To create both **internal** and **external** load balancers we would typically follow a set of steps within a cloud environment (like AWS, Azure, or GCP) or within your on-premise infrastructure (using tools like NGINX, HAProxy, or similar).

For this example, let’s assume you're working with **AWS**, and you want to set up **Application Load Balancers (ALB)** for both internal and external access. I'll walk you through the steps and verify their functionality.

**Step-by-Step Guide**

**1. Create External Load Balancer (Public ALB)**

* **Purpose:** This load balancer is accessible from the public internet and routes requests to your public-facing web servers.

**Steps:**

* Go to the AWS Management Console.
* Navigate to **EC2 > Load Balancers**.
* Click **Create Load Balancer**.
* Choose **Application Load Balancer**.
* **Name:** Enter a name (e.g., my-external-alb).
* **Scheme:** Select **Internet-facing** (this means the load balancer is public).
* **Listeners:** Choose HTTP or HTTPS (depending on your setup).
* **VPC:** Choose the VPC where your instances are deployed.
* **Subnets:** Select public subnets for the load balancer.
* **Security Groups:** Attach an existing security group or create a new one (ensure the necessary ports like 80/443 are open).
* **Target Group:** Create a new target group, select the target type (e.g., instance, IP, or Lambda function), and configure health checks.
  + Set the health check path, such as / for a basic health check.
* **Review and Create**: Review your configuration and click **Create**.

**2. Create Internal Load Balancer (Private ALB)**

* **Purpose:** This load balancer routes traffic between private resources (e.g., EC2 instances that don’t need to be exposed to the internet).

**Steps:**

* Repeat the above steps, but:
  + Select **Internal** under the **Scheme** option (private).
  + Choose private subnets for the internal load balancer.

**3. Configure Auto Scaling Group**

* If you want your load balancers to scale automatically, you can configure an **Auto Scaling Group** that integrates with your load balancers.
* Go to the **Auto Scaling Groups** section in the EC2 console, and create a new group with a policy to scale based on load.

**4. Verify the Load Balancer Configuration**

**a. Test External Load Balancer (Public ALB):**

1. **DNS Resolution:** After creating the ALB, AWS will assign a DNS name. You can find it in the **Description** section of the load balancer in the AWS console.
   * Example: my-external-alb-1234567890.us-west-2.elb.amazonaws.com.
2. **Test Connectivity:**
   * Open a browser or use curl to access the DNS name. Example:

bash

CopyEdit

curl http://my-external-alb-1234567890.us-west-2.elb.amazonaws.com

* + The result should be the response from your web server.

**b. Test Internal Load Balancer (Private ALB):**

1. **VPC Connectivity:** The internal load balancer should only be accessible from within the VPC or connected networks (e.g., VPN or Direct Connect).
2. **Test Connectivity:**
   * SSH into an instance within the same VPC (preferably within a private subnet).
   * Use curl to test the private DNS of the load balancer:

bash

CopyEdit

curl http://my-internal-alb-1234567890.us-west-2.elb.amazonaws.com

* + The result should be the response from the target servers behind the internal load balancer.

**5. Verify Target Health**

* Go to **EC2 > Target Groups** in the AWS Management Console.
* Select the target group associated with your load balancer.
* Check if the targets (EC2 instances) show as **healthy**. If the targets are not healthy, verify the health check configurations.

**6. Monitor Load Balancer Logs**

* Enable **Access Logs** for detailed logging of requests going through the load balancer. This can be helpful for troubleshooting.
* Navigate to **Load Balancers > (select your ALB) > Attributes** and enable access logging.

**7. Troubleshooting**

* **Check Security Groups:** Ensure the security groups associated with your load balancer and EC2 instances allow traffic on the required ports (e.g., 80, 443 for HTTP/HTTPS).
* **Check NACLs (Network Access Control Lists):** Ensure that the network ACLs are not blocking access to the load balancer.

**8. Clean-Up**

If you are done testing, make sure to **delete** your load balancers and related resources (e.g., target groups) to avoid unnecessary costs.