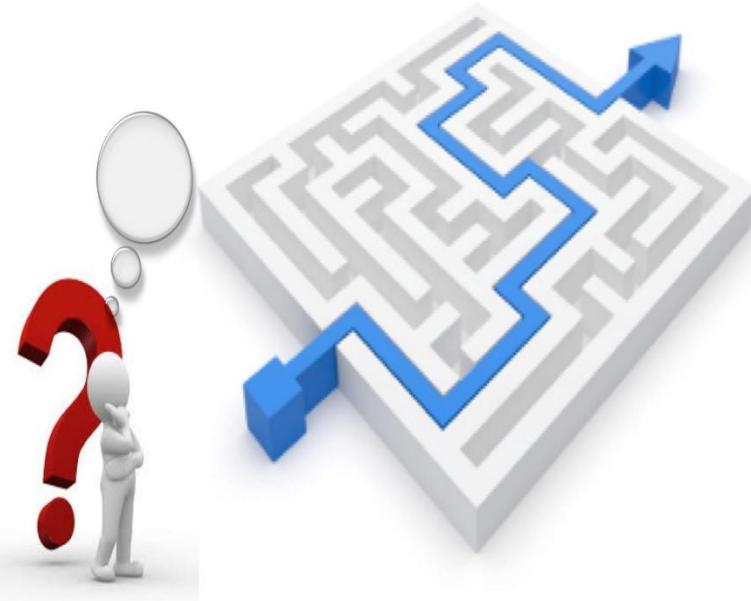
Sources of data (examples)

- IoT sensors data
- Log files from customers of mobile and web apps
- eCommerce purchases
- In-game player activities
- Social media networks
- Financial trading floors and stock markets
- Telemetry



Amazon Kinesis

Kinesis Data Streams



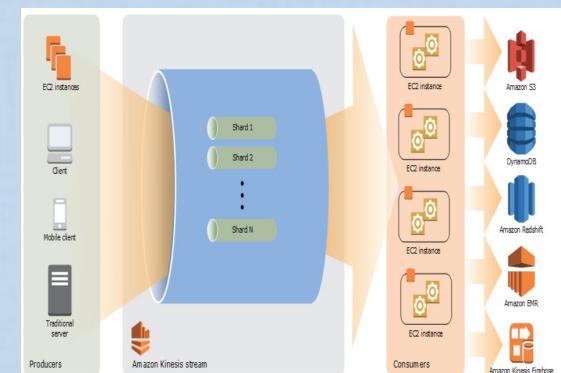
AWS Services

AWS Kinesis Streams

Use Amazon Kinesis Streams to collect and process large stream of data records in real time.

Custom data-processing applications, known as *Amazon Kinesis Streams applications* read data from an *Kinesis stream* as data records.

- These applications use the Kinesis Client Library and run on Amazon EC2 instances.
- The processed records can be:
 - Sent to dashboards,
 - Used to generate alerts,
 - Used to Dynamically change pricing and advertising strategies,
 - Saved to a variety of other AWS services (EMR, DynamoDB, or Redshift)
- Kinesis can make this huge streams of data available for processing By AWS services or customer applications in less than a second

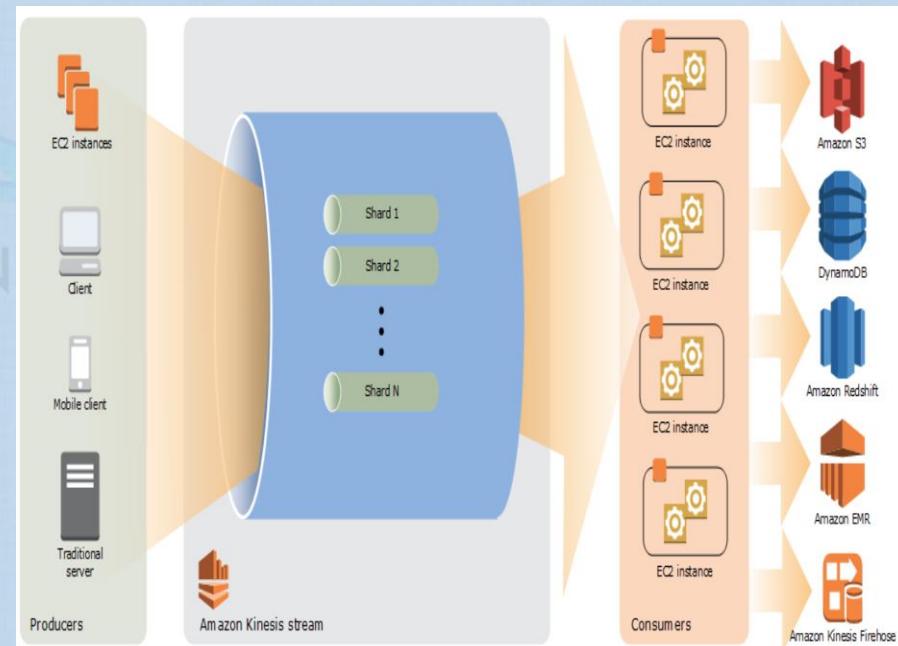


source:aws.amazon.com

AWS Services

AWS Kinesis Streams

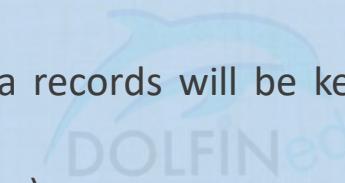
- **Producers:** put records into Amazon Kinesis Streams
- **Consumers:** get records from Amazon Kinesis Streams and process them
- **Kinesis Streams Application:** is a consumer of a stream that commonly runs on a fleet of EC2 instances
- **Shard:** is a uniquely identified group of data records in a stream. A stream is composed of one or more shards.
 - A shard is the base throughput unit of a stream
 - it can take up to 1Mb/s input and 2 Mb/s output
- **Record:** the data unit stored in a Kinesis stream



source:aws.amazon.com

AWS Kinesis Streams _ Retention Period

- Record: is the unit of data stored in a Kinesis stream
- A shard can support 1000 PUT records/sec
- **Retention Period:**
 - The time for which data records will be kept and made accessible to Kinesis streams applications
 - Default is 24 hours (1 day)
 - Can be extended to 7 days (168 hrs) for additional charges



AWS Kinesis Streams - Encryption

- **Server-side encryption**
 - Amazon Kinesis Streams can automatically encrypt sensitive data as a producer enters it into a stream.
 - Kinesis Streams uses KMS master keys for encryption
 - To read from or write to an encrypted stream, producer and consumer applications must have permission to access the master key
- **Note**
 - Using server-side encryption incurs KMS costs

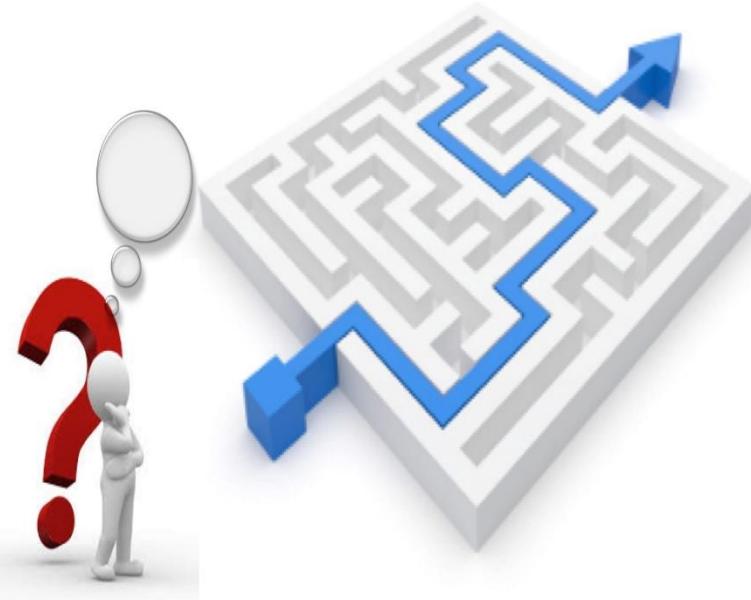


AWS Kinesis Streams

- Amazon Kinesis Streams manages the infrastructure, storage, networking, and configuration needed to stream your data at the level of your data throughput.
 - You do not need to worry about provisioning, deployment, ongoing-maintenance of hardware, software, or other services for your data streams.
- Amazon Kinesis Streams synchronously replicates data across three availability zones, providing high availability and data durability.
- Kinesis Streams use cases:
 - Accelerate log & data feed intakes
 - Real time metric and reporting analytics
 - Complex stream processing (successive stages of stream processing)

Amazon Kinesis

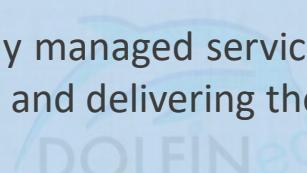
Kinesis Firehose



AWS Services

AWS Kinesis Firehose

- Use **Amazon Kinesis Firehose** to deliver real-time streaming data to destinations such as Amazon S3 and Amazon Redshift. Is the easiest way to load streaming data into AWS
 - Kinesis streams can be used as the source(s) to Kinesis Firehose
 - You can configure Kinesis Firehose to transform your data before delivering it.
- Amazon Kinesis Firehose is a fully managed service for automatically capturing real-time data stream from producers (sources) and delivering them to destinations such as:
 - AWS S3
 - AWS Redshift,
 - AWS Elasticsearch Service (ES)
 - and Splunk.
- With Kinesis Firehose, you don't need to write applications or manage resources.



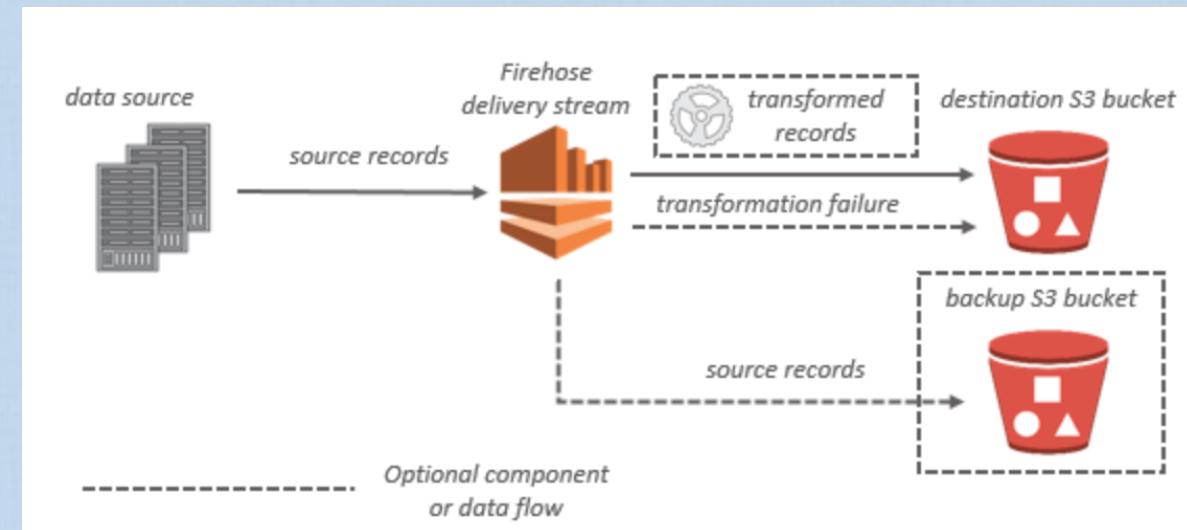
AWS Kinesis Data Firehose

- It can batch, compress, and encrypt the data before loading it, minimizing the amount of storage used at the destination and increasing security.
- Amazon Kinesis Data Firehose manages all underlying infrastructure, storage, networking, You do not have to worry about provisioning, deployment, ongoing maintenance.
 - Firehose also scales elastically without requiring any intervention.
- Amazon Kinesis Data Firehose synchronously replicates data across three facilities in an AWS Region, providing high availability and durability for the data as it is transported to the destinations.
- Each delivery stream stores data records **for up to 24 hours in case** the delivery destination is unavailable.
- Firehose can invoke an AWS Lambda function to transform incoming data before delivering it to destinations.

AWS Services

AWS Kinesis Firehose

- For AWS S3 destinations, streaming data is delivered to your S3 bucket.
- If data transformation is enabled, you can optionally back up source data to another Amazon S3 bucket.
- ElasticSearch case is similar

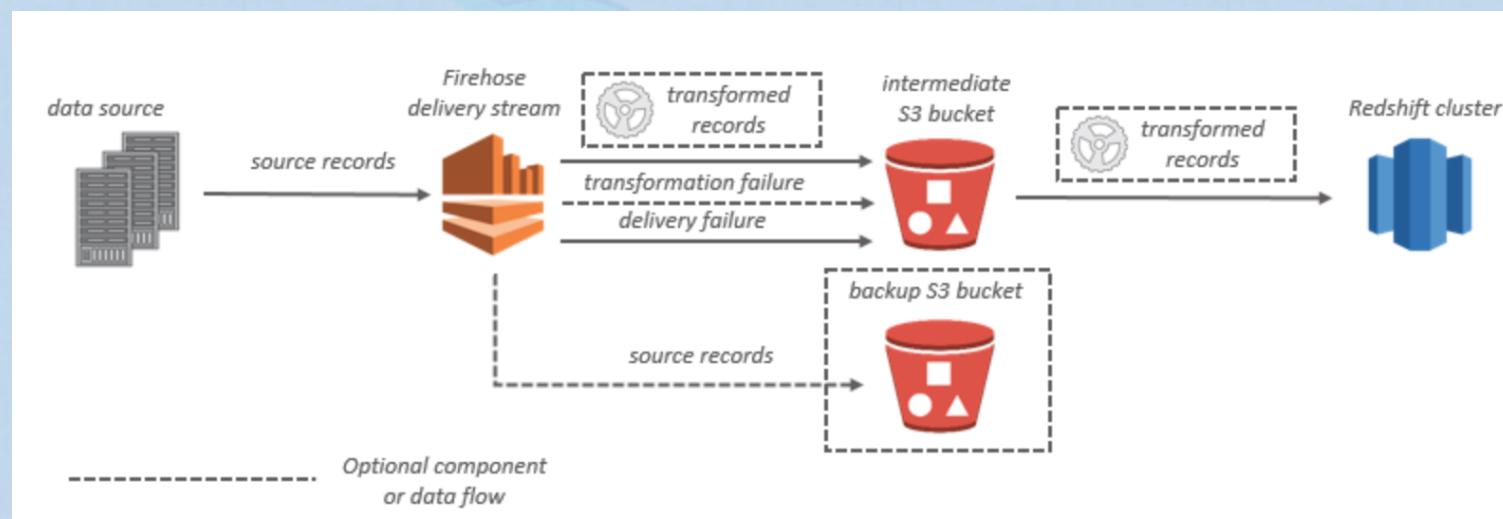


source:aws.amazon.com
DOLFIN ED

AWS Services

AWS Kinesis Firehose

- For AWS Redshift destinations, streaming data is delivered to your S3 bucket first.
 - Kinesis Firehose then issues an Amazon Redshift **COPY** command to load data from your S3 bucket to your Amazon Redshift cluster
 - If data transformation is enabled, you can optionally back up source data to another Amazon S3 bucket



source:aws.amazon.com

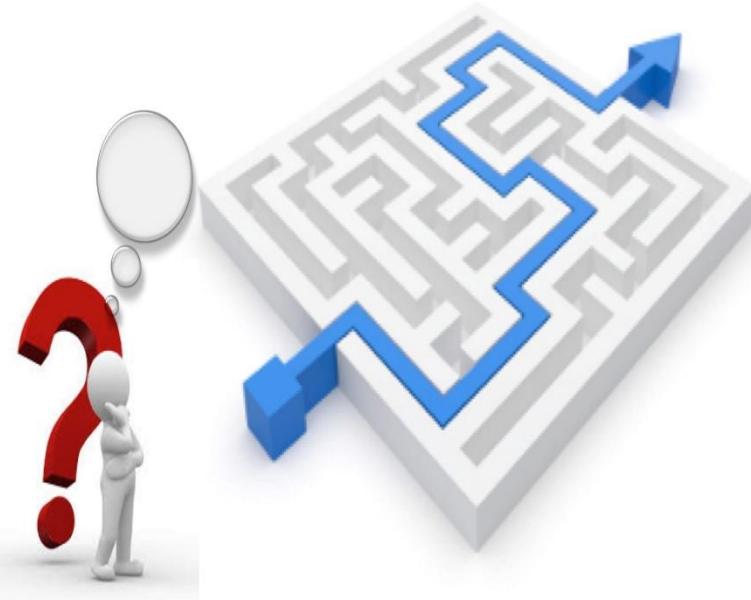
AWS Kinesis Firehose – Data Encryption

- **Server-side encryption**
 - If you have sensitive data, you can enable server-side data encryption when you use Amazon Kinesis Firehose.
 - However, this is only possible if you DO NOT use a Kinesis stream as your data source.
 - When you configure a Kinesis stream as the data source of a Kinesis Firehose delivery stream, the data is stored in the Kinesis stream.
 - Enable Kinesis Streams to encrypt the data in this case.



Amazon Kinesis

Kinesis Analytics



AWS Kinesis Analytics

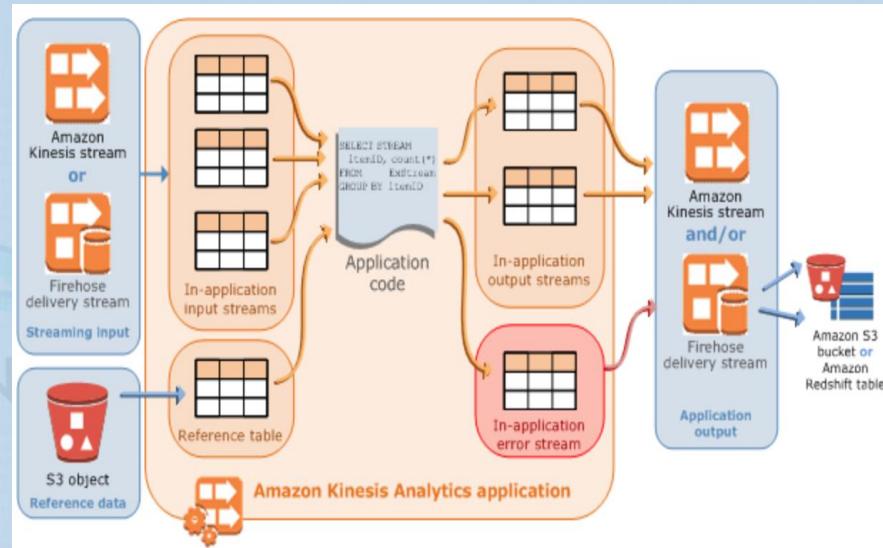
- Use **Amazon Kinesis Analytics** to process and analyze streaming data with standard SQL
- The service enables you to quickly author and run powerful SQL code against streaming sources
- The service supports ingesting data from Amazon Kinesis Streams and Amazon Kinesis Firehose streaming sources.
- You can also configure destinations where you want Amazon Kinesis Analytics to persist the results.
- Amazon Kinesis Analytics supports Amazon Kinesis Firehose (Amazon S3, Amazon Redshift, and Amazon Elasticsearch Service), and Amazon Kinesis Streams as destinations.



AWS Services

AWS Kinesis Analytics

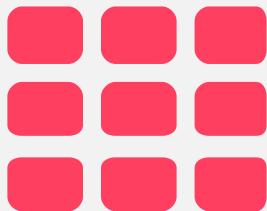
- Amazon Kinesis Analytics needs permissions to read records from a streaming source and write application output to the external destinations.
- You use IAM roles to grant these permissions.
- Note here the sources and the fact that the application is SQL code based



source:aws.amazon.com
DOLFIN

AWS Kinesis Analytics

- Amazon Kinesis Analytics enables you to quickly author SQL code that continuously reads, processes, and stores data in near real time.
- Using standard SQL queries on the streaming data, you can construct applications that transform and gain insights into your data.
- Use cases:
 - **Generate time-series analytics** – You can calculate metrics over time windows, and then stream values to Amazon S3 or Amazon Redshift through an Amazon Kinesis Firehose delivery stream.
 - **Feed real-time dashboards** – You can send aggregated and processed streaming data results downstream to feed real-time dashboards.
 - **Create real-time metrics** – You can create custom metrics and triggers for use in real-time monitoring, notifications, and alarms.



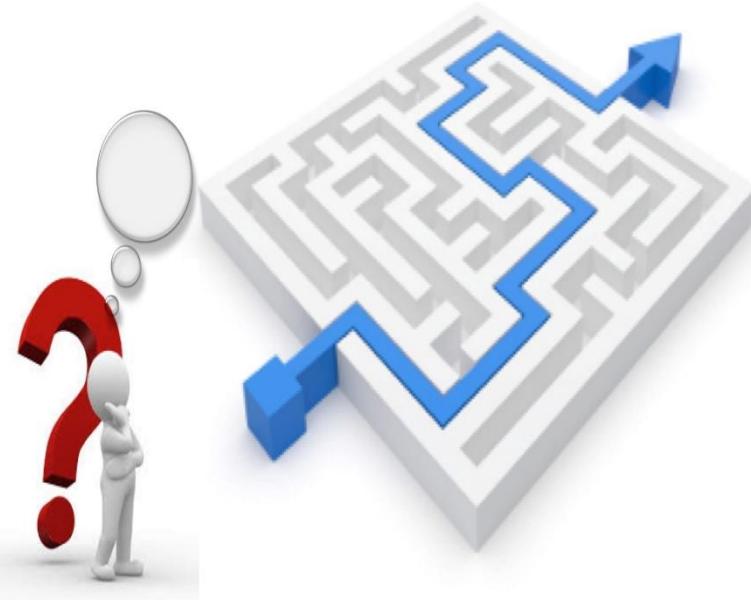
AWS SERVICES REDSHIFT

You Can Do It Too!



Amazon REDSHIFT

Introduction



AWS Services

Data Warehouse

- A data **warehouse** is a relational **database** that is designed for query and analysis rather than for transaction processing.
 - It usually contains historical data derived from transaction data, but it can include data from other sources.
 - To perform analytics you need a data warehouse not a regular database
- **OLAP** (On-line Analytical Processing) is characterized by relatively low volume of transactions. Queries are often very complex and involve aggregations.
- A data warehouse can be a layer on top of other OLTP databases
 - The data warehouse takes data from these database(s) and creates a layer that is optimum and dedicated to perform analytics
- RDS (MySQL..etc) is an **OLTP** database, where there is detailed and current data, and a schema used to store transactional data
 - Usually constrained to a single application



AWS Services : Redshift

- Redshift, is an AWS, fully managed, **Petabyte scale** data warehouse service in the cloud
- Amazon Redshift gives you fast querying capabilities **over structured data** using familiar SQL-based clients and business intelligence (BI) tools using standard ODBC and JDBC connections.
- Queries are distributed and parallelized across multiple physical resources.
- Amazon Redshift uses replication and continuous backups to enhance availability and improve data durability and can automatically recover from component and node failures.



source: aws.amazon.co

AWS Services : Redshift

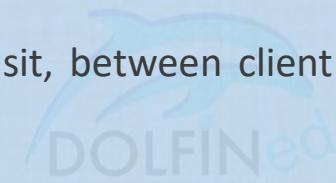
- Redshift is a SQL based data warehouse used for analytics applications (Analytics DB)
 - Example use cases: Sales Reporting, Health Care analytics
 - It is suited for OLAP-based use cases (On Line Analytics Processing)
 - **Can store huge amount of data (a database), but can't ingest huge amounts of data in real time** (not like what Kinesis can do)
- You can launch & Configure it from AWS Console or through AWS Redshift APIs
- Redshift can:
 - Fully recover from a node or component failure
 - It automatically patches and performs data backup
 - Backups can be stored for a user defined retention period
 - Is 10 times faster than traditional SQL RDBMS



AWS Services

Redshift – Data Security

- Supports encryption of data “at rest” using hardware accelerated AES-256 bits
 - By default, AWS Redshift takes care of encryption key management
 - You can choose to manage your own keys through HSM (Hardware Security Modules), or AWS KMS (Key Management Service)
- Supports SSL Encryption, in-transit, between client applications and Redshift data warehouse cluster
- You can’t have direct access to your AWS Redshift cluster nodes, however, you can through the applications themselves



AWS Services

Redshift Performance

- Redshift uses a variety of features to achieve much faster performance compared to traditional SQL DBs
 - Columnar Data Storage:
 - Data is stored sequentially in columns instead of rows
 - Columnar based DB is ideal for data warehousing and analytics
 - Requires far fewer I/Os, which greatly enhances performance
 - Advanced Compression
 - Data is stored sequentially in columns which allows for much better compression
 - Less storage space
 - Redshift automatically selects compression scheme
 - Massive Parallel Processing (MPP) : Data and query loads are distributed across all nodes

Redshift

- No upfront commitment, you can start small and grow as required
 - You can start with a single, 160GB, Redshift data warehouse node
- For a multi-node deployment (Cluster), you need a leader node and compute node(s)
 - The leader node manages client connections and receives queries
 - The compute nodes store data and perform queries and computations
 - You can have up to 128 compute nodes in a cluster



Redshift – Backup Retention

- Amazon Redshift automatically patches and backs up (Snapshots) your data warehouse, storing the backups for a user-defined retention period in AWS S3.
 - It keeps the backup by default for one day (24 hours) but you can configure it from 0 to 35 days
 - Automatic backups are stopped if you choose retention period of 0
 - You have access to these automated snapshots during the retention period
- If you delete the cluster,
 - You can choose to have a final snapshot to use later
 - Manual backups are not deleted automatically, if you do not manually delete them, you will be charged standard S3 storage rates
- AWS Redshift currently supports only one AZ (no Multi-AZ option)
- You can restore from your backup to a new Redshift cluster in the same or a different AZ
 - This is helpful in case the AZ hosting your cluster fails



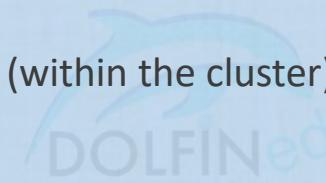
Redshift Monitoring

- Metrics for compute utilization, storage utilization, and read/write traffic to your Amazon Redshift data warehouse cluster, are available free of charge via the AWS console or AWS Cloudwatch APIs
- You can also add additional, user-defined, metrics via AWS CloudWatch custom metric functionality.



Redshift - Availability and Durability

- Redshift automatically **replicates all** your data within your data warehouse cluster
 - To other drives within the cluster to maintain copies of your original data
- Redshift always keeps three copies of your data:
 - The original one
 - A replica on compute nodes (within the cluster)
 - A backup copy on S3
- Cross Region Replication
 - Redshift can asynchronously replicate your snapshots to S3 in another region for DR



source: www.amazon.com

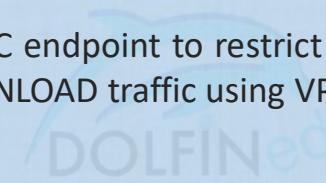
Redshift Spectrum

- Amazon Redshift Spectrum allows you to directly run SQL queries against exabytes of data in Amazon S3.
- You are charged for the number of bytes scanned by Redshift Spectrum, rounded up to the next megabyte, with a 10MB minimum per query.
- There are no charges for Data Definition Language (DDL) statements like CREATE/ALTER/DROP TABLE statements for managing partitions, and failed queries.
- With Redshift Spectrum, you can run multiple Amazon Redshift clusters accessing the same data in Amazon S3.
- You can use one cluster for standard reporting and another for data science queries. Your marketing team can use their own clusters different from your operations team.
- Redshift Spectrum automatically distributes the execution of your query to several Redshift Spectrum workers out of a shared resource pool to read and process data from Amazon S3, and pulls results back into your Amazon Redshift cluster for any remaining processing.
- Redshift Spectrum supports Amazon S3's Server Side Encryption (SSE) using your account's default key managed used by the AWS Key Management Service (KMS).



Redshift Enhanced VPC Routing

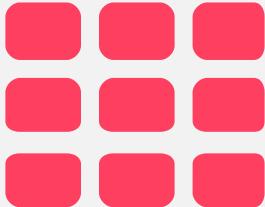
- Use Redshift's Enhanced VPC Routing to force all of the COPY and UNLOAD traffic to go through the VPC privately through endpoints.
- This way it becomes possible to tightly manage the flow of data between Amazon Redshift clusters and all of your data sources.
- You can also add a policy to your VPC endpoint to restrict unloading data only to a specific S3 bucket in your account, and monitor all COPY and UNLOAD traffic using VPC flow logs.



Redshift – Billing

You pay for:

- **Compute node hours** – Compute node hours are the total number of hours you run across all your compute nodes for the billing period. You are billed for 1 unit per node per hour. You are not charged for the leader node
- **Backup Storage** – Backup storage is the S3 storage associated with your automated and manual snapshots for your data warehouse.
 - Be careful if you change your backup retention period (extra charges)
- **Data transfer** – There is no data transfer charge for data transferred to, or from, Amazon Redshift and Amazon S3 within the same AWS Region.
 - For all other data transfers into and out of Amazon Redshift, you will be billed at standard AWS data transfer rates.



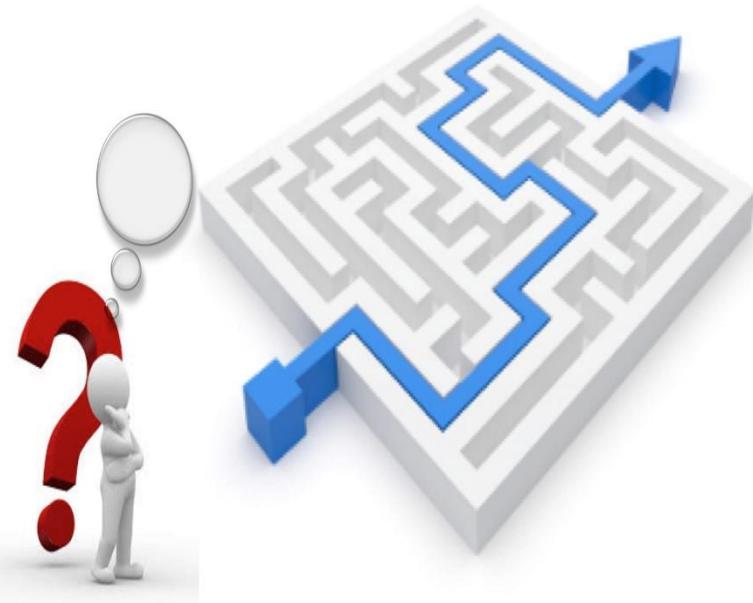
AWS SERVICES ELASTIC CONTAINER SERVICE (ECS)

You Can Do It Too!



Amazon Elastic Container Service

Introduction



AWS EC2 Container Service

AWS ECS

- Amazon Elastic Container Service (Amazon ECS) is a highly scalable, fast, container management service that makes it easy to run, stop, and manage Docker containers on a cluster.
- You can host your cluster on a **serverless infrastructure that is managed by Amazon ECS** by launching your services or tasks using the **Fargate launch type**.
- For more control you can host your tasks on a cluster of Amazon Elastic Compute Cloud (Amazon EC2) instances **that you manage** by using the EC2 launch type.
- Amazon ECS lets you:
 - Launch and stop container-based applications with simple API calls,
 - Allows you to get the state of your cluster from a centralized service,
 - Gives you access to many familiar Amazon EC2 features.

AWS EC2 Container Service

AWS ECS

- You can use Amazon ECS to schedule the placement of containers across your cluster based on your resource needs, isolation policies, and availability requirements.
- Amazon ECS eliminates the need for you to operate your own cluster management and configuration management systems or worry about scaling your management infrastructure.
- Amazon ECS can be used to create a consistent deployment and build experience, manage, and scale batch and Extract-Transform-Load (ETL) workloads, and build sophisticated application architectures on a microservices model.

AWS EC2 Container Service

AWS ECS

Features of Amazon ECS

- Amazon ECS is a **regional** service that simplifies running application containers in a highly available manner across multiple Availability Zones within a region.
- You can create Amazon ECS clusters within a new or existing VPC.
- After a cluster is up and running, you can **define task definitions and services** that specify which Docker container images to run across your clusters.
- Container images are stored in and pulled from container registries, which may exist within or outside of your AWS infrastructure.

AWS EC2 Container Service

AWS ECS

Containers and Images

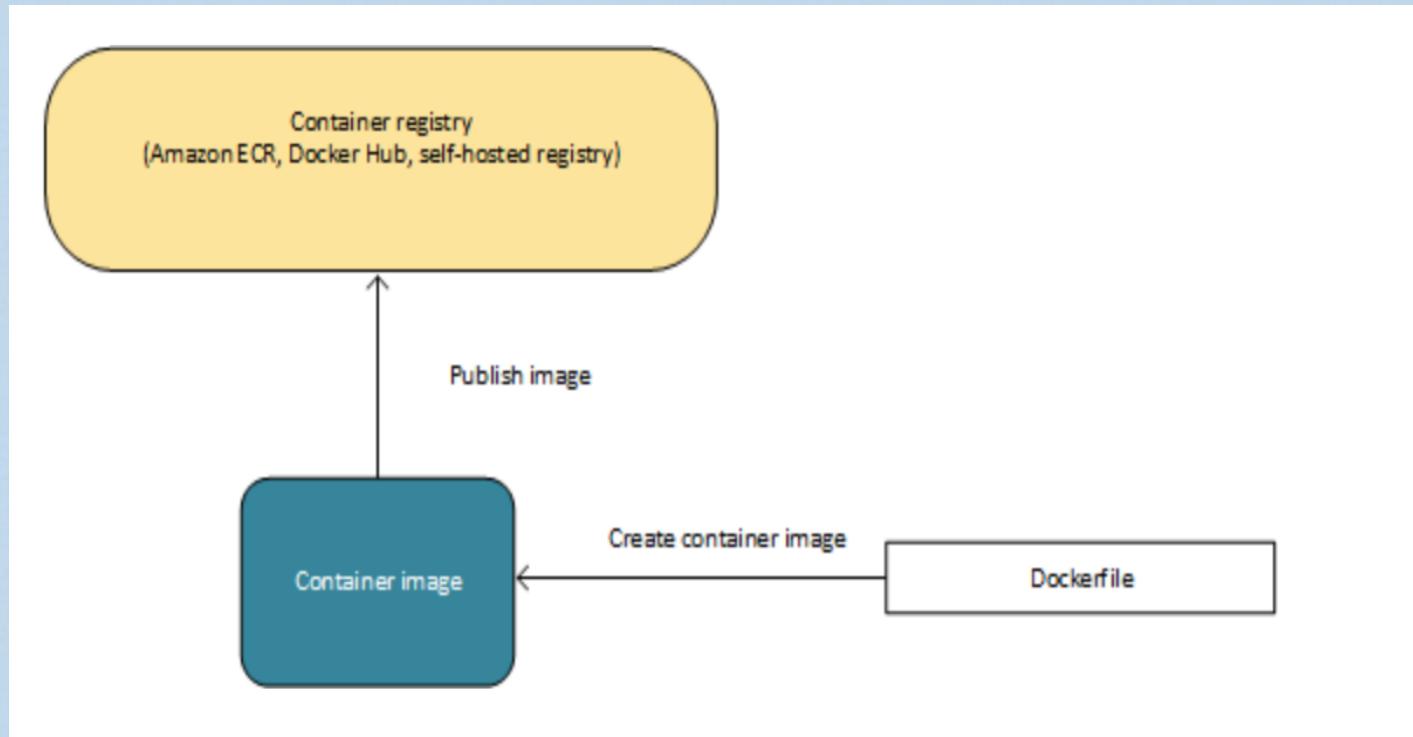
- To deploy applications on Amazon ECS, your application components must be architected to run in *containers*.
- A Docker container is a standardized unit of software development, containing everything that your software application needs to run: code, runtime, system tools, system libraries, etc. Containers are created from a read-only template called an *image*.
- Images are typically built from a Dockerfile, a plain text file that specifies all of the components that are included in the container. These images are then stored in a *registry* from which they can be downloaded and run on your cluster.

DOLFINed

AWS EC2 Container Service

AWS ECS

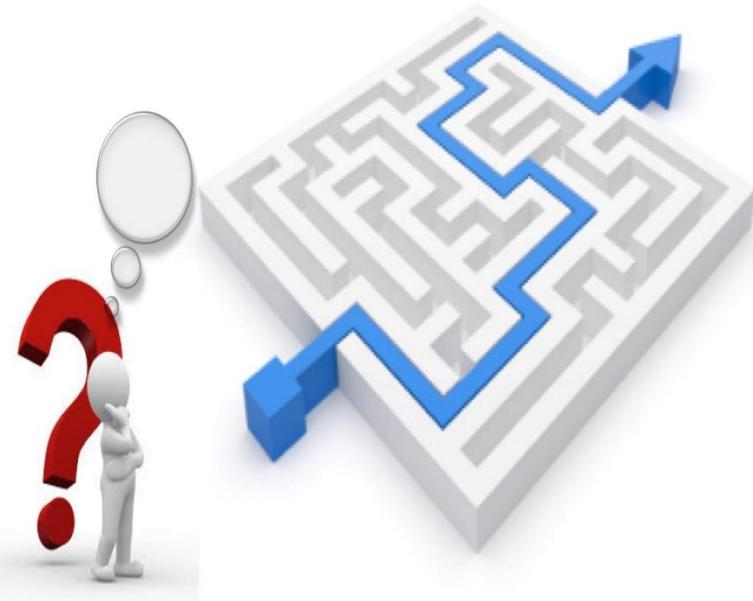
- Docker Image creation workflow:



source: aws.amazon.com

Amazon Elastic Container Service

ECS Launch Types



AWS EC2 Container Service

AWS ECS Launch Types

- **Fargate Launch Type**

The Fargate launch type allows you to run your containerized applications without the need to provision and manage the backend infrastructure. Just register your task definition and Fargate launches the container for you.

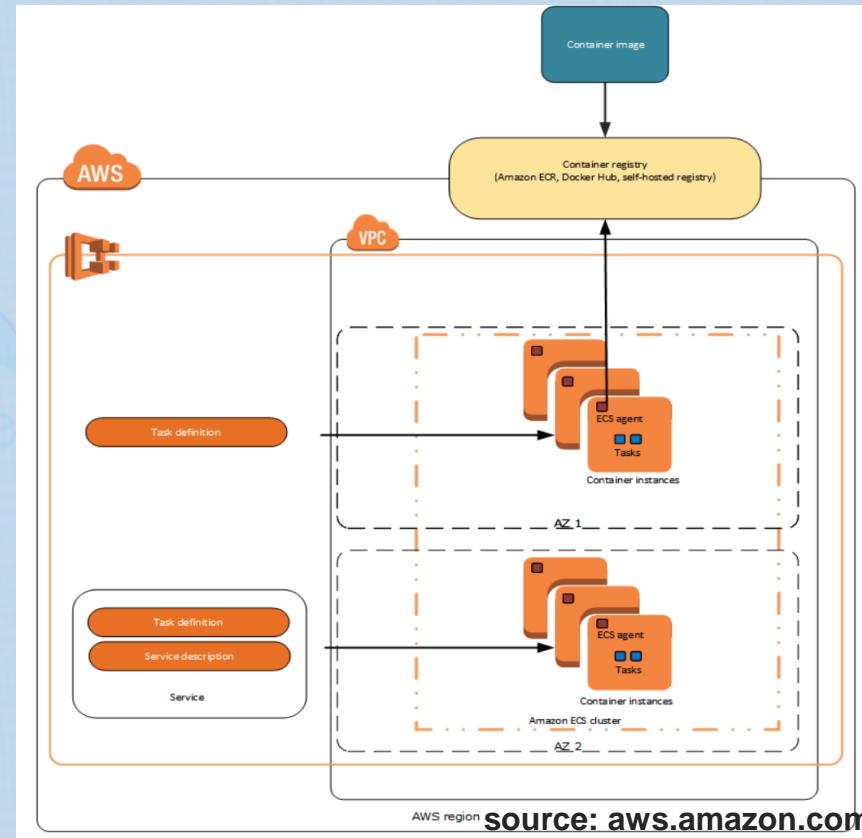
- You are hosting your cluster on a **serverless infrastructure that is managed by Amazon ECS** by launching your services or tasks using the **Fargate launch type**.

AWS EC2 Container Service

AWS ECS

Amazon ECS launch types:

- EC2 Launch Type**
- The EC2 launch type allows you to run your containerized applications on a cluster of Amazon EC2 instances that you manage.
 - Provides for more control
 - You need to manage the EC2 fleet (cluster)
- The Fargate launch type only supports using container images hosted in Amazon ECR or publicly on Docker Hub.
- Private repositories are currently only supported using the EC2 launch type.



source: aws.amazon.com

AWS EC2 Container Service

AWS ECS

Amazon ECS Container Instances

- An Amazon ECS container instance is an Amazon EC2 instance that is running the Amazon ECS container agent and has been registered into a cluster.
- When you run tasks with Amazon ECS, your tasks using the EC2 launch type are placed on your active container instances.
- **Note**
Tasks using the Fargate launch type are deployed onto AWS-managed infrastructure so this topic does not apply.



AWS EC2 Container Service

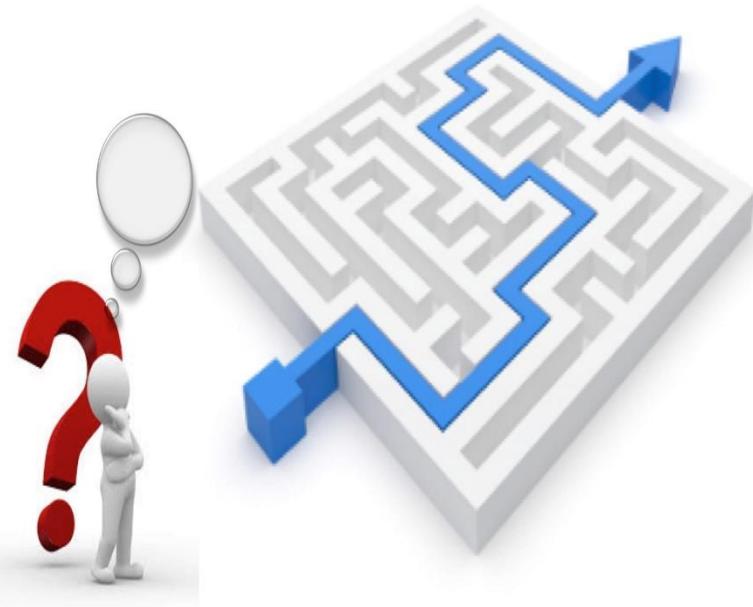
AWS ECS Clusters

- When you run tasks using Amazon ECS, you place them on a *cluster*, which is a logical grouping of resources.
- If you use the Fargate launch type with tasks within your cluster, Amazon ECS manages your cluster resources.
- If you use the EC2 launch type, then your clusters will be a group of container instances you manage.
- Amazon ECS downloads your container images from a registry that you specify, and runs those images within your cluster.



Amazon Elastic Container Service

ECS Task Definition & Tasks



AWS EC2 Container Service

AWS ECS – Task Definition

Task Definitions

- To prepare your application to run on Amazon ECS, you create a *task definition*.
- The task definition is a text file, in JSON format, that describes one or more containers, up to a **maximum of ten**, that form your application. It can be thought of as a blueprint for your application.
- Amazon ECS task definitions use Docker images to launch containers on the container instances in your clusters.
- An ECS Task:
 - Is the instantiation of a task definition within an ECS cluster.
 - After you have created a task definition for your application within Amazon ECS, you can specify the number of tasks that will run on your cluster.

source: [aws.amazon.co](https://aws.amazon.com)


AWS EC2 Container Service

AWS ECS Container Registry (AWS ECR)

Amazon Elastic Container Registry

- Amazon ECR is a managed AWS Docker registry service. Customers can use the familiar Docker CLI to push, pull, and manage images.

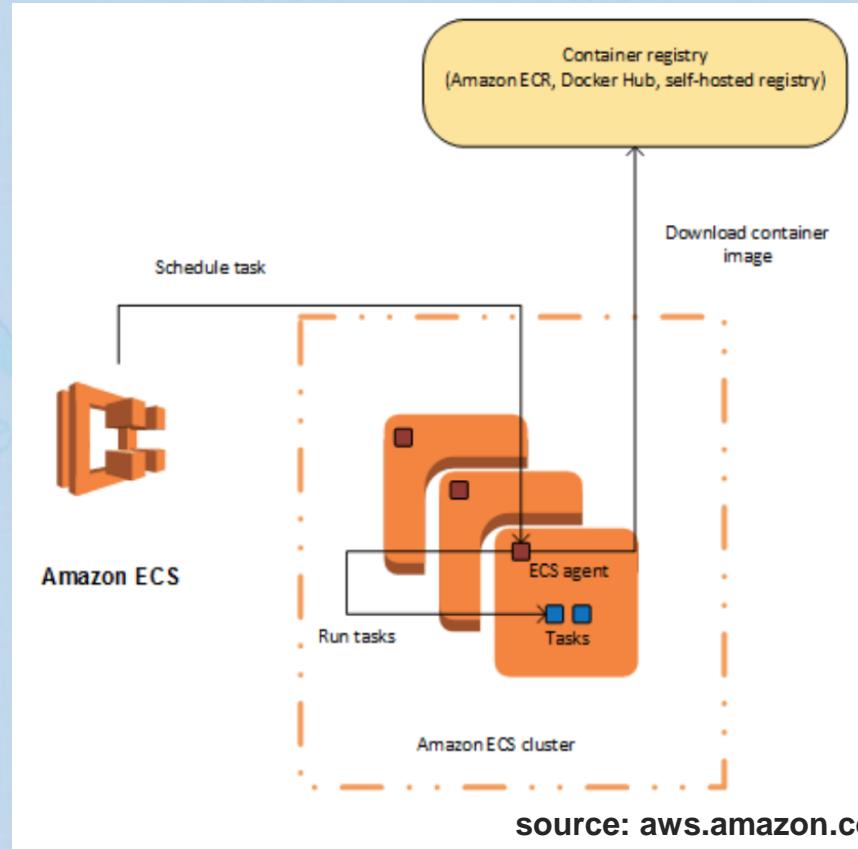


source: [aws.amazon.co](https://aws.amazon.com)
The AWS logo consists of a stylized blue and white cloud-like shape followed by the word "AWS" in a bold, sans-serif font.

AWS EC2 Container Service

AWS ECS – Container Agent

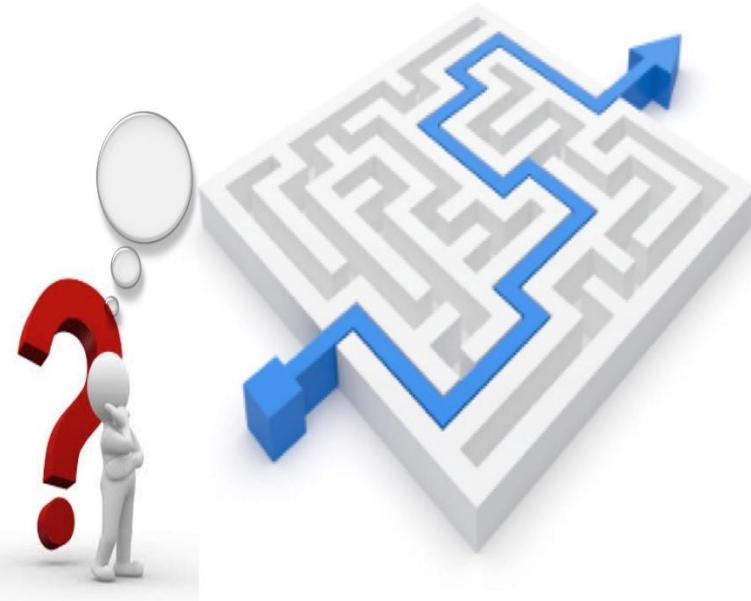
- The *container agent* runs on each infrastructure resource within an Amazon ECS cluster.
- It sends information about the resource's current running tasks and resource utilization to Amazon ECS, and starts and stops tasks whenever it receives a request from Amazon ECS.



source: [aws.amazon.co](http://aws.amazon.com)

Amazon Elastic Container Service

ECS Task definitions and IAM Roles

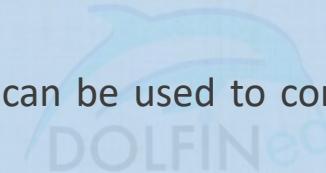


AWS EC2 Container Service

AWS ECS – IAM Roles and Task Roles

Amazon ECS IAM Policies, Roles, and Permissions

- Amazon ECS container instances make calls to the Amazon ECS and Amazon EC2 APIs on your behalf, so they need to authenticate with your credentials.
 - This authentication is accomplished by creating an IAM role for your container instances and associating that role with your container instances when you launch them.
- Basically, in Amazon ECS, IAM can be used to control access at the container instance level using IAM roles
 - This is required if you are to control using ECS Launch type



AWS EC2 Container Service

AWS ECS – IAM Roles for Tasks

- You must create an IAM policy for your tasks to use that specifies the permissions that you would like the containers in your tasks to have.
 - You must also create a role for your tasks to use before you can specify it in your task definitions.
- You can create the role using the **Amazon EC2 Container Service Task Role** service role in the IAM console.



AWS EC2 Container Service

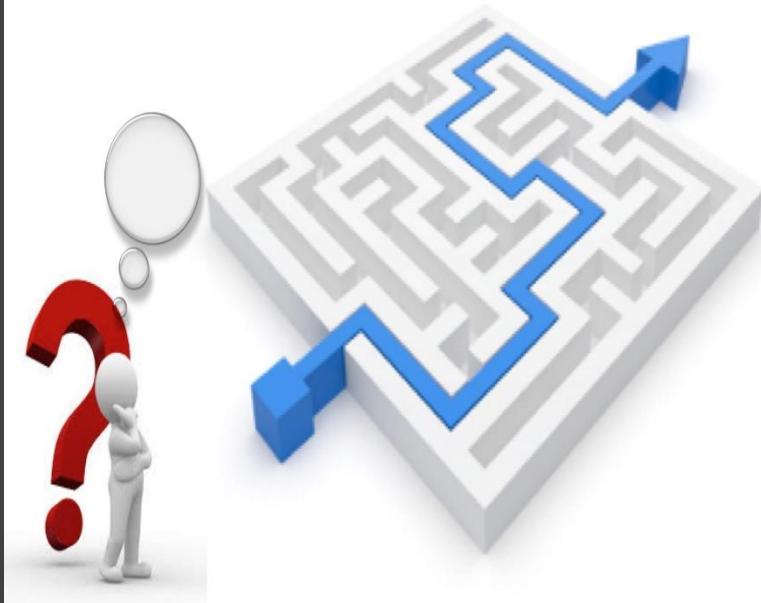
AWS ECS

Benefits of Using IAM Roles for Tasks

- **Credential Isolation:** A container can only retrieve credentials for the IAM role that is defined in the task definition to which it belongs; a container never has access to credentials that are intended for another container that belongs to another task.
- **Authorization:** Unauthorized containers cannot access IAM role credentials defined for other tasks.
- **Auditability:** Access and event logging is available through CloudTrail to ensure retrospective auditing.
- Task credentials have a context of taskArn that is attached to the session, so CloudTrail logs show which task is using which role.

Amazon Elastic Container Service

ECS Task Definitions and ALB



AWS Application Load Balancer

Load balancing to ECS hosted Micro-Services

Microservices as Targets with Your Application Load Balancer

- You can use a micro-services architecture to structure your application as services that you can develop and deploy independently.
- You can install one or more of these services on each EC2 instance, with each service accepting connections on a different port.
- You can use a single Application Load Balancer to route requests to all the services for your application.
- When you register an EC2 instance with a target group, you can register it multiple times;
 - For each service, register the instance using the port for the service.



AWS Application Load Balancer

Load balancing to ECS hosted Micro-Services

An ECS Services is:

- A service allows you to run and maintain a specified number (the "desired count") of simultaneous instances of a task definition in an ECS cluster.
 - Underneath which we define the desired number of Tasks to be run

Service Load Balancing

- Your Amazon ECS service can optionally be configured to use Elastic Load Balancing to distribute traffic evenly across the tasks in your service.
- Application Load Balancers offer several features that make them particularly attractive for use with Amazon ECS services:
 - Application Load Balancers allow containers to use dynamic host port mapping
 - Such that multiple tasks from the same service are allowed per container instance



AWS Application Load Balancer

AWS ALB

- Application Load Balancers support path-based routing and priority rules
 - Such that multiple services can use the same listener port on a single Application Load Balancer.
- The Application Load Balancer **integrates with EC2 Container Service (ECS) using Service Load Balancing.**
 - ECS Instances can be registered with the ALB using multiple ports
 - This allows for requests to be routed to multiple containers on a single container instance
 - Amazon ECS will automatically register tasks with the ALB using a dynamic container-to-host port mapping
 - This allows for dynamic mapping of services to ports as specified in the ECS task definition.
 - The ECS task scheduler will automatically add these tasks to the ALB.



AWS Application Load Balancer

AWS ALB – ALB and ECS Services

Application Load Balancers offer several features that make them particularly attractive for use with Amazon ECS services:

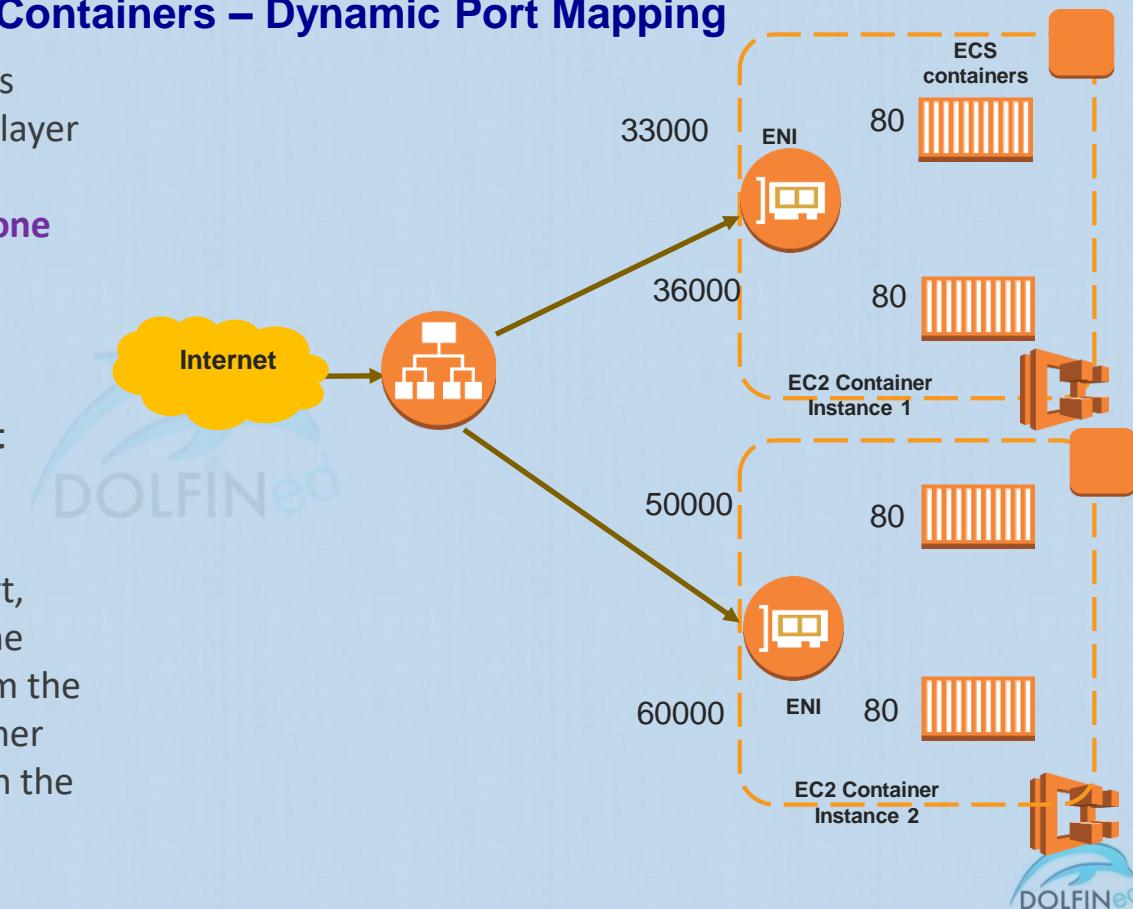
- Application Load Balancers allow containers to use dynamic host port mapping
 - Such that **multiple tasks (using the same port) from the same service are allowed per container instance**
 - You can use dynamic port mapping to support multiple tasks from a single service on the same container instance.
- Application Load Balancers support path-based routing and priority rules, such that **multiple services can use the same listener port** on a single Application Load Balancer
- In dynamic port mapping, **Amazon ECS manages updates to your services by automatically registering and deregistering containers with the ALB using the instance ID and port for each container.**

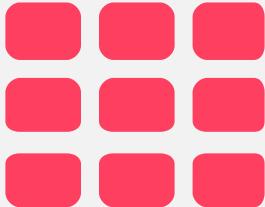


AWS Application Load Balancer

AWS ALB with Containers – Dynamic Port Mapping

- An Application Load Balancer makes routing decisions at the application layer (HTTP/HTTPS), supports path-based routing, and **can route requests to one or more ports on each container instance in your cluster.**
- Application Load Balancers support dynamic host port mapping.**
- If your task's container definition specifies port 80 for a container port, and port 0 for the host port, then the host port is dynamically chosen from the ephemeral port range of the container instance (such as 32768 to 61000 on the latest Amazon ECS-optimized AMI).





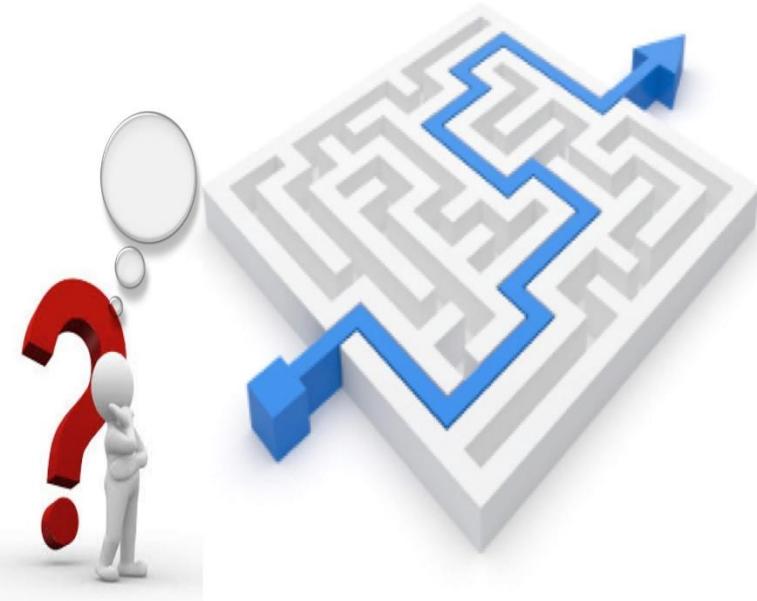
IDENTITY AND ACCESS MANAGEMENT (IAM)

You Can Do It Too!



Identity and Access Management

Introduction to IAM



Review Topic : AWS IAM

IAM

- AWS Identity and Access Management (IAM) is a web service that helps you securely control access to AWS resources. You use IAM to control who is authenticated (signed in) and authorized (has permissions) to use resources.
 - IAM provides the infrastructure necessary to control authorization and access control for your account.



Review Topic : AWS IAM

IAM Features

- **Shared access to your AWS account**
 - You can grant other people permission to administer and use resources in your AWS account without having to share your access credentials (password or access key).
- **Granular permissions**
 - You can grant different permissions to different people for different resources.
- **Secure access to AWS resources for applications that run on AWS**

Review Topic : AWS IAM

IAM Features

- **Multi-factor authentication (MFA)**
 - You can add two-factor authentication to your account and to individual users for extra security.
 - With MFA you or your users must provide not only a password or access key to work with your account, but also a code from a specially configured device.
- **Identity federation**
 - You can allow users who already have passwords elsewhere—for example, in your corporate network or with an internet identity provider—to get temporary access to your AWS account.
- **Identity information for assurance**
 - If you use AWS CloudTrail, you receive log records that include information about those who made requests for resources in your account. That information is based on IAM identities.

source: aws.amazon.co

Review Topic : AWS IAM

IAM Features

- **PCI DSS Compliance**
 - IAM supports the processing, storage, and transmission of credit card data by a merchant or service provider, and has been validated as being compliant with Payment Card Industry (PCI) Data Security Standard (DSS).
- **Integrated with many AWS services**
- **Eventually Consistent**
 - IAM, like many other AWS services, is Eventually consistent.
 - IAM achieves high availability by replicating data across multiple servers within Amazon's data centers around the world.



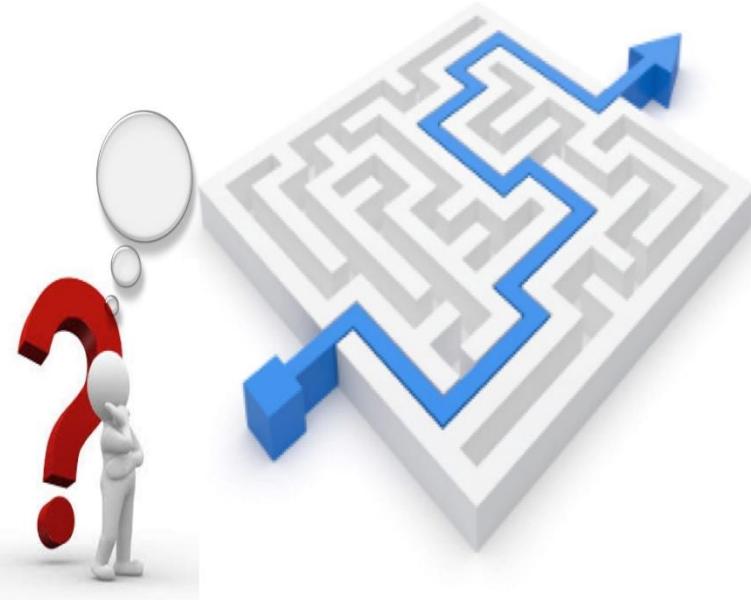
Review Topic : AWS IAM

IAM Features

- **Free to use**
 - AWS Identity and Access Management is a feature of your AWS account offered at no additional charge.
 - You will be charged only for use of other AWS products by your IAM users.
- **AWS Security Token Service**
 - Is an included feature of your AWS account offered at no additional charge.
 - You are charged only for the use of other AWS services that are accessed by your AWS STS temporary security credentials.

Identity and Access Management

IAM Identities



Review Topic : AWS IAM

IAM Identities

Identities (Users, Groups, and Roles)

- *IAM identities*, is what you create under your AWS account to provide authentication for people, applications, and processes in your AWS account.
- Identities represent the user, and can be authenticated and then authorized to perform actions in AWS.
- Each of these can be associated with one or more *Policies* to determine what actions a user, role, or member of a group can do with which AWS resources and under what conditions.
- *IAM groups*, which are collections of IAM users that you can manage as a unit.
- *IAM roles* is very similar to an IAM user but has no password or access keys assigned to it.

source: aws.amazon.com



Review Topic : AWS IAM

IAM Identities – IAM Users

IAM Users:

- An IAM user is an entity that you create in AWS. It represents the person or service who uses the IAM user to interact with AWS.
 - An IAM user can represent an **actual person or an application** that requires AWS access to perform actions on AWS resources
- For any user you can assign them:
 - A **username and password** to access the AWS console
 - An **access key ID (Access key and Secret Key)** that they can use for **programmatic access (issuing requests)** to your AWS resources

Review Topic : AWS IAM

IAM Groups

IAM Groups:

- An IAM Group is a collection of IAM users
- It is a way to assign permissions/policies to multiple users at once
- A group is not truly an "identity" in IAM because it cannot be identified as a Principal in a permission policy.
- Groups can't be nested; they can contain only users, not other groups.

Review Topic : AWS IAM

IAM Identities – IAM Roles

IAM Roles:

- An IAM Role is very similar to a user, in that **it is an identity with permission policies** that determine what the identity can and cannot do in AWS.
- **An IAM Role does not have any credentials (password or access keys) associated with it.**
- Instead of being uniquely associated with one person, a role is intended to be assumable by anyone who needs it (and it authorized to use it).
 - An IAM user can assume a role to temporarily take on different permissions for a specific task.
 - An IAM Role can be assigned to a Federated user who signs in by using an external identity provider instead of IAM.
 - AWS uses details passed by the identity provider to determine which role is mapped to the federated user.

Review Topic : AWS IAM

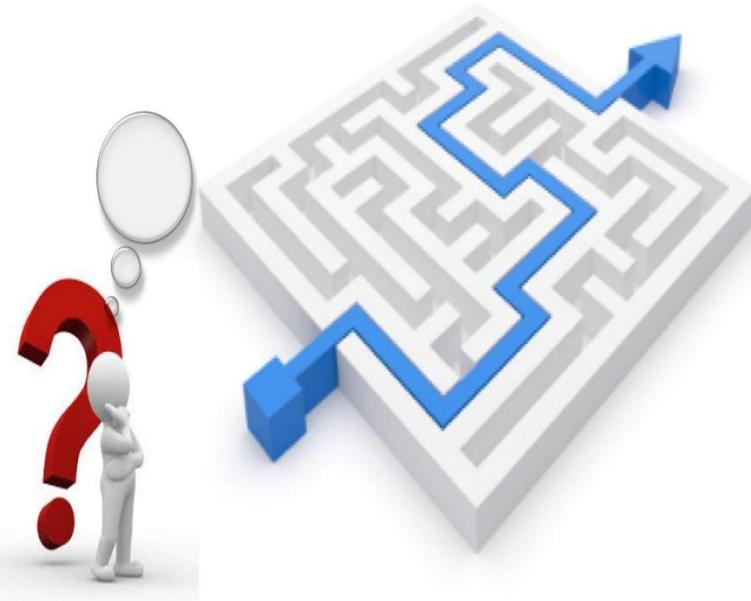
IAM Temporary Credentials through Secure Token Service (STS)

IAM Temporary Credentials:

- Temporary credentials are primarily used with IAM roles, but there are also other uses.
 - You can request temporary credentials that have a more restricted set of permissions than your standard IAM user.
 - This prevents you from accidentally performing tasks that are not permitted by the more restricted credentials.
- A benefit of temporary credentials is that they expire automatically after a set period of time.
 - You have control over the duration that the credentials are valid.

Identity and Access Management

IAM User Credentials



Review Topic : AWS IAM

Users and Credentials

You can access AWS in different ways depending on the user credentials:

- **Console Password:**
 - A password that the user can type to sign in to interactive sessions such as the AWS Management Console.
- **Access Keys:**
 - A combination of an access key ID and a secret access key.
 - You can assign two to a user at a time.
 - These can be used to make programmatic calls to AWS when using the API in program code or at a command prompt when using the AWS CLI or the AWS PowerShell tools.

Review Topic : AWS IAM

Users and Credentials

You can access AWS in different ways depending on the user credentials:

- **Server Certificates:**
 - SSL/TLS certificates that you can use to authenticate with some AWS services.
 - AWS recommends that you use AWS Certificate Manager (ACM) to provision, manage, and deploy your server certificates.
 - Use IAM only when you must support HTTPS connections in a region that is not supported by ACM.
- **SSH access keys for use with AWS CodeCommit:**
 - An SSH public key in the OpenSSH format that can be used to authenticate with AWS CodeCommit.

Review Topic : AWS IAM

IAM

Create credentials for the user, depending on the type of access the user requires:

- **Programmatic access:**
 - The IAM user might need to make API calls or use the AWS CLI or the Tools for Windows PowerShell.
 - In that case, create an access key (an access key ID and a secret access key) for that user.
- **AWS Management Console access:**
 - If the user needs to access AWS resources from the AWS Management Console, create Create a password for the user.
- As a best practice, do not create credentials of a certain type for a user who will never need that kind of access.

Review Topic : AWS IAM

IAM Users – Password Management

- You can
 - Allow all IAM users in the account to change their own password
 - Allow only a selected IAM users to change their own password
 - For this case, Disable the option for all users to change their own passwords and you use an IAM policy to grant permissions to only some users to change their own passwords and optionally other credentials like their own access keys.



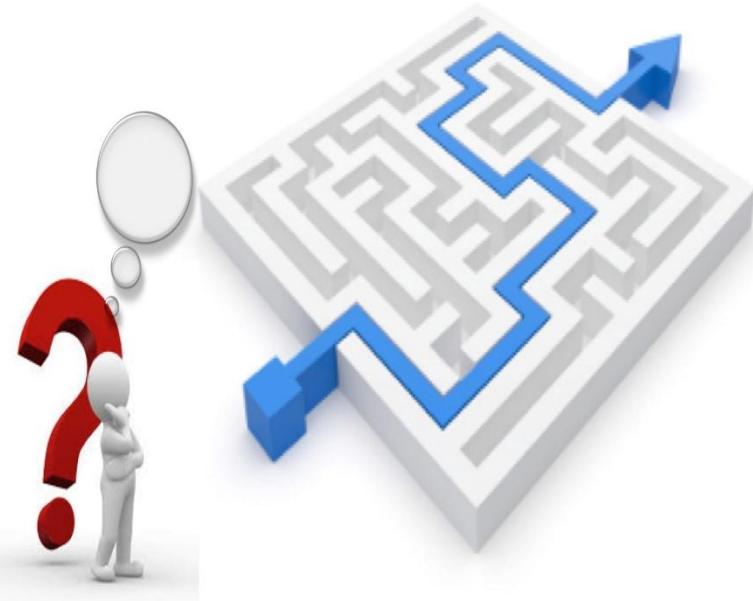
Review Topic : AWS IAM

IAM Users – Access Keys

- Users need their own access keys to make programmatic calls to AWS from the HTTPS API, CLI, SKDs, Tools for Windows PowerShell
- You can create, modify, view, or rotate access keys (access key IDs and secret access keys) for IAM users.
- When you create an access key, IAM returns the access key ID and secret access key.
- You should save these in a secure location and give them to the user.
 - To ensure the security of your AWS account, the secret access key is accessible only at the time you create it.
 - If a secret access key is lost, you must delete the access key for the associated user and create a new key.

Identity and Access Management

IAM Elements



Review Topic : AWS IAM

Accessing IAM

- You can access IAM via:
 - **AWS Console** (Requires a username and password for access), and
 - **Programmatic access** through SDKs, HTTPS API, and CLI (require the Access Key and Secret Access key).



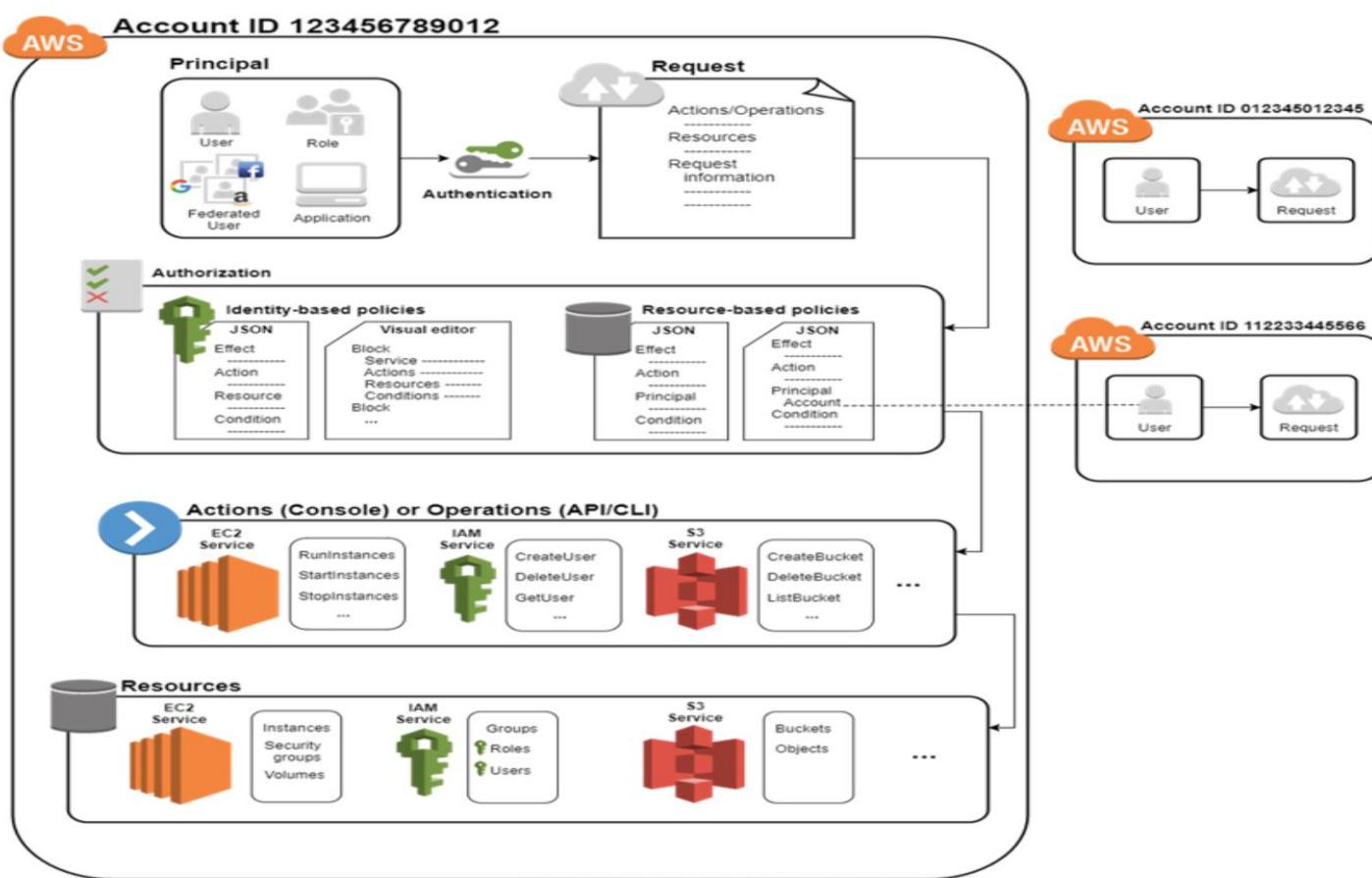
Review Topic : AWS IAM

IAM Elements

- The IAM infrastructure includes the following elements:
 - Principal
 - Request
 - Authentication
 - Authorization
 - Actions
 - Resources



Review Topic : AWS IAM



source: aws.amazon.com

Review Topic : AWS IAM

IAM Elements - Principal

Principal:

- A principal is an entity that can take an action on an AWS resource.
- Your administrative IAM user is your first *principal*.
- You can allow users and services to assume a role.
- You can support federated users or programmatic access to allow an application to access your AWS account.
- IAM Users, Roles, Federated users, and Applications **are all AWS principals**.
- IAM Groups can not be used as a Principal in an IAM Policy

Review Topic : AWS IAM

IAM Elements - Request

A Request:

- When a principal sends a request via Console, CLI, SDKs, or APIs including the following:
 - Actions (or operations) that the principal wants to perform
 - Resources upon which the actions are performed
 - Principal information, including the environment from which the request was made

Request Context:

- Before AWS can evaluate and authorize a request, AWS gathers the request information:
 - Principal (the requester), which is determined based on the authorization data.
 - This includes the aggregate permissions that are associated with that principal.
 - Environment data, such as the IP address, user agent, SSL enabled status, or the time of day.
 - Resource data, or data related to the resource that is being requested.
 - This can include information such as a DynamoDB table name, S3 bucket, or a tag on an Amazon EC2 instance.

Review Topic : AWS IAM

IAM Elements - Authentication

- A principal sending a request must be authenticated (signed in to AWS) to send a request to AWS.
 - Some AWS services, like AWS S3, allow requests from anonymous users.
- To authenticate **from the console**, you must sign in with your **user name and password**.
- To authenticate from the **API or CLI**, you must provide your **access key and secret key**.
- You might also be required to provide additional security information.
 - AWS recommends that you use multi-factor authentication (MFA) to increase the security of your account.

Review Topic : AWS IAM

IAM Elements - Authorization

- To Authorize requests, IAM uses values from the request context to check for matching policies and determine whether to allow or deny the request.
 - IAM policies are stored in IAM as JSON documents and specify the permissions that are allowed or denied
 - *user (Identity) -based policies*, specify permissions allowed/denied for principals
 - *Resource-based policies*, specify permissions allowed/denied for resources
- **Evaluation Logic**
 - IAM checks each policy that matches the context of your request.
 - If a single policy includes a denied action, IAM denies the entire request and stops evaluating. This is called an *explicit deny*.
 - Requests are *denied by default, thus*, IAM authorizes your request only if every part of your request is allowed by the matching policies.



Review Topic : AWS IAM

IAM Elements - Authorization

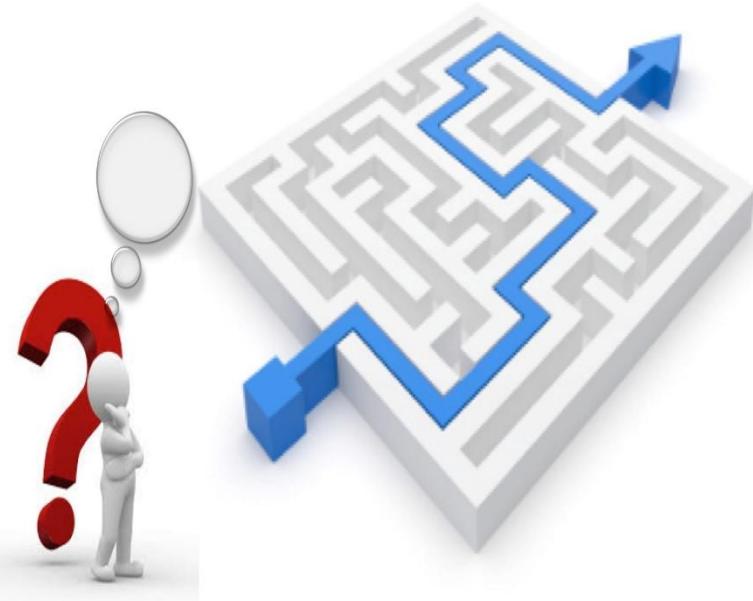
- The evaluation logic follows these rules:
 - By default, all requests are denied.
 - An explicit allow overrides this default.
 - An explicit deny overrides any allows.

Note

- By default, only the AWS account root user has access to all the resources in that account.
 - So if you are not signed in as the root user, you must have permissions granted by a policy.

Identity and Access Management

IAM Elements



Review Topic : AWS IAM

IAM Policies – Creating one

You can create a new IAM policy in the AWS Management Console using one of the following methods:

- **Import** – You can import a managed policy within your account and then edit the policy to customize it to your specific requirements.
 - A managed policy can be an AWS managed policy, or a customer managed policy that you created previously.
- **Visual editor** – You can construct a new policy from scratch in the visual editor. If you use the visual editor, you do not have to understand JSON syntax.
- **JSON** – In the **JSON** tab, you can create a policy using JSON syntax.

Review Topic : AWS IAM

IAM Elements - Actions

- Actions are defined by a service, and are the things that you can do to a resource, such as viewing, creating, editing, and deleting that resource.
- Any actions or resources that are not explicitly allowed are denied by default.
- After your request has been authenticated and authorized, AWS approves the actions in your request.
- IAM supported actions for a user resource include,
 - CreateUser, DeleteUser, GetUser, UpdateUser
- To allow a principal to perform an action, you must include the necessary actions in a policy that applies to the principal or the affected resource.

Review Topic : AWS IAM

IAM Elements - Resources

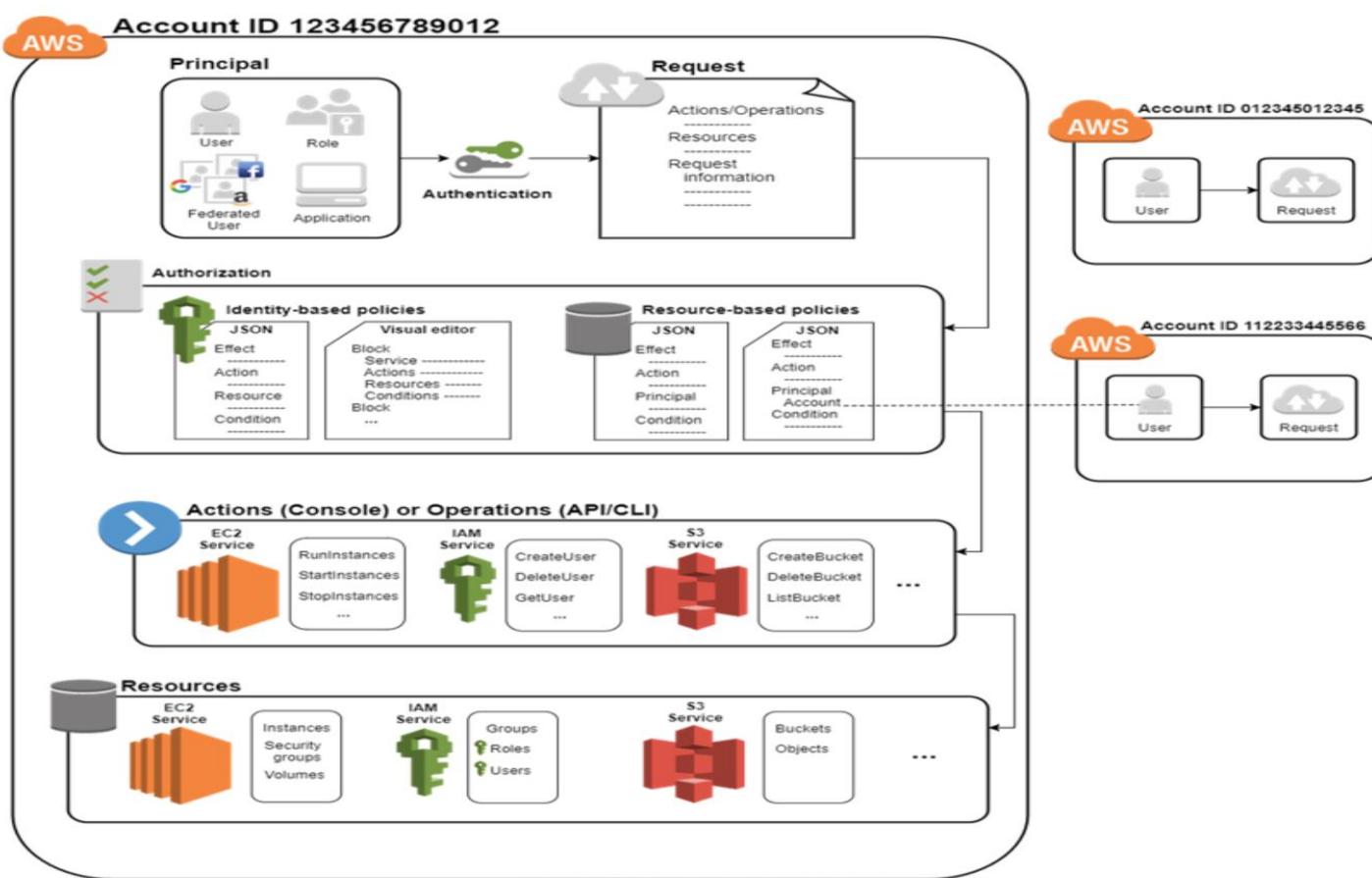
- A resource is an entity that exists within a service.
 - Examples are, EC2 instances, S3 bucket, IAM user, DynamoDB table
 - Each AWS service defines a set of actions that can be performed on each resource.
- After AWS approves the actions in your request, those actions can be performed on the related resources within your account.
- If you create a request to perform an unrelated action on a resource, that request is denied.
 - For example, if you request to delete an IAM role but provide an IAM group resource, the request fails.
- When you provide permissions using an **identity-based policy in IAM**, then you provide permissions to access resources only within the same account.

Review Topic : AWS IAM

IAM Cross-Account Access

- Identity-based policies is not the way to provide cross-account access
- If you need to make a request in a different account (Account B),
 - The resource in that account must have an attached resource-based policy that allows access from your account.
 - Otherwise, you must assume a role (Identity based policy) within that account with the permissions that you need.
 - This role will be created by the other account to allow you required access
 - The IAM user (you as the requester interested to access the resource in the other account) will make an API call to assume that role in the other AWS account (Account B)

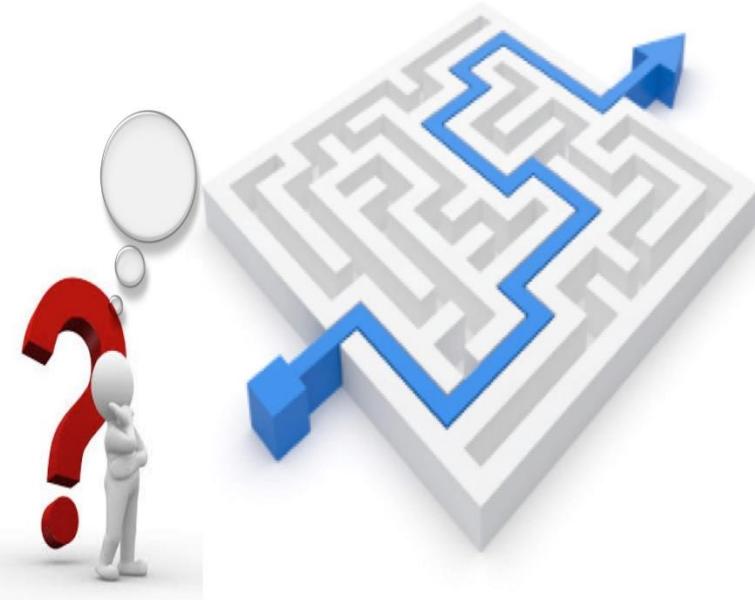
Review Topic : AWS IAM



source: aws.amazon.com

Identity and Access Management

IAM Authorization and Policies
User-based & Resource-based Policies



Review Topic : AWS IAM

IAM – Access Management

Permissions and Policies

- The access management portion of AWS Identity and Access Management (IAM) helps you to define what a user or other entity is allowed to do in an account, often referred to as *authorization*.
- Permissions are granted through policies that are created and then attached to users, groups, or roles.

Policies and Users

- By default, IAM users can't access anything in your account.
- You grant permissions to a user by creating a *policy*, which is a document that defines the effect, actions, resources, and optional conditions.
- **Any actions or resources that are not explicitly allowed are denied by default.**

source: aws.amazon.com

Review Topic : AWS IAM

IAM – Access Management – User-based policies

The following example shows a policy that grants permission to Perform Amazon DynamoDB actions (dynamodb:*) on the Books table in the account 123456789012 within the us-east-2 region.

```
{  
    "Version": "2012-10-17",  
    "Statement": {  
        "Effect": "Allow",  
        "Action": "dynamodb:*",  
        "Resource": "arn:aws:dynamodb:us-east-2:123456789012:table/Books"  
    }  
}
```

- When you attach this policy to a user, group, or role, then the IAM entity has those DynamoDB permissions.
- Typically, users in your account have multiple policies that together represent the permissions for that user.
- Notice that there is no Principal in the policy, since it gets attached to a user or Role



source: aws.amazon.com

Review Topic : AWS IAM

IAM – Multiple Policies

- Users or groups can have multiple policies attached to them that grant different permissions.
 - In the case of multiple policies attached to a user (or a group)
 - The users' permissions are calculated based on the combination of policies.
 - The basic principle still applies: If the user has not been granted an explicit permission for an action and a resource, the user does not have those permissions.



Review Topic : AWS IAM

IAM – Resource Based Policies

- In some cases (Like S3 buckets), you can attach a policy to a resource in addition to attaching it to a user or group. This is called a **resource-based policy**
- A *resource-based policy* contains slightly different information than a user-based policy.
 - In a resource-based policy you specify what actions are permitted and what resource is affected.
 - You **also explicitly list who is allowed access to the resource (a Principal)**
 - In a user-based policy, the "who" is established by whomever the policy is attached to.
- Resource-based policies include a Principal element that specifies who is granted the permissions.

Review Topic : AWS IAM

IAM – Resource Based Policies

An S3 bucket policy that allows an IAM user named bob in AWS account 777788889999 to put objects into the bucket called example-bucket. The **Principal element is set to** the Amazon Resource Name (ARN) of the IAM user bob. This indicates that the resource (in this case, the S3 bucket) is accessible to that IAM user but no one else.

```
{  
    "Version": "2012-10-17",  
    "Statement": {  
        "Effect": "Allow",  
        "Principal": {"AWS": "arn:aws:iam::777788889999:user/bob"},  
        "Action": [  
            "s3:PutObject",  
            "s3:PutObjectAcl"  
        ],  
        "Resource": "arn:aws:s3:::example-bucket/*"  
    }  
}
```

source: [aws.amazon.co](https://aws.amazon.com)

Identity and Access Management

IAM Roles, IAM Service Roles & IAM Service-linked Roles



Review Topic : AWS IAM

IAM Roles

- An **IAM Role**, *is a set of permissions* that grant access to actions and resources in AWS.
 - These permissions *are attached to the role, not to an IAM user or group.*
 - instead of being uniquely associated with one person, a role is intended to be assumable by anyone who needs it.
- A role does not have standard long-term credentials (password or access keys) associated with it.
 - Instead, if a user assumes a role, Temporary Security Credentials are created dynamically and provided to the user.
- Roles can be assumed/used by any of the following:
 - An IAM user in the same, or different, AWS account as the role
 - A web service offered by AWS such as Amazon EC2
 - An external user authenticated by an external identity provider (IdP) service that is compatible with SAML 2.0 or OpenID Connect (OIDC), or a custom-built identity broker.

Review Topic : AWS IAM

IAM – Assuming/Using a Role

There are two ways to assume/use a role:

- **Interactively in the IAM console,**
 - IAM users in your account using the IAM console can *switch to* a role to temporarily use the permissions of the role in the console.
 - The users give up their original permissions and take on the permissions assigned to the role.
 - When the users exit the role, their original permissions are restored.
- **Programmatically with the AWS CLI, Tools for Windows PowerShell, or API.**
 - An application or a service offered by AWS (like Amazon EC2) can *assume* a role by requesting temporary security credentials for a role with which to make programmatic requests to AWS.
 - You use a role this way so that you don't have to share or maintain long-term security credentials (for example, by creating an IAM user) for each entity that requires access to a resource.

Review Topic : AWS IAM

IAM Role and Resource Based Policies – The difference

- A **resource-based policy** specifies who (in the form of a *list of AWS account ID numbers*) can access that resource.
- Cross-account access with a **resource-based policy** has an advantage over that with an **IAM role**.
 - With a resource that is accessed through a resource-based policy, the user still works in the trusted account and does not have to give up his or her user permissions in place of the role permissions.
 - In other words, the user continues to have access to resources in the trusted account at the same time as he or she has access to the resource in the trusting account.
 - This is useful for tasks such as copying information to or from the shared resource in the other account.
- The disadvantage is that not all services support resource-based policies.



source: aws.amazon.co

Review Topic : AWS IAM

IAM Role – Service Roles

Creating a Role to Delegate Permissions to an AWS Service

- Many AWS services require that you use roles to control what that service can access.
- **AWS service role**
 - Is a role that a service assumes to perform actions on your behalf.
 - When you set up most AWS service environments, you must define a role for the service to assume.
 - This service role must include all the permissions required for the service to access the AWS resources that it needs.
 - Service roles vary from service to service, but many allow you to choose your permissions, as long as you meet the documented requirements for that service.
 - You can create, modify, and delete a service role from within IAM.

Review Topic : AWS IAM

IAM Role for EC2 instances

- Roles don't have their own permanent set of credentials the way IAM users do.
- You can specify a role for the instance at launch or after.
 - Applications that run on the EC2 instance can use the role's credentials when they access AWS resources.
 - The role's permissions determine what the application can do.
 - In case of Amazon EC2, AWS IAM automatically provides temporary security credentials that are attached to the role and then makes them available for the EC2 instance to use on behalf of its applications.
 - The temporary security credentials that are available on the instance are automatically rotated for you, by AWS, before they expire so that a valid set is always available.
 - AWS makes new credentials available at least five minutes before the expiration of the old credentials.
- For cases other than AWS EC2 Roles, You need to request the temporary credentials first,

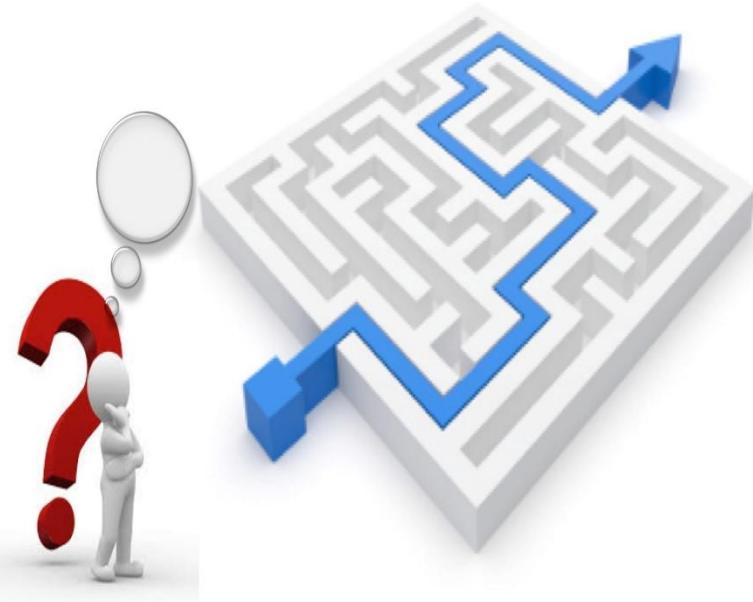
Review Topic : AWS IAM

IAM Role – Service-linked Roles

- AWS service-linked role
 - Is a unique type of service role that is linked directly to an AWS service.
 - Service-linked roles are predefined by the service and include all the permissions that the service requires to call other AWS services on your behalf.
 - The linked service also defines how you create, modify, and delete a service-linked role.
 - A service might automatically create or delete the role. It might allow you to create, modify, or delete the role as part of a wizard or process in the service.
 - Or it might require that you use IAM to create or delete the role.
 - Regardless of the method, service-linked roles make setting up a service easier because you don't have to manually add the necessary permissions.

Identity and Access Management

IAM Role Delegation

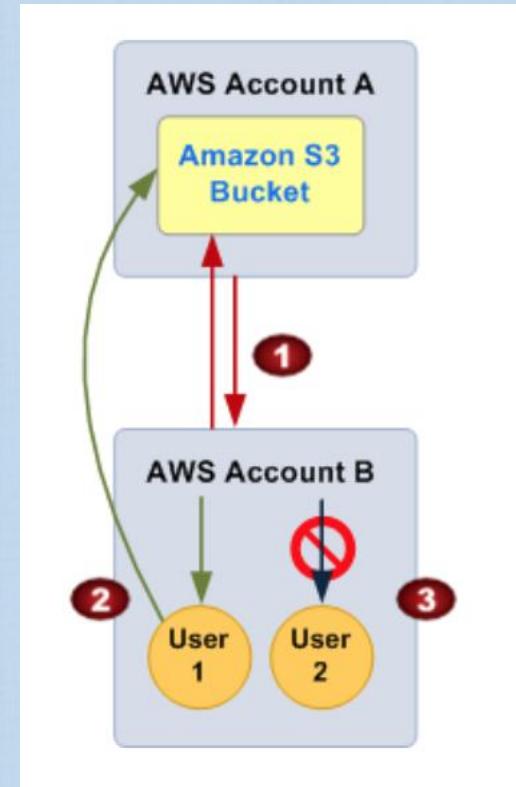


Review Topic : AWS IAM

IAM Role Delegation

Delegation

- Is the granting of permission to someone to allow access to resources that you control.
- Delegation involves setting up a trust between the account that owns the resource (**the trusting account**), and the account that contains the users that need to access the resource (**the trusted account**).
- The trusted and trusting accounts can be any of the following:
 - The same account.
 - Two accounts that are both under your (organization's) control.
 - Two accounts owned by different organizations.



source: [aws.amazon.co](https://aws.amazon.com)

Review Topic : AWS IAM

IAM Role Delegation

- To delegate permission to access a resource, you create an **IAM role** that has two policies attached.
 - The **permissions policy** (JSON format) where the actions and resources the role can use are defined.
 - It grants the user of the role the needed permissions to carry out the intended tasks on the resource.
 - The **trust policy** (JSON format) specifies which trusted accounts can grant its users permissions to assume the role.
 - It defines who can assume the role .
 - This **trusted entity** is included in the policy as the **principal element** in the document.
- When you **create a trust policy**, you cannot specify a wildcard (*) as a principal.
 - The trust policy on the role in the **trusting account** is one-half of the permissions.
 - The **other half is a permissions policy attached to the user in the trusted account** that allows that user to switch to or assume the role.

Review Topic : AWS IAM

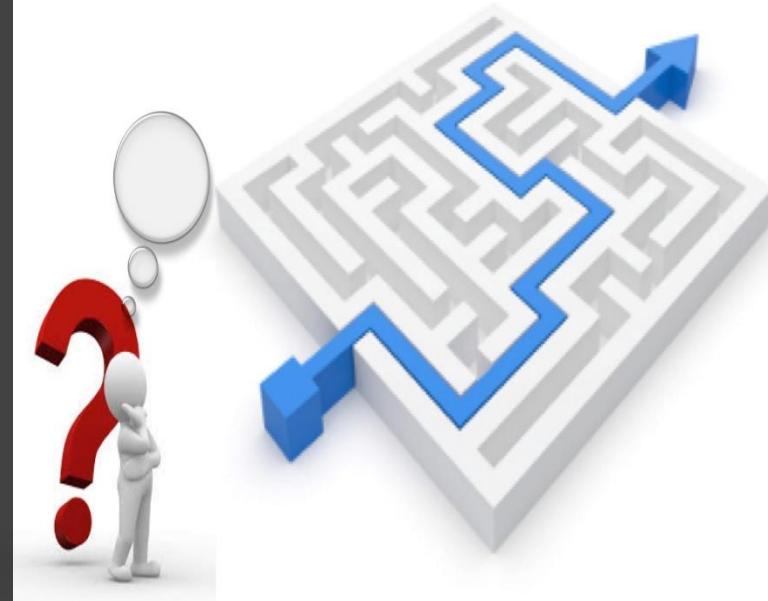
IAM Principal

- **Principal**
 - An entity in AWS that can perform actions and access resources.
 - A principal can be an AWS account root user, an IAM user, or a role.
- If you reference an AWS account as principal, it generally means any principal defined within that account.



Identity and Access Management

IAM Role for Cross-Account Access



Review Topic : AWS IAM

IAM Role for Cross-Account Access

- You might need to allow users from another AWS account to access resources in your AWS account.
- If so, don't share security credentials, such as access keys, between accounts.
 - Instead, use IAM roles.
 - You can define a role in the trusting account, that specifies what permissions the IAM users in the other, trusted, account are allowed.
 - You can also designate which **AWS accounts** have the IAM users that can assume the role.
 - We do not define users in trust policies, rather, AWS accounts.

Review Topic : AWS IAM

IAM Roles and Cross-Account Access

Role for cross-account access

- Granting access to resources in one account to a trusted principal in a different account.
- Roles are the primary way to grant cross-account access.
 - However, with some of the web services offered by AWS you can attach a resource-based policy directly to a resource (instead of using a role as a proxy).
 - You can use them to grant principals in another AWS account (including users) access to the resource.



Review Topic : AWS IAM

AWS Services that support Resource—Based Policies

- The following services support **resource-based policies** for the specified resources:
 - Amazon Simple Storage Service (S3) buckets,
 - Amazon Glacier vaults,
 - Amazon Simple Notification Service (SNS) topics, and
 - Amazon Simple Queue Service (SQS) queues.



This lecture focuses primarily on : **IAM ID
Federation and Authentication**



Break it down into smaller pieces

Building Block way



Explain/Understand each in depth

Deep knowledge about each building block, its functionality, limitations, ways to integrate it, is key in architecting or building solutions on the platform



Build solutions however you like

Once you get the deep understanding, you can use that knowledge to build solutions that best meets the proposed requirements.



Review Topic : AWS IAM

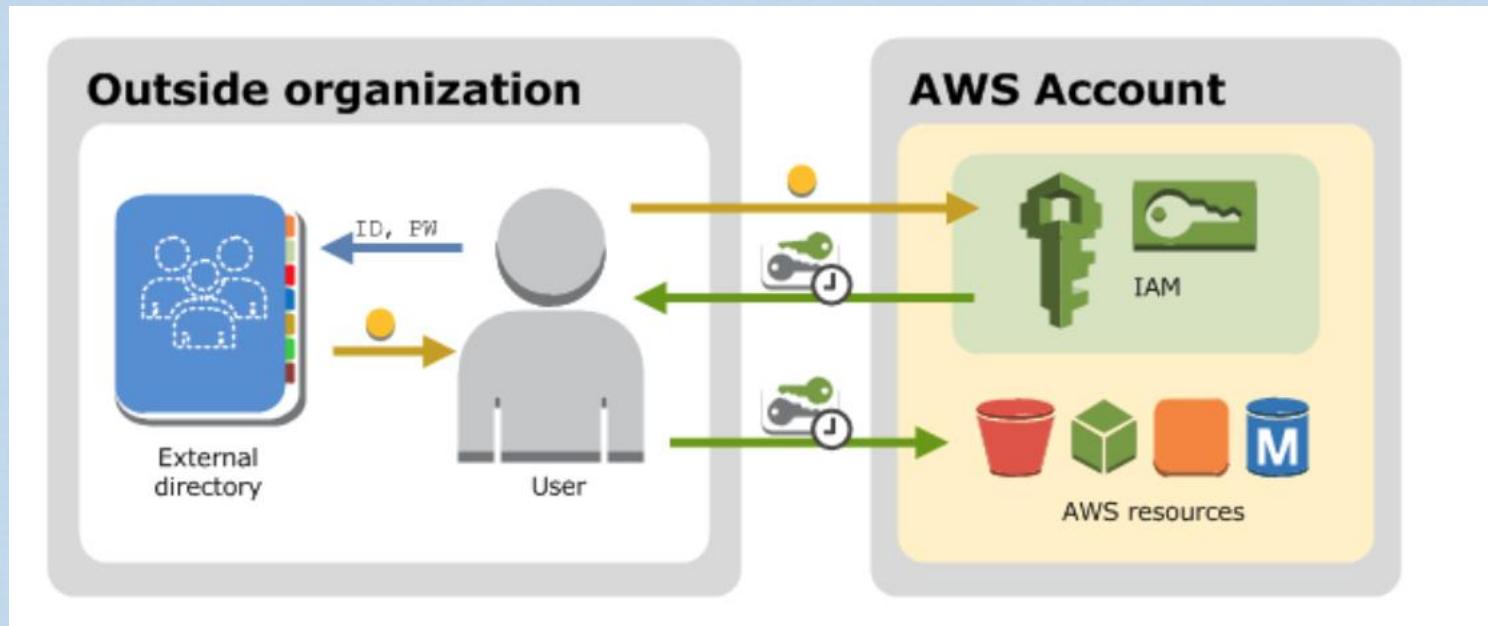
IAM – Identity Federation

Federating Existing Users

- If your account users already have a way to be authenticated, such as authentication through your corporate network,
 - You can *federate* those user identities into AWS.
 - A user who has already logged to the corporate using their corporate identity,
 - The corporate can replace their existing identity with a temporary identity in your AWS account.
 - This user can work in the AWS Management Console.
 - Similarly, an application that the user is working with can make programmatic requests using permissions that you define.

Review Topic : AWS IAM

IAM – ID Federation



source: aws.amazon.com
DOLFINED

Review Topic : AWS IAM

IAM – When to use ID Federation

Federation is particularly useful in these cases:

i) Your users already have identities in a corporate directory.

- If your corporate directory is compatible with **Security Assertion Markup Language 2.0 (SAML 2.0)**,
 - You can configure your corporate directory to provide single-sign on (SSO) access to the AWS Management Console for your users.
- If your corporate directory is not compatible with SAML 2.0,
 - You can create an identity broker application to provide single-sign on (SSO) access to the AWS Management Console for your users (Custom Federation Broker).
- If your corporate directory is Microsoft Active Directory, you can use AWS Directory Service to establish trust between your corporate directory and your AWS account.

Review Topic : AWS IAM

IAM – When to use ID Federation

ii) Your users already have Internet identities.

- If you are creating a mobile app or web-based app that can let users identify themselves through an Internet identity provider like Login with Amazon, Facebook, Google, or **any OpenID Connect (OIDC) compatible identity provider**, the app can use web federation to access AWS.
- AWS recommends using **AWS Cognito** for identity federation with Internet identity providers

Review Topic : AWS IAM

IAM Users and Single Sign-On (SSO)

- IAM users in your account have access only to the AWS resources that you specify in the policy that is attached to the user or to an IAM group that the user belongs to.
 - To work in the console, users must have permissions to perform the actions that the console performs, such as listing and creating AWS resources.
- If your organization has an existing identity system, you might want to create a single sign-on (SSO) option.
 - SSO gives users access to the AWS Management Console for your account without requiring them to have an IAM user identity.
 - AWS Labs can make use of this control
- SSO also eliminates the need for users to sign in to your organization's site and to AWS separately.



source: [aws.amazon.co](https://aws.amazon.com)

Review Topic : AWS IAM

IAM - Access management for Federated Users

Federated Users and Roles

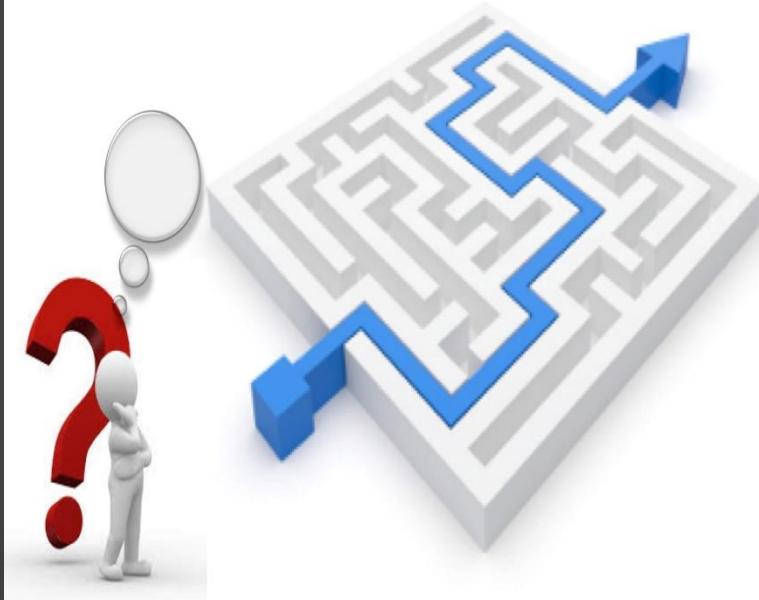
- Federated users don't have permanent identities in your AWS account the way that IAM users do.
- To assign permissions to federated users, you can create an entity referred to as a *role* and define permissions for the role.
- When a federated user signs in to AWS, the user is associated with the role and is granted the permissions that are defined in the role.



source: [aws.amazon.co](https://aws.amazon.com)

Identity and Access Management

IAM Role vs IAM Users – When to use



Review Topic : AWS IAM

IAM Roles – When to use them instead of IAM Users

You're creating an app that runs on a mobile phone and that makes requests to AWS.

- *Don't create an IAM user and distribute the user's access key with the app.*
- **Instead**, use an identity provider like Login with Amazon, Amazon Cognito, Facebook, or Google to authenticate users and map the users to an IAM role.
- The app can use the role to get temporary security credentials that have the permissions specified by the policies attached to the role.

t|

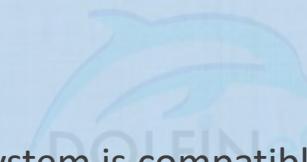


Review Topic : AWS IAM

IAM Roles – When to use them instead of IAM Users

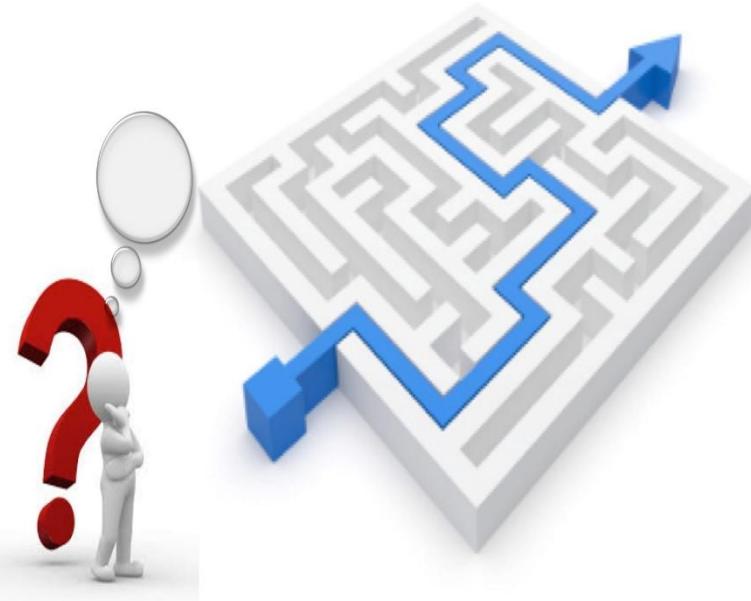
Users in your company are authenticated in your corporate network and want to be able to use AWS without having to sign in again—that is, you want to allow users to federate into AWS.

- *Don't create IAM users.*
- *Instead,* Configure a federation relationship between your enterprise identity system and AWS.
- You can do this in two ways:
 - If your company's identity system is compatible with SAML 2.0, you can establish trust between your company's identity system and AWS (SAML 2.0-based Federation)
 - Create and use a custom proxy server that translates user identities from the enterprise into IAM roles that provide temporary AWS security credentials (Customer Federation Broker)



Identity and Access Management

IAM Best Practices



Review Topic : AWS IAM

IAM Best Practices

- Lock away your AWS account Root user access keys (an access key ID and secret access key)
- Create Individual IAM users
- Use AWS Defined Policies to Assign permissions whenever possible (AWS ready policies)
- Use Groups to assign permissions to IAM users
- Grant Least Privilege
- Use Access levels to Review IAM permissions (AWS categorizes each service action into one of four *access levels* based on what each action does: List, Read, Write, or Permissions management.)
- Configure a Strong Password policy for Users

Review Topic : AWS IAM

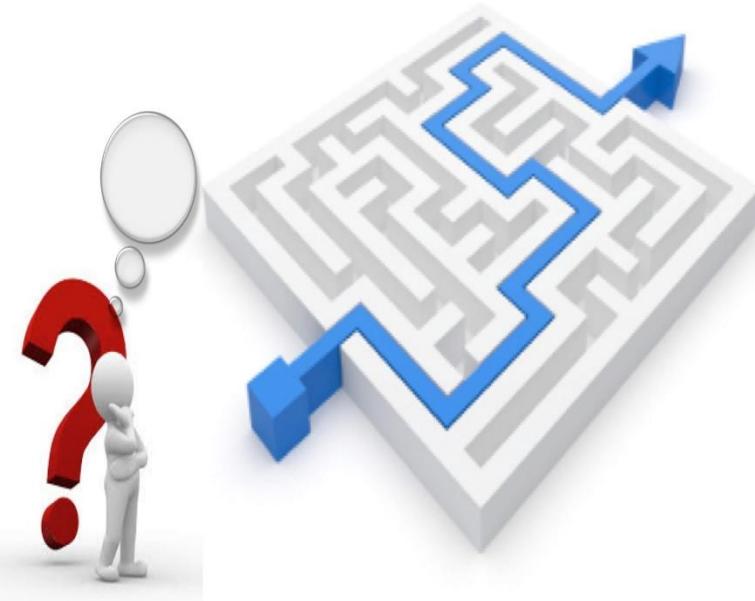
IAM Best Practices

- Enable MFA for Privileged Users
- Use Roles for Applications that run on AWS EC2 instances
- Delegate by using Roles instead of sharing Credentials
- Rotate Credentials Regularly
- Remove Unnecessary Credentials
- Use Policy Conditions for Extra Security
- Monitor Activity in Your AWS Account



Identity and Access Management

IAM Logging

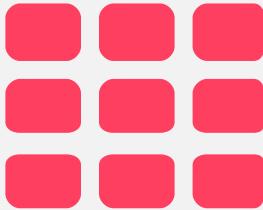


Review Topic : AWS IAM

IAM – CloudTrail logging

Logging sign-in details in CloudTrail

- If you enable CloudTrail to log sign-in events to your logs, you need to be aware of how CloudTrail chooses where to log the events.
- If your users sign-in directly to a console, they are redirected to either a global or a regional sign-in endpoint, based on whether the selected service console supports regions.
- For example, the main console home page supports regions, so if you sign-in to the following URL: <https://alias.signin.aws.amazon.com/console>
 - you are redirected to a "default" regional sign-in endpoint
 - <https://us-east-1.signin.aws.amazon.com>, resulting in a regional CloudTrail log entry in that region's log:



AWS ACTIVE DIRECTORY SERVICES

You Can Do It Too!



AWS Active Directory Services

Option	Fully Managed by AWS	SSO support	RDS MS SQL support	MFA Support	Best Use case	Snapshots for Backup and DR	Same Policies on premise and in AWS	Can authenticate to AWS console w/out SAML2.0	Cloud Trail and SNS integration	Supports Multi AZ
AWS Active Directory Service for MS AD	Yes - Standard < 5000 users, and Enterprise >5000 users	Yes, SAML 2.0 , No replication option to On Premise AD (VPN)	Yes	Yes	> 5000 users, or when Trust with on-premise MS AD	Yes (Automated and Manual)	Can be achieved with Trust	Yes	Yes	Yes (default)
AD Connector	Small <=500 users and Large <=5000 Users	Yes, however, No LDAP DB in the cloud, (Requires VPN or DX)	No	Yes	When you want to use your On-premise Existing AD directory with your AWS application	N/A	No policies in AWS, everything is on premise	Yes, allows users to log in to AWS using their AD credentials		N/A
Simple AD	Yes	Yes (Kerberos based), No trust relations or SAML based federation	No	No	<= 5000 users, Best for a low cost Active Directory compatible service in AWS	Yes (Automated and Manual)		Yes		Deployed on 2 EC2 instances in 2 different subnets in a VPC
You own MS AD on EC2 instances (unmanaged)	No , Small <=500 users and Large <=5000 Users	MS ADs in AWS can join On PremiseAD (Replication) Not trust (VPN) But will need trust with AS for SSO		Yes	Low scale, Low cost, basic features AD like requirement, and for LDAP aware applications	Not automated, you need to do your own snapshots of the EBS volumes	Yes	You need SAML2.0 for SSO	N/A	Your own Design