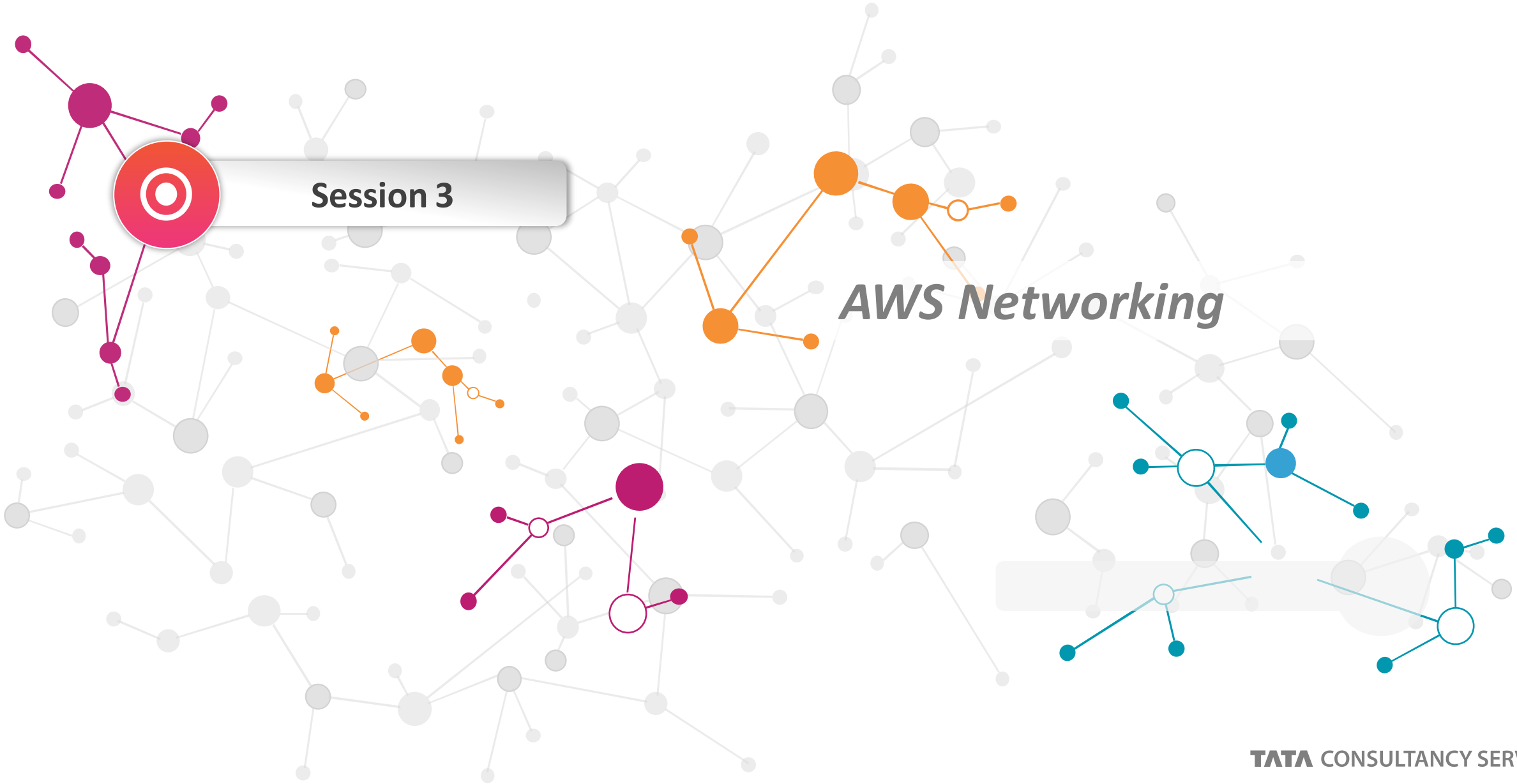
AWS Academy

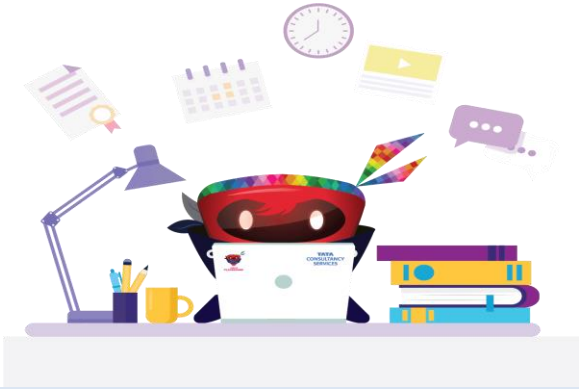
North America
Talent Development Team

Date


Session 3
AWS Networking

- *Virtual Private Cloud*
- *Subnet*
- *Elastic IP*
- *NAT Gateways*
- *Direct Connect*
- *Elastic Load Balancer*
- *Route53*

AWS SAA BootCamp

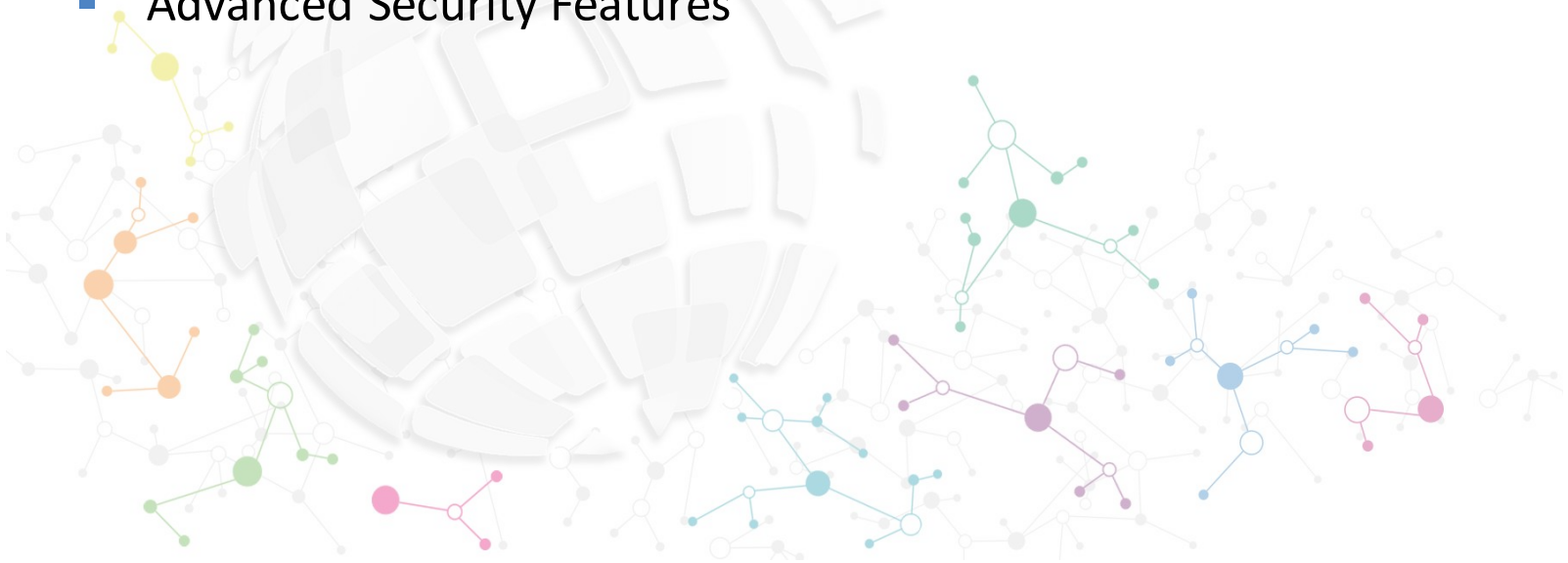


AWS Networking

Amazon VPC

Amazon VPC

- Virtual network topology that you define
- Your own logically isolated section of AWS
- Complete control of your networking environment
  - IP ranges
  - Subnets
  - Routing tables
  - Gateways
- Multiple Connectivity Options
- Advanced Security Features

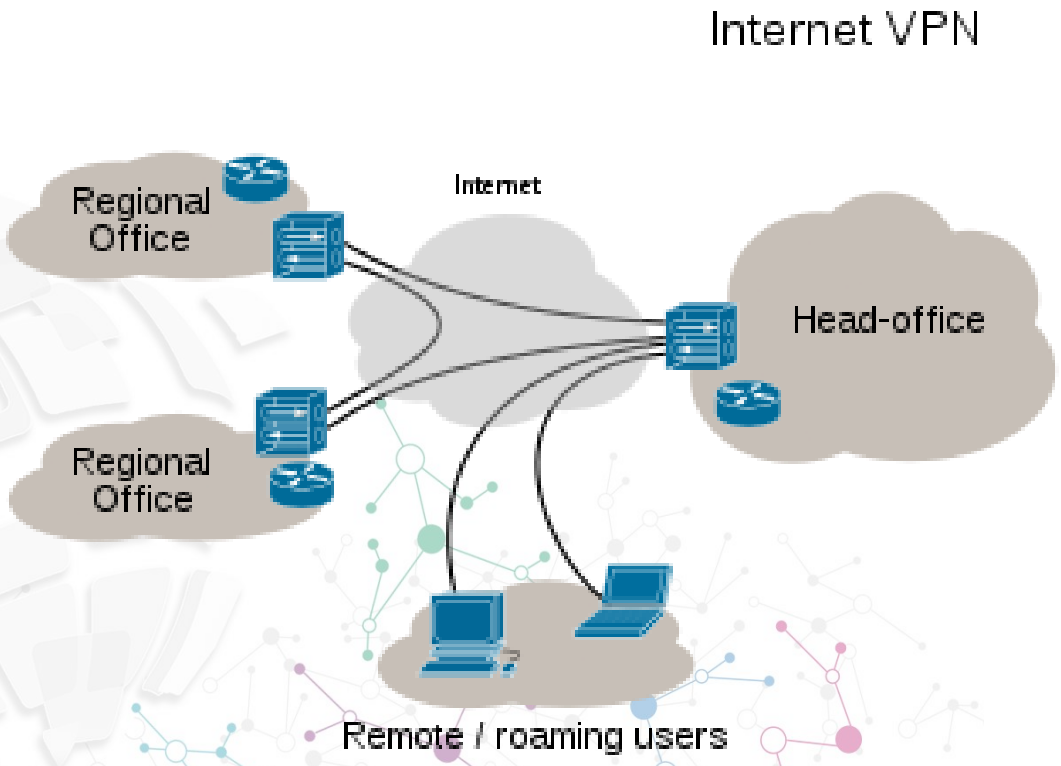


Plan your VPC IP space before creating it

- Consider future AWS region expansion
- Consider future connectivity to corporate networks
- Consider subnet design
- VPC can be between /16 and /28
- CIDR cannot be modified once created, Can add additional CIDR block
- Overlapping IP spaces = future headache

Concept of VPN

A **virtual private network (VPN)** extends a private network across a public network, and enables users to send and receive data across shared or public networks as if their computing devices were directly connected to the private network. Applications running on a computing device, e.g., a laptop, desktop, smartphone, across a VPN may therefore benefit from the functionality, security, and management of the private network. Encryption is a common, though not an inherent, part of a VPN connection.



Introduction to VPC

AWS Definition:

Amazon Virtual Private Cloud (Amazon VPC) lets you provision a logically isolated section of the AWS Cloud where you can launch AWS resources in a virtual network that you define. You have complete control over your virtual networking environment, including selection of your own IP address range, creation of subnets, and configuration of route tables and network gateways. You can use both IPv4 and IPv6 in your VPC for secure and easy access to resources and applications.

Simplified definition:

“A private sub-session of AWS network in which you can place AWS resources (such EC2, Databases). You will be having full control over who have access to the AWS resources that you place inside your VPC.”

Introduction to VPC

- Creating a VPC allows you to create a logically isolated section of AWS Cloud for your use.
- Complete control over virtual networking
  - Define you own IP address range (Private or Internal IP addresses)
  - Divide your cloud's private IP address range into one or more subnets to facilitate running applications and services in your VPC.
  - Define which subnets are private and which are public.

- Multiple layers of security
  - Network ACL
  - Security groups

Amazon VPC Components

An Amazon VPC consists of the following components:

- Subnets
- Route Tables
- Dynamic Host Configuration Protocols (DHCP) options set
- Security groups
- Network Access control List (NACL)

An Amazon VPC has the following optional components:

- Internet Gateways (IGWs)
- Elastic IP(EIP) addresses
- Elastic Network Interfaces (ENIs)
- VPC Endpoints
- VPC Peering
- Network Address Translation (NATs) instances and NAT gateways
- Virtual Private Gateway (VPG), Customer Gateways (CGWs), and Virtual Private Networks (VPNs)

Amazon VPC - CIDR

- When you create an Amazon VPC, you must specify the IPv4 address range by choosing a Classless Inter-Domain Routing (CIDR) block, such as 10.0.0.0/16
- You should always create unique IP address schema to avoid overlapping of IP addresses across networks that you may connect in future
- An Amazon VPC address range may be as large as /16 (65,536 available addresses) or as small as /28 (16 available addresses) and should not overlap any other network with which they are to be connected.
- CIDR blocks define subnets (for example, 10.0.1.0/24 and 192.168.0.0/24). The smallest subnet that you can create is /28 (16 IP addresses).
- AWS reserves the first four IP addresses and the last IP address of every subnet are not available for you to use. For example, a subnet defined as a /28 has 16 available IP addresses; subtract the 5 IPs needed by AWS to yield 11 IP addresses for your use within the subnet.

Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR	DHCP options set	Main Route table	Main Network ACL	Tenancy
	vpc-061a4b7c	available	172.31.0.0/16	-	dopt-4651a23c	rtb-98924ee6	acl-5eef5723	default

CIDR - Fundamentals

	128	64	32	16	8	4	2	1	255	
N	0	0	0	0	1	0	1	0	10	
N	0	0	0	0	0	0	0	0	0	
N	0	0	0	0	0	0	0	0	0	256
H	N	N	N	N	0	0	0	0	0	256

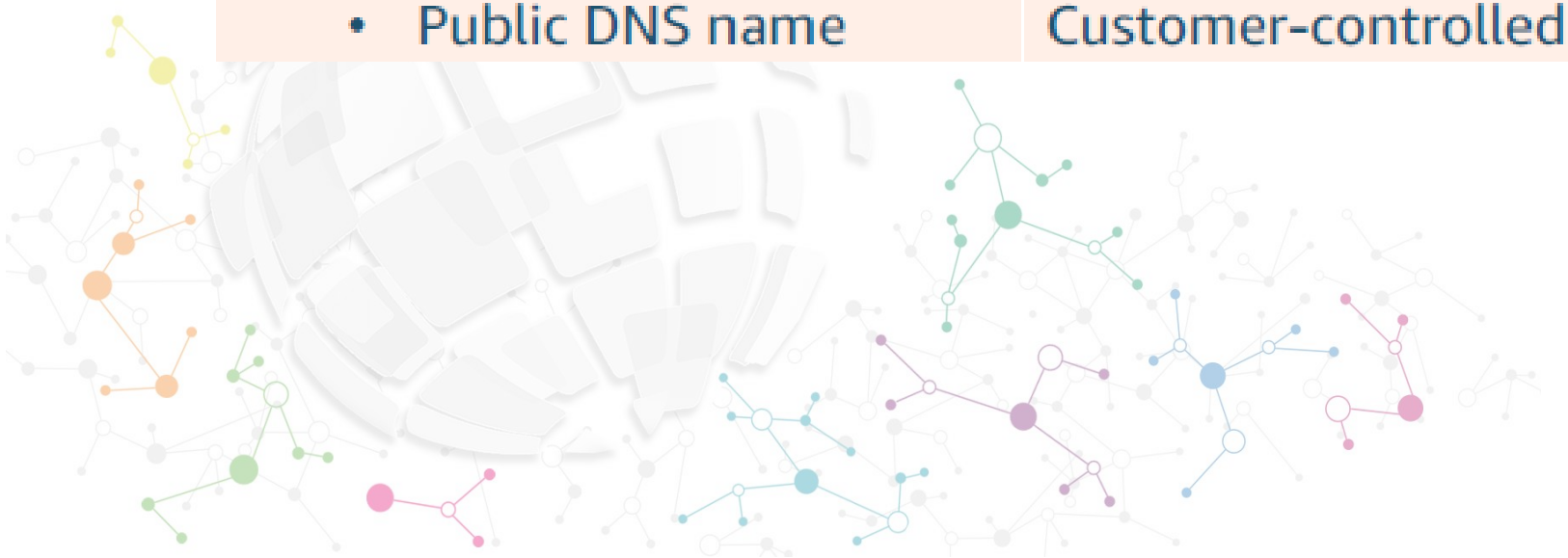
10.0.0.0/8 = 1,67,77,216 IPs (256*256*256)
10.0.0.0/16 = 65536 IPs (256x256)
10.0.0.0/24= 256 IPs
10.0.0.0/28 = 16 Ips

Largest VPC range /16 (65536, ideally having 65531 IPs)
Smallest is /28 = 16 IPs (5 IPs for AWS, 11 IPs)

IPv4
4x8= 32
0….255= 256

IP Addressing

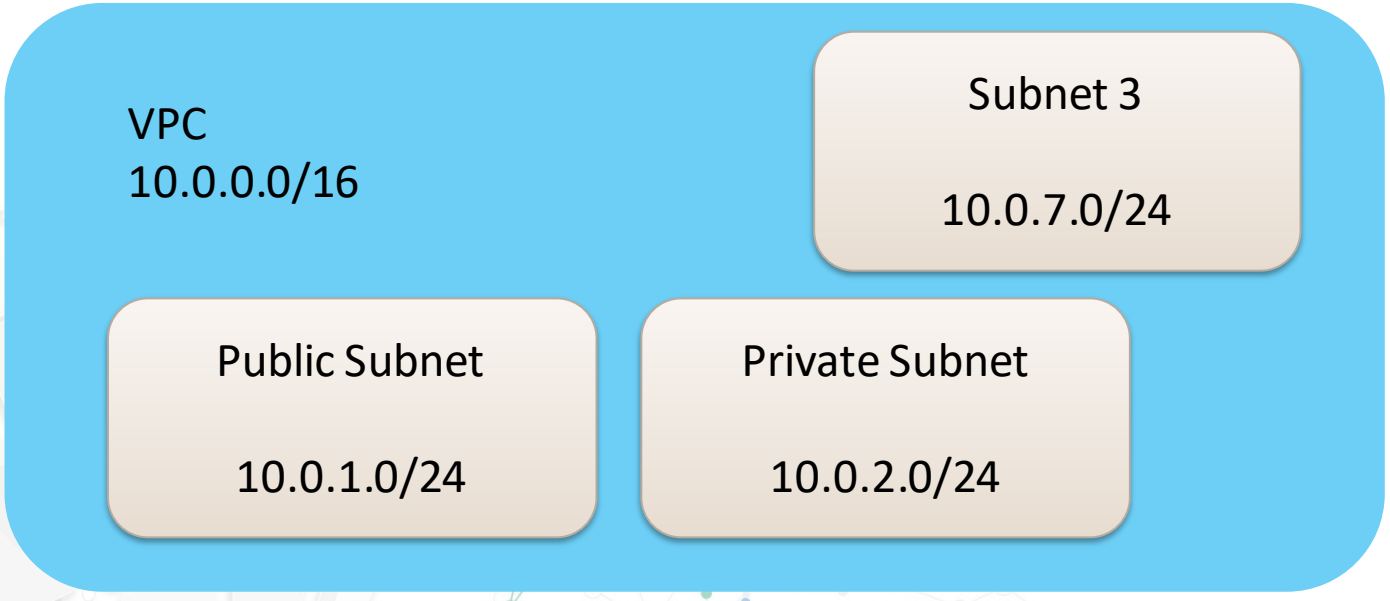
Default VPC	Virtual Private Cloud
Dynamic Private IP	Dynamic or Static Private IP Address
Dynamic Public IP	None by default (can be created with publicIP=true)
Optional Static Public IP (EIP)	Optional Static Public IP (EIP)
AWS-provided DNS names • Private DNS name • Public DNS name	AWS-provided public DNS lookup AWS-provided private DNS names Customer-controlled DNS options



Subnets

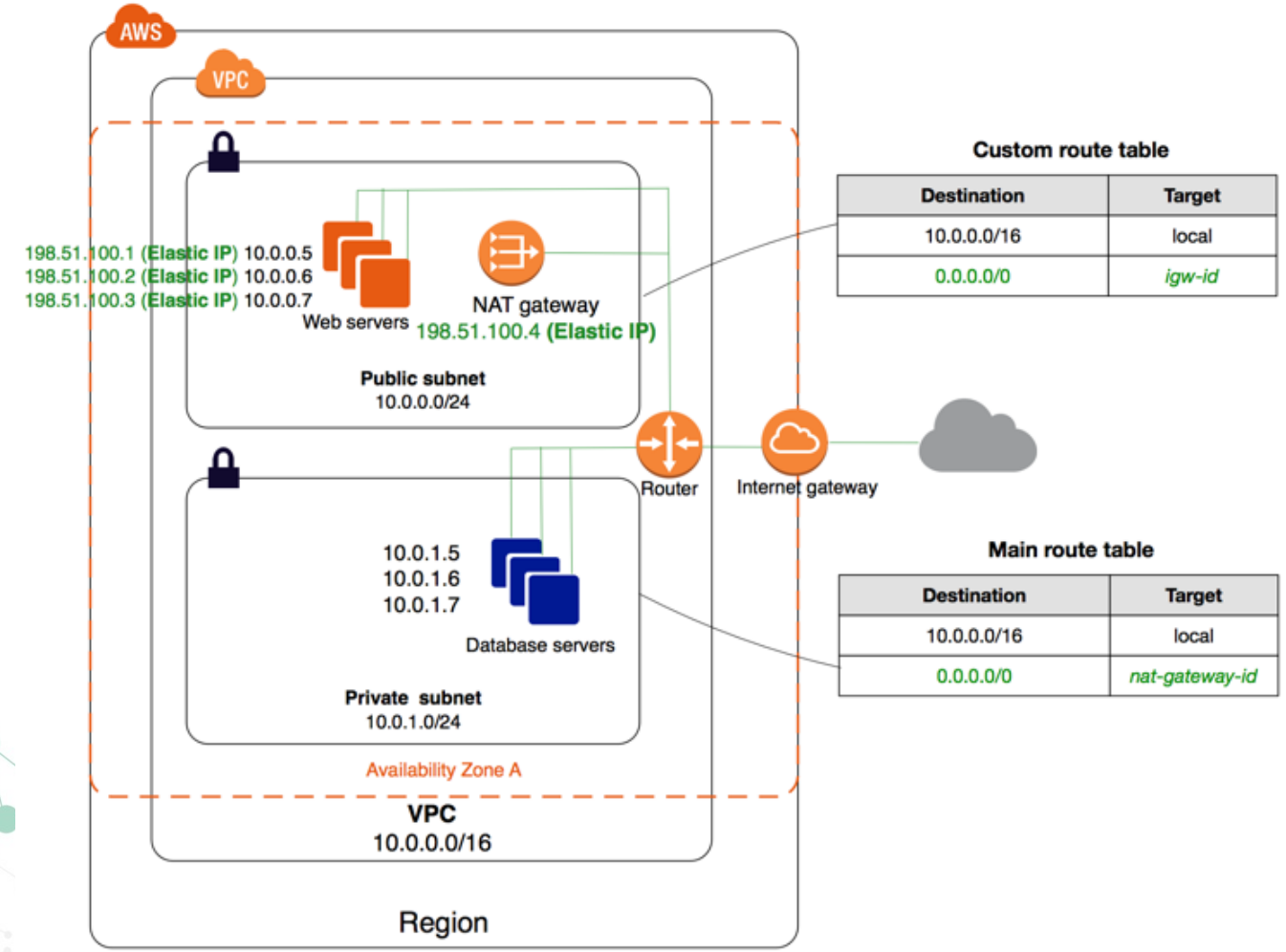
- The network range that you define for your VPC can be sub divided into smaller ranges and thus you create subnets

- Subnet -A range of IP address in your VPC

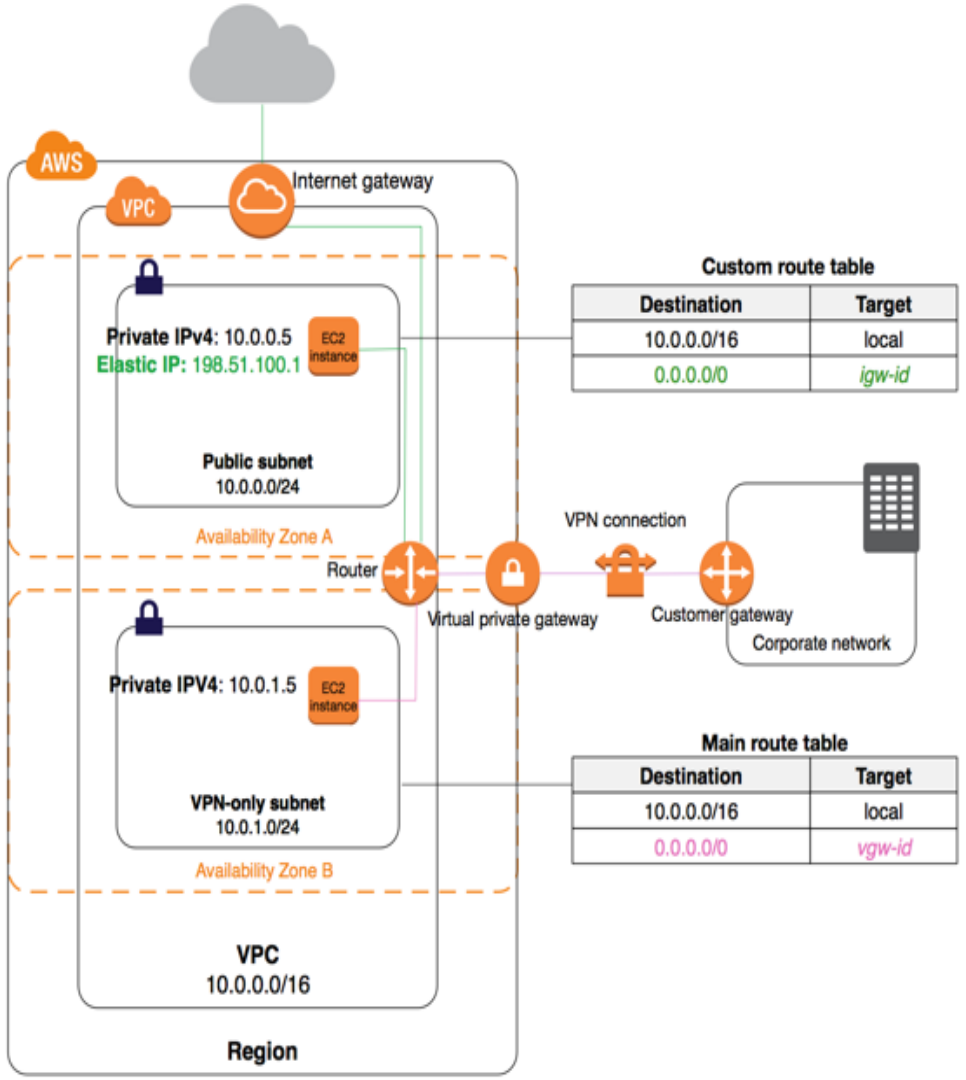


Subnets Contd..

- Subnets categorized into two types -Private and Public Subnet
  - A **Public Subnet** is a subnet that's associated with a route table that has a route to an Internet gateway.
  - If a subnet does not have a route table with route to an internet gateway, then subnet is known as **Private Subnet**



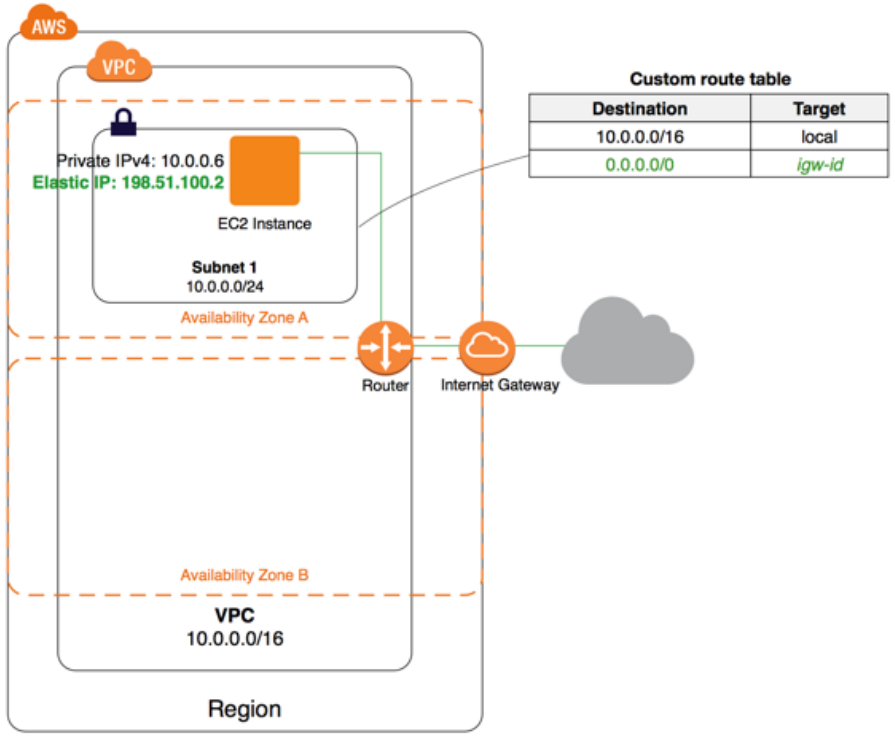
Route Tables



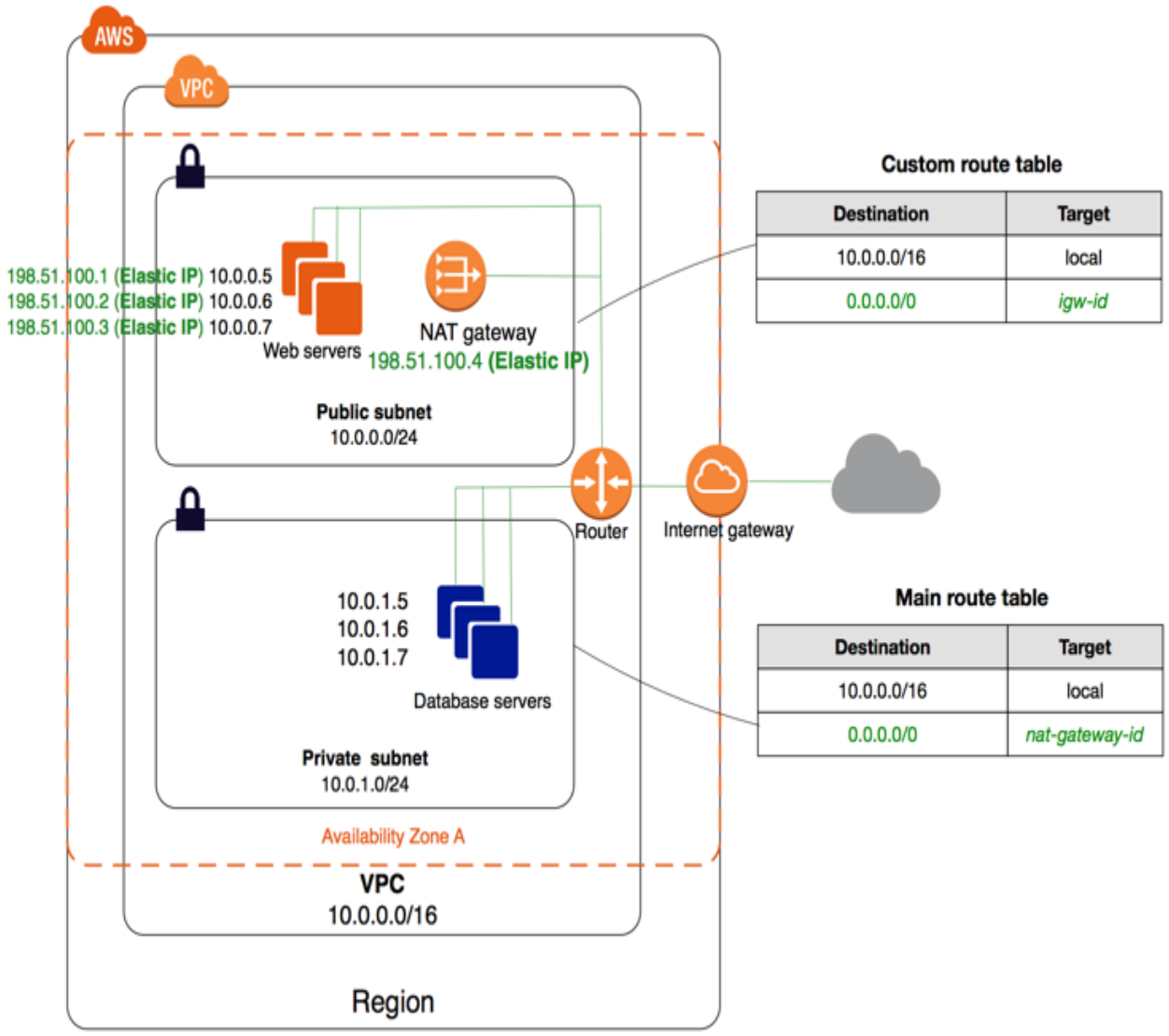
- A public subnet is a subnet that's associated with a route table that has a route to an Internet gateway.
- Each subnet in your VPC must be associated with a route table, the table controls the routing for the subnet.
- A subnet can only be associated with one route table at a time, but you can associate multiple subnets with the same route table.

Internet Gateway

- An internet gateway is a horizontally scaled, redundant, and highly available VPC component that allows communication between instances in your VPC and the internet. It therefore imposes no availability risks or bandwidth constraints on your network traffic.
- An internet gateway serves two purposes: to provide a target in your VPC route tables for internet-routable traffic, and to perform network address translation (NAT) for instances that have been assigned public IPv4 addresses.
- An internet gateway supports IPv4 and IPv6 traffic



NAT Gateway



Custom route table

Destination	Target
10.0.0.0/16	local
0.0.0.0/0	*igw-id*

Main route table

Destination	Target
10.0.0.0/16	local
0.0.0.0/0	*nat-gateway-id*

- NAT Gateway provides access to Internet for your EC2 instances launched in the Private subnet of your VPC
- However you cannot access your instances from the Internet

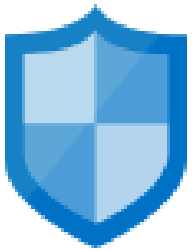
- High availability – built-in redundancy
- High bandwidth – up to 10Gbps
- Fully Managed by AWS
- Assign an EIP to each NAT Gateway
- View NAT gateways' traffic using Flow Logs
- NAT gateways support TCP, UDP, and ICMP protocols
- Network ACLs apply to NAT gateway's traffic
- CloudTrail Support

Network Access Control List (NACL)

- NACLs act as a firewall for controlling traffic in and out of subnets
- NACLs comprise a numbered list of rules which are evaluated in order, starting with the lowest numbered rule
- Separate Inbound and Outbound rules
- You can define Allow and Deny rules
- NACL is attached to a subnet and only one NACL can be attached to a subnet at any given time
- An NACL can be associated with multiple subnets.
- NACLs are stateless –which means you have to define both Inbound and Outbound rules. e.g. if you want to allow HTTP traffic on port 80 into your subnet, then you need
  - Inbound rule to Allow traffic to port 80 from source IP range of 0.0.0.0/0 (whatever your desired source IP range)
  - oOutbound rule to Allow traffic to port range 32768-65535 to 0.0.0.0/0 (whatever your desired source IP range)

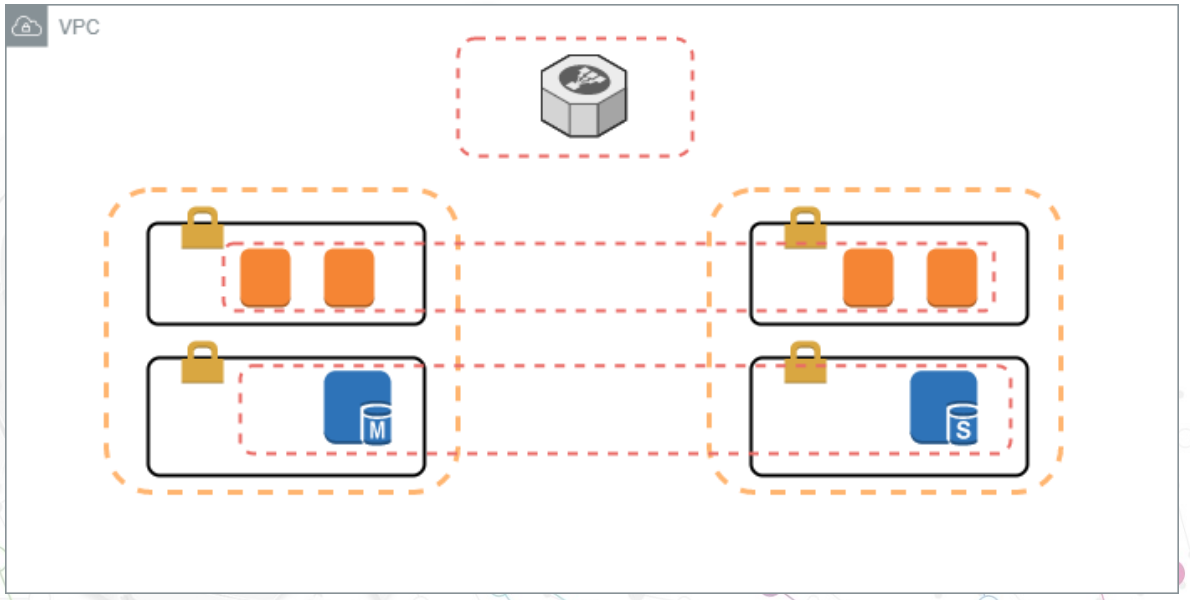
Security Groups

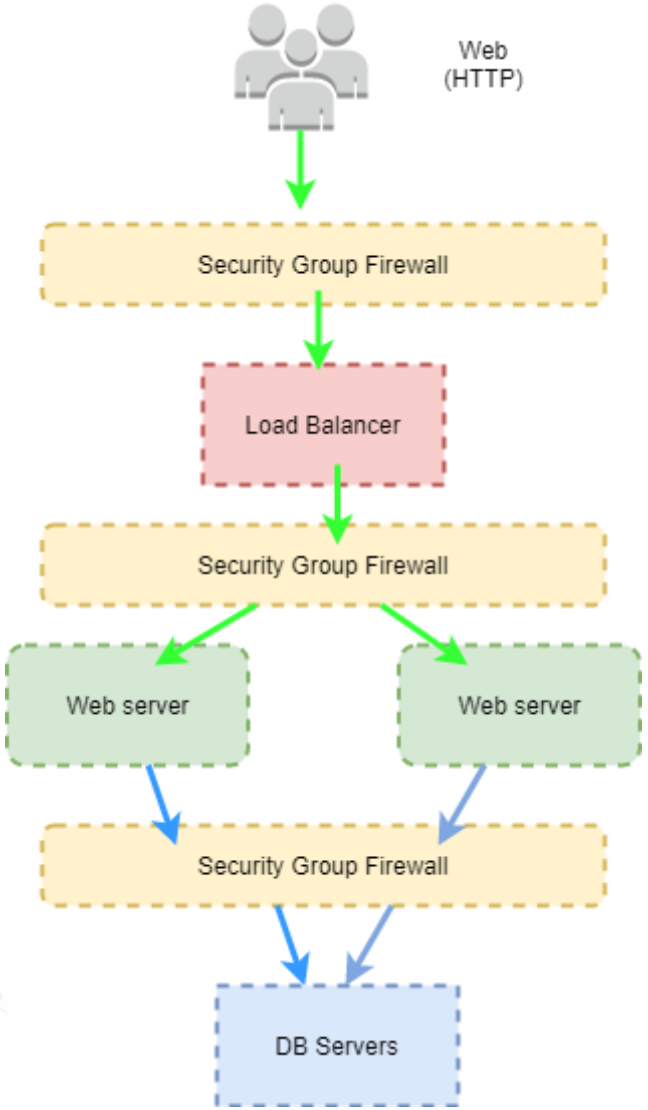
- A Security Group acts as a firewall that controls the traffic allowed to and fro with an instance
- When you launch an instance, you may assign one or more Security Groups. Effective rules for the instance are collation of all rules from all attached Security Groups.
- There are INBOUND and OUTBOUND rules for a Security Group. Security Group rules are ALLOW rules and there are no DENY rules
- The new rules are automatically applied to all instances to which the security group is assigned
- Any updates made to Security Group rules or any Security groups attached to/detached from instance take effect immediately [there is no delay / latency for the rules to take effect]
- Security Group is Stateful–define rules only for REQUEST traffic and it's RESPONSE is automatically allowed
- Typically you create Security Groups based on roles [e.g. Security Group for web servers, for database servers, for management servers etc.] and use one or multiple of these to attach to instances based on their roles



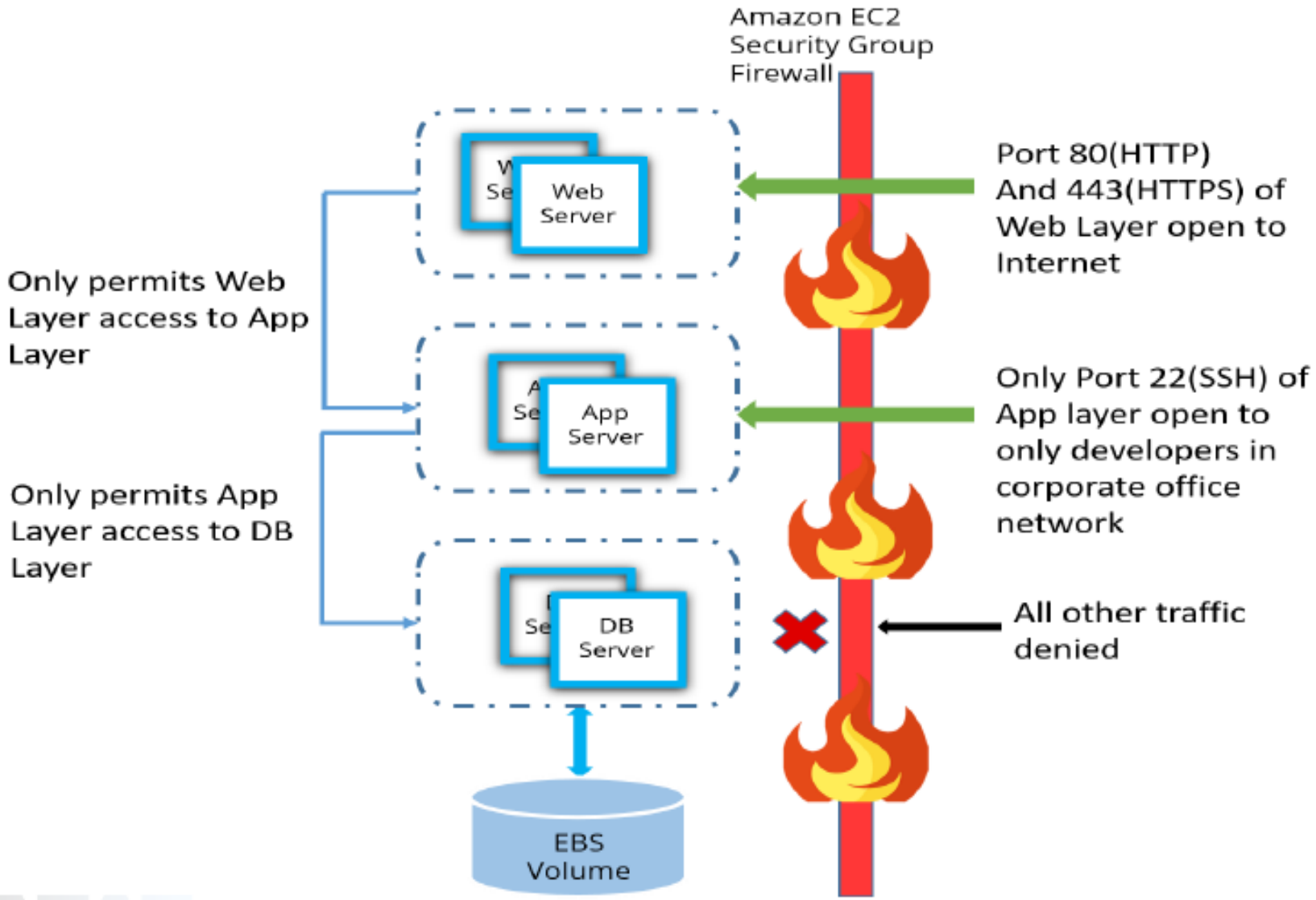
Network Building Blocks| Network Control - Security Groups

- Security Groups
  - Per instance
  - Stateful





Security Groups



Network Access Control List (NACL) - Examples

Summary | **Inbound Rules** | Outbound Rules | Subnet Associations | Tags

Allows inbound traffic. Because network ACLs are stateless, you must create inbound and outbound rules.

Edit

View: All rules

Rule #	Type	Protocol	Port Range	Source	Allow / Deny
200	HTTP (80)	TCP (6)	80	0.0.0.0/0	ALLOW
*	ALL Traffic	ALL	ALL	0.0.0.0/0	DENY

Summary | Inbound Rules | **Outbound Rules** | Subnet Associations | Tags

Allows outbound traffic. Because network ACLs are stateless, you must create inbound and outbound rules.

Edit

View: All rules

Rule #	Type	Protocol	Port Range	Destination	Allow / Deny
200	Custom TCP Rule	TCP (6)	32768-65535	0.0.0.0/0	ALLOW
*	ALL Traffic	ALL	ALL	0.0.0.0/0	DENY

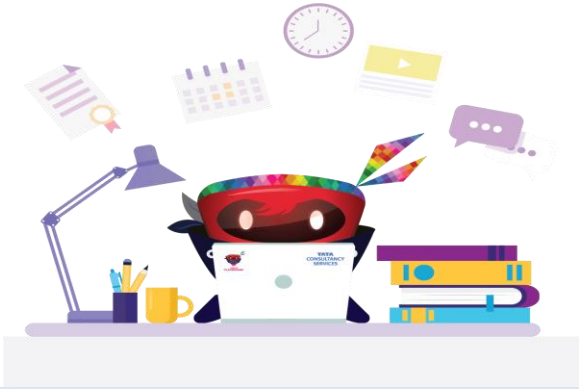
Comparison of Security Group & Network ACLs

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
Security Group evaluate all rules before deciding whether to allow traffic	Network ACL process rules in number order when deciding whether to allow traffic
Applies to an instance only 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)

Elastic IP

- An Elastic IP address is a static, public IPv4 address designed for dynamic cloud computing.
- You can associate an Elastic IP address with any instance or network interface for any VPC in your account.
- With an Elastic IP address, you can mask the failure of an instance by rapidly remapping the address to another instance in your VPC.
- You can retain control of the Elastic IP address until you release it back to AWS

AWS SAA Boot Camp



AWS Networking

Direct Connect

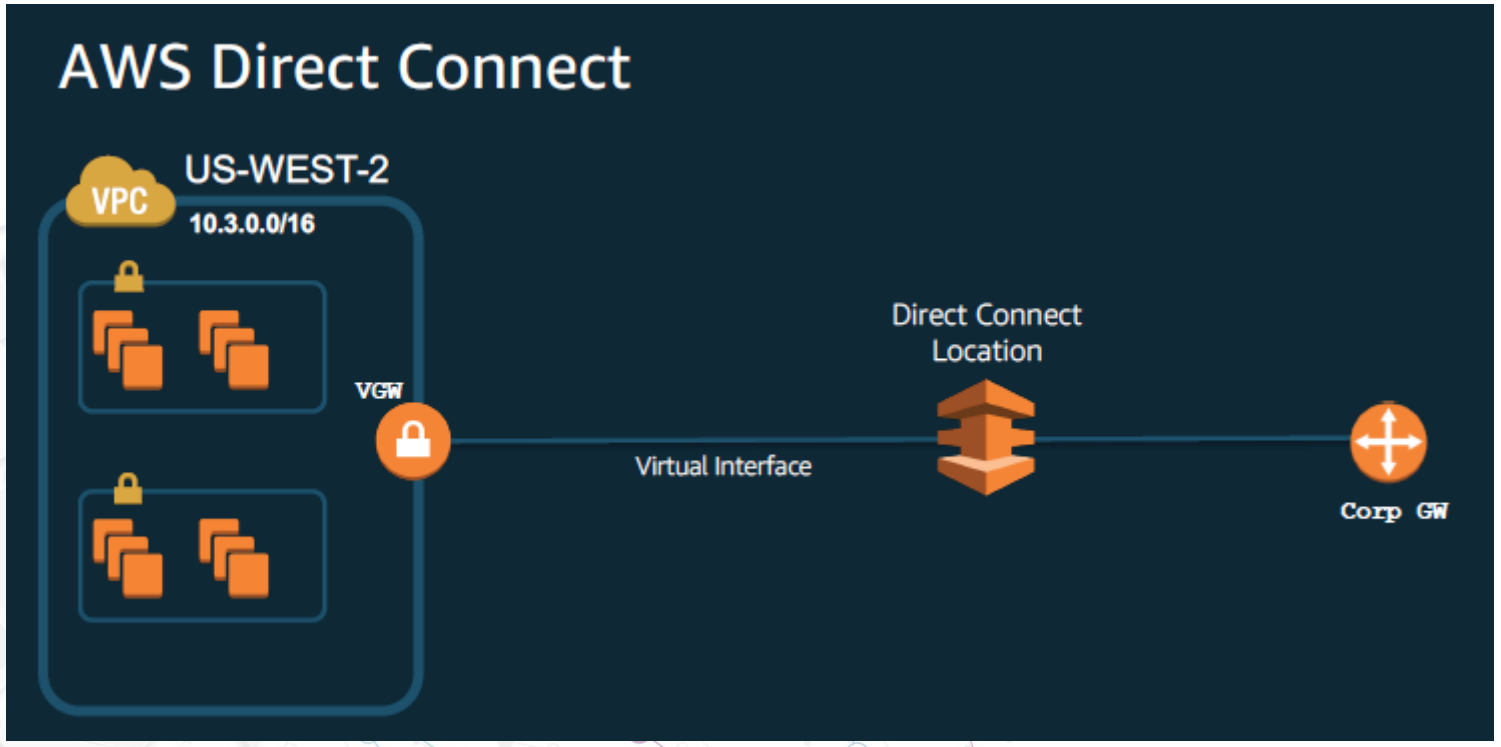
AWS Direct Connect

- 1 Gbps or 10 Gbps fiber cross connect
  - 50M, 100M, 200M, 300M, 400M, & 500M available through APN Partners
  - Single VIF per connection through APN Partners. To connect to more than 1 VPC, need additional connections.
- Consistent Network Performance
- Lower latency compared to a VPN connection
- Private connectivity into your VPC
- Connect Multiple AWS Regions using Direct Connect Gateway.
- Multiple AWS accounts can share a connection
- Reduced data-out rates
- Public & Private Virtual Interface available through Direct Connect
  - Private VIF allows you to connect to your VPC. Private ASN can be used for BGP
  - Public VIF allows you to connect to Public AWS services like S3, DynamoDB, etc through DX. Public ASN number is required for use with BGP.

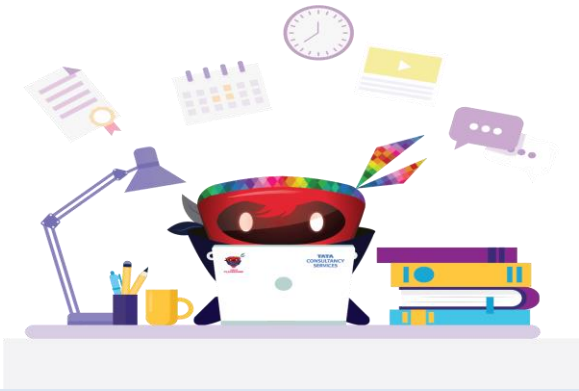
VIF – Virtual Interface
BGP – Border Gateway Protocol
DX – Direct Connect
ASN – Autonomous System Number

AWS Direct Connect| Continued

- Site redundancy can be achieved by leveraging two DX cross connect locations.
- If you have multiple DX connections going to the same DX cross connect location, you can request that to be terminated on separate physical devices (router).



AWS SAA Boot Camp

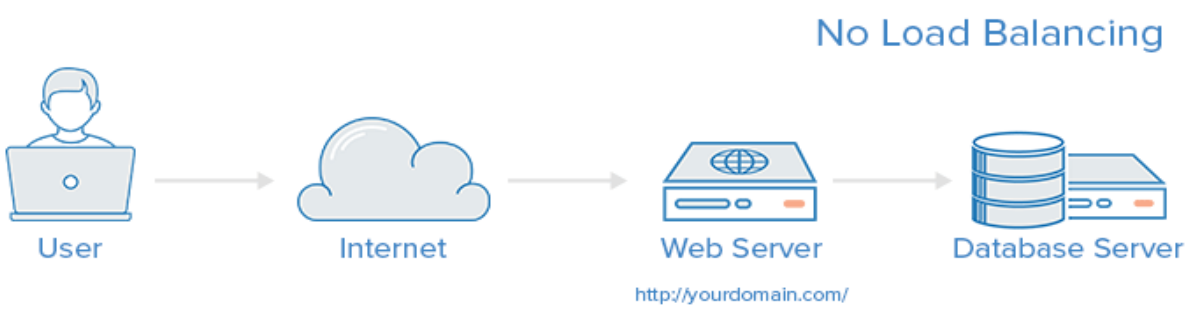


AWS Networking

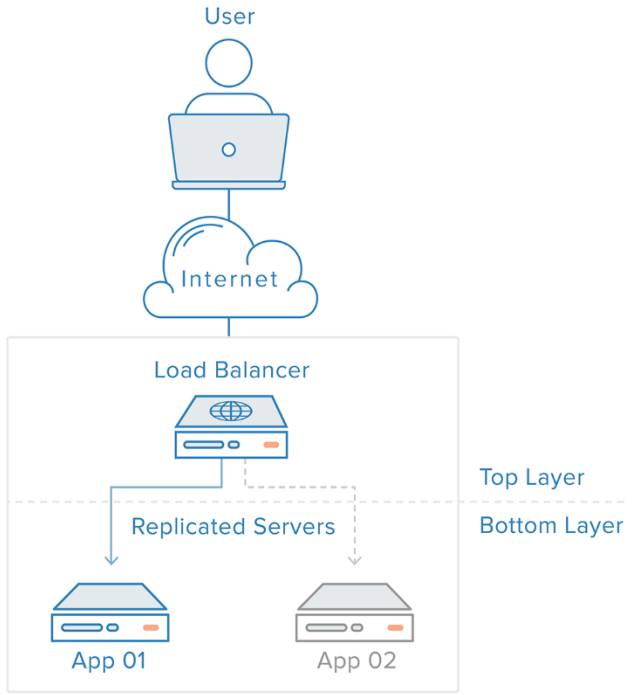
Elastic Load Balancer

What is Load balancing?

- Distribute Workloads across multiple compute resources
- Optimize resource usage
- Avoid overload of a single resource
- Increases reliability and availability of your application
- Provide single internet service from multiple servers
- Load Balancer receives request and forwards it to backend server
- Can check health of application on the backend servers



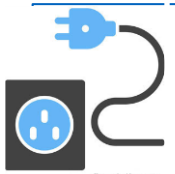
- You can either manage your own virtual load balancers on Amazon EC2 instances or leverage an AWS Cloud service called **Elastic Load Balancing**, which provides a managed load balancer for you.



- ELB allows you to distribute traffic across a group of Amazon EC2 instances in one or more Availability Zones in a Region, enabling you to achieve high availability for your applications.
- AWS manages the availability of ELB and capacity of ELB as traffic to ELB increases

Algorithm

- Uses round robin algorithm for request routing to back-end EC2 instances



Supported Routing and Load Balancing of

- Hypertext Transfer Protocol (HTTP)
- Hypertext Transfer Protocol Secure (HTTPS)
- Transmission Control Protocol (TCP)
- Secure Sockets Layer (SSL)
- SSL offload

Sticky Sessions

- Enable Sticky session for the ELB with either application generated cookies or ELB generated cookie with Time-to-live (TTL) for the cookie
- Sticky session allows to stick a user's session to one back-end server

Health checks

- Health checks allow monitoring the availability of the application running on the back-end EC2 instances and to take instances Out-of-service if they are not healthy and return In-service once they are healthy

Cross Zone Load balancing

- Provides Cross zone load balancing capability to evenly distribute load across all the instances in all the AZs enabled for the ELB

Scalability

- Auto scaling can be configured with ELB to automatically launch additionally instance if workload increases and add these to the ELB to distribute load across this additional capacity

Monitoring

- Cloud watch alarm can be used to monitor ELB metrics

Log Management

- ELB access log can be enabled on ELB and configured to deliver the logs to S3 bucket

Elastic Load Balancer (ELB)

Connection Draining

- The number of seconds to allow existing traffic to continue to flow to the back-end instance
- (default: 300 sec ; min:1 sec, max: 3600 secs)

Connection timeout

- The number of seconds a connection (from client to ELB or from ELB to back-end instance) can be idle before the load balancer closes the connection
- (default: 60 sec; min:1 sec, max: 3600 secs)

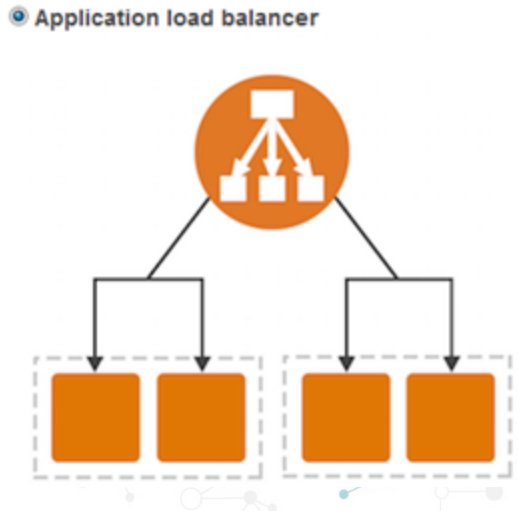
Elastic Load Balancer (ELB)

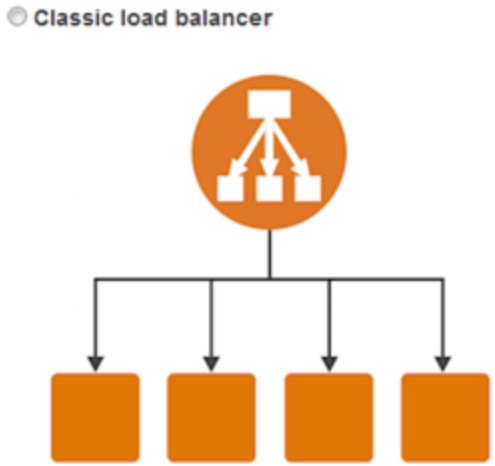
Application – Layer7 (ALB)
Presentation – Layer6
Session Layer5
Transport Layer4
Network - Layer 3(NLB)
Data – Layer2
Physical – Layer1

Elastic Load Balancing supports three types of load balancers:

- Application Load Balancers
- Network Load Balancers
- Classic Load Balancers

There is a key difference in how the load balancer types are configured. With Application Load Balancers and Network Load Balancers, you register targets in target groups, and route traffic to the target groups. With *Classic Load Balancers, you register instances with the load balancer.*





Elastic Load Balancer| Deployment options

When creating ELB, you can deploy them as:

Internet-Facing Load Balancers:

- You can use Internet-facing load balancer that takes requests from clients over the Internet and distributes them to Amazon EC2 instances that are registered with the load balancer. Has external DNS name

Internal Load Balancers

- You can use internal load balancers to route traffic to your Amazon EC2 instances in VPCs with private. Has internal DNS name

HTTPS Load Balancers

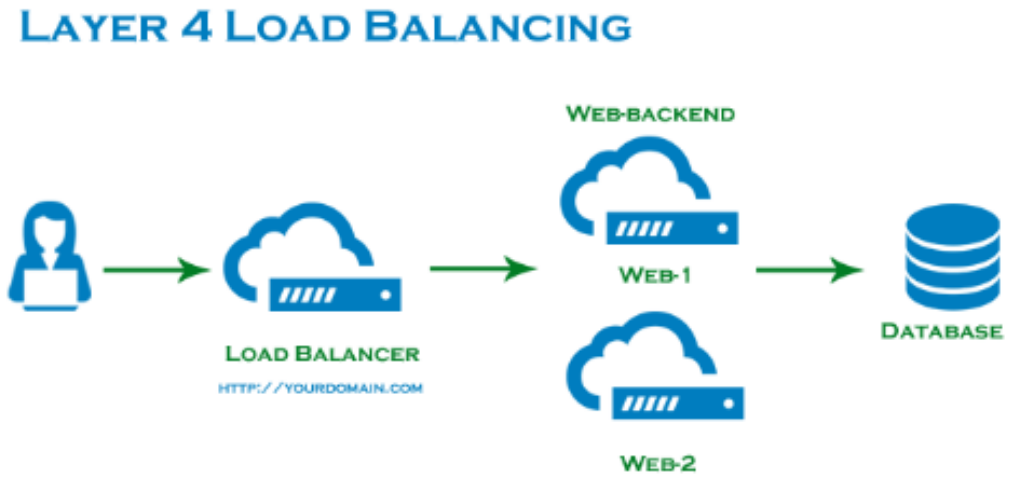
- You can create a load balancer that uses the SSL/Transport Layer Security (TLS) protocol for encrypted connections (also known as SSL offload). This feature enables traffic encryption between your load balancer and the clients that initiate HTTPS sessions, and for connections between your load balancer and your back-end instances.

ELB Types | Classic Load Balancer

- The AWS Classic Load Balancer(CLB) operates at Layer 4(transport) of the OSI model. What this means is that the load balancer routes traffic between clients and backend servers based on IP address and TCP port.

- AWS ELB -CLB also supports Layer 7 specific features such as X-Forwarder and sticky sessions

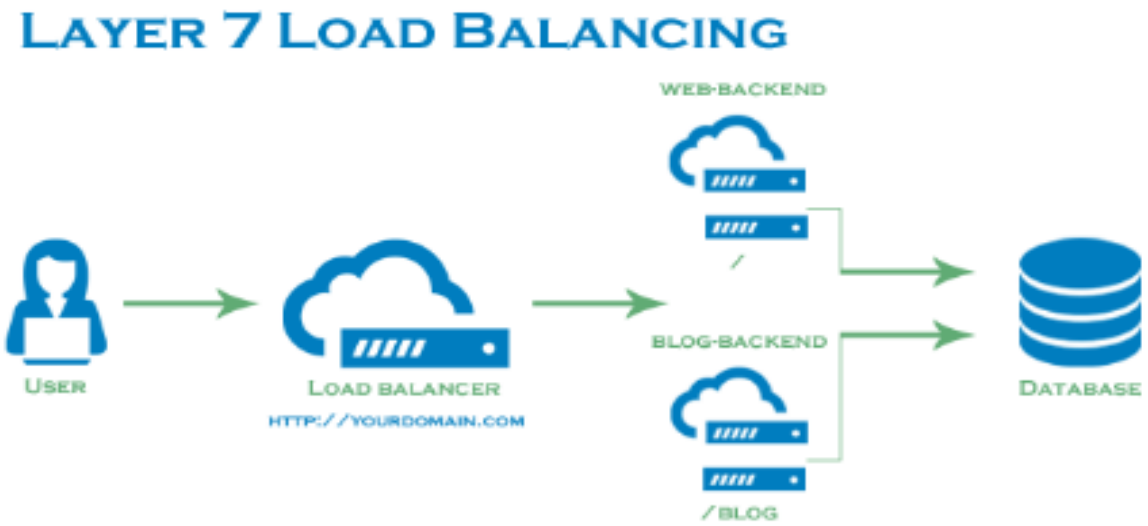
- In this example, the port on which the load balancer routes to the target server will often be port 80 (HTTP) or 443 (HTTPS).



ELB Types | Application Load Balancer

- AWS Application Load Balancer (ALB) operates at Layer 7(application) of the OSI model. At Layer 7, the ELB has the ability to inspect application-level content, not just IP and port.

- Example, an ELB at a given IP will receive a request from the client on port 443 (HTTPS). The Application Load Balancer will process the request, not only by receiving port, but also by looking at the destination URL.



Application Load Balancer | Enhanced features

Additional request support Protocols

Access Logs

Path and Host-based routing

Deletion Protection & Request Tracing

Application Load Balancer

Native IPv6 support

Dynamic ports

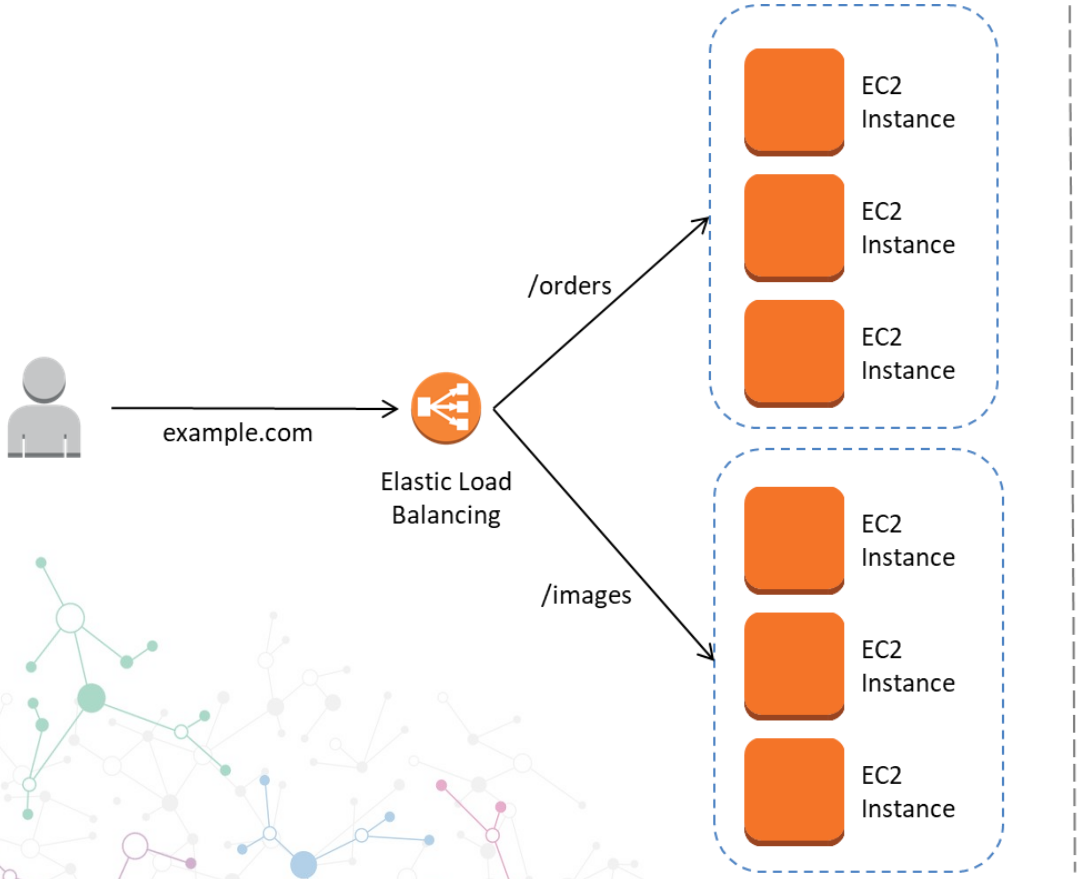
AWS WAF

CloudWatch Metrics

Health Checks

Application Load Balancer

- Multiple services can share a single load balancer using path-based routing. In the example given here, the client could request any of the following URLs:
  - *http://www.example.com/orders*
  - *http://www.example.com/images*

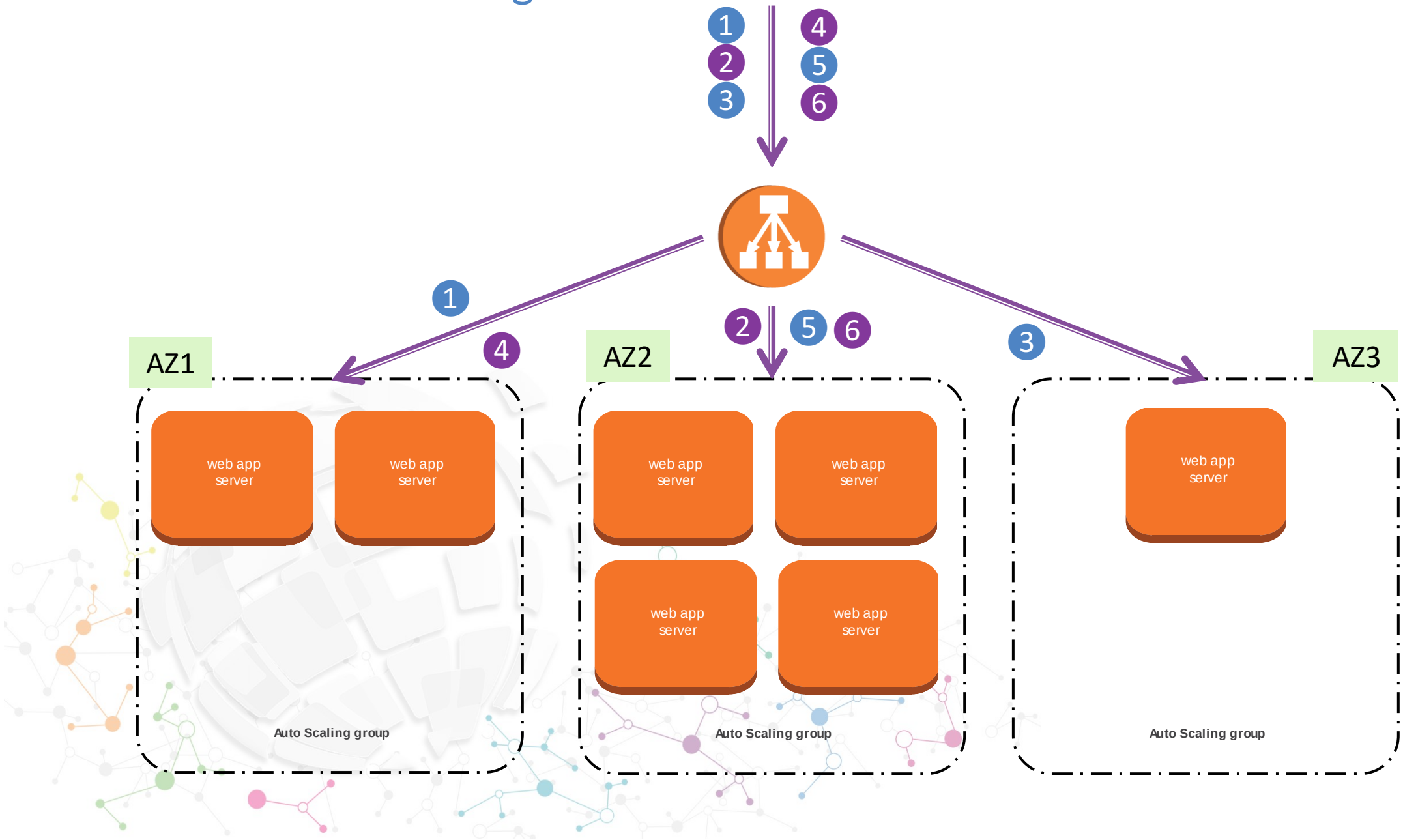


Application Load Balancer allows for multiple applications to be hosted behind a single load balancer

Cross Zone Load Balancing

- Configure Cross-Zone Load Balancing for Your Classic Load Balancer
- By default, your Classic Load Balancer distributes incoming requests evenly across its enabled Availability Zones. For example, if you have ten instances in Availability Zone us-west-2a and two instances in us-west-2b, the requests are distributed evenly between the two Availability Zones. As a result, the two instances in us-west-2b serve the same amount of traffic as the ten instances in us-west-2a. To ensure that your load balancer distributes incoming requests evenly across all instances in its enabled Availability Zones, enable cross-zone load balancing.
- Cross-zone load balancing reduces the need to maintain equivalent numbers of instances in each enabled Availability Zone, and improves your application's ability to handle the loss of one or more instances. However, we still recommend that you maintain approximately equivalent numbers of instances in each enabled Availability Zone for higher fault tolerance.
- For environments where clients cache DNS lookups, incoming requests might favor one of the Availability Zones. Using cross-zone load balancing, this imbalance in the request load is spread across all available instances in the region, reducing the impact of misbehaving clients.

Cross Zone Load Balancing



Elastic Load Balancer |Pricing

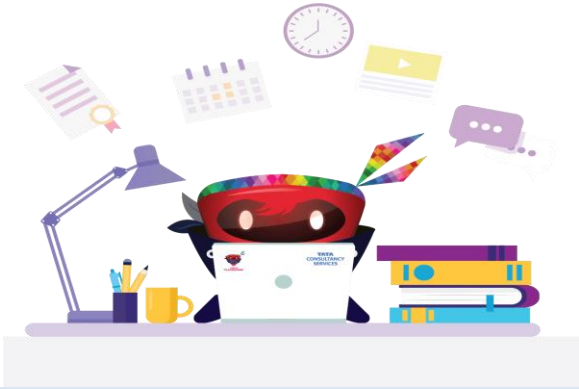
- ELB –Classic Load Balancer Pricing
  - ❑ *With Elastic Load Balancing, you only pay for what you use. You are charged for each hour or partial hour your load balancer is running and for each GB of data transferred through your load balancer. You will be charged at the end of each month for your Elastic Load Balancing resources actually consumed.*

    Refer: https://aws.amazon.com/elasticloadbalancing/classicloadbalancer/pricing/

- ELB –Application Load Balancer Pricing
  - ❑ *With the Application Load Balancer, you only pay for what you use. You are charged for each hour or partial hour your Application Load Balancer is running and the number of Load Balancer Capacity Units (LCU) used per hour.*

    Refer: https://aws.amazon.com/elasticloadbalancing/applicationloadbalancer/pricing/

AWS SAA Boot Camp

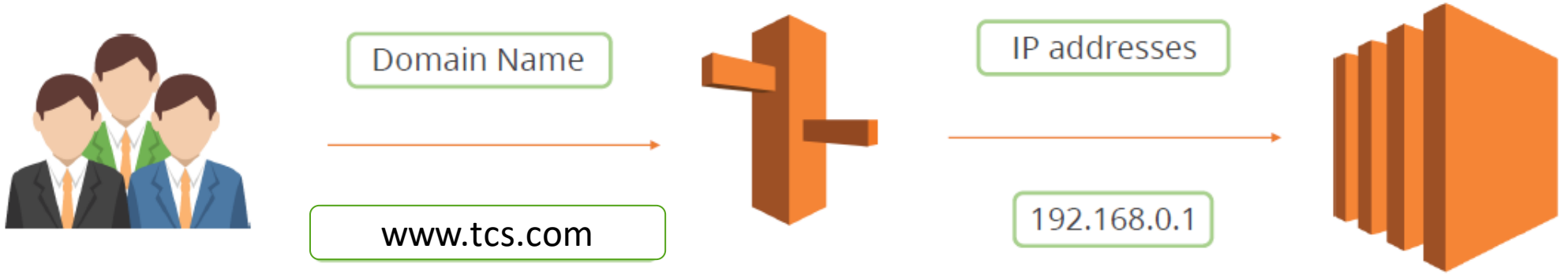


AWS Networking

Route 53

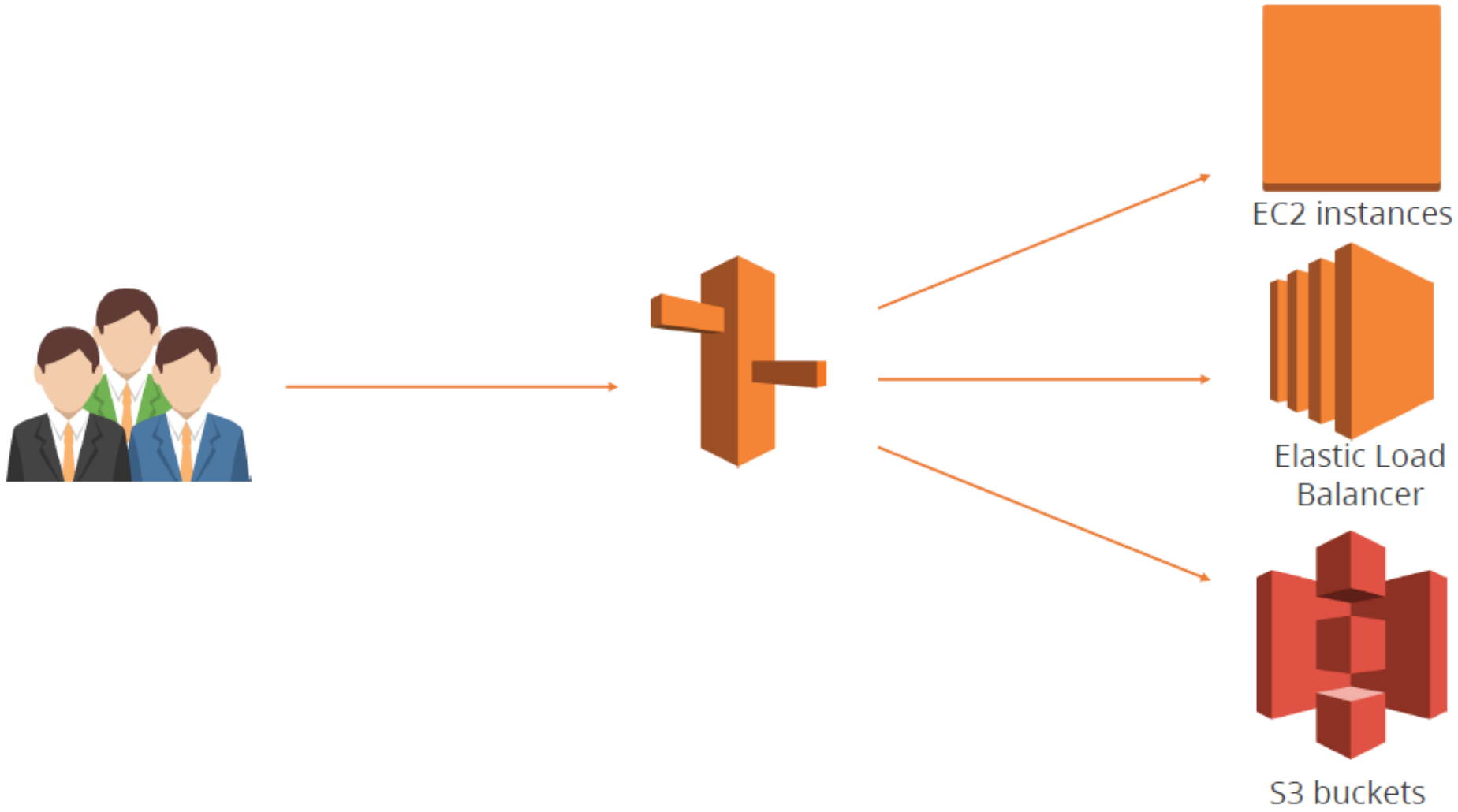
Domain Name Servers(DNS)

DNS provides a directory of domain names and translates them to IP addresses.



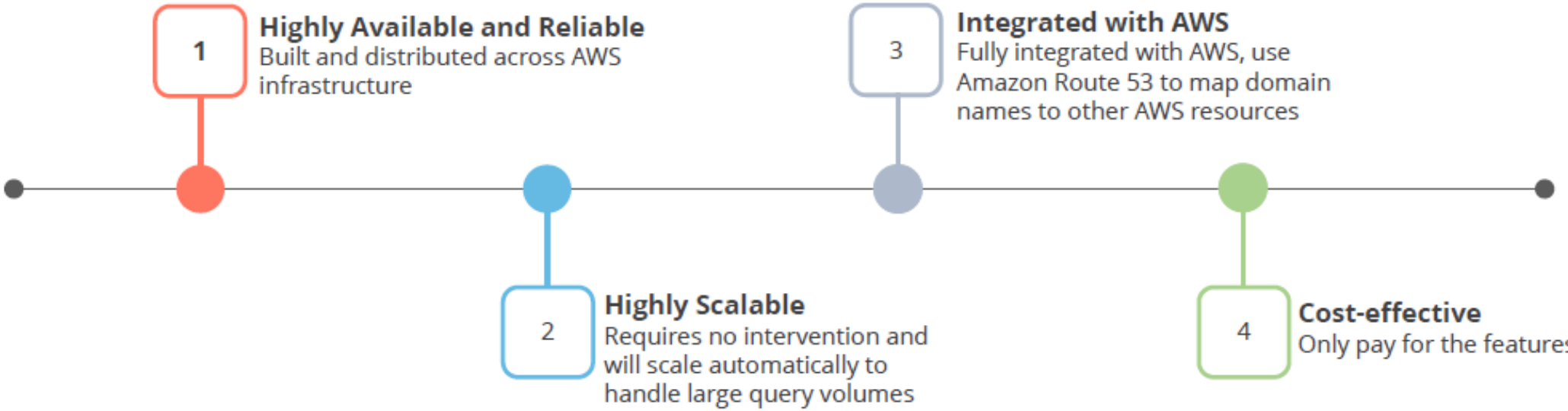
Route 53 Uses

You can use Route 53 to route user traffic to AWS resources like EC2 instances, ELB, or S3 buckets.



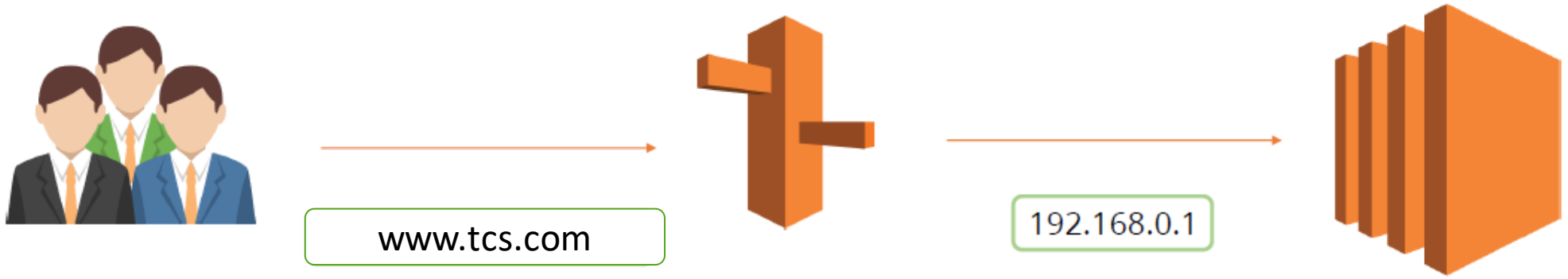
Route 53 Benefits

Following are the Route 53 benefits:



DNS Uses

DNS is used to translate domain names into IP addresses.



DNS - Terminologies

1. Domain Name

Human-friendly name for an Internet resource

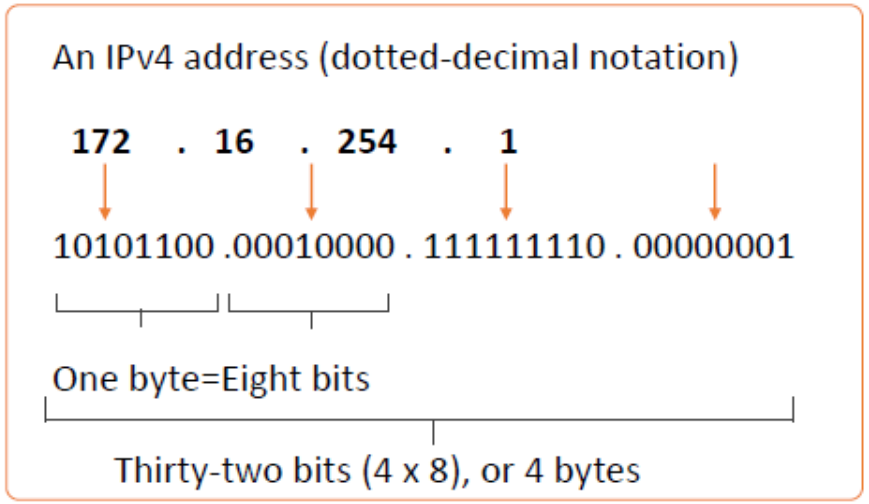
192.168.0.1

www.tcs.com

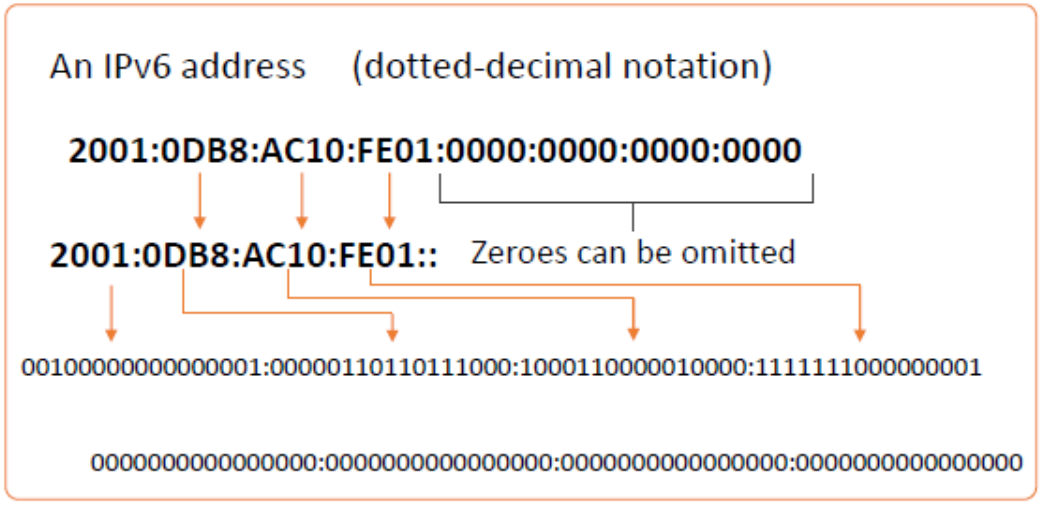
DNS Terminologies

2. IP Address

IP address is a network addressable location.
Each IP address has to be unique within its network. In your AWS VPC, you can have IP addresses like 10.0.1.0, but for websites, the network is the Internet, so a unique IP address is required.



10.0.1.0



DNS Terminologies

3. Top-Level Domain

A Top-Level Domain is the portion of the domain name furthest to the right.

DNS Terminologies

4. Hosts

The domain owner can define individual hosts within a domain that refers to separate services or computers.

DNS Terminologies

5. Subdomains

Subdomains are the parts that are underneath the top-level domain.

DNS Terminologies

6. Fully Qualified Domain Name (FQDN)

A Fully Qualified Domain Name, or FQDN, also called an absolute domain, is the complete domain name for a specific computer on the Internet.

DNS Terminologies

7. Name Server

A Name server is a computer or service that translates domain names to IP Addresses.

DNS Terminologies

8. Zone Files

Zone Files reside in name servers and are text files that contain the mappings between domain names and IP addresses.

```
$ORIGIN example.com
$TTL 86400
@       IN      SOA     dns1.example.com.     hostmaster.example.com. (
                        2001062501 ; serial
                        21600      ; refresh after 6 hours
                        3600       ; retry after 1 hour
                        604800     ; expire after 1 week
                        86400 )    ; minimum TTL of 1 day

        IN      NS      dns1.example.com.
        IN      NS      dns2.example.com.

        IN      MX      10      mail.example.com.
        IN      MX      20      mail2.example.com.

                IN      A       10.0.1.5

server1         IN      A       10.0.1.5
server2         IN      A       10.0.1.7
dns1            IN      A       10.0.1.2
dns2            IN      A       10.0.1.3

ftp             IN      CNAME   server1
mail            IN      CNAME   server1
mail2           IN      CNAME   server2
www             IN      CNAME   server2
```

DNS Terminologies

9. Start of Authority (SOA)

A Start of Authority, or SOA, record is mandatory for every domain.

DNS Terminologies

10.Time-To-Live (TTL)

Time-to-Live, or TTL is the length of time (in seconds) that a DNS record is cached on a DNS server or on your PC before it rechecks the details.

DNS Terminologies

11.Records

A record maps a resource to a name.

www.tcs.com → 192.182.239.21
www.google.com → 212.21.32.4

DNS Terminologies | Record Types

- A Record
- CNAME
- NS Record
- Alias Record

DNS Terminologies | Record Types

- A Record
- CNAME
- NS Record
- Alias Record

An “A Record” matches a domain (or subdomain) to an IP address.

```
example.com     A       12.34.56.78
```

DNS Terminologies | Record Types

- A Record
- CNAME
- NS Record
- Alias Record

Canonical Name (CNAME) record matches a domain or subdomain to a different domain.

```
alias.com       CNAME   example.com.
```

DNS Terminologies | Record Types

- A Record
- CNAME
- NS Record
- Alias Record

A NameServer Record (NS Record) stores information about the name servers for a domain.

```
example.com     NS      ns1.linode.com.
example.com     NS      ns2.linode.com.
```

DNS Terminologies | Record Types

- A Record
- CNAME
- NS Record
- Alias Record

An Alias Record is an AWS-created record and used only within AWS. It is similar to a CNAME, however, it's used to map DNS names to ELB, S3 buckets, and CloudFront distributions within your hosted zone.

Amazon Route 53| Routing policies

A routing policy determines how Amazon Route 53 responds to queries.
There are five available methods:

- Simple
- Weighted
- Latency
- Failover
- Geolocation

Amazon Route 53| Routing policies

Simple | Weighted | Latency | Failover | Geolocation

"Simple" is the default routing policy for a single resource.

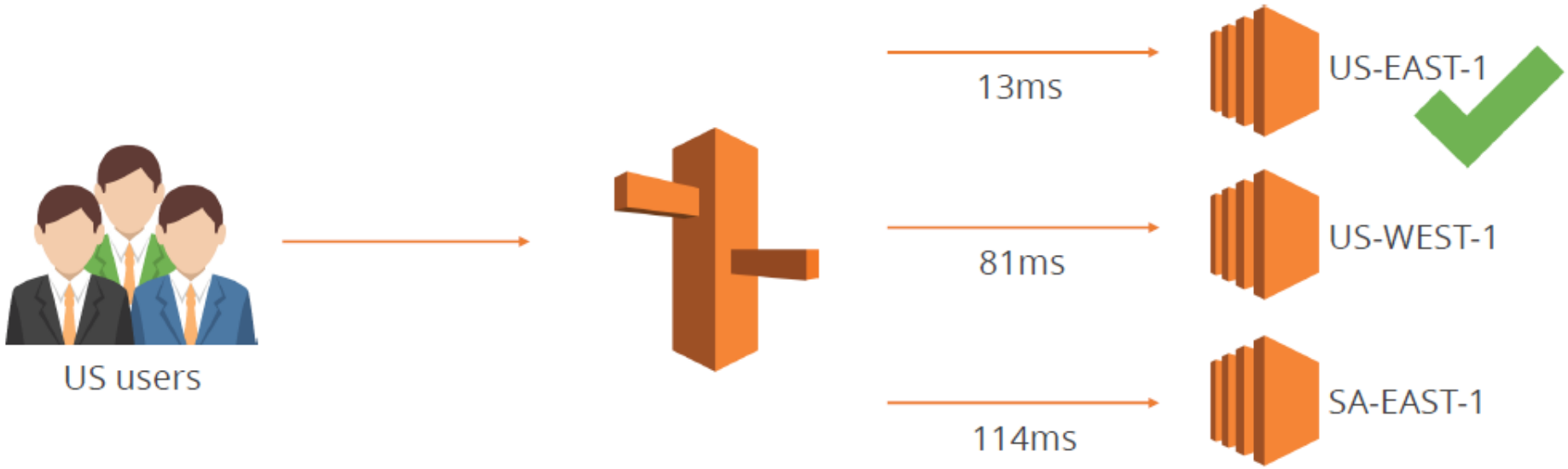
Simple | **Weighted** | Latency | Failover | Geolocation

"Weighted" routing policy can split traffic based on different weights assigned.

Amazon Route 53| Routing policies

Simple | Weighted | Latency | Failover | Geolocation

"Latency" routing policy allows you to route traffic based on the lowest network latency for your end user.



Amazon Route 53| Routing policies

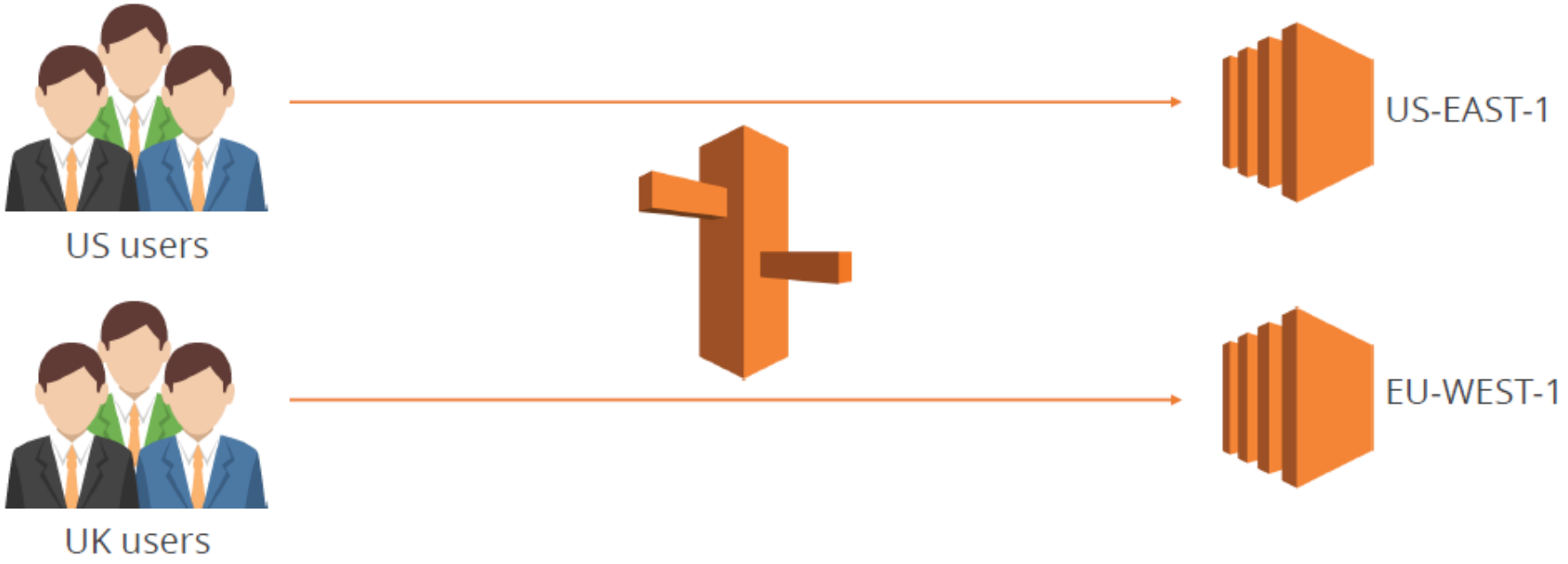
Simple | Weighted | Latency | Failover | Geolocation

"Failover" routing policy allows you to have an active/passive setup.

Amazon Route 53| Routing policies

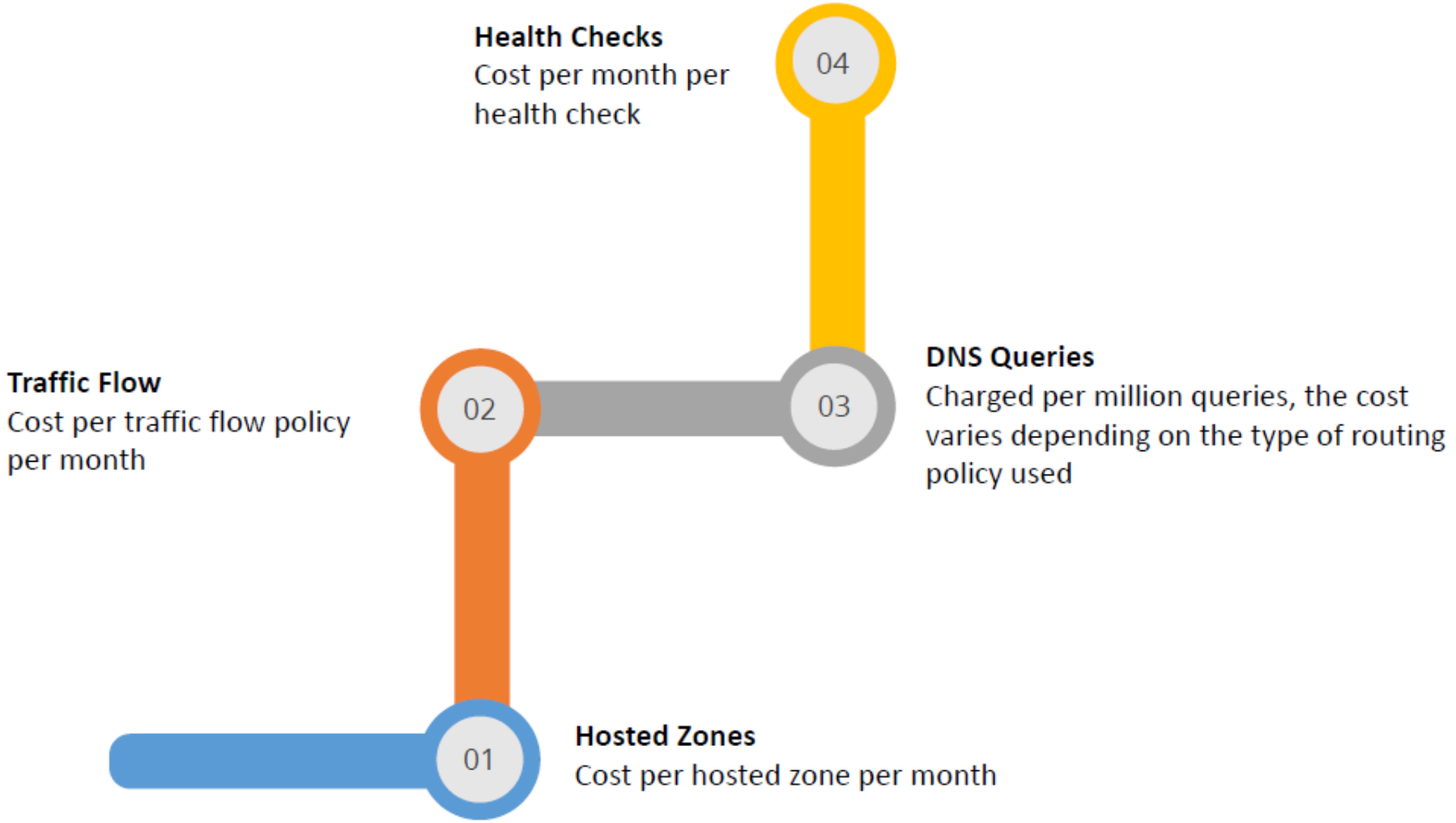
Simple | Weighted | Latency | Failover | Geolocation

"Geolocation" routing policy routes traffic based on the geographic location of your users.



Amazon Route 53 | Cost Overview

The diagram presents an overview of the costs associated with Route 53:



Thank You!

TATA CONSULTANCY SERVICES