Week 7 Networking

Dr Zhi Zhang

Overview

- Networking concepts
- Elastic load balancing

Network

- A network can be defined as a group of computers and other devices connected so as to be able to exchange data.
- Each of the devices on the network is a node and each node has a unique address.
- An address consists of numeric values that are easy for a device to work with, but not for humans to remember.
  - e.g., 204.160.241.98

IP Address

- It is an Internet Protocol address, which is a numerical label assigned to each device connected to a network that uses the Internet Protocol for communication.
- An IPv4 (Internet Protocol version 4) address has 4 bytes (32 bits) separated by periods.
  - e.g.,: 192.168.1.10
  - the first R bits correspond to the network portion.
  - the remaining H bits (32 – R) are used for the host portion.
  - subnet mask: determines the values for R and H.
    - e.g., IPv4 address "192.168.1.10" with a subnet mask of "255.255.255.0".
    - Question: what are the network portion and host portion for this IP address?
      - the network portion: "192.168.1" and the host portion: "10"
      - A different notation: 192.168.1.10/24
        - CIDR (Classless Inter-Domain Routing) notation

IP Address

- It is an Internet Protocol address, which is a numerical label assigned to each device connected to a network that uses the Internet Protocol for communication.
- An IPv4 address has 4 bytes separated by periods.
  - e.g.,: 192.168.1.10
  - the first R bits correspond to the network portion.
  - the remaining H bits are used for the host portion.
  - subnet mask: determines the values for R and H.
    - **it is used to cluster IP addresses into different subnets.**
      - e.g., 130.95.141.192 with a subnet mask of 255.255.255.192
      - question: how many possible IP addresses does this subnet have?
        - CIDR notation: 130.95.141.192/26
        - host number: 2^(32-26) = 64

A short summary

- To route a network packet on the Internet, its destination, i.e., IP address, is needed.
- IP address is decided by its network address and host address.
- Subnet mask is used for IP address management.

CIDR (Classless Inter-Domain Routing)

- CIDR is a notation to represent IP addresses and their associated subnet masks.
- In CIDR notation, an IP address is followed by a slash ("/") and a number. This number represents the bit length of the network portion.

CIDR Practice

- A: 172.16.17.30/20
- B: 172.16.28.15/20
- Question: Is B in the same subnet with A?
- For A:

  20
  - 172.16.17.30: 10101100.00010000.00010001.00011110
  - 20 specifies the bit length of network portion
  - first IP address in this subnet: **10101100.00010000.0001**0000.00000000 = 172.16.16.0
  - last IP address in this subnet: **10101100.00010000.0001**1111.11111111 = 172.16.31.255
- For B, it falls in the range of the subnet above.

CIDR Practice

- A: 172.16.17.30/20
- B: 172.16.28.15/20
- Question: Is B in the same subnet with A?
- For A:

  20
  - IP address: 172.16.17.30: 10101100.00010000.00010001.00011110
  - Subnet mask: 255.255.240.0: **11111111.11111111.1111**0000.00000000
  - First IP address in this subnet can be done via a bitwise AND on the IP address and subnet mask
    - 172.16.17.30: 10101100.00010000.00010001.00011110
    - 255.255.240.0: 11111111.11111111.11110000.00000000

      -------------------------|AND|---------------------------
    - First address: 10101100.00010000.00010000.00000000 = **172.16.16.0**
  - Last address is done using a bitwise OR on the address and the complement of the subnet mask
    - 172.16.17.30: 10101100.00010000.00010001.00011110
    - 255.255.240.0: 00000000.00000000.00001111.11111111

      ------------------------|OR|-------------------------
    - Last address: 10101100.00010000.00011111.11111111 = **172.16.31.255**

Network

- A network can be defined as a group of computers and other devices connected so as to be able to exchange data.
- Each of the devices on the network is a node and each node has a unique address.
- Addresses are numeric values that are easy for computers to work with, but not for humans to remember.
  - e.g., 204.160.241.98
- **Some networks provide unique domain names that humans can remember.**
  - e.g., www.javasoft.com, corresponding to an IP address.

Domain Name

- It uses alphanumeric characters and symbols separated by periods to create a hierarchical naming structure. This structure is organized from right to left.
  - For www.javasoft.com,
    - "com" is the top-level domain (TLD).
    - " javasoft" is the second-level domain (SLD).
    - "www" is a subdomain of " javasoft."

DNS (Domain Name System)

- Domain names are also known as mnemonic textual Internet addresses.
- DNS servers are responsible for translating mnemonic textual Internet addresses into the numeric addresses.

- Question: what is the difference between a domain name and a URL?
  - A quick answer: a domain name is a part of a URL.

Domain name and URL

- Domain name: a human-readable hierarchical structure, which consists of alphanumeric characters and symbols separated by periods.
- URL (Uniform Resource Locator): a complete address to locate a specific resource on a given website. It consists of:
  - a network protocol,
  - a domain name,
  - an additional path or query parameters.
  - e.g., https://www.example.com/products/category?id=123&sort=asc
    - the protocol is "https", the domain name is "www.example.com", the path is "/products/category";
    - the query parameters are "?id=123&sort=asc", which requests specific items of the category of products via "id=123" in an ascending order via "sort=asc".

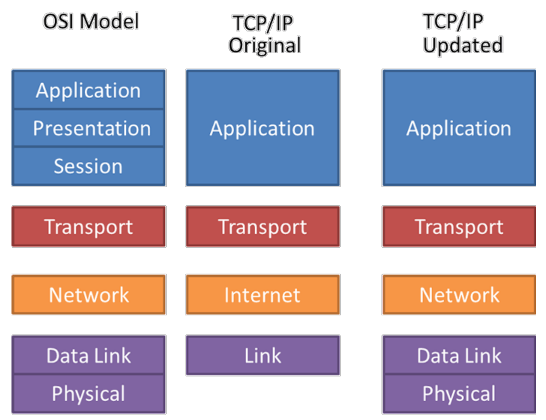
Port

- An IP address identifies a host machine on the Internet.
- An IP port identifies a specific application protocol running on an Internet host machine.
- A port is identified by a number, the port number.
- There are some port numbers which are allocated for specific application protocols.
  - e.g.,

Application Protocols	Default Port Numbers
HTTP	80
HTTPS	443
Telnet	23
SSH	22

Protocol

- A set of rules that govern how data is transmitted over a network.
  - Examples: TCP (Transmission Control Protocol), IP (Internet Protocol), HTTP (Hypertext Transfer Protocol)
- **Each protocol is designed based on a layered model.**
  - A real-world model : **TCP/IP**.
  - A conceptual model : **OSI (Open Systems Interconnection)**



https://fiberbit.com.tw/tcpip-model-vs-osi-model/

5-layer TCP/IP model

- **Application Layer** provides network services directly to applications. It hosts various application-specific protocols.
  - Examples of protocols for different applications: HTTP and HTTPS for web browsing, Telnet (Telecommunication Network) for remote access, SSH (Secure Shell) for secure remote access.

5-layer TCP/IP model

- **Transport Layer** provides protocols to establish and manage end-to-end network connection between applications running on different hosts.
  - Particularly, it keeps track of the applications in the above layer by assigning port numbers to them.
  - Examples of protocols for network connection: TCP (Transmission Control Protocol) for reliable connections, UDP (User Datagram Protocol) for fast communication.

5-layer TCP/IP model

- **Network Layer** is responsible for routing packets of data to reach their destination.
  - Particularly, it manages packet addressing and determines the best path.
  - Examples of protocols for packet routing: IP (Internet Protocol), ICMP (Internet Control Message Protocol)

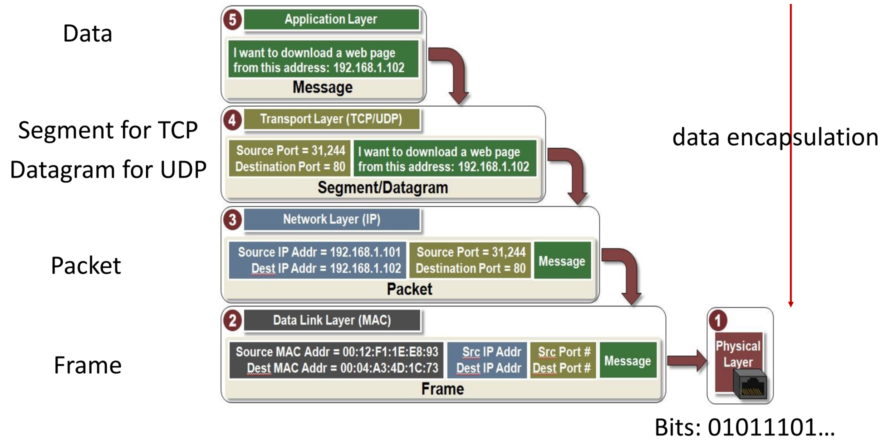
5-layer TCP/IP model

- **Data Link Layer** provides node-to-node data transmission.
  - Particularly, it uses MAC (Media Access Control) addresses to uniquely identify and address individual network devices.
  - Examples of protocols for node addressing: Ethernet, Wi-Fi

5-layer TCP/IP model

- **Physical Layer** deals with data transmission over a physical medium.
  - Examples of medium: Optical fibers, Wireless radio waves

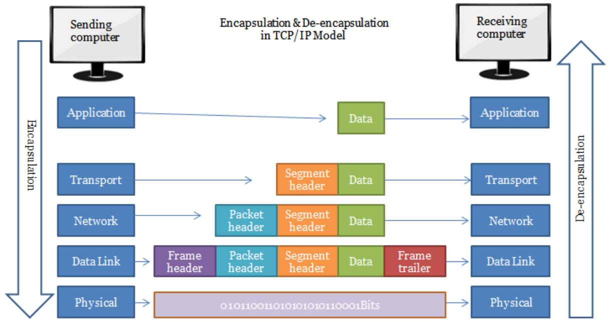
Transmit Data with TCP/IP model



Data transmission in a network

- The data transmitted, goes back and forth through the layers, with a protocol implemented in each layer adding or removing its own header/trailer.
  - **data encapsulation** (sending data): At each layer in the sending device, the data is encapsulated with a header/trailer specific to a protocol in layer. The process continues from the topmost layer to the bottommost layer.
  - **data decapsulation** (receiving data): The bottommost layer in the receiving device receives the encapsulated data. At each layer, relevant header/trailer is removed and the remaining data is passed to an upper layer. This process continues up to the topmost layer.

Data encapsulation/decapsulation in TCP/IP



https://nglthu.github.io/Networking/

ELB (Elastic Load Balancing)

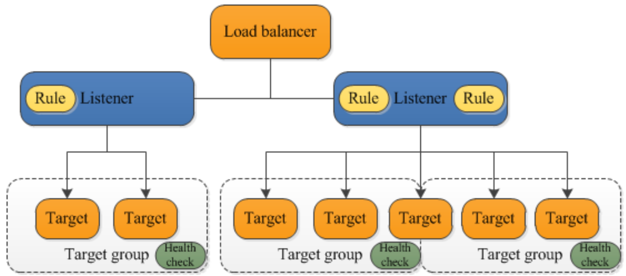
- ELB automatically distributes incoming network traffic across multiple targets, into one or more Amazon Availability Zones.
  - The targets can be EC2 instances, containers, IP addresses, etc.

Benefits of ELB

- Increases the availability and fault tolerance of our applications.
  - compute resources (targets) can be added or removed, allowing **horizontal scaling**
  - What are horizonal scaling and vertical scaling?
- Enables health check of our compute resources.
  - Healthy: a compute resource is responsive and functioning as expected.

How ELB works

- An ELB serves as the single point of contact for clients, distributing incoming requests across multiple target groups of multiple targets.
- A listener checks for network requests from clients, using the network protocol and port that we configure (the configuration is defined in a rule).
- Each target group routes requests to one or more registered targets.

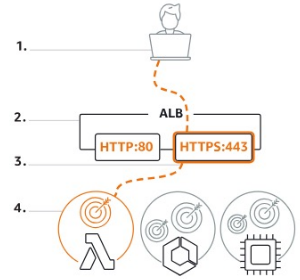


ALB (Application Load Balancer)

- ALB is a primary type of ELB.
  - it works at the application layer of the OSI model.

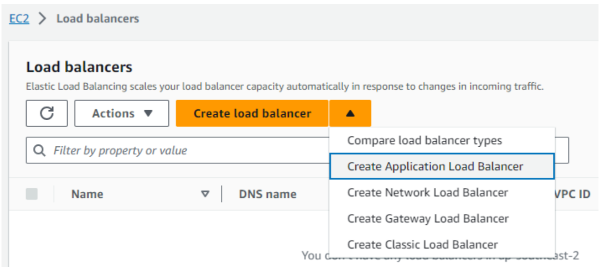
How ALB works

1. Clients make HTTP or HTTPS requests to our application.
2. The listeners in our load balancer receive requests matching the protocol and port that we configure.
3. The receiving listener evaluates the incoming requests against the rules we have configured, and routes the request to the appropriate target group.
4. The targets in one or more target groups receive requests based on the routing rules configured in the listeners.



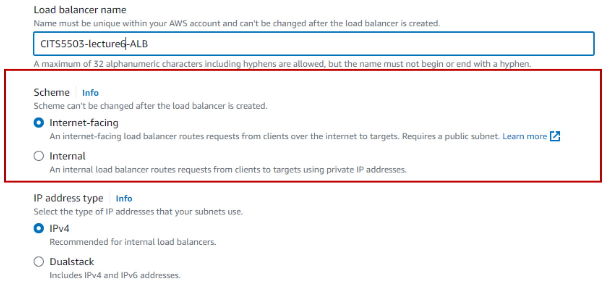
Set up an ALB

- Navigate to EC2 Dashboard and Click "Load balancers"



Set up an ALB

- Basic configuration

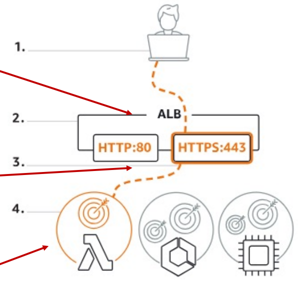


Scheme

- Internet-facing indicates the ALB has a public IP address.
- Internal means the ALB has a private IP address.

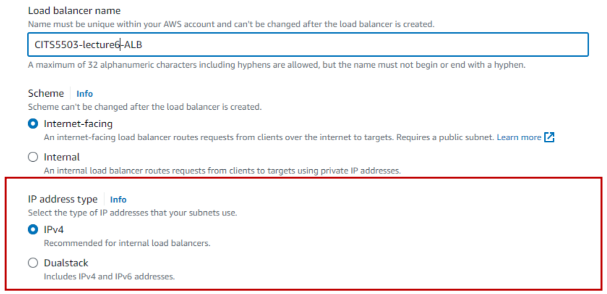
- Both schemes route the client's requests to targets' private IP addresses.

- Targets do NOT need public IP addresses to receive requests from the ALB.



Set up an ALB

- Basic configuration



IP address type

- IPv4 (Internet Protocol version 4)
- Dualstack includes IPv4 and IPv6 (Internet Protocol version 6)

- IPv4: it is widely used in real-world.
  - What is the main reason why IPv6 is needed?
    - A quick answer: all IPv4 addresses will be exhausted in the foreseeable future.

IP address type

- IPv4
- Dualstack includes IPv4 and IPv6 (Internet Protocol version 6)

- IPv4: it is widely used in real-world.
  - What is the main reason why IPv6 is needed?
    - It uses an address format with 4 bytes (32 bits)
    - It allows for about 4.3 billion unique IP addresses: 2^8 * 2^8 * 2^8 * 2^8 = 2^32
    - The number of internet-connected devices has been increasing all the time.
  - How to address the IPv4 exhaustion problem?
    - Fundamental solution: IPv6.
    - Mitigation: network address translation (NAT) and private IP address ranges.

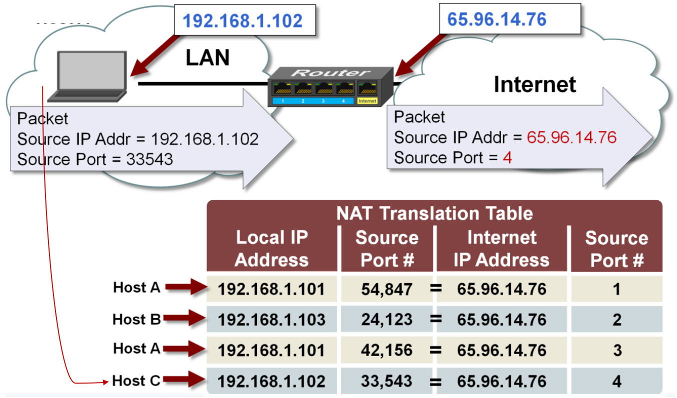
NAT and private IP address range

- NAT: it is a way to map multiple private IPv4 addresses inside an internal network to a public IPv4 address before transmitting data to the internet.
  - All the devices in an internal network use a single public IPv4 address.

- Private IPv4 address range: it is a reserved IP address block that is not routable on the internet.
  - It is used for internal communication among devices within the same internal/private network.
  - Three primary address ranges:

Range	CIDR	Total Addresses
10.0.0.0 to 10.255.255.255	10.0.0.0/8	2^24 (24 = 32 – 8)
172.16.0.0 to 172.31.255.255	172.16.0.0/12	2^20 (20 = 32 – 12)
192.168.0.0 to 192.168.255.255	192.168.0.0/16	2^16 (16 = 32 – 16)

How NAT works



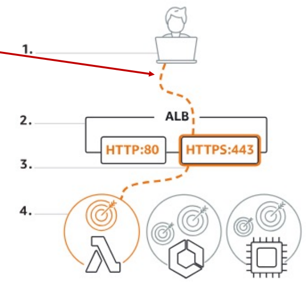
IPv6

- It is most recent version of IP and uses an address format with 16 bytes (128 bits)
  - It is expressed in hexadecimal notation with colons, e.g., 2001:0db8:85a3:0000:0000:8a2e:0370:7334
  - It allows for a much larger number of addresses: 2^128.

IP address type

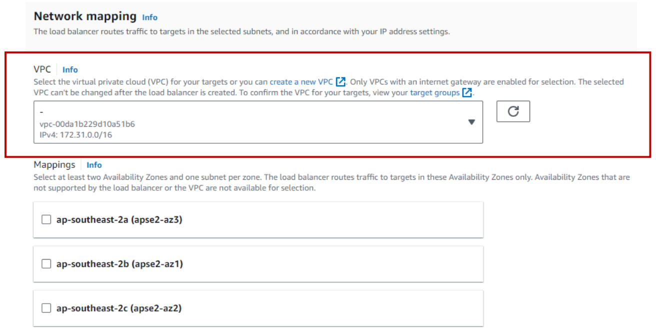
- Choose IPv4 if the client chooses an IPv4 address to communicate with the ALB.
- Choose Dualstack if the client uses an IPv6 address.

- IPv4 is only supported if the ALB uses the internal scheme.



Set up an ALB

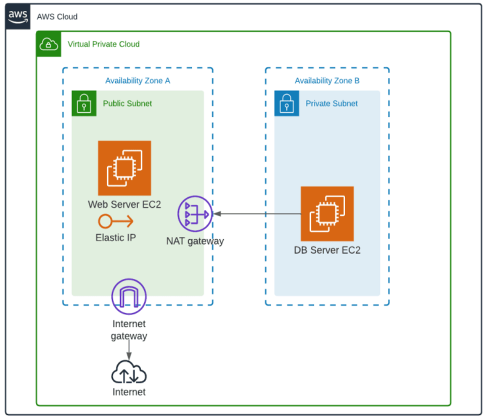
- Network configuration



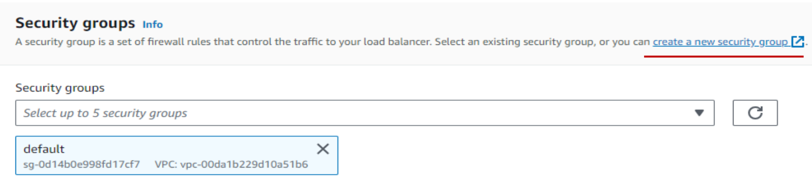
VPC

- VPC (virtual private cloud): it is a virtual network dedicated to an AWS account.
  - It is logically isolated from other virtual networks in the AWS Cloud.
- Internet gateway: provides VPC with internet access
  - Manage outbound traffic to the internet and inbound traffic to the AWS resources.
- Subnet: it is a range of IP addresses in a VPC.
  - It is used to divide a VPC into multiple logical sub-networks.
  - It can be either public or private sub-network.
  - It must reside entirely within one Availability Zone.

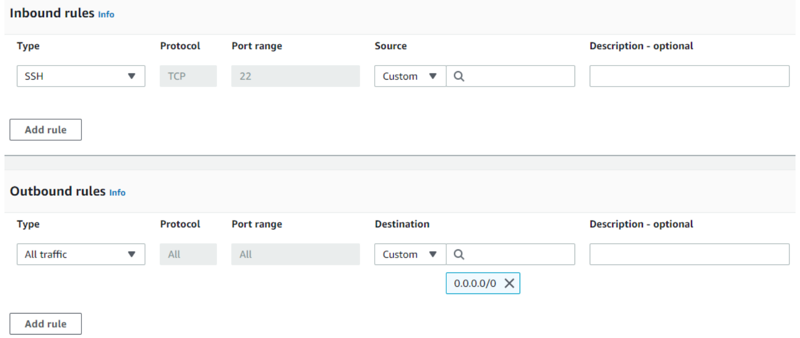
An example



Security group



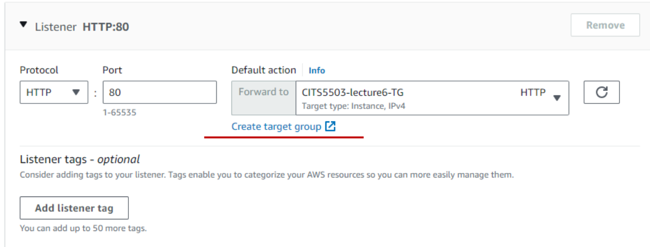
Security group



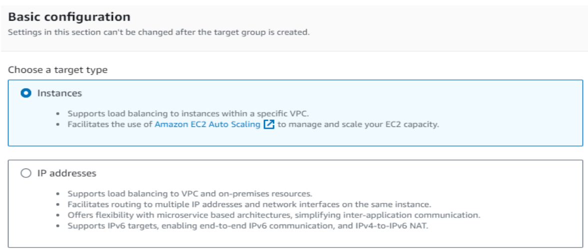
[a] Create the load balancer and specify the two region subnets and a security group (note that the security group should authorise inbound traffic for HTTP, which is used by the following step [d])

Listeners and routing

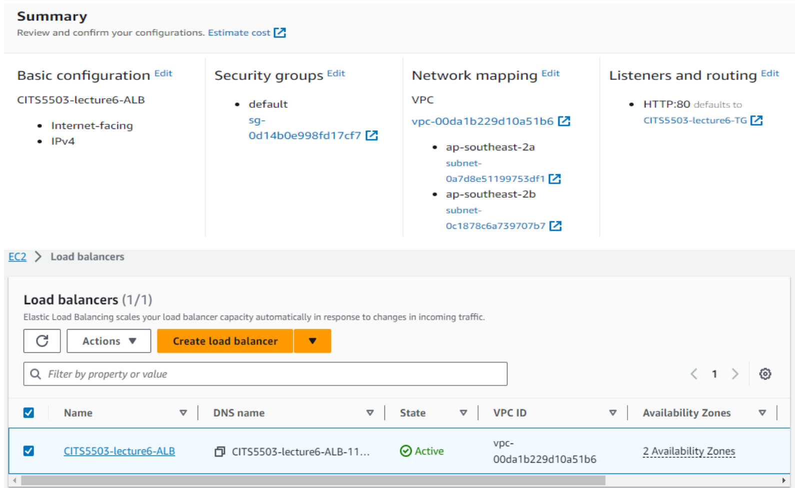
- Listener: it is a process that checks for connection requests using the protocol and port we configure.
- A Listener in ALB supports protocols and ports:
  - Protocols: HTTP, HTTPS with Ports: 1-65535

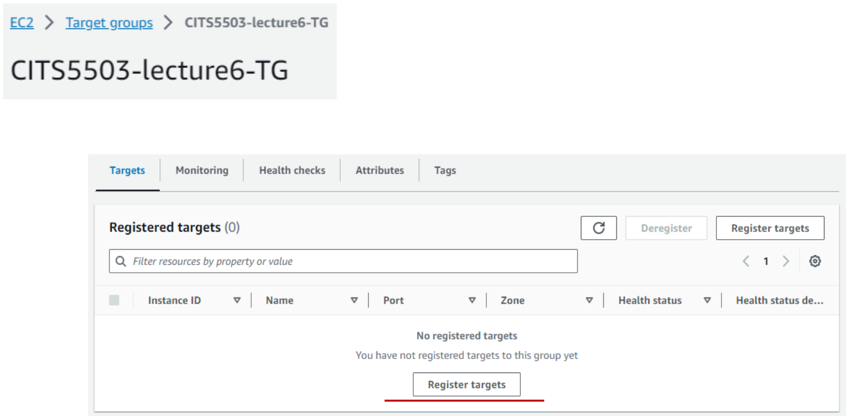


Create target group



- Instance: specify targets via EC2 instance IDs.
- IP Address: specify targets via IP addresses or IP address ranges.
  - IP addresses can be related to other AWS resources such as containers.





Practice Questions

- [13 marks] Q1: Discuss 3 reasons why you would use Application Load Balancing and how this would be set up to load balance a Python Django application. Specifically, describe the configuration of the Listener and Target Group running the Python Django application.

Reasons

- [2 marks] **High fault-tolerance: the** ALB can distribute traffic to multiple targets in multiple groups, making the Django application healthy and improving its fault-tolerant.
- [2 marks] **High scalability**: As the ALB can distribute traffic evenly, the Django application can be scaled horizontally. For example, when traffic increases, more targets can be added, and the ALB can distribute traffic to them.
- [2 marks] **Good match**: Django is a web framework accepting http and https requests. The main responsibility of ALB is to optimize http and https traffic.

Practice Questions

- [13 marks] Q1: Discuss 3 reasons why you would use Application Load Balancing and how this would be set up to load balance a Python Django application. Specifically, describe the configuration of the Listener and Target Group running the Python Django application.

How?

- [4 marks] **Configure a Target Group:**
  - Create a target group that will host the targets running the Django application.
  - Set health check protocol on which Django application is running, e.g., HTTP.
  - Register targets running the Django application within the target group.
- [3 marks] **Configure Listeners:**
  - Create HTTP and/or HTTPS listeners within the ALB. Particularly, port should be specified, e.g., port 80 for HTTP and port 443 for HTTPS.