Elastic Load Balance

Getting Started

Issue   01

Date   2023-03-30

Contents

1 Overview

You can create a load balancer to distribute traffic across backend servers.

Two examples are given to show how you can quickly create a load balancer to distribute incoming traffic across backend servers.

- **Entry level**: An application deployed on separated ECSs needs to handle a large number of requests. Health checks are required to monitor the health of the servers to ensure that incoming traffic is routed only to healthy servers to eliminate SPOFs and improve service availability.

Figure 1-1 Entry level
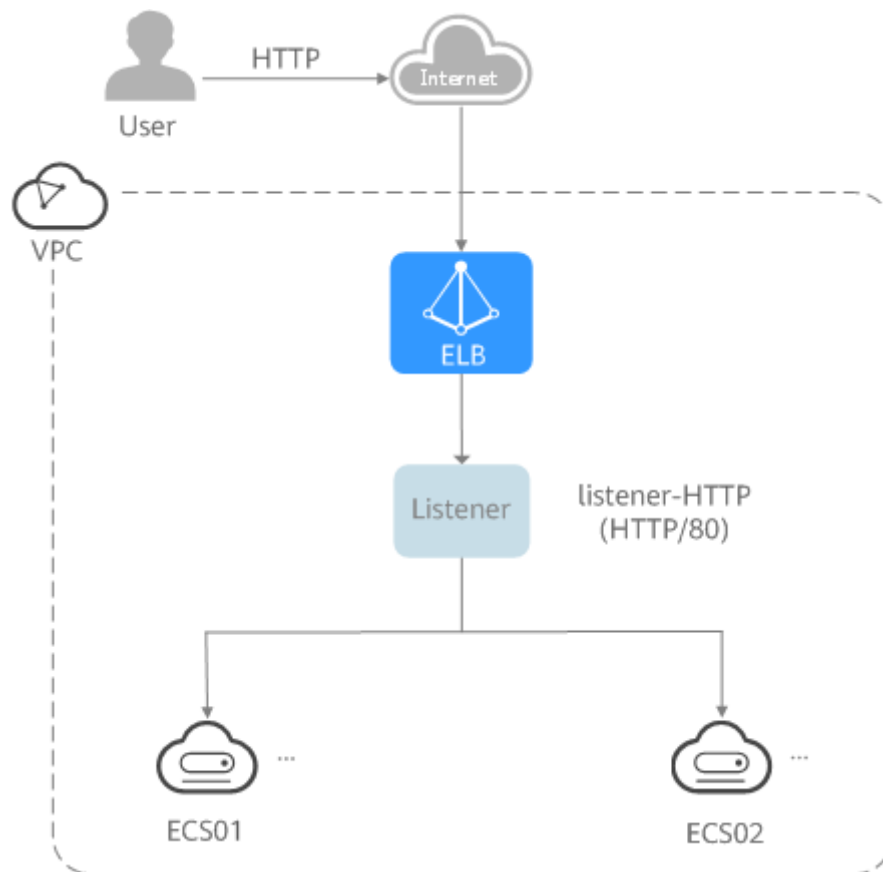
As the incoming traffic increases, you can add more servers to balance the load across backend servers.

●   **Advanced level**: An application deployed on separated servers uses one domain name but different URLs to provide services, and requests are routed to different servers based on the URLs. Forwarding policies are required to forward requests from different URLs to the servers in the corresponding backend server groups.

Figure 1-2 Advanced level
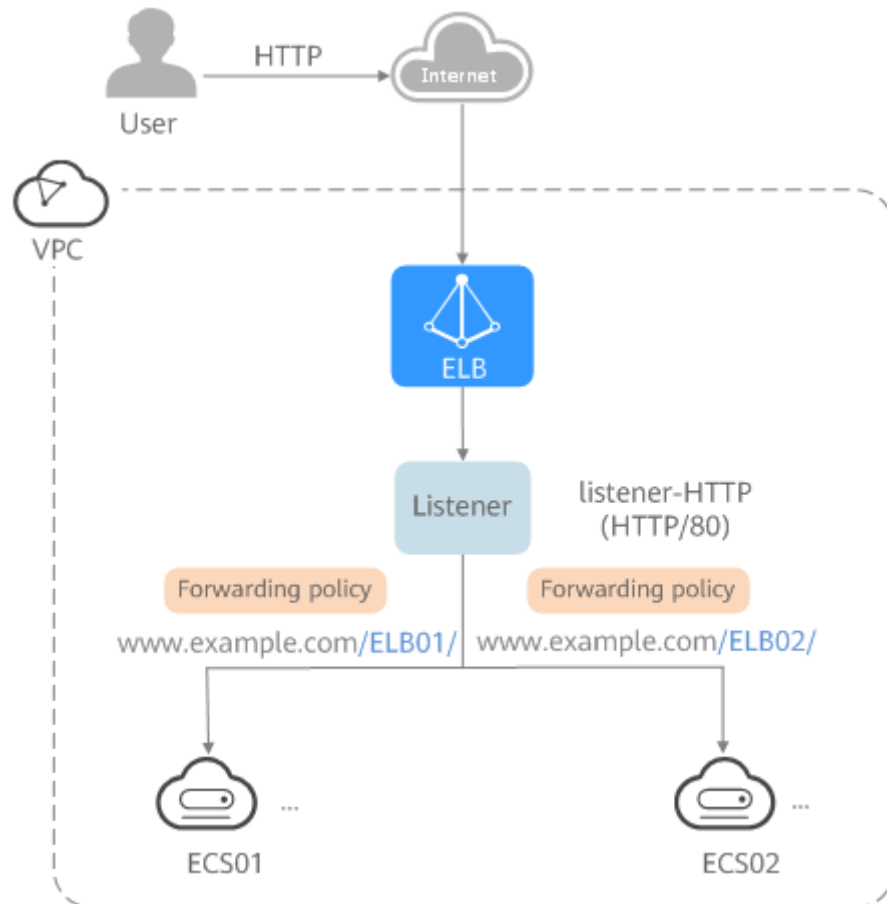
As the incoming traffic increases, you can add more backend servers to the two backend server groups. You can also configure health checks to monitor the health of backend servers to ensure that incoming traffic is routed only to healthy backend servers.

2 Process Flowchart

Figure 1 shows how you can use basic functions of ELB to route requests when you are still new to ELB, and Figure 2 shows how you can use domain name or URL-based forwarding of ELB to route requests more efficiently.

Figure 2-1 Getting started - entry level

Figure 2-2 Getting started - advanced level

3 Preparations for Using ELB

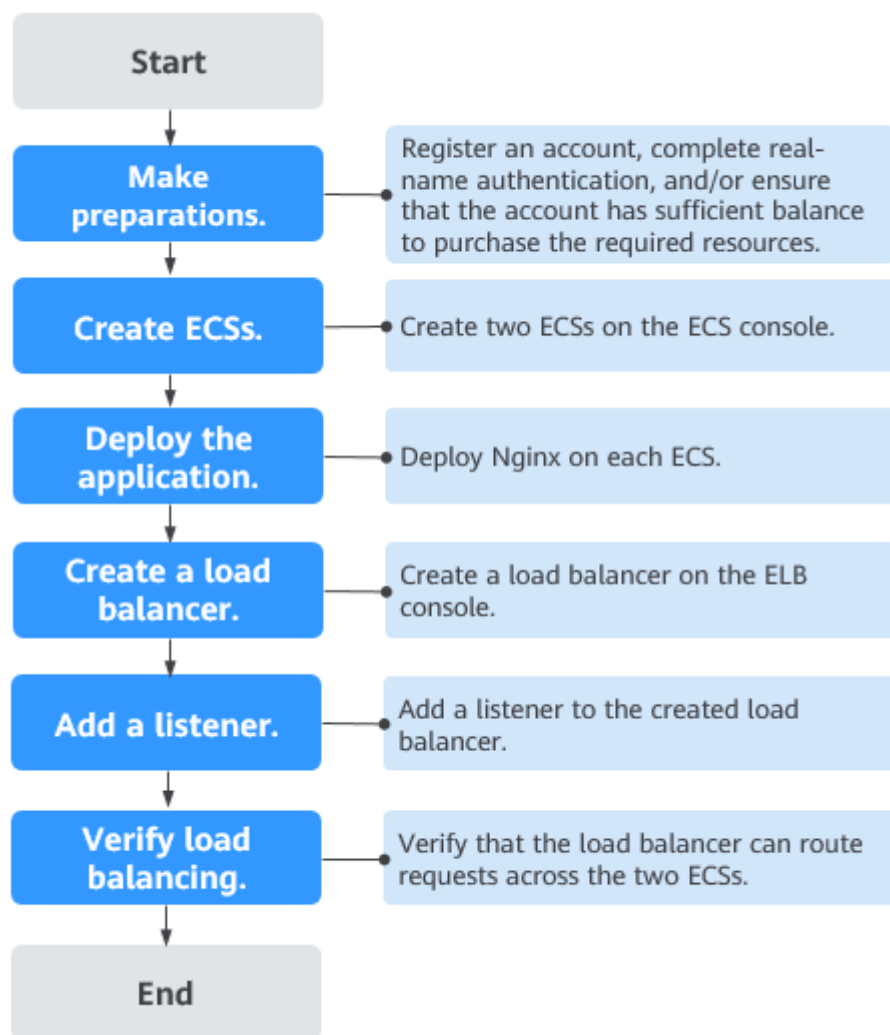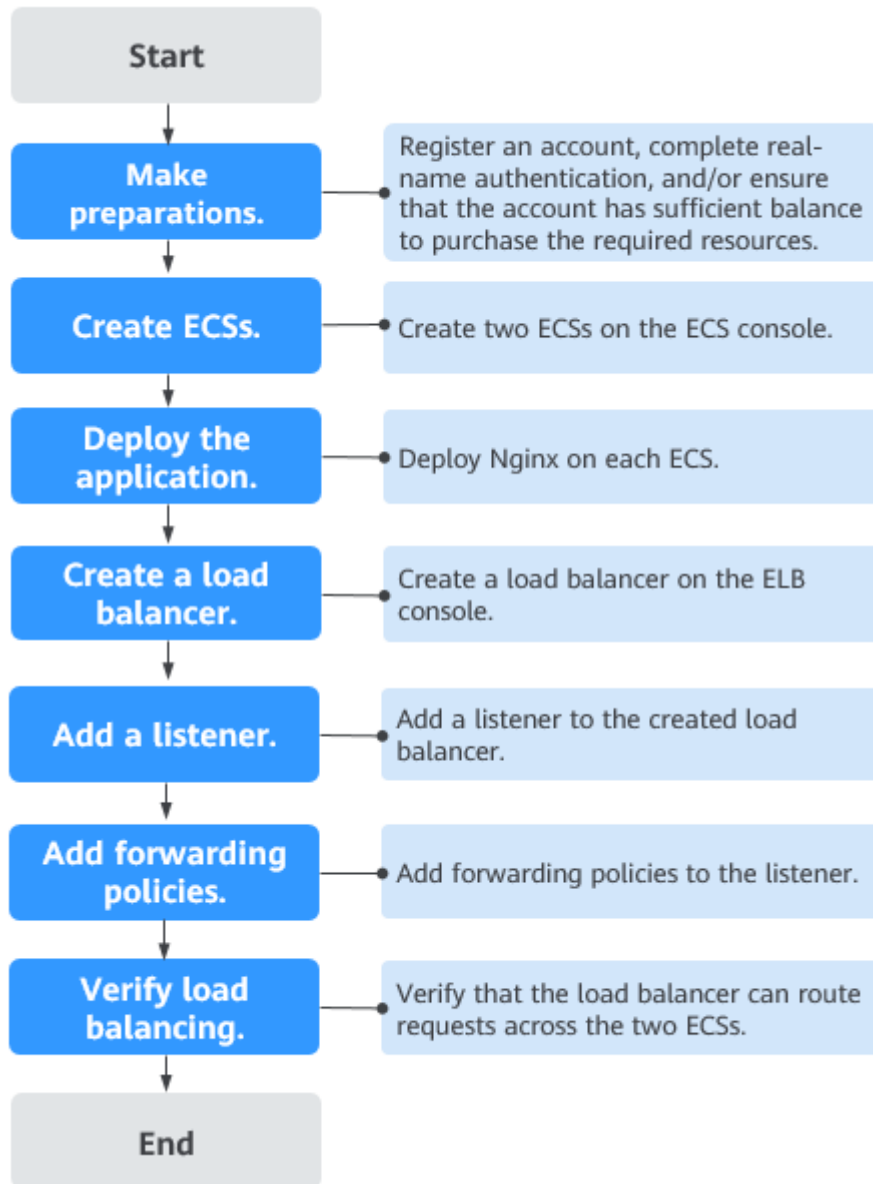Before using ELB to route requests, you need to make some preparations.

- **Registering a HUAWEI CLOUD account and completing real-name authentication**
- **Topping Up Your Account**
- **Creating an IAM User**

Registering a HUAWEI CLOUD Account and Completing Real-Name Authentication

If you already have a HUAWEI CLOUD account and completed real-name authentication, skip this part. If you do not have a HUAWEI CLOUD account, perform the following steps to register one:

1. Visit the **HUAWEI CLOUD official website** and click **Register**.
2. On the displayed **Register** page, register an account as prompted.

   If the registration is successful, the system automatically redirects you to your personal information page.
3. Complete real-name authentication by following the instructions in **Individual Real-Name Authentication**.

Topping Up Your Account

Dedicated load balancers are not free and can be purchased in the pay-per-use billing mode. For details, see **Billing (Dedicated Load Balancers)**.

Ensure that your account has sufficient balance to buy the resources you need.

Creating an IAM User

If you want to allow multiple users to manage your resources without sharing your password or private key, you can create IAM users and grant permissions to the users. These users can use specified links and their own accounts to access the cloud platform and help you manage resources efficiently. You can also configure account security policies to ensure the security of these accounts.

If you have registered an account, you can create an IAM user on the IAM console.

For example, to create an ELB administrator, perform the following steps:

1. Enter your account name and password to log in to the management console.

2. In the upper right corner of the page, hover the mouse over the username and select **Identity and Access Management**.

3. In the navigation pane on the left, choose **Users**. In the right pane, click **Create User**.

4. Enter the user information on the **Create User** page.

   – **Username**: Enter **elb_administrator**.

   – **Email Address**: Email address bound to the IAM user. This parameter is mandatory if the access type is specified as **Set by user**.

   – (Optional) **Mobile Number**: Mobile number of the IAM user.

   – (Optional) **Description**: Enter a description, for example, **ELB administrator**.

5. Select **Management console access** for **Access Type** and **Set now** for **Password**. Enter a password, and click **Next**.

   Figure 3-1 Selecting the access type

   ☐ NOTE

   An ELB administrator can log in to the management console and manage users. It is good practice to select **Set now** for **Password** when you create an ELB administrator for yourself. If you create an ELB administrator for another user, select **Set by user** for **Password** so that the user can set their own password.

6. (Optional) Add the user to the **admin** user group and click **Create**.

   User group **admin** has all the permissions. If you want to grant fine-grained permissions to IAM users, see **Creating a User and Granting Permissions**.

   Check whether the IAM user is displayed in the user list. You can click the IAM user login link above the list and use the created user to log in to the console.

4 Using Load Balancers (Entry Level)

Scenarios

You have a web application, which often needs to handle heavy traffic and is deployed on two ECSs for load balancing.

You can create a load balancer to distribute traffic evenly across the two ECSs, which eliminates SPOFs and makes your application more available.

Prerequisites

- You have added security group rules to allow traffic from the ports used by the two ECSs. (Alternatively, you can enable all ports first and then disable the ports that are no longer used.)

Creating ECSs

ECSs are used as backend servers.

Each ECS needs an EIP for accessing the Internet, and the EIP is used for configuring the application on the ECS. You can determine whether to bind an EIP to each ECS based on your requirements.

1. Log in to the management console.

2. In the upper left corner of the page, click and select the desired region and project.

3. Hover on in the upper left corner to display **Service List** and choose **Compute** > **Elastic Cloud Server**.

4. Click **Buy ECS**, configure the parameters, and click **Next**.

    The following table lists the specifications of the two ECSs.

    Table 4-1 ECS specifications

Item	Example Value
Name	ECS01 and ECS02

Item	Example Value
OS	CentOS 7.2 64bit
vCPUs	2
Memory	4 GB
System disk	40 GB
Data disk	100 GB
Bandwidth	5 Mbit/s

5. Submit your request.

Deploying the Application

Deploy Nginx on the two ECSs and edit two HTML pages so that a page with message "Welcome to ELB test page one!" is returned when ECS01 is accessed, and the other page with message "Welcome to ELB test page two!" is returned when ECS02 is accessed.

1. **Log in to the ECSs.**

2. Install and start Nginx.

    a. Run the **wget** command to download the Nginx installation package for your operating system in use. CentOS 7.6 is used as an example here.
    ```
    wget http://nginx.org/packages/centos/7/noarch/RPMS/nginx-release-centos-7-0.el7.ngx.noarch.rpm
    ```

    b. Run the following command to create the Nginx yum repository:
    ```
    rpm -ivh nginx-release-centos-7-0.el7.ngx.noarch.rpm
    ```

    c. Run the following command to install Nginx:
    ```
    yum -y install nginx
    ```

    d. Run the following commands to start Nginx and configure automatic Nginx enabling upon ECS startup:
    ```
    systemctl start nginx
    systemctl enable nginx
    ```

    e. Enter **http://**_EIP bound to the ECS_ in the address box of your browser. If the following page is displayed, Nginx has been installed.

    Figure 4-1 Nginx installed successfully
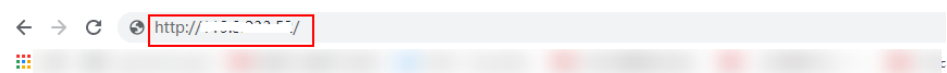
3. Modify the HTML page of ECS01.

Modify the **index.html** file in the default root directory of Nginx **/usr/share/nginx/html** to identify access to ECS01.

a. Open the **index.html** file.

vim /usr/share/nginx/html/index.html

b. Press **i** to enter editing mode.

c. Modify the **index.html** file to be as follows:

```
…
  <body>
     <h1>Welcome to <strong>ELB</strong> test page one!</h1>

     <div class="content">
        <p>This page is used to test the <strong>ELB</strong>!</p>

        <div class="alert">
           <h2>ELB01</h2>
           <div class="content">
              <p><strong>ELB test (page one)!</strong></p>
              <p><strong>ELB test (page one)!</strong></p>
              <p><strong>ELB test (page one)!</strong></p>
           </div>
        </div>
     </div>
  </body>
```

d. Press **Esc** to exit editing mode. Then, enter **:wq** to save the settings and exit the file.

4. Modify the HTML page of ECS02.

Modify the **index.html** file in the default root directory of Nginx **/usr/share/nginx/html** to identify access to ECS02.

a. Open the **index.html** file.

vim /usr/share/nginx/html/index.html

b. Press **i** to enter editing mode.

c. Modify the **index.html** file to be as follows:

```
…
  <body>
     <h1>Welcome to <strong>ELB</strong> test page two!</h1>

     <div class="content">
        <p>This page is used to test the <strong>ELB</strong>!</p>

        <div class="alert">
           <h2>ELB02</h2>
           <div class="content">
              <p><strong>ELB test (page two)!</strong></p>
              <p><strong>ELB test (page two)!</strong></p>
              <p><strong>ELB test (page two)!</strong></p>
           </div>
        </div>
     </div>
  </body>
```
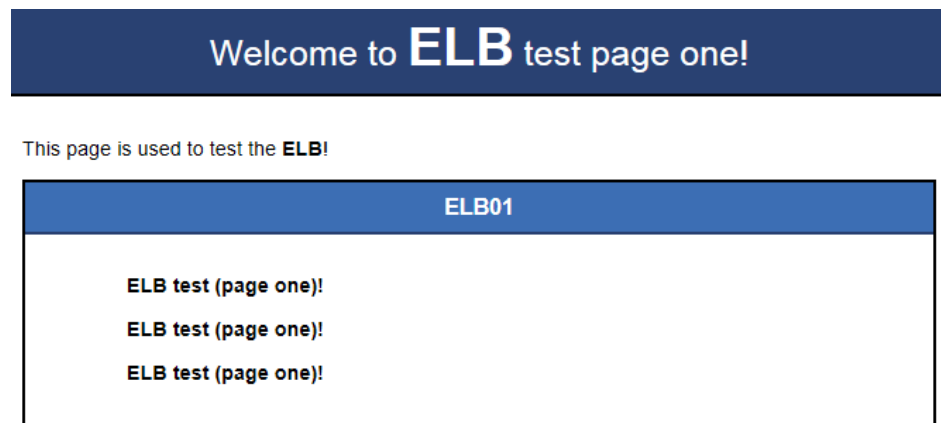
d. Press **Esc** to exit editing mode. Then, enter **:wq** to save the settings and exit the file.

5. Use your browser to access **http://**ECS01 EIP and **http://**ECS02 EIP to verify that Nginx has been deployed.

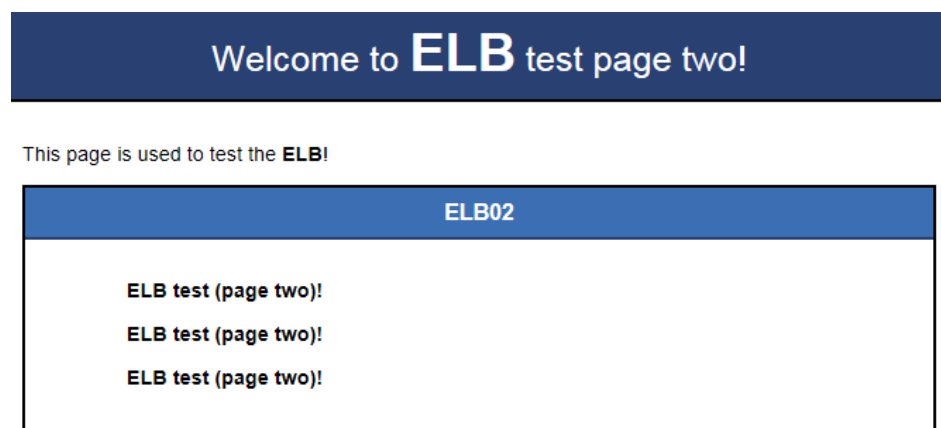If the modified HTML pages are displayed, Nginx has been deployed.

– HTML page of ECS01

Figure 4-2 Nginx successfully deployed on ECS01

–   HTML page of ECS02

Figure 4-3 Nginx successfully deployed on ECS02

Creating a Load Balancer

The load balancer needs an EIP to access the application deployed on the ECSs over the Internet. You can determine whether to bind an EIP to the load balancer based on your requirements. For details, see **Load Balancing on a Public or Private Network**.

1.   In the upper left corner of the page, click and select the desired region and project.

2.   Hover on in the upper left corner to display **Service List** and choose **Network** > **Elastic Load Balance**.

3.   Click **Buy Elastic Load Balancer** and then configure the parameters.

4.   Click **Next**.

5.   Confirm the configuration and submit your request.

6.   View the newly created load balancer in the load balancer list.

Adding a Listener

Add a listener to the created load balancer. When you add the listener, create a backend server group, configure a health check, and add the two ECSs to the created backend server group.

Figure 4-4 Traffic forwarding
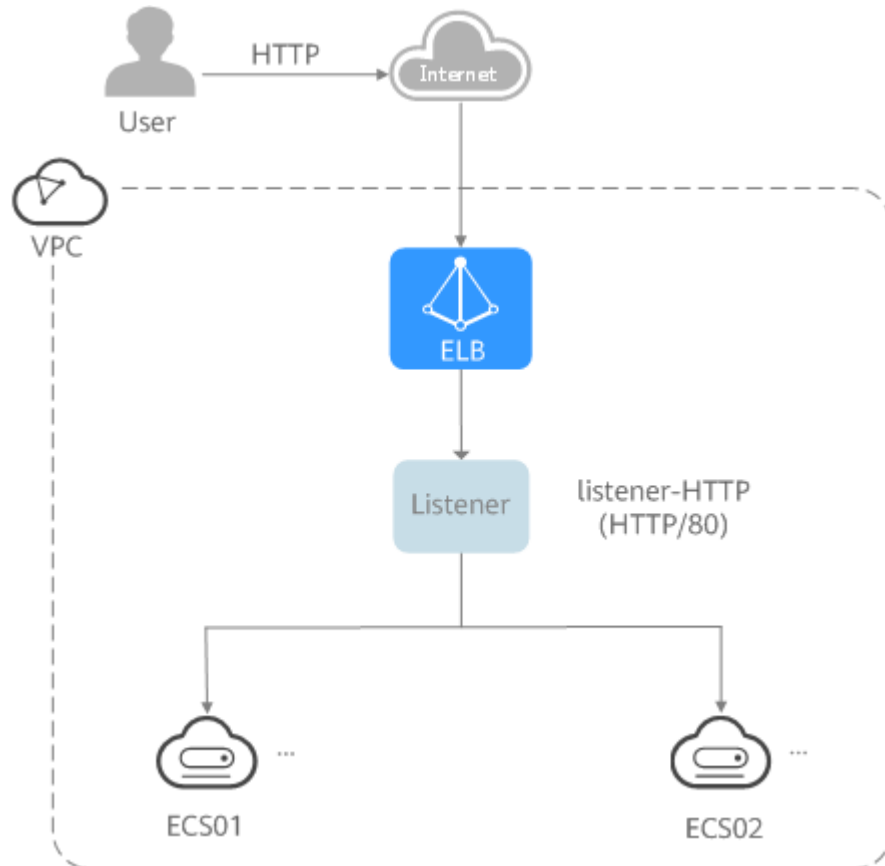
1. Hover on ☰ in the upper left corner to display **Service List** and choose **Network** > **Elastic Load Balance**.
2. Locate the created load balancer (**elb-01**) and click its name.
3. Under **Listeners**, click **Add Listener**.
4. Configure the listener and click **Next**.
   - **Name**: Enter a name, for example, **listener-HTTP**.
   - **Frontend Protocol/Port**: Select a protocol and enter a port for the load balancer to receive requests. For example, set it to **HTTP** and **80**.

Figure 4-5 Adding a listener

5. Create a backend server group and configure a health check.

   – Backend server group

     ▪ **Name**: Enter a name, for example, **server_group-ELB**.

     ▪ **Load Balancing Algorithm**: Select an algorithm that the load balancer will use to route requests, for example, **Weighted round robin**.

   – Health check

     ▪ **Protocol**: Select a protocol for the load balancer to perform health checks on backend servers. If the load balancer uses TCP, HTTP, or HTTPS to receive requests, the health check protocol can be TCP or HTTP. Here we use HTTP as an example.

     ▪ **Domain Name**: Enter a domain name that will be used for health checks, for example, **www.example.com**.

     ▪ **Port**: Enter a port for the load balancer to perform health checks on backend servers, for example, **80**.

6. Select the servers you want to add, set the backend port, and click **Finish**.

   – Backend servers: Select **ECS01** and **ECS02**.

   – Backend port: Set it to **80**. Backend servers will use this port to communicate with the load balancer.

Verifying Load Balancing

After the load balancer is configured, you can access the domain name to check whether the two ECSs are accessible.

1. Modify the **C:\Windows\System32\drivers\etc\hosts** file on your PC to map the domain name to the load balancer EIP.

   View the load balancer EIP on the **Summary** page of the load balancer.

Figure 4-6 hosts file on your PC

```
# localhost name resolution is handled within DNS itself.
#   127.0.0.1       localhost
#   ::1             localhost

11████████14   www.example.com
```

2. On the CLI of your PC, run the following command to check whether the domain name is mapped to the load balancer EIP:

   ping www.example.com

   If data packets are returned, the domain name has been mapped to the load balancer EIP.

3. Use your browser to access **http://www.example.com**. If the following page is displayed, the load balancer has routed the request to ECS01.

Figure 4-7 Accessing ECS01
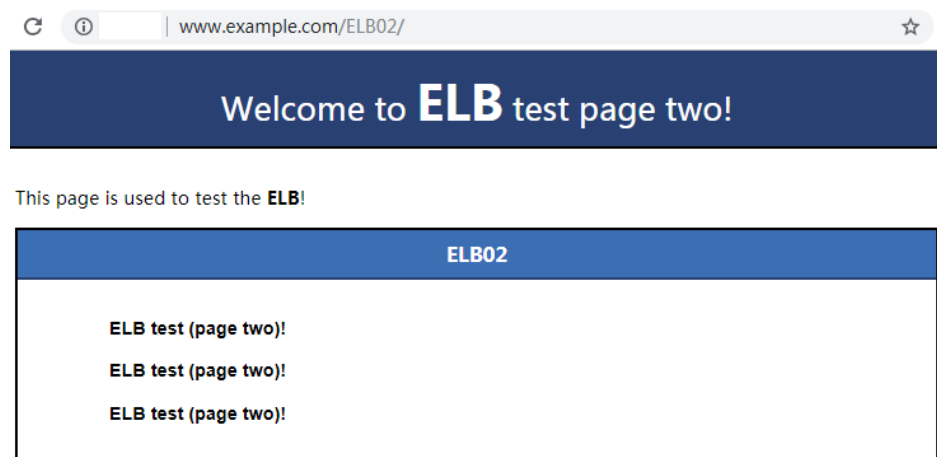

4. Use your browser to access **http://www.example.com**. If the following page is displayed, the load balancer has routed the request to ECS02.

Figure 4-8 Accessing ECS02

5 Using Load Balancers (Advanced Level)

Scenarios

You have two web applications that are deployed on separated ECSs but use the same domain name for access. You can set different URLs to process requests.

To forward requests based on URLs, you need to create a load balancer, add an HTTP or HTTPS listener, and add forwarding policies to specify the URLs.

An HTTP listener is used as an example to describe how to route requests from two URLs (**/ELB01** and **/ELB02**) of the same domain name (www.example.com) to different backend servers.

Prerequisites

- You have added security group rules to allow traffic from the ports used by the two ECSs. (Alternatively, you can enable all ports first and then disable the ports that are no longer used.)

Creating ECSs

ECSs are used as backend servers to process requests. Each ECS needs an EIP for accessing the Internet and configuring the application on the ECS.

1. Log in to the management console.

2. In the upper left corner of the page, click ⊙ and select the desired region and project.

3. Hover on ☰ in the upper left corner to display **Service List** and choose **Compute** > **Elastic Cloud Server**.

4. The following table lists the specifications of the two ECSs.

Table 5-1 ECS specifications

Item	Example Value
Name	ECS01 and ECS02

Item	Example Value
OS	CentOS 7.2 64bit
vCPUs	2
Memory	4 GB
System disk	40 GB
Data disk	100 GB
Bandwidth	5 Mbit/s

5. Submit your request.

Deploying the Application

Deploy Nginx on the two ECSs and edit two HTML pages so that a page with message "Welcome to ELB test page one!" is returned when ECS01 is accessed, and the other page with message "Welcome to ELB test page two!" is returned when ECS02 is accessed.

1. **Log in to the ECSs.**
2. Install and start Nginx.

    a. Run the **wget** command to download the Nginx installation package for your operating system in use. CentOS 7.6 is used as an example here.
   ```
    wget http://nginx.org/packages/centos/7/noarch/RPMS/nginx-release-centos-7-0.el7.ngx.noarch.rpm
    ```

    b. Run the following command to create the Nginx yum repository:
    ```
    rpm -ivh nginx-release-centos-7-0.el7.ngx.noarch.rpm
    ```

    c. Run the following command to install Nginx:
    ```
    yum -y install nginx
    ```

    d. Run the following commands to start Nginx and configure automatic Nginx enabling upon ECS startup:
    ```
    systemctl start nginx
    systemctl enable nginx
    ```

    e. Enter **http://**_EIP bound to the ECS_ in the address box of your browser. If the following page is displayed, Nginx has been installed.

    **Figure 5-1** Nginx installed successfully
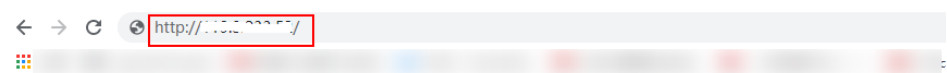
    

3. Modify the HTML page of ECS01.

Move the **index.html** file from the default root directory of Nginx **/usr/share/ nginx/html** to the **ELB01** directory and modify the file to identify access to ECS01.

a. Create the **ELB01** directory and copy the **index.html** file to this directory:

**mkdir /usr/share/nginx/html/ELB01**

**cp /usr/share/nginx/html/index.html /usr/share/nginx/html/ELB01**/

b. Open the **index.html** file.

**vim /usr/share/nginx/html/ELB01/index.html**

c. Press **i** to enter editing mode.

d. Modify the **index.html** file to be as follows:

```
…
  <body>
    <h1>Welcome to <strong>ELB</strong> test page one!</h1>

    <div class="content">
      <p>This page is used to test the <strong>ELB</strong>!</p>

      <div class="alert">
        <h2>ELB01</h2>
        <div class="content">
          <p><strong>ELB test (page one)!</strong></p>
          <p><strong>ELB test (page one)!</strong></p>
          <p><strong>ELB test (page one)!</strong></p>
        </div>
      </div>
    </div>
  </body>
```

e. Press **Esc** to exit editing mode. Then, enter **:wq** to save the settings and exit the file.

4. Modify the HTML page of ECS02.

Move the **index.html** file from the default root directory of Nginx **/usr/share/ nginx/html** to the **ELB02** directory and modify the file to identify access to ECS02.

a. Create the **ELB02** directory and copy the **index.html** file to this directory:

mkdir /usr/share/nginx/html/ELB02

cp /usr/share/nginx/html/index.html /usr/share/nginx/html/ELB02/

b. Open the **index.html** file.

vim /usr/share/nginx/html/ELB02/index.html

c. Press **i** to enter editing mode.

d. Modify the **index.html** file to be as follows:

```
…
  <body>
    <h1>Welcome to <strong>ELB</strong> test page two!</h1>

    <div class="content">
      <p>This page is used to test the <strong>ELB</strong>!</p>

      <div class="alert">
        <h2>ELB02</h2>
        <div class="content">
          <p><strong>ELB test (page two)!</strong></p>
          <p><strong>ELB test (page two)!</strong></p>
          <p><strong>ELB test (page two)!</strong></p>
        </div>
      </div>
```
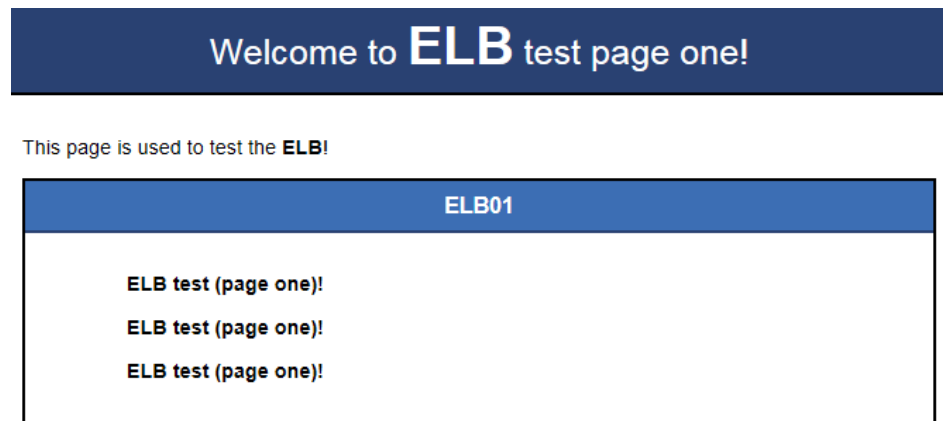
```
        </div>
    </body>
```

    e.   Press **Esc** to exit editing mode. Then, enter **:wq** to save the settings and exit the file.

5.   Use your browser to access **http://**_ECS01 EIP_**/ELB01/** and **http://**_ECS02 EIP_**/ELB02/** to verify that Nginx has been deployed.

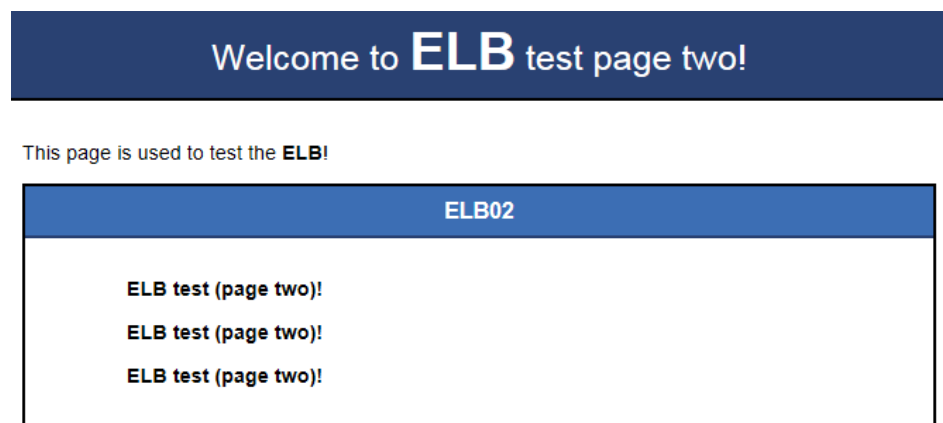If the modified HTML pages are displayed, Nginx has been deployed.

–   HTML page of ECS01

Figure 5-2 Nginx successfully deployed on ECS01

–   HTML page of ECS02

Figure 5-3 Nginx successfully deployed on ECS02

Creating a Load Balancer

The load balancer needs an EIP to access the application deployed on the ECSs over the Internet. You can determine whether to bind an EIP to the load balancer based on your requirements. For details, see **Load Balancing on a Public or Private Network**.

1.   In the upper left corner of the page, click 📍 and select the desired region and project.
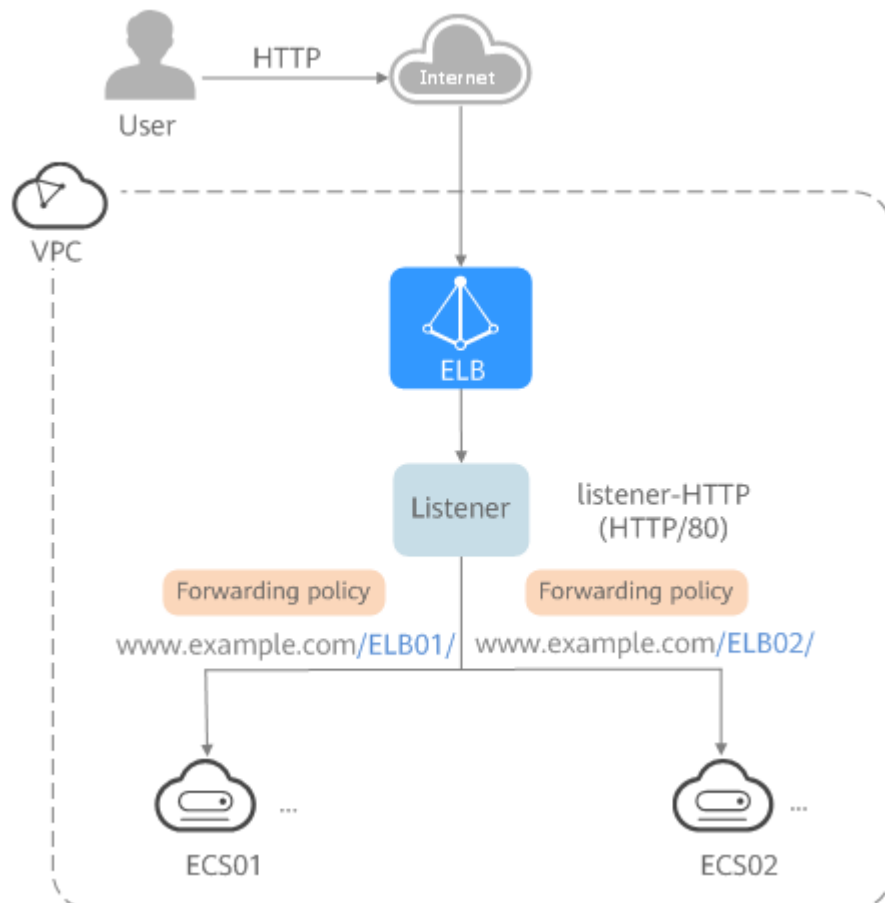
2. Hover on ☰ in the upper left corner to display **Service List** and choose **Network** > **Elastic Load Balance**.

3. Click **Buy Elastic Load Balancer** and then configure the parameters.

4. Click **Next**.

5. Confirm the configuration and submit your request.

6. View the newly created load balancer in the load balancer list.

Adding a Listener

Add a listener to the created load balancer. When you add the listener, create a backend server group, configure a health check, and add the two ECSs to the created backend server group.

Configure two forwarding policies to forward HTTP requests to the two ECSs, for example, requests from **www.example.com/ELB01/** to ECS01, and those from **www.example.com/ELB02/** to ECS02.

Figure 5-4 Traffic forwarding

1. Hover on ☰ in the upper left corner to display **Service List** and choose **Network** > **Elastic Load Balance**.

2. Locate the created load balancer and click its name.

3. Under **Listeners**, click **Add Listener**.

4. Configure the listener and click **Next**.

– **Name**: Enter a name, for example, **listener-HTTP**.

– **Frontend Protocol/Port**: Select a protocol and enter a port for the load balancer to receive requests. For example, set it to **HTTP** and **80**.

5. Create a backend server group, configure a health check, and click **Finish**.

– Backend server group

▪ **Name**: Enter a name, for example, **server_group-ELB**.

▪ **Load Balancing Algorithm**: Select an algorithm that the load balancer will use to route requests, for example, **Weighted round robin**.

– Health check

▪ **Protocol**: Select a protocol for the load balancer to perform health checks on backend servers. If the load balancer uses TCP, HTTP, or HTTPS to receive requests, the health check protocol can be TCP or HTTP. Here we use HTTP as an example. Note that the protocol cannot be changed after the listener is added.

▪ **Domain Name**: Enter a domain name that will be used for health checks, for example, www.example.com.

▪ **Port**: Enter a port for the load balancer to perform health checks on backend servers, for example, **80**.

Adding Forwarding Policies

1. Click ☰ on the right of the newly added listener and select **Configure Forwarding Policy**.

Alternatively, click **Forwarding Policies** on the right of the page.

2. Configure the forwarding policy and click **Next**.

– **Name**: Enter a name for the forwarding policy, for example, **forwarding_policy-ELB01**.

– **Domain Name**: Enter a domain name that will be used to forward the requests, for example, www.example.com. The domain name in the request must exactly match that specified in the forwarding policy.

– **URL**: You can also specify a URL to forward the requests, for example, **/ELB01/**.

– **URL Matching Rule**: Select a rule for matching the specified URL string with the URL in the request. Three options are available, **Exact match**, **Prefix match**, and **Regular expression match**. **Exact match** enjoys the highest priority, and **Regular expression match** the lowest priority. Select **Exact match** here.

3. Click **Add Forwarding Policy** and configure a forwarding policy.

– **Domain name**: Enter a domain name that will be used to forward the requests, for example, www.example.com. The domain name in the request must exactly match that specified in the forwarding policy.

- – **URL**: You can also specify a URL to forward the requests, for example, **/ELB01/**.

- – **URL matching rule**: Select a rule for matching the specified URL string with the URL in the request. Three options are available, **Exact match**, **Prefix match**, and **Regular expression match**. **Exact match** enjoys the highest priority, and **Regular expression match** the lowest priority. Select **Exact match** here.

- – **Action**: Select **Forward to a backend server group**.

- – **Backend Server Group**: Select **server_group-ELB01**.

4. Add the backend server group and configure a health check.

- – Backend server group

  - ▪ **Name**: Enter a name, for example, **server_group-ELB01**.

  - ▪ **Load Balancing Algorithm**: Select an algorithm that the load balancer will use to route requests, for example, **Weighted round robin**.

- – Health check

  - ▪ **Protocol**: Select a protocol for the load balancer to perform health checks on backend servers. If the load balancer uses TCP, HTTP, or HTTPS to receive requests, the health check protocol can be TCP or HTTP. Here we use HTTP as an example. Note that the protocol cannot be changed after the listener is added.

  - ▪ **Domain Name**: Enter a domain name that will be used for health checks, for example, **www.example.com**.

  - ▪ **Port**: Enter a port for the load balancer to perform health checks on backend servers, for example, **80**.

5. Select the newly added forwarding policy. On the **Backend Server Groups** tab page on the right, click **Add**.

6. Select the server you want to add, set the backend port, and click **Finish**.

- – Backend server: ECS01

- – Backend port: Set it to **80**. Backend servers will use this port to communicate with the load balancer.

7. Repeat #en-us_elb_02_0001/li1279175813279 to **6** to add another forwarding policy, create a backend server group, and add ECS02 to the backend server group.

Verifying Load Balancing

After the load balancer is configured, you can access the domain name or the specified URL to check whether the two ECSs are accessible.

1. Modify the **C:\Windows\System32\drivers\etc\hosts** file on your PC to map the domain name to the load balancer EIP.

   View the load balancer EIP on the **Summary** page of the load balancer.

Figure 5-5 hosts file on your PC

```
# localhost name resolution is handled within DNS itself.
#   127.0.0.1       localhost
#   ::1             localhost

11███████14   www.example.com
```

2. On the CLI of your PC, run the following command to check whether the domain name is mapped to the load balancer EIP:

   ping www.example.com

   If data packets are returned, the domain name has been mapped to the load balancer EIP.

3. Use your browser to access **http://www.example.com/ELB01/**. If the following page is displayed, the load balancer has routed the request to ECS01.
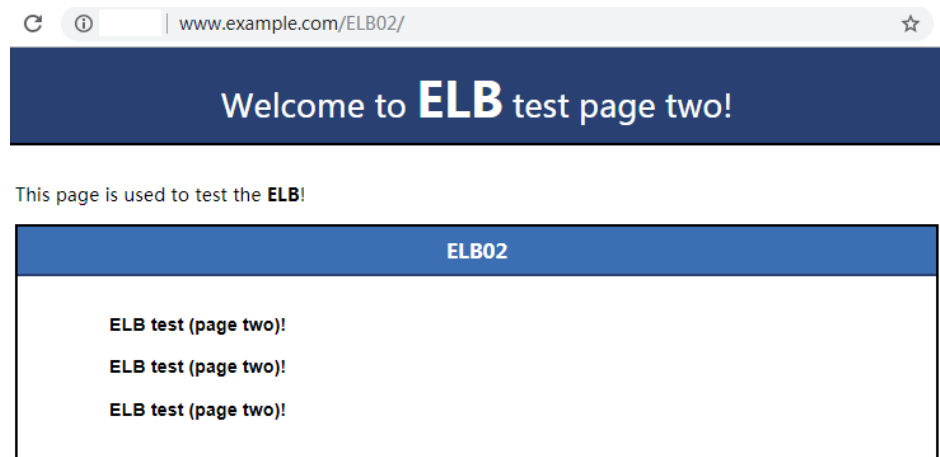
Figure 5-6 Accessing ECS01
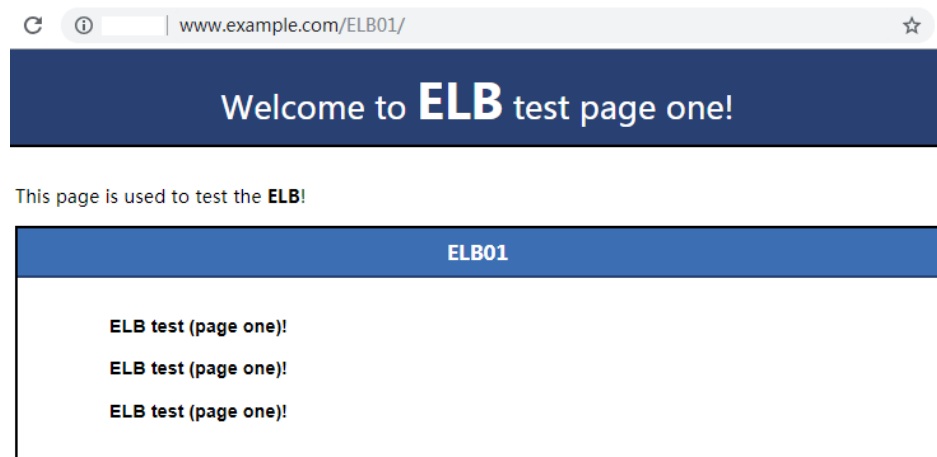
☐ NOTE

   ELB01/ indicates that the default directory named **ECS01** is accessed, while **ELB01** indicates the file name. Therefore, the slash (/) following **ELB01** must be retained.

4. Use your browser to access **http://www.example.com/ELB02/**. If the following page is displayed, the load balancer has routed the request to ECS02.

Figure 5-7 Accessing ECS02

6 Change History

Released On	Description
2023-03-30	This issue is the first official release.