ALIBABA CLOUD

Alibaba Cloud

Server Load Balancer

Tutorials

Document Version: 20220316

Alibaba Cloud

Legal disclaimer

Alibaba Cloud reminds you to carefully read and fully understand the terms and conditions of this legal disclaimer before you read or use this document. If you have read or used this document, it shall be deemed as your total acceptance of this legal disclaimer.

1. You shall download and obtain this document from the Alibaba Cloud website or other Alibaba Cloud-authorized channels, and use this document for your own legal business activities only. The content of this document is considered confidential information of Alibaba Cloud. You shall strictly abide by the confidentiality obligations. No part of this document shall be disclosed or provided to any third party for use without the prior written consent of Alibaba Cloud.

2. No part of this document shall be excerpted, translated, reproduced, transmitted, or disseminated by any organization, company or individual in any form or by any means without the prior written consent of Alibaba Cloud.

3. The content of this document may be changed because of product version upgrade, adjustment, or other reasons. Alibaba Cloud reserves the right to modify the content of this document without notice and an updated version of this document will be released through Alibaba Cloud-authorized channels from time to time. You should pay attention to the version changes of this document as they occur and download and obtain the most up-to-date version of this document from Alibaba Cloud-authorized channels.

4. This document serves only as a reference guide for your use of Alibaba Cloud products and services. Alibaba Cloud provides this document based on the "status quo", "being defective", and "existing functions" of its products and services. Alibaba Cloud makes every effort to provide relevant operational guidance based on existing technologies. However, Alibaba Cloud hereby makes a clear statement that it in no way guarantees the accuracy, integrity, applicability, and reliability of the content of this document, either explicitly or implicitly. Alibaba Cloud shall not take legal responsibility for any errors or lost profits incurred by any organization, company, or individual arising from download, use, or trust in this document. Alibaba Cloud shall not, under any circumstances, take responsibility for any indirect, consequential, punitive, contingent, special, or punitive damages, including lost profits arising from the use or trust in this document (even if Alibaba Cloud has been notified of the possibility of such a loss).

5. By law, all the contents in Alibaba Cloud documents, including but not limited to pictures, architecture design, page layout, and text description, are intellectual property of Alibaba Cloud and/or its affiliates. This intellectual property includes, but is not limited to, trademark rights, patent rights, copyrights, and trade secrets. No part of this document shall be used, modified, reproduced, publicly transmitted, changed, disseminated, distributed, or published without the prior written consent of Alibaba Cloud and/or its affiliates. The names owned by Alibaba Cloud shall not be used, published, or reproduced for marketing, advertising, promotion, or other purposes without the prior written consent of Alibaba Cloud. The names owned by Alibaba Cloud include, but are not limited to, "Alibaba Cloud", "Aliyun", "HiChina", and other brands of Alibaba Cloud and/or its affiliates, which appear separately or in combination, as well as the auxiliary signs and patterns of the preceding brands, or anything similar to the company names, trade names, trademarks, product or service names, domain names, patterns, logos, marks, signs, or special descriptions that third parties identify as Alibaba Cloud and/or its affiliates.

6. Please directly contact Alibaba Cloud for any errors of this document.

Document conventions

Style	Description	Example
Danger	A danger notice indicates a situation that will cause major system changes, faults, physical injuries, and other adverse results.	**Danger:** Resetting will result in the loss of user configuration data.
Warning	A warning notice indicates a situation that may cause major system changes, faults, physical injuries, and other adverse results.	**Warning:** Restarting will cause business interruption. About 10 minutes are required to restart an instance.
Notice	A caution notice indicates warning information, supplementary instructions, and other content that the user must understand.	**Notice:** If the weight is set to 0, the server no longer receives new requests.
Note	A note indicates supplemental instructions, best practices, tips, and other content.	**Note:** You can use Ctrl + A to select all files.
>	Closing angle brackets are used to indicate a multi-level menu cascade.	Click **Settings**> **Network**> **Set network type**.
Bold	Bold formatting is used for buttons , menus, page names, and other UI elements.	Click **OK**.
`Courier font`	Courier font is used for commands	Run the `cd /d C:/window` command to enter the Windows system folder.
Italic	Italic formatting is used for parameters and variables.	`bae log list --instanceid` *Instance_ID*
[] or [a|b]	This format is used for an optional value, where only one item can be selected.	`ipconfig [-all|-t]`
{} or {a|b}	This format is used for a required value, where only one item can be selected.	`switch {active|stand}`

Table of Contents

1.ALB

1.1. Configure HTTPS to encrypt communication

This topic describes how to configure HTTPS for Application Load Balancer (

ALB

) to achieve encrypted communication.

Scenarios

As a large number of enterprises migrate their services to the cloud, high security of cloud services is demanded, especially by governments and financial enterprises. To ensure service security, more and more enterprises require encryption for both frontend communication and backend communication. This requires ALB to ensure communication security when clients send requests to ALB and when ALB forwards requests to backend servers.

ALB

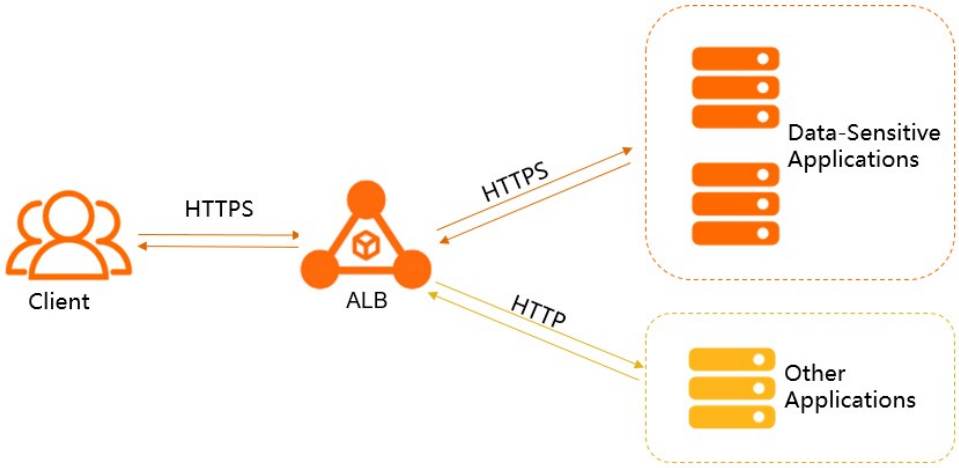
supports HTTPS encryption to ensure communication security when clients send requests to

ALB

and when

ALB

forwards requests to backend servers.



Configure HTTPS

1. Log on to the ALB console.
2. In the top navigation bar, select the region where the

   ALB

   instance is deployed.
3. In the left-side navigation pane, choose **ALB > Server Groups**.
4. Set the following parameters and click **Create**.

- **VPC**: Select the virtual private cloud (VPC) to which the ALB instance belongs.
- **Backend Server Protocol**: Select **HTTPS**.
- For more information about the parameters, see Create a server group in Manage server groups.

5. In the **Server group created** dialog box, click **Add Backend Server**. On the **Server Groups** page, find the server group that you created and click **Modify Backend Server** in the **Actions** column.
6. On the **Backend Servers** tab, click **Add Backend Server**.
7. In the **Add Backend Server** panel, specify the type of backend server, select the backend server that you want to add, and then click **Next**.
8. Set the server port to **443**, set the weight, and then click **OK**.
9. For more information about how to create HTTPS listeners, see Add an HTTPS listener.

> **Note** In the **Select Server Group** wizard, select the server group that you created.

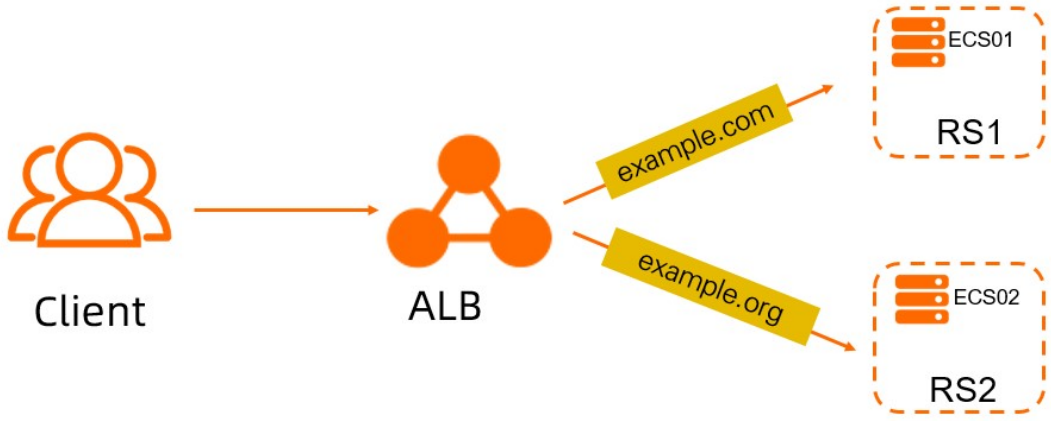
1.2. Add multiple domain names for an NGINX service to serve HTTPS requests

This topic describes how to associate multiple certificates with an HTTPS listener of an Application Load Balancer (ALB) instance to distribute requests destined for different domain names to different NGINX services deployed on backend servers.

Scenario

After ALB receives an HTTPS request, ALB matches the requested domain name against the certificates that you uploaded. If one of the certificates is matched, ALB sends the request to a backend server based on the forwarding rule that you configured for the domain name and then returns the corresponding certificate to the client. If no certificate is matched, ALB sends the request to a backend server in the default server group and returns the default certificate to the client. The following configurations are used in this example:

- The default certificate: default. The default server group: RS1.
- The domain name `example1.com` is associated with the additional certificate example1. Requests destined for `example1.com` are forwarded to RS1.
- The domain name `example.org` is associated with the additional certificate example2. Requests destined for `example.org` are forwarded to RS2.



Prerequisites

- An

  ALB

  instance is created. For more information, see Create an ALB instance.
- RS1 and RS2 are created. For more information, see Manage server groups.
- An Elastic Compute Service (ECS) instance is added to each server group. In this example, ECS01 is added to RS1 and ECS02 is added to RS2. Different NGINX services are deployed on the ECS instances.
- You have purchased the required certificates from Alibaba Cloud. If the certificates are purchased from a third party service provider, you must upload them to SSL Certificates Service. In addition, make sure that the certificates are associated with your domain names. For more information about how to create a certificate, see Apply for a certificate. In this example, the following certificates are used:
  - The default certificate.
  - The additional certificate example1 that is associated with `example.com`.
  - The additional certificate example2 that is associated with `example.org`.

Background information

The number of additional certificates that can be associated with an

ALB

instance: 10 for a basic ALB instance and 25 for a standard ALB instance. The default certificate is not included in this quota.

Step 1: Create an HTTPS listener

1. Log on to the ALB console.
2. In the top navigation bar, select the region where the

   ALB

   instance is deployed.
3. On the **Instances** page, find the ALB instance that you want to manage and click **Create Listener**

in the **Actions** column.

4. On the **Configure Listener** page, set the parameters of the listener and click **Next**.

   The following configurations are used in this example. For more information about the other parameters and how to create an HTTPS listener, see Add an HTTPS listener.

   - **Select Server Certificate**: In this example, the **default** certificate is selected.
   - **Select Server Group**: In this example, RS1 is selected.

Step 2: Add an additional certificate

1. On the **Instances** page, find the ALB instance that you want to manage and click its ID.
2. On the **Listener** tab, find the HTTPS listener that you created and click **Manage Certificate** in the **Actions** column.
3. On the **Certificates** tab, click **Add Extended Validation Certificate**.
4. In the **Add Extended Validation Certificate** dialog box, select the certificate example1 and click **OK**. Repeat the preceding steps to add the certificate example2.

Step 3: Create forwarding rules

1. On the **Instances** page, find the ALB instance that you want to manage and click its ID.
2. On the **Listener** tab, find the HTTPS listener that you created and click **View/Modify Forwarding Rule** in the **Actions** column.
3. On the **Forwarding Rules** tab, click **Add New Rule**.
4. Set the parameters of the forwarding rule and click **OK**. In this example, the following configurations are used:
   - If the requested **Domain Name** is `example.com`, then **Forward** the request to RS1. **Weight**: 100.
   - If the requested **Domain Name** is `example.org`, then **Forward** the request to RS2. **Weight**: 100.

   > **Note**
   > - An ECS instance with a higher weight receives more requests. In this example, the default value 100 is used.
   > - Valid values: 1 to 100.

Step 4: Create CNAME records

Create CNAME records to map `example.com` and `example.org` to the publicly accessible domain name of the

ALB

instance.

1. Log on to the ALB console.
2. In the top navigation bar, select the region where the ALB instance is deployed.
3. Find the

   ALB

instance that you want to manage and copy the domain name.

4. To create a CNAME record, perform the following operations:
   i. Log on to the Alibaba Cloud DNS console.
   ii. On the **Manage DNS** page, click **Add Domain Name**.
   iii. In the **Add Domain Name** dialog box, enter the domain name of your host and click **OK**.

> **Notice** Before you create the CNAME record, you must use a TXT record to verify the ownership of the domain name.

   iv. In the **Actions** column of the domain name that you want to manage, click **Configure**.
   v. On the **DNS Settings** page, click **Add Record**.
   vi. In the **Add Record** panel, set the following parameters and click **Confirm**.

Parameter	Description
Type	Select **CNAME** from the drop-down list.
Host	Enter the prefix of the domain name of your host.
ISP Line	Select Default.
Value	Enter the CNAME. The CNAME is the domain name of the ALB instance that you copied in Step 3.
TTL	Select the time-to-live (TTL) value of the record on the DNS server. In this example, the default value is used.

> **Note**
> - Newly created CNAME records immediately take effect. The time that is required for a modified CNAME record to take effect is determined by the TTL value. The default TTL value is 10 minutes.
> - If the CNAME record that you want to create conflicts with an existing record, we recommend that you specify another domain name.

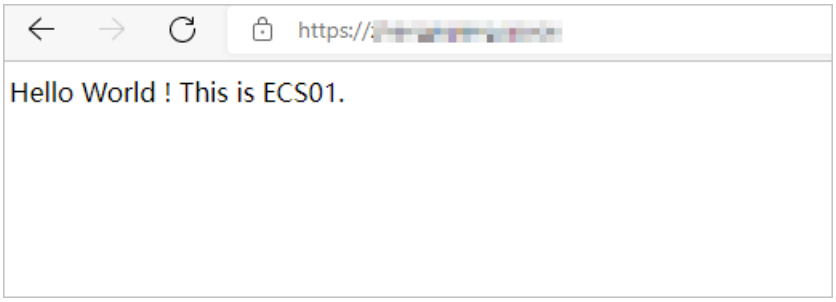
Step 5: Test connectivity

Enter `example.com` and `example.org` in the address bar of a browser to check whether you can access

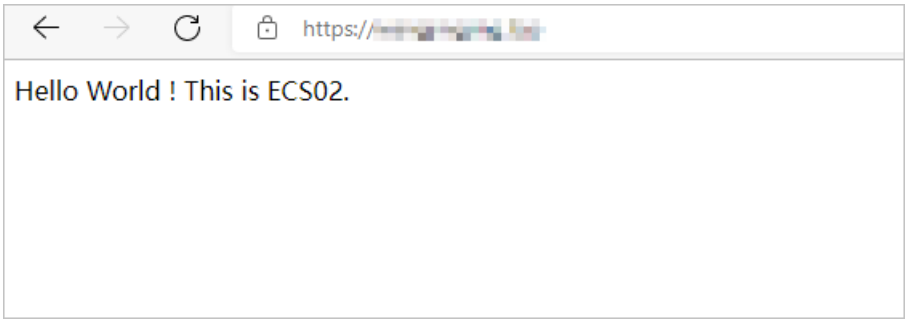
ALB

. In this example, NGINX is used to set up two static websites on ECS01 and ECS02.

- In the address bar of the browser, enter the domain name `example.com`, which is associated with the additional certificate example1. If you can access the domain name, it indicates that the request is sent to ECS01 in RS1 based on the forwarding rule that you configured. Test result:



- In the address bar of the browser, enter the domain name `example.org`, which is associated with the additional certificate example2. If you can access the domain name, it indicates that the request is sent to ECS02 in RS2 based on the forwarding rule that you configured. Test result:



1.3. Redirect HTTP requests to an HTTPS listener

HTTPS is an extension of HTTP and is used for secure communication. This topic describes how to redirect HTTP requests to an HTTPS listener in the Application Load Balancer (

ALB

) console.

Prerequisites

- An HTTP listener is created. For more information, see Add an HTTP listener.
- An HTTPS listener is created. For more information, see Add an HTTPS listener.

Redirect HTTP requests to an HTTPS listener

1. Log on to the ALB console.
2. In the top navigation bar, select the region where the

   ALB

   instance is deployed.
3. On the **Instances** page, click the ID of the ALB instance.
4. On the **Listener** tab, find the HTTP listener and click **View/Modify Forwarding Rule** in the **Actions** column.
5. Choose **Forwarding Rules > Inbound Forwarding Rules**, and click **Add New Rule**.
6. In the **Add Forwarding Rules** section, set the parameters.

In this example, the requests whose value in the Accept-Language header field is Chinese are redirected to https://www.example.com.

Action: Select **Redirect**.

- **Protocol**: Select **HTTPS**.
- **Port**: Enter the port number of the HTTPS listener.

For more information about the parameters, see Manage forwarding rules for a listener.

7. Click **OK**. After you complete the preceding steps, the requests whose value in the Accept-Language header field is Chinese are redirected from

   ALB

   domain names to https://www.example.com.

1.4. Mirror production traffic to a staging environment by using the traffic mirroring feature of ALB

This topic describes how to mirror production traffic to a staging environment by using the traffic mirroring feature of

ALB

.

Scenarios

To test a new feature before it can be officially released without interrupting the current service, the test must be performed separately from the production environment.

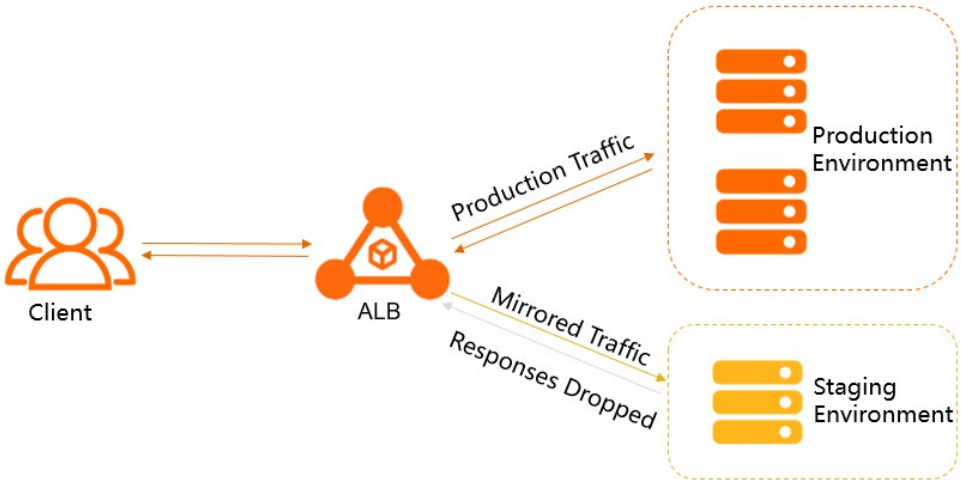
ALB

provides the traffic mirroring feature, which allows you to mirror production traffic to a group of backend servers for testing purposes. To prevent service interruptions, responses from testing servers are automatically dropped by

ALB

. You can use the traffic mirroring feature in the following scenarios:

- Test new features and service performance.
- Copy production traffic for testing without the need to create additional data.
- Reproduce errors for troubleshooting.



Limits

- The traffic mirroring feature is available only for accounts included in the whitelist. To use this feature, .
- Internet-facing

  ALB

  and internal-facing

  ALB

  instances support the traffic mirroring feature.
- Alibaba Cloud provides basic and standard

  ALB

  instances. Only standard

  ALB

  instances support the traffic mirroring feature.

Prerequisites

- A server group (Server Group A) is created to process production traffic and a server group (Server Group B) is created to process mirrored traffic. The backend protocol is set to HTTP. For more information, see Manage server groups.
- A listener is created. For more information, see Add an HTTP listener, Add an HTTPS listener, or Add a QUIC listener.

Mirror network traffic

1. Log on to the ALB console.
2. In the top navigation bar, select the region where the

   ALB

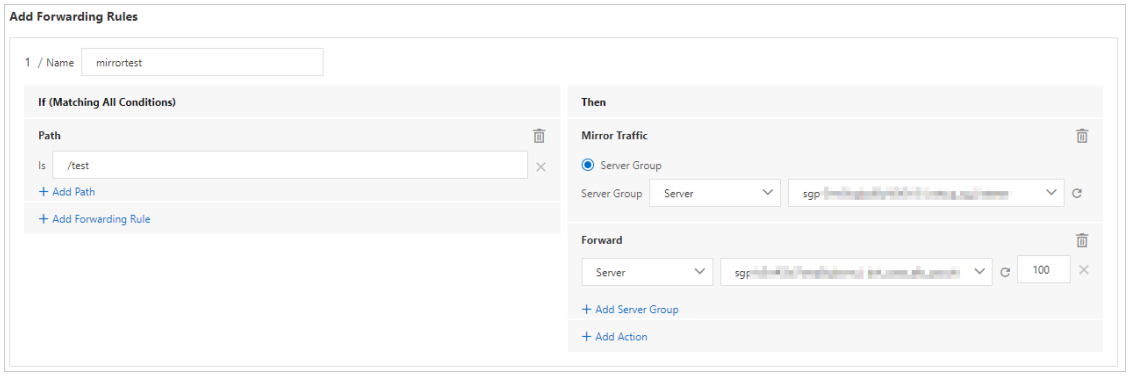
   instance is deployed.
3. On the **Instances** page, click the ID of the ALB instance that you want to manage.
4. On the **Listener** tab, find the listener that you want to manage and click **View/Modify Forwarding Rule** in the **Actions** column.

5. Choose **Forwarding Rules > Inbound Forwarding Rules** and click **Add Forwarding Rules**.
6. In the **Add Forwarding Rules** section, set the parameters of the forwarding rule.

   In this example, network traffic from the **/test Path** is mirrored to servers in Server Group B.

   Action: Set **Traffic Mirror** to Server Group B and **Forward** to Server Group A.

   > **Notice**
   >
   > ALB
   >
   > drops the responses from Server Group B. You must set **Forward** to Server Group A to prevent service interruptions.



7. Click **OK**. After you complete the configurations, network traffic sent to Server Group A can be mirrored to Server Group B.

References

Manage forwarding rules for a listener

1.5. Use ALB to implement canary releases

Canary releases allow applications of old and new versions to be deployed in parallel. You can apply the new version to a small percentage of users before rolling out to the entire user base. With canary releases, you can control the percentage of users that are exposed to the old and new versions based on your business requirements.

Scenario

When you build an application, you want to speed up the release process while ensuring user experience. To reduce the risk of errors in the new version disrupting your workflow, you want to implement canary releases and allow the new version to receive service traffic from a subset of users. Then, you can evaluate the new version against metrics such as logs, performance, and stability based on feedback from the canary users. You can incrementally roll out the new version to the entire user base or roll back to the old version based on the evaluation.

Preparations

1. Deploy the old and new versions on different backend servers. To ensure service availability, you can deploy each version on multiple backend servers.
2. Create an

   ALB

instance. For more information, see Create an ALB instance.

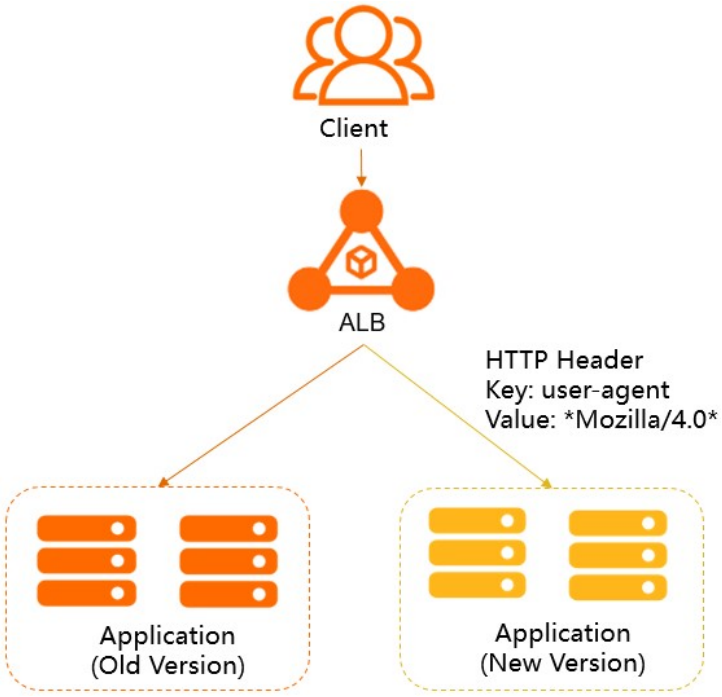
3. Create server groups. Then, add the backend servers where the old and new version are deployed to different server groups. For more information, see Manage server groups.
4. Create a listener. Then, add the backend servers where the old version is deployed to the default server group of the listener. For more information, see Add an HTTP listener, Add an HTTPS listener, and Add a QUIC listener.

   After you complete the preceding tasks, all requests are forwarded to the old version. You can use one or more of the following methods to implement canary releases. In this example, the new version is added to Server Group A and the old version is added to Server Group B.

   - Canary releases based on HTTP headers
   - Canary releases based on cookies
   - Canary release based on server groups

Canary releases based on HTTP headers

In this example, HTTP requests that contain the User-Agent header *Mozilla/4.0* are forwarded to the new version.



1. Log on to the ALB console.
2. In the top navigation bar, select the region where the

   ALB

   instance is deployed.
3. On the **Instances** page, click the ID of the Application Load Balancer (ALB) instance that you want to manage.
4. On the **Listener** tab, find the listener and click **View/Modify Forwarding Rule** in the **Actions** column.
5. Choose **Forwarding Rules > Inbound Forwarding Rules** and click **Add Forwarding Rules**.

6. Set the parameters and click **OK**.
   - **Forwarding Condition**: Select **HTTP Header** from the drop-down list, set the key to *user-agent*, and then set the value to **Mozilla/4.0**.
   - **Forwarding Action**: Select **Forward** from the drop-down list and select Server Group A.

   For more information about the parameters, see Manage forwarding rules for a listener.
7. You can configure more forwarding rules to increase the percentage of user traffic distributed to the new version. If the new version works as expected, you can discontinue the old version and forward all requests to the new version.

Canary releases based on cookies

This example shows how to configure ALB to forward requests to the new version based on cookies that are specified in key-value pairs.

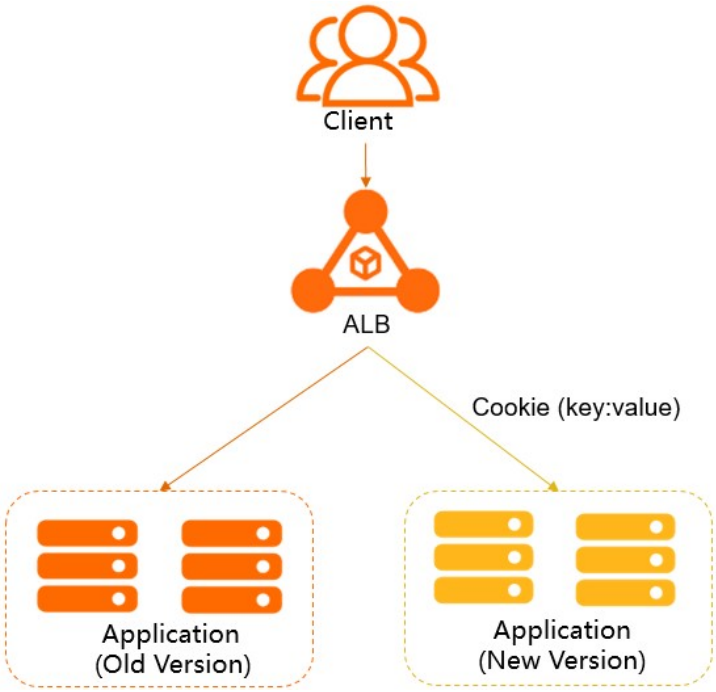
Basic

ALB

instances do not support this feature. You must upgrade to the

ALB

edition before you can use this feature. For more information, see Modify the configurations of an ALB instance.



1. Log on to the ALB console.
2. In the top navigation bar, select the region where the

   ALB

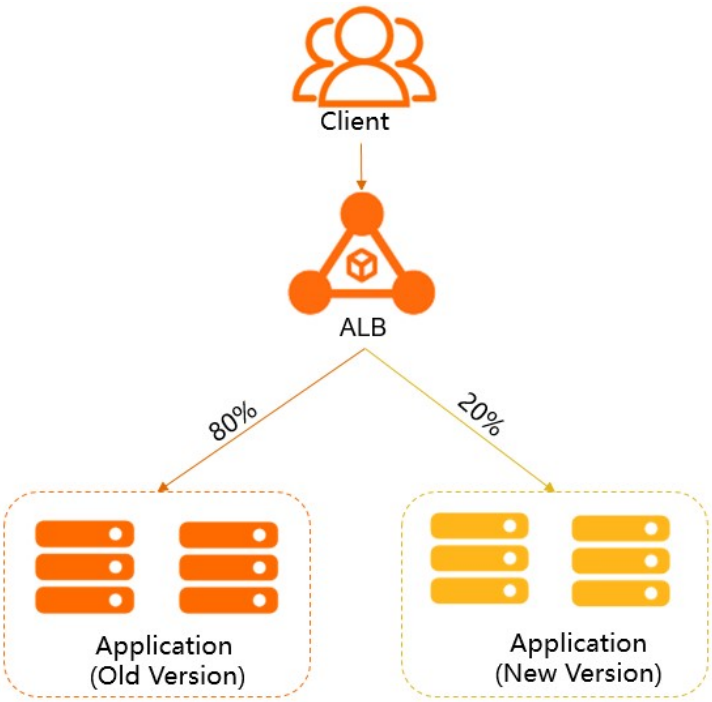
   instance is deployed.
3. On the **Instances** page, click the ID of the ALB instance that you want to manage.
4. On the **Listener** tab, find the listener that you want to manage and click **View/Modify Forwarding Rule** in the **Actions** column.

5. Choose **Forwarding Rules > Inbound Forwarding Rules** and click **Add Forwarding Rules**.
6. Set the parameters and click **OK**.
   - **Forwarding Condition**: Select **Cookie** from the drop-down list and set the value to *key: value*.
   - **Forwarding Action**: Select **Forward** from the drop-down list and select Server Group A.

   For more information about the parameters, see Manage forwarding rules for a listener.
7. You can configure more forwarding rules to increase the percentage of user traffic distributed to the new version. If the new version works as expected, you can discontinue the old version and forward all requests to the new version.

Canary release based on server groups

In this example, 80% of the requests destined for *www.test.com* are forwarded to the old version and 20% of the requests are forwarded to the new version.



1. Log on to the ALB console.
2. In the top navigation bar, select the region where the

   ALB

   instance is deployed.
3. On the **Instances** page, click the ID of the ALB instance that you want to manage.
4. On the **Listener** tab, find the listener that you want to manage and click **View/Modify Forwarding Rule** in the **Actions** column.
5. Choose **Forwarding Rules > Inbound Forwarding Rules** and click **Add Forwarding Rules**.
6. Set the parameters and click **OK**.
   - **Forwarding Condition**: Select **Domain Name** from the drop-down list and set the value to *www.test.com*.
   - **Forwarding Action**: Select **Forward** from the drop-down list, and select Server Group B (set the weight to 80) and Server Group A (set the weight to 20).

For more information about the parameters, see Manage forwarding rules for a listener.

7. You can change the weights of the server groups to meet your business requirements. If the new version works as expected, you can discontinue the old version and forward all requests to the new version.

1.6. Use QUIC to accelerate the delivery of video and audio content

Quick UDP Internet Connection (QUIC) is a network protocol that can accelerate access from clients, especially in scenarios where network connections are weak, or connections are frequently switched between Wi-Fi and cellular networks. QUIC can achieve connection multiplexing without reconnections, accelerate access, and secure data transfer.

Context

QUIC provides the same level of security as SSL, and supports connection multiplexing and zero round trip time resumption (0-RTT). In scenarios of weak connections, high network latency, and packet loss, QUIC ensures service availability. QUIC can implement different congestion control algorithms at the application layer regardless of the operating system or kernel. Compared with TCP, QUIC supports flexible adjustments based on service requirements. QUIC is a suitable alternative when TCP optimization encounters bottlenecks.

As short videos and live streaming services become more and more popular, streaming platforms require high bandwidth and low network latency to meet their business requirements. QUIC can minimize network latency, solve video buffering, accelerate the delivery of audio and video content, and secure data transfer.

Application Load Balancer (ALB)

supports the following versions of QUIC: Q46, Q44, Q43, Q39, Q36, and Q35.

Scenarios

When you use Chrome to access

ALB

,

ALB

distributes requests to a backend server based on the domain name `example.com` that you associate with a listener. The listener that is used to distribute requests varies in the following scenarios:

- If both an HTTPS listener and a QUIC listener are added to the

  ALB

  instance, requests are distributed by the QUIC listener by default. Therefore, after you enter the domain name `example.com` in your Chrome browser, the QUIC listener distributes the request to the default server group RS1.
- If the QUIC listener is unavailable, the associated HTTPS listener takes over to serve your workloads. In this case, after you enter the domain name `example.com` in your Chrome browser, the HTTPS listener of

  ALB

distributes the request to the default server group RS1.

Client requirements

- You can directly initiate QUIC requests from a Chrome browser to

  ALB

  .
- If you use another client, make sure that the client is integrated with a network library such as lsquic-client or Cronet that supports QUIC.
- Before you use Chrome to access a QUIC listener, make sure that your browser supports the QUIC version used by ALB.
  - ALB

    supports Q46 and previous versions of QUIC, which are used by Chrome 74-81.
  - Chrome browsers later than Chrome 74-81 use Q50 and later versions of QUIC. If you use these browsers, you must downgrade the Chrome browser to a previous version before you can access

    ALB

    .

Prerequisites

- An

  ALB

  instance is created. For more information, see Create an ALB instance.
- A server group named RS1 is created. For more information, see Manage server groups.
- An Elastic Compute Service (ECS) instance named ECS01 is added to the server group RS1. A NGINX video service is hosted on ECS01.
- An SSL certificate that is associated with the domain name `example.com` is configured on the

  ALB

  instance.

Step 1: Create a QUIC listener

1. Log on to the ALB console.
2. In the top navigation bar, select the region where the

   ALB

   instance is deployed.
3. On the **Instances** page, find the ALB instance that you want to manage and click its ID. On the **Listener** tab, click **Create Listener**.
4. On the **Configure Listener** wizard page, set the following parameters and click **Next**.

Parameter	Description
Select Listener Protocol	Select a protocol for the listener. In this example, **QUIC** is selected.

Parameter	Description
Listening Port	Enter the port on which the ALB instance listens. The ALB instance uses the port to receive requests and forward the requests to backend servers. Valid values: 1 to 65535. **Note** The ports on which an ALB instance listens must be unique.
Listener Name	Enter a name for the listener.
Advanced	Click **Modify** to configure advanced settings.
Request Timeout	Specify the request timeout period. Unit: seconds. Valid values: 1 to 180. If no response is received from the backend server within the request timeout period, ALB returns an HTTP 504 error to the client.
Enable Gzip Compression	Specify whether to enable Gzip compression for a specified file type. Gzip supports the following file types: `text/xml`, `text/plain`, `text/css`, `application/javascript`, `application/x-javascript`, `application/rss+xml`, `application/atom+xml`, and `application/xml`.
Add HTTP Header Fields	You can add the following HTTP header fields: ○ `SLB-ID`: Add the header field to retrieve the ID of the ALB instance. ○ `X-Forwarded-Proto`: Add the header field to retrieve the listener protocol used by the ALB instance. ○ `X-Forwarded-Port`: Add the header field to retrieve the ports on which the ALB instance listens.

5. In the **Configure SSL Certificate** wizard, select the server certificate and click **Next**.
6. In the **Select Server Group** wizard, specify **Server Type**, and then select a server group. Confirm the information about the backend servers and click **Next**.
7. In the **Configuration Review** wizard, confirm the configuration and click **Submit**.

Step 2: Create an HTTPS listener

When you create an HTTPS listener, enable QUIC upgrade and associate the QUIC listener that you created with the HTTPS listener.

1. On the **Instances** page, find the QUIC listener that you created in Step 1 and click its ID.
2. On the **Listener** tab, click **Create Listener**.
3. On the **Configure Listener** wizard page, set the following parameters and click **Next**.

Parameter	Description
Listener Protocol	Select the protocol of the listener. **HTTPS** is selected in this example.
Listener Port	Enter the port on which the ALB instance listens. The ALB instance listens on the port and forwards requests to backend servers. **443** is entered in this example. In most cases, port 80 is used for HTTP and port 443 is used for HTTPS. Valid values: 1 to 65535. **Note** The ports on which an ALB instance listens must be unique.
Listener Name	Enter a name for the listener. The name must be 2 to 256 characters in length. The name can contain only Chinese characters and the characters in the following string: `/^([^\x00-\xff]|[\w.,;/@-]){2,256}$/` .
Advanced Settings	Click **Modify** to configure advanced settings.
Enable HTTP/2	Specify whether to enable HTTP/2.
Idle Connection Timeout Period	Specify the timeout period of idle connections. Unit: seconds. Valid values: 1 to 60. If no request is received within the specified timeout period, ALB closes the current connection. ALB creates a new connection when a new connection request is received. **Note** This feature is unavailable for HTTP/2 requests.
Connection Request Timeout Period	Specify the request timeout period. Unit: seconds. Valid values: 1 to 180. If no response is received from the backend server within the request timeout period, ALB returns an HTTP 504 error to the client.

Parameter	Description
Enable Gzip Compression	Specify whether to enable Gzip compression for specific file types. Gzip supports the following file types: `text/xml`, `text/plain`, `text/css`, `application/javascript`, `application/x-javascript`, `application/rss+xml`, `application/atom+xml`, `application/xml`, and `application/json`.
Add HTTP Header Fields	You can add the following HTTP header fields: ○ `X-Forwarded-For`: obtains the real IP address of the client. ○ `SLB-ID`: obtains the ID of the ALB instance. ○ `X-Forwarded-Proto`: obtains the listener protocol of the ALB instance. ○ `X-Forwarded-Clientcert-subjectdn`: obtains information about the owner of the client certificate. ○ `X-Forwarded-Clientcert-issuerdn`: obtains information about the authority that issues the client certificate. ○ `X-Forwarded-Clientcert-fingerprint`: obtains the fingerprint of the client certificate. ○ `X-Forwarded-Clientcert-clientverify`: obtains the verification result of the client certificate. ○ `X-Forwarded-Port`: obtains the port on which the ALB instance listens. ○ `X-Forwarded-Client-Port`: obtains the port over which a client communicates with the ALB instance.
QUIC Update	Select whether to enable the QUIC update feature. If you enable QUIC update, select a QUIC listener and associate the listener with the ALB instance.

4. In the **Configure SSL Certificate** wizard, select the server certificate and click **Next**.

> **Note** To configure TLS security policies, click **Modify** next to **Advanced Settings**.

5. In the **Select Server Group** wizard, specify **Server Type** and select a server group from the drop-down list, confirm the backend servers, and then click **Next**.
6. In the **Configuration Review** wizard, confirm the configuration and click **Submit**.

Step 3: Create a CNAME record

Create a CNAME record to map `example.com` to the publicly accessible domain name of the ALB instance.

1. Log on to the ALB console.
2. In the top navigation bar, select the region where the ALB instance is deployed.

3. Find the ALB instance that you want to manage and copy the domain name.
4. To create a CNAME record, perform the following operations:
   i. Log on to the Alibaba Cloud DNS console.
   ii. On the **Manage DNS** page, click **Add Domain Name**.
   iii. In the **Add Domain Name** dialog box, enter the domain name of your host and click **OK**.

> **Notice** Before you create the CNAME record, you must use a TXT record to verify the ownership of the domain name.

   iv. In the **Actions** column of the domain name that you want to manage, click **Configure**.
   v. On the **DNS Settings** page, click **Add Record**.
   vi. In the **Add Record** panel, set the following parameters and click **Confirm**.

Parameter	Description
Type	Select **CNAME** from the drop-down list.
Host	Enter the prefix of the domain name of your host.
ISP Line	Select Default.
Value	Enter the CNAME. The CNAME is the domain name of the ALB instance that you copied in Step 3.
TTL	Select the time-to-live (TTL) value of the record on the DNS server. In this example, the default value is used.

> **Note**
> - Newly created CNAME records immediately take effect. The time that is required for a modified CNAME record to take effect is determined by the TTL value. The default TTL value is 10 minutes.
> - If the CNAME record that you want to create conflicts with an existing record, we recommend that you specify another domain name.

Step 4: Verify the result

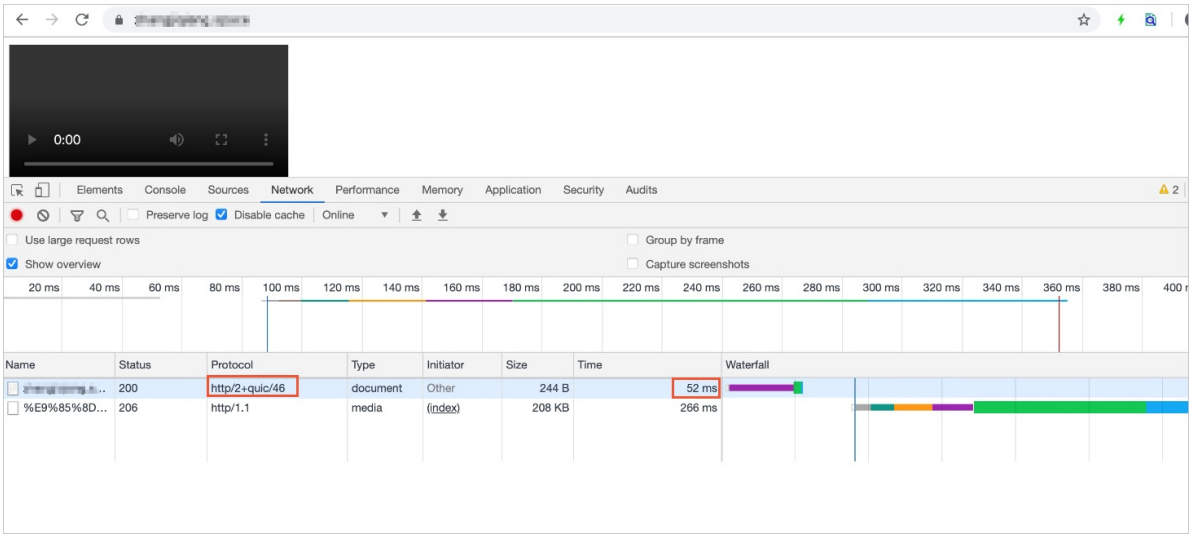
Enter `example.com` in the Chrome browser to access the ALB instance. In this example, NGINX is used to deploy a video service on ECS01 in RS1.

- If both an HTTPS listener and a QUIC listener are added, after you enter `example.com` in the Chrome browser and press `F12`, you can view that **Protocol** displays **http/2+quic/46** and **Time** displays

52ms.

> **Note** http/2+quic/46 indicates that the Q46 protocol is used.

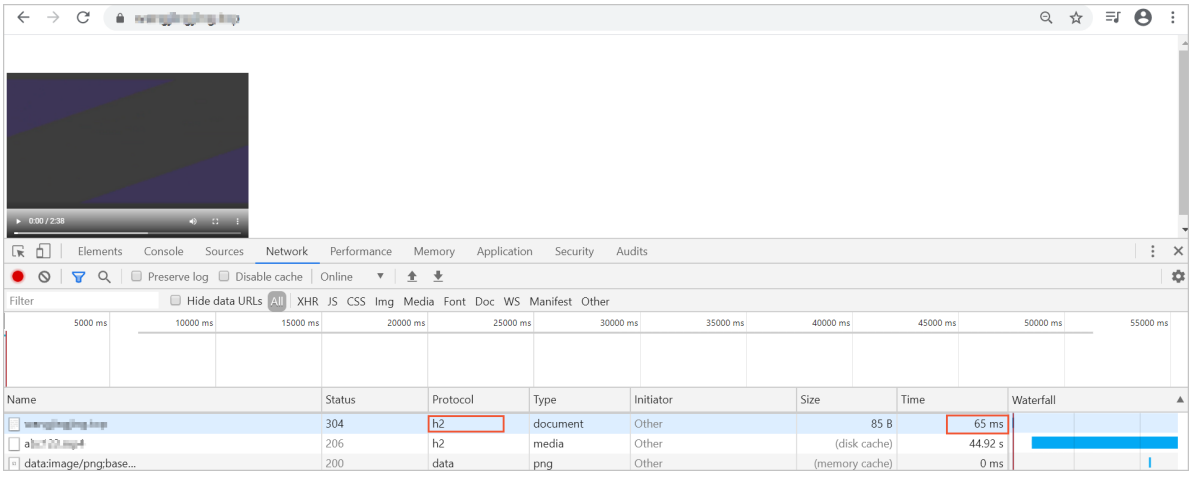
The following figure shows the result.



- If the QUIC listener is unavailable, after you enter `example.com` in the Chrome browser and press `F12`, you can view that **Protocol** displays **h2** and **Time** displays **65ms**.

> **Note** h2 indicates that the HTTPS protocol is used.

The following figure shows the result.



The test result shows that QUIC accelerates the delivery of video content.

1.7. Customize TLS policies to improve website security

When you configure an HTTPS listener, you can select a Transport Layer Security (TLS) policy of a late version to improve the security of your services and websites. A TLS policy consists of the TLS version and corresponding cipher suites.

Context

With the popularity of HTTPS, more and more enterprises and individuals use HTTPS to deploy websites. HTTPS is the future of Internet development.

However, some HTTPS websites have a high security rating, whereas other HTTPS websites have low security rating. The typical reason why some HTTPS websites have a low security rating is that the servers use TLS policies of early versions. Servers that use TLS policies of early versions have a large number of security vulnerabilities and are vulnerable to attacks.

Application Load Balancer (ALB)

supports custom TLS policies. You can specify a custom TLS version and corresponding cipher suites. This improves the security of your services and websites.

Limits

Basic Edition Application Load Balancer (

ALB

) instances do not support custom TLS policies. To use custom TLS policies, you must upgrade

ALB

to Standard Edition. For more information, see Modify the configurations of an ALB instance.

Create an HTTPS listener and use a custom TLS policy

1. In the region where the

   ALB

   instance is deployed, create a custom TLS policy. For more information, see TLS security policies.
2. Create an HTTPS listener and select the custom TLS policy that you created. For more information, see Add an HTTPS listener.

Customize the TLS policy of the HTTPS listener

1. In the region where the

   ALB

   instance is deployed, create a custom TLS policy. For more information, see TLS security policies.
2. In the left-side navigation pane, choose **ALB > Instances**.
3. Find the ALB instance that you want to manage and click its ID.
4. On the **Listener** tab, find the HTTPS listener that you want to manage and click its ID.
5. In the **SSL Certificates** section, click next to **TLS Security Policy**.
6. In the **Edit TLS Security Policy** dialog box, select the custom TLS policy that you created, and click **Save**.

1.8. Add VPC-connected backend servers to an ALB instance across regions

This topic describes how to add VPC-connected backend servers to an Application Load Balancer (ALB) instance across regions. If an

ALB

instance and a backend server that you want to add to the ALB instance belong to virtual private clouds (VPCs) in different regions, you must use transit routers of Cloud Enterprise Network (CEN) to route network traffic from the

ALB

instance to the backend server.

Scenario

The following scenario is used as an example. A company created a VPC named VPC1 in the China (Chengdu) region and deployed an

ALB

instance in the VPC. In addition, the company created a VPC named VPC2 in the China (Hangzhou) region and deployed Elastic Compute Service (ECS) instances in VPC2. The company wants the

ALB

instance to access the ECS instances in VPC2, which is deployed in a different region. To achieve this, VPC1 and VPC2 must be attached to a CEN instance. This way, the ECS instances can be specified as the backend servers of the

ALB

instance. This method ensures data transfer efficiency and reduces latency. This method is suitable for the online gaming and finance industries.

Scenario: Add backend servers in a different region

The following table describes how networks are planned in this example. You can plan the CIDR blocks based on your business requirements. Make sure that the CIDR blocks do not overlap with each other.

Region	VPC	vSwitch	Zone	CIDR block
China (Chengdu)	VPC1 Primary CIDR block: 172.16.0.0/12	VSW1	Chengdu Zone A	172.16.0.0/24
		VSW2	Chengdu Zone B	172.16.6.0/24
China (Hangzhou)	VPC2 Primary CIDR block: 192.168.0.0/16	VSW3	Hangzhou Zone H	192.168.8.0/24
		VSW4	Hangzhou Zone I	192.168.7.0/24

Precautions

- This feature is supported only in the China (Chengdu) region and available only for users whose accounts are included in the whitelist. Therefore, if you want to specify VPC-connected ECS instances as the backend servers of an

  ALB

instance in a different region, make sure that the ALB instance is deployed in the China (Chengdu) region and that your account is included in the whitelist. To use this feature, or contact your sales manager.

- Internet-facing

  ALB

  and internal-facing

  ALB

  instances support this feature.
- Alibaba Cloud provides two editions of

  ALB

  instances: basic and standard ALB instances. Only standard ALB instances support this feature.
- If you want to specify VPC-connected ECS instances as the backend servers of an

  ALB

  instance in a different region, you must specify the ECS instances by IP address.
- VPC1 and VPC2 must be attached to the same CEN instance.
- The Enterprise Edition transit routers that are associated with the VPCs automatically create an elastic network interface (ENI) on the vSwitch in each zone so that network traffic can be routed from the VPCs to the transit routers. When you create the VPCs, you must create a vSwitch in each zone of the Enterprise Edition transit routers so that network traffic can be routed from the VPCs to the transit routers. For more information, see Regions and zones that support Enterprise Edition transit routers.

Preparations

- A VPC (VPC1) is created in the China (Chengdu) region. Another VPC (VPC2) is created in the China (Hangzhou) region. Two vSwitches (VSW1 and VSW2) are created in VPC1. VSW1 is deployed in Zone A. VSW2 is deployed in Zone B. Two vSwitches (VSW3 and VSW4) are created in VPC2. VSW3 is deployed in Zone H. VSW4 is deployed in Zone I. For more information, see Create and manage a VPC.
- An ECS instance named ECS1 is created in VPC2 and an application service is deployed on ECS1. For more information, see Create an instance by using the wizard.
- An

  ALB

  instance is created in VPC1. For more information, see Create an ALB instance.
- A CEN instance is created and a bandwidth plan is associated with the CEN instance. For more information, see Create a CEN instance and 使用带宽包.

Procedure

Procedure for adding backend servers in a different region

Step 1: Create a server group for the

ALB

instance

Create an IP server group and add the IP addresses of the ECS instances that you want to specify as backend servers to the server group.

1. Log on to the ALB console.
2. In the left-side navigation pane, choose **ALB > Server Groups**.
3. On the **Server Groups** page, click **Create Server Group**, set the following parameters and click **Create**.

Parameter	Description
Server Group Type	Select the type of server group that you want to create. In this example, **IP** is selected. **Note** If you want to specify VPC-connected ECS instances as the backend servers of an ALB instance in a different region, you must specify the ECS instances by IP address. In addition, your account must be included in the whitelist.
Server Group Name	Enter a name for the server group. The name must be 2 to 128 characters in length, and can contain letters, digits, periods (.), underscores (_), and hyphens (-). The name must start with a letter.
VPC	Select a VPC from the drop-down list. In this example, **VPC1** is selected.
Backend Server Protocol	Select a backend protocol. **HTTP** is selected in this example. **Note** HTTPS listeners of a basic ALB instance can be associated only with server groups that use HTTP.
Scheduling Algorithm	Select a scheduling algorithm. Default value: **Weight Round Robin**. In this example, the default scheduling algorithm is used.
Resource Group	Select the resource group to which the server group belongs.
Session Persistence	Specify whether to enable session persistence.
Configure Health Check	Specify whether to enable health checks. In this example, health checks are enabled.
Advanced Settings	In this example, the default advanced settings are used. For more information, see Manage server groups.

4. On the **Server Groups** page, find the server group that you want to manage and click **Modify Backend Server** in the **Actions** column.
5. On the **Backend Servers** tab, click **Add IP Address**.

6. In the **Add Backend Server** panel, enter the private IP address of the ECS instance (ECS1 in this example) that you want to specify as a backend server, enable **Remote IP Address**, and then click **Next**.
7. Specify the port and weight of the IP address and click **OK**. In this example, the port is set to 80 and the default weight is used.

Step 2: Add a listener to the

ALB

instance

1. Log on to the ALB console.
2. In the top navigation bar, select the region where the

   ALB

   instance is deployed. In this example, **China (Chengdu)** is selected.
3. On the **Instances** page, find the

   ALB

   instance in VPC1 and click **Create Listener** in the **Actions** column to open the configuration wizard.
4. In the **Configure Listener** step, set the following parameters and click **Next**.

Parameter	Description
Listener Protocol	Select the protocol of the listener. **HTTP** is selected in this example.
Listener Port	Specify the port on which the ALB instance listens. The ALB instance listens on the port and forwards requests to backend servers. Valid values: 1 to 65535. In this example, the value is set to **80**.
Listener Name	Specify a name for the listener.
Advanced Settings	In this example, the default advanced settings are used.

5. In the **Select Server Group** step, select **IP** from the **Server Group** drop-down list, select a server group, and then click **Next**.
6. In the **Confirm** step, confirm the configurations and click **Submit**.

Step 3: Attach the VPCs to the CEN instance

1. Log on to the CEN console.
2. On the **Instances** page of the CEN console, click the ID of the CEN instance that you want to manage.
3. On the details page of the CEN instance, click the ⊕ icon next to **VPC**.
4. On the **Connection with Peer Network Instance** page, set the following parameters and click **OK**.

Parameter	Description
Network Type	In this example, **VPC** is selected.
Region	Select the region where the network instance is deployed. In this example, **China (Chengdu)** is selected.
Transit Router	The system automatically creates a transit router in the selected region. In this example, an Enterprise Edition transit router is created. For more information about transit routers, see How transit routers work.
Select the primary and secondary zones for the transit router	Select the primary and secondary zones of the transit router. In this example, **Primary Zone** is set to **Chengdu Zone A** and **Secondary Zone** is set to **Chengdu Zone B**.
Resource Owner ID	Specify whether the network instance belongs to the current or another account. In this example, **Your Account** is selected.
Billing Method	In this example, **Pay-As-You-Go** is selected.
Attachment Name	Enter a name for the connection. The name must be 2 to 128 characters in length and can contain digits, underscores (_), and hyphens (-). It must start with a letter.
Networks	Select the ID of the VPC that you want to connect. In this example, VPC1 is selected.
VSwitch	Select a vSwitch for the primary zone and secondary zone. In this example, VSW1 is selected for the primary zone and VSW2 is selected for the secondary zone.
Advanced Settings	By default, advanced settings are enabled. In this example, the default advanced settings are used.

5. After you attach VPC1 to the CEN instance, click **Create More Connections** and repeat Substep 4 of Step 3 to attach VPC2 to the CEN instance.

   In this example, the following configurations are used. For the other configurations, the default settings are used.

   - **Region** is set to **China (Hangzhou)**.
   - **Primary Zone** is set to **Hangzhou Zone H** and **Secondary Zone** is set to **Hangzhou Zone I**.
   - VPC2 is selected for **Networks**.
   - For **VSwitch**, VSW3 is used for the primary zone and VSW4 is used for the secondary zone.

Step 4: Create a cross-region connection

1. Log on to the CEN console.
2. On the **Instances** page of the CEN console, click the ID of the CEN instance that you want to manage.

3. On the **Basic Settings > Transit Router** tab, find the transit router that you want to manage, and then click **Create Connection** in the **Actions** column.

   You can choose the transit router that is associated with VPC1 or the transit router that is associated with VPC2. In this example, the transit router associated with VPC1 is used.

4. On the **Connection with Peer Network Instance** page, set the following parameters and click **OK**.

Parameter	Description
Network Type	Select **Cross-region**.
Region	Select the region where the specified transit router is deployed. In this example, **China (Chengdu)** is selected.
Transit Router	Select the transit router deployed in the selected region. If no transit router is found in the selected region, the system automatically creates a transit router.
Attachment Name	Specify a name for the cross-region connection. The name must be 2 to 128 characters in length, and can contain digits, underscores (_), and hyphens (-). It must start with a letter.
Bandwidth Allocation Mode	In this example, **Allocate from Bandwidth Plan** is selected.
Peer Region	Select the region where the peer transit router is deployed. In this example, **China (Hangzhou)** is selected.
Transit Router	Select the peer transit router. If no transit router is found in the selected region, the system automatically creates a transit router.
Bandwidth Plan	Select a bandwidth plan that is associated with the CEN instance.
Bandwidth	Specify a valid bandwidth value. Unit: Mbit/s.
Advanced Settings	In this example, the default advanced settings are used.

Step 5: Add routes to the system route table of VPC1

Check whether the system route table of VPC1 contains a route that points to the VPC1 connection. If no route points to the VPC1 connection, perform the following operations to add a route that points to the VPC1 connection.

1. Log on to the VPC console.
2. On the **VPCs** page, click the ID of VPC1.
3. On the details page, click the **Resources** tab and then click the number below **Route Table**.
4. On the **Route Tables** page, find the route table whose **Route Table Type** is **System** and click its ID.
5. On the **Route Entry List > Custom Route** tab, click **Add Route Entry**.

6. In the **Add Route Entry** panel, set the following parameters and click **OK**.

Parameter	Description
Name	Enter a name for the route entry.
Destination CIDR Block	Enter the CIDR block that you want to access. In this example, the IP address of ECS1 is entered, which is *192.168.7.54*.
Next Hop Type	Select the next hop type. **Transit Router** is selected in this example.
Transit Router	Select a transit router. In this example, the transit router that is associated with VPC1 is selected.

Step 6: Configure back-to-origin routes

View the back-to-origin route of the

ALB

instance. Add the back-to-origin route to the system route table of VPC2 and the route table of the transit router that is associated with VPC1.

1. To view the back-to-origin route of an

   ALB

   instance, perform the following operations:

   i. Log on to the ALB console.

   ii. In the top navigation bar, select the region where the ALB instance is deployed. In this example, **China (Chengdu)** is selected.

   iii. On the **Instances** page, click the ID of the

   ALB

   instance in VPC1.

   iv. On the **Instance Details** tab, click **View** next to **Back-to-origin Route**.

   View the back-to-origin route of an ALB instance

2. To add the back-to-origin route of

   ALB

   to the system route table of VPC2, perform the following operations:

   i. Log on to the VPC console.

   ii. On the **VPCs** page, click the ID of VPC2.

   iii. On the details page, click the **Resources** tab and then click the number below **Route Table**.

   iv. On the **Route Tables** page, find the route table whose **Route Table Type** is **System** and click its ID.

   v. On the **Route Entry List > Custom Route** tab, click **Add Route Entry**.

vi. In the **Add Route Entry** panel, set the following parameters and click **OK**.

Parameter	Description
Name	Enter a name for the route entry.
Destination CIDR Block	Enter the CIDR block that you want to access. In this example, the destination CIDR block of the back-to-origin route of the ALB instance is entered, which is obtained from Substep 1 of Step 6. If the ALB instance has multiple back-to-origin routes, repeat the preceding operations to add all of the back-to-origin routes.
Next Hop Type	Select the next hop type. **Transit Router** is selected in this example.
Transit Router	Select a transit router. In this example, the transit router that is associated with VPC2 is selected.

Add the routes in the following table for VPC2.

Destination CIDR block	Next hop
100.XX.XX.0/25	The transit router associated with VPC2
100.XX.XX.128/25	The transit router associated with VPC2
100.XX.XX.64/26	The transit router associated with VPC2
100.XX.XX.128/26	The transit router associated with VPC2
100.XX.XX.192/26	The transit router associated with VPC2
100.XX.XX.0/26	The transit router associated with VPC2

3. To add the back-to-origin route of

   ALB

   to the transit router associated with VPC1, perform the following operations:

   i. Log on to the CEN console.

   ii. On the **Instances** page of the CEN console, click the ID of the CEN instance that you want to manage.

   iii. Choose **Basic Settings > Transit Router**, find the transit router associated with VPC1 and click its ID.

   iv. On the **Route Table** tab, click the ID of the route table to which you want to add the back-to-origin route, click the **Route Entry** tab, and then click **Add Route Entry**.

v. In the **Add Route Entry** dialog box, set the following parameters and click **OK**.

Parameter	Description
Route Table	By default, the current route table is selected.
Transit Router	By default, the current transit router is selected.
Name	Enter a name for the route. The name must be 0 to 128 characters in length. letters, digits, commas (,), periods (.), semicolons (;), forward slashes (/), at signs (@), underscores (_), and hyphens (-)..
Destination CIDR	Enter the destination CIDR block of the route. In this example, the destination CIDR block of the back-to-origin route of the ALB instance is entered, which is obtained from Substep 1 of Step 6. If the ALB instance has multiple back-to-origin routes, repeat the preceding operations to add all of the back-to-origin routes.
Blackhole Route	Default value: No.
Next Hop	Specify the next hop. In this example, the VPC1 connection is selected.
Description	Enter a description for the route. The description must be 2 to 256 characters in length. letters, digits, commas (,), periods (.), semicolons (;), forward slashes (/), at signs (@), underscores (_), and hyphens (-)..

Add the routes in the following table for the transit router associated with VPC1.

Destination CIDR block	Next hop
100.XX.XX.0/25	VPC1 connection
100.XX.XX.128/25	VPC1 connection
100.XX.XX.64/26	VPC1 connection
100.XX.XX.128/26	VPC1 connection
100.XX.XX.192/26	VPC1 connection
100.XX.XX.0/26	VPC1 connection

Step 7: Test network connectivity

1. Log on to the ECS instance that is deployed in VPC1. For more information, see Connect to an ECS instance.
2. Run the `wget http://`*domain name of the ALB instance* command to check whether the ECS instance in VPC1 can access ECS1 in VPC2 through

ALB

.

In this example, the following command is used:

```
wget http://alb-fo89znps6q********.internal.cn-chengdu.alb.aliyuncs.com
```

If you can receive echo reply packets, it indicates that the connection is established.

Connectivity test

1.9. Connect an on-premises server to ALB

This topic describes how to connect an on-premises server to Application Load Balancer (ALB). You can use services such as Cloud Enterprise Network (CEN) transit routers to connect an on-premises server to an

ALB

instance so that the

ALB

instance can distribute network traffic to the on-premises server.

Scenario

The following scenario is used as an example. A company created a virtual private cloud (VPC) in the China (Chengdu) region and deployed an

ALB

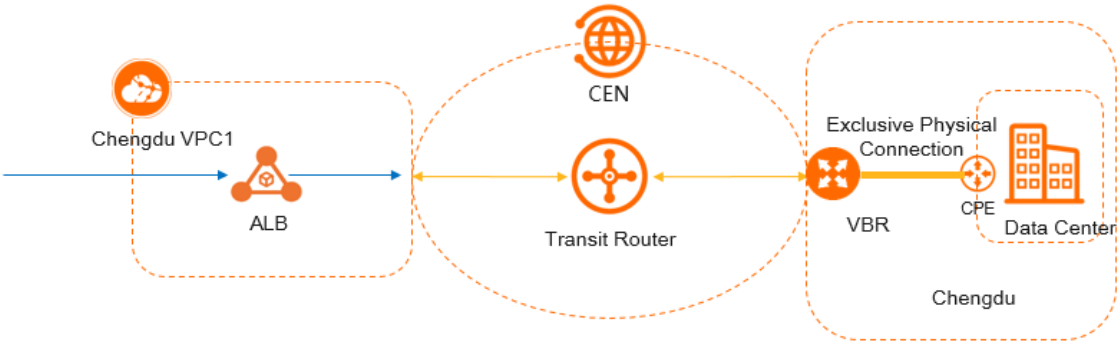
instance in the VPC. The VPC is referred to as VPC1 in this example. The company wants to connect a data center to Alibaba Cloud through a virtual border router (VBR) in the region where the

ALB

instance is deployed and then use CEN to connect the VBR and VPC1. This way, the

ALB

instance in VPC1 can distribute network traffic to the on-premises server.



The following table describes how networks are planned in this example. You can plan CIDR blocks based on your business requirements. Make sure that the CIDR blocks do not overlap with each other.

China (Chengdu)	vSwitch	Zone	CIDR block
VPC1 Primary CIDR block: 172.16.0.0/12	VSW1	Chengdu Zone A	172.16.0.0/24
	VSW2	Chengdu Zone B	172.16.6.0/24
VBR	N/A	N/A	IPv4 CIDR block for the gateway device in the data center: 10.0.0.2/30 IPv4 CIDR block for the VBR: 10.0.0.1/30
Data center	VSW3	N/A	192.168.20.0/24

Precautions

- To connect an on-premises server to an

  ALB

  instance, you must be included in the whitelist. To apply to be included in the whitelist, or contact your customer service manager from Alibaba Cloud.
- You can connect an on-premises server to an Internet-facing

  ALB

  or internal-facing

  ALB

  instance.
- To connect an on-premises server to an

  ALB

  instance, you must specify the on-premises server as a backend server of the ALB instance by IP address.
- The Enterprise Edition transit routers that are associated with the VPCs automatically attach an elastic network interface (ENI) to the vSwitch in each zone so that network traffic can be routed from the VPCs to the transit routers. When you create the VPCs, you must specify a vSwitch for each zone of the Enterprise Edition transit routers so that network traffic can be routed from the VPCs to the transit routers. For more information, see Regions and zones that support Enterprise Edition transit routers.

Prerequisites

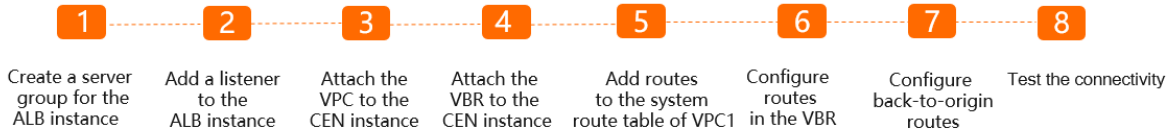
- VPC1 is created in the China (Chengdu) region. Two vSwitches (VSW1 and VSW2) are deployed in VPC1. VSW1 is deployed in Zone A. VSW2 is deployed in Zone B. For more information, see Create and manage a VPC.
- An

  ALB

  instance is created in VPC1. For more information, see Create an ALB instance.

- An Elastic Compute Service (ECS) instance is deployed in VPC1. Application services are hosted on ECS1. For more information, see Create an instance by using the wizard.
- A CEN instance is created. For more information, see Create a CEN instance.
- A connection over an Express Connect circuit is established. A VBR is created. For more information, see Create a dedicated connection over an Express Connect circuit and Create a VBR.

Procedure



Step 1: Create a server group for the

ALB

instance

Create an IP server group and add the IP address of the on-premises server that you want to connect to the server group.

1. Log on to the ALB console.
2. In the left-side navigation pane, choose **ALB > Server Groups**.
3. On the **Server Groups** page, click **Create Server Group**, set the following parameters and click **Create**.

Parameter	Description
Server Group Type	Select the type of server group that you want to create. In this example, **IP** is selected. **Note** To specify an on-premises server as a backend server of an ALB instance, you must add the server by IP address. In addition, your Alibaba Cloud account must be included in the whitelist.
Server Group Name	Enter a name for the server group. The name must be 2 to 128 characters in length, and can contain letters, digits, periods (.), underscores (_), and hyphens (-). The name must start with a letter.
VPC	Select a VPC from the drop-down list. In this example, **VPC1** is selected.
Backend Server Protocol	Select a backend protocol. **HTTP** is selected in this example. **Note** HTTPS listeners of a basic ALB instance can be associated only with server groups that use HTTP.

Parameter	Description
Scheduling Algorithm	Select a scheduling algorithm. Default value: **Weight Round Robin**. In this example, the default scheduling algorithm is used.
Resource Group	Select the resource group to which the server group belongs.
Session Persistence	Specify whether to enable session persistence.
Configure Health Check	Specify whether to enable health checks. In this example, health checks are enabled.
Advanced Settings	In this example, the default advanced settings are used. For more information, see Manage server groups.

4. On the **Server Groups** page, find the server group that you want to manage and click **Modify Backend Server** in the **Actions** column.
5. On the **Backend Servers** tab, click **Add IP Address**.
6. In the **Add Backend Server** panel, enter the IP address of the on-premises server, enable **Remote IP Address**, and then click **Next**.
7. Specify the port and weight of the IP address and click **OK**. In this example, the port is set to 80 and the default weight is used.

Step 2: Add a listener to the

ALB

instance

1. Log on to the ALB console.
2. In the top navigation bar, select the region where the

   ALB

   instance is deployed. In this example, **China (Chengdu)** is selected.
3. On the **Instances** page, find the

   ALB

   instance in VPC1 and click **Create Listener** in the **Actions** column to open the configuration wizard.
4. In the **Configure Listener** step, set the following parameters and click **Next**.

Parameter	Description
Listener Protocol	Select the protocol of the listener. **HTTP** is selected in this example.
Listener Port	Specify the port on which the ALB instance listens. The ALB instance listens on the port and forwards requests to backend servers. Valid values: 1 to 65535. In this example, the value is set to **80**.
Listener Name	Specify a name for the listener.

Parameter	Description
Advanced Settings	In this example, the default advanced settings are used.

5. In the **Select Server Group** step, select **IP** from the **Server Group** drop-down list, select a server group, and then click **Next**.
6. In the **Confirm** step, confirm the configurations and click **Submit**.

Step 3: Attach the VPC to the CEN instance

1. Log on to the CEN console.
2. On the **Instances** page of the CEN console, click the ID of the CEN instance that you want to manage.
3. On the details page of the CEN instance, click the ⊕ icon next to **VPC**.
4. On the **Connection with Peer Network Instance** page, set the following parameters and click **OK**.

Parameter	Description
Network Type	In this example, **VPC** is selected.
Region	Select the region where the network instance is deployed. In this example, **China (Chengdu)** is selected.
Transit Router	The system automatically creates a transit router in the selected region. In this example, an Enterprise Edition transit router is created. For more information about transit routers, see How transit routers work.
Select the primary and secondary zones for the transit router	Select the primary and secondary zones of the transit router. In this example, **Primary Zone** is set to **Chengdu Zone A** and **Secondary Zone** is set to **Chengdu Zone B**.
Resource Owner ID	Specify whether the network instance belongs to the current or another account. In this example, **Your Account** is selected.
Billing Method	In this example, **Pay-As-You-Go** is selected.
Attachment Name	Enter a name for the connection. The name must be 2 to 128 characters in length and can contain digits, underscores (_), and hyphens (-). It must start with a letter.
Networks	Select the ID of the VPC that you want to connect. In this example, VPC1 is selected.
VSwitch	Select a vSwitch for the primary zone and secondary zone. In this example, VSW1 is selected for the primary zone and VSW2 is selected for the secondary zone.

Parameter	Description
Advanced Settings	By default, advanced settings are enabled. In this example, the default advanced settings are used.

Step 4: Attach the VBR to the CEN instance

1. After you attach VPC1 to the CEN instance, click **Create More Connections**.
2. On the **Connection with Peer Network Instance** page, set the following parameters and click **OK**.

Parameter	Description
Network Type	In this example, **Virtual Border Router (VBR)** is selected.
Region	Select the region where the network instance is created. In this example, **China (Chengdu)** is selected.
Transit Router	The system automatically selects the transit router in the current region. In this example, the transit router in the China (Chengdu) region is selected.
Resource Owner ID	Specify whether the network instance belongs to the current or another Alibaba Cloud account. In this example, **Your Account** is selected.
Attachment Name	Enter a name for the connection. The name must be 2 to 128 characters in length and can contain digits, underscores (_), and hyphens (-). It must start with a letter.
Networks	Select the ID of the VBR that you want to connect. In this example, the VBR that you created is selected.
Advanced Settings	By default, advanced settings are enabled. In this example, the default advanced settings are used. For more information, see 创建VBR连接.

Step 5: Add routes to the system route table of VPC1

Check whether the system route table of VPC1 contains a route that points to the VPC1 connection. If no route points to the VPC1 connection, perform the following operations to add a route that points to the VPC1 connection.

1. Log on to the VPC console.
2. On the **VPCs** page, click the ID of VPC1.
3. On the details page, click the **Resources** tab and then click the number below **Route Table**.
4. On the **Route Tables** page, find the route table whose **Route Table Type** is **System** and click its ID.
5. On the **Route Entry List > Custom** tab, click **Add Route Entry**.
6. In the **Add Route Entry** panel, set the following parameters and click **OK**.

Parameter	Description
Name	Enter a name for the route.
Destination CIDR Block	Enter the CIDR block that you want to access. In this example, the CIDR block of the on-premises server is entered, which is *192.168.20.0/24*.
Next Hop Type	Select the next hop type. **Transit Router** is selected in this example.
Transit Router	Select a transit router. In this example, the transit router that is associated with VPC1 is selected.

Step 6: Configure routes in the VBR

Add a route that points to the data center to the route table of the VBR.

1. Log on to the Express Connect console.
2. In the top navigation bar, select the region and click **Virtual Border Routers (VBRs)** in the left-side navigation pane.
3. On the **Virtual Border Routers (VBRs)** page, find the VBR that you want to manage and click its ID.
4. On the details page of the VBR, click the **Routes** tab and click **Add Route**.
5. In the **Add Route Entry** panel, set the following parameters and click **OK**.

Parameter	Description
Next Hop Type	Select the type of next hop. In this example, **Physical Connection Interface** is selected.
Destination CIDR Block	In this example, the CIDR block of the on-premises sever is entered, which is *192.168.20.0/24*.
Next Hop	Select an Express Connect circuit.

Step 7: Configure back-to-origin routes

View the back-to-origin route of the

ALB

instance. Add the back-to-origin route to the route table of the transit router that is associated with VPC1 and to the route table in the data center.

1. To view the back-to-origin route of an

   ALB

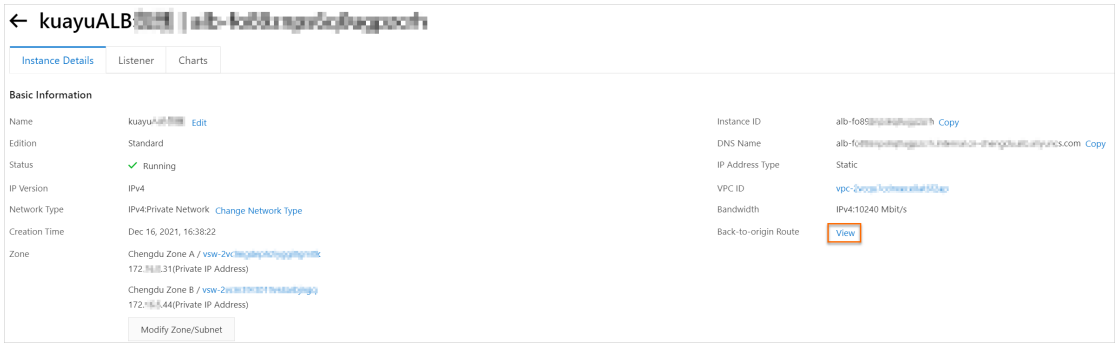
   instance, perform the following operations:

   i. Log on to the ALB console.

   ii. In the top navigation bar, select the region where the ALB instance is deployed. In this example, **China (Chengdu)** is selected.

iii. On the **Instances** page, click the ID of the ALB instance in VPC1.

iv. On the **Instance Details** tab, click **View** next to **Back-to-origin Route**.



2. To add the back-to-origin route of ALB to the transit router associated with VPC1, perform the following operations:

i. Log on to the CEN console.

ii. On the **Instances** page, click the ID of the CEN instance that you want to manage.

iii. Choose **Basic Settings > Transit Router**, find the transit router associated with VPC1 and click its ID.

iv. On the **Route Table** tab, click the ID of the route table to which you want to add the back-to-origin route, click the **Route Entry** tab, and then click **Add Route Entry**.

v. In the **Add Route Entry** dialog box, set the following parameters and click **OK**.

Parameter	Description
Route Table	By default, the current route table is selected.
Transit Router	By default, the current transit router is selected.
Name	Enter a name for the route. The name must be 0 to 128 characters in length, and can contain letters, digits, commas (,), periods (.), semicolons (;), forward slashes (/), at signs (@), underscores (_), and hyphens (-).
Destination CIDR	Enter the destination CIDR block of the route. In this example, the destination CIDR block of the back-to-origin route of the ALB instance is entered, which is obtained from Substep 1 of Step 6. If the ALB instance has multiple back-to-origin routes, repeat the preceding operations to add all of the back-to-origin routes.
Blackhole Route	Default value: No.
Next Hop	Specify the next hop. In this example, the transit router that is associated with VPC1 is selected.
Description	Enter a description for the route. The description must be 2 to 256 characters in length and can contain letters, digits, commas (,), periods (.), semicolons (;), forward slashes (/), at signs (@), underscores (_), and hyphens (-).

Add the routes in the following table for the transit router associated with VPC1.

Destination CIDR block	Next hop
100.XX.XX.0/25	VPC1 connection
100.XX.XX.128/25	VPC1 connection
100.XX.XX.64/26	VPC1 connection
100.XX.XX.128/26	VPC1 connection
100.XX.XX.192/26	VPC1 connection
100.XX.XX.0/26	VPC1 connection

3. Perform the following operations to add the back-to-origin route of the

   ALB

   instance to the data center.

   Add the back-to-origin route of the

ALB

instance to the gateway device in the data center. The following content describes the route configuration in this example. If the ALB instance has multiple back-to-origin routes, repeat the preceding operations to add all of the back-to-origin routes.

> **Note** The route configuration in this example is only for reference. The route configuration may vary based on the gateway device.

```
ip route 100.XX.XX.0/25 255.255.255.128 IP address for the VBR
```

Step 8: Test the connectivity

1. Log on to the ECS instance that is deployed in VPC1. For more information, see Connect to an ECS instance.
2. Run the `wget http://domain name` command to check whether the ECS instance in VPC1 can access the on-premises server.

   In this example, the following command is used:

   ```
   wget http://alb-fo89znps6q********.internal.cn-chengdu.alb.aliyuncs.com
   ```

   If you can receive echo reply packets, the connection is established.

2.CLB

2.1. Configure an HTTPS listener for one-way authentication

To configure an HTTPS listener for one-way authentication, you only need to upload a server certificate.

Step 1: Upload a server certificate

Before you configure an HTTPS listener for one-way authentication, you must purchase a server certificate and upload the server certificate to the certificate management system of Classic Load Balancer (CLB). You do not need to configure the backend Elastic Compute Service (ECS) instances.

1. Log on to the CLB console.
2. In the left-side navigation pane, click **Certificates**. Then, click **Create Certificate**.
3. Select Certificate Source: **Upload Third-party Certificate**
4. Set the following parameters:
   - Certificate Name: The name must be 1 to 80 characters in length and can contain letters, digits, hyphens (-), forward slashes (/), periods (.), underscores (_), and asterisks (*).
   - Region: Select **China (Hangzhou)**.

     > **Note** The region that you select must be the same as the region where the CLB instance is deployed.

   - Certificate Type: Select **Server Certificate**.
   - Public Key Certificate and Private Key: Paste the content of the server certificate and the private key in the fields. You can click **Example** to view the valid certificate formats. The certificate that you want to upload must be in the PEM format. For more information, see Certificate requirements.
5. Click **OK**.

Step 2: Configure a CLB instance

1. Log on to the CLB console.
2. On the **Instances** page, click **Create CLB**.
3. Set the required parameters, click **Buy Now**, and then complete the payment.

   Set Instance Type to **Internet** and Region to **China (Hangzhou)**. For more information, see Create a CLB instance.
4. Go back to the **Instances** page and select the **China (Hangzhou)** region.
5. Find the CLB instance that you created and click its ID or click **Configure Listener** in the Actions column.
6. On the **Listener** tab, click **Add Listener**.
7. On the **Protocol and Listener** wizard page, set the following parameters:

- **Select Listener Protocol**: Select HTTPS.
- **Listening Port**: Enter 443.
- **Scheduling Algorithm**: Select Round-Robin (RR).

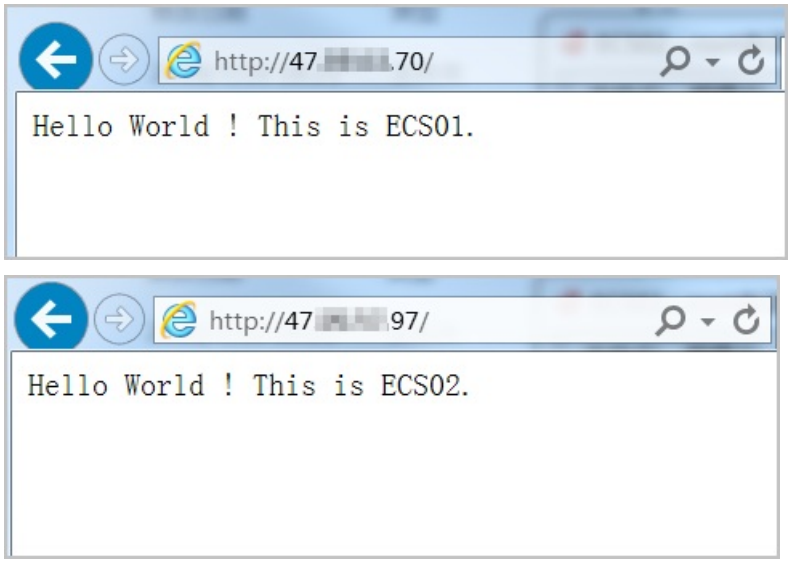
8. Click **Next**. On the **SSL Certificates** wizard page, select the server certificate that you uploaded and a TSL security policy.
9. Click **Next**. On the Backend Servers wizard page, click **Default Server Group** and click **Add More**. Add ECS instances and set the port to 80.
10. Use the default values for other parameters and click **Next**. On the Confirm wizard page, click **Submit**.

Step 3: Test the CLB service

1. Go back to the **Instances** page and view the health check status.

   If **Normal** is displayed, this indicates that the backend servers can receive requests from listeners.

2. Enter the public IP address of the CLB instance in the browser.



2.2. Configure an HTTPS listener for mutual authentication

To configure an HTTPS listener for mutual authentication, you must upload a server certificate and a CA certificate.

In this example, a self-signed CA certificate is used to sign the client certificate. Perform the following steps to configure the HTTPS listener:

1. Prepare a server certificate.
2. Generate a CA certificate by using OpenSSL.
3. Generate a client certificate.
4. Upload the server certificate and the CA certificate.
5. Install the client certificate.
6. Configure an HTTPS listener for mutual authentication.

7. Test the CLB service.

Step 1: Prepare a server certificate

A server certificate is used by the client browser to check whether the certificate sent by the server is signed and issued by a trusted authority. You can purchase a server certificate from SSL Certificates Service of Alibaba Cloud or another service provider.

Step 2: Generate a CA certificate by using Open SSL

1. Run the following commands to create a *ca* folder in the */root* directory and then create four subfolders in the *ca* folder:

```
sudo mkdir ca
 cd ca
 sudo mkdir newcerts private conf server
```

   where:

   - *newcerts* stores the digital certificate signed by the CA certificate.
   - *private* stores the private key of the CA certificate.
   - *conf* stores the configuration files used for simplifying parameters.
   - *server* stores the server certificate.

2. Create an *openssl.conf* file that contains the following information in the *conf* directory:

```
[ ca ]
 default_ca = foo
 [ foo ]
 dir = /root/ca
 database = /root/ca/index.txt
 new_certs_dir = /root/ca/newcerts
 certificate = /root/ca/private/ca.crt
 serial = /root/ca/serial
 private_key = /root/ca/private/ca.key
 RANDFILE = /root/ca/private/.rand
 default_days = 365
 default_crl_days= 30
 default_md = md5
 unique_subject = no
 policy = policy_any
 [ policy_any ]
 countryName = match
 stateOrProvinceName = match
 organizationName = match
 organizationalUnitName = match
 localityName = optional
 commonName      = supplied
 emailAddress    = optional
```

3. Run the following commands to generate a private key:

```
  cd /root/ca
   sudo openssl genrsa -out private/ca.key
```

A similar output is displayed.

4. Run the following command, enter the required information as prompted, and then press Enter to generate a *csr* file.

```
sudo openssl req -new -key private/ca.key -out private/ca.csr
```

> **Note** Common Name specifies the domain name of the CLB instance.

5. Run the following command to generate a *crt* file:

```
sudo openssl x509 -req -days 365 -in private/ca.csr -signkey private/ca.key -out priv
ate/ca.crt
```

6. Run the following command to set the initial sequence number of the CA key. The key can be any four characters:

```
sudo echo FACE > serial
```

7. Run the following command to create a CA key library:

```
sudo touch index.txt
```

8. Run the following command to create a certificate revocation list for removing the client certificate:

```
sudo openssl ca -gencrl -out /root/ca/private/ca.crl -crldays 7 -config "/root/ca/con
f/openssl.conf"
```

Output:

```
Using configuration from /root/ca/conf/openssl.conf
```

Step 3: Generate a client certificate

1. Run the following command to create the *users* directory in the *ca* directory to store client keys:

```
sudo mkdir users
```

2. Run the following command to create a client key:

```
sudo openssl genrsa -des3 -out /root/ca/users/client.key 1024
```

> **Note** When you create the key, enter a passphrase for the key to prevent unauthorized access. Enter the same passphrase twice.

3. Run the following command to create a *csr* file for the client key:

```
sudo openssl req -new -key /root/ca/users/client.key -out /root/ca/users/client.csr
```

Enter the passphrase in the previous step and other required information as prompted.

> **Note** In this example, the passphrase for the client certificate is `A challenge password` and the passphrase for `client.key` is test. The password for the client certificate can be the same as that of the root certificate or server certificate.

4. Run the following command to use the CA key from Step 2 to sign the client key:

```
sudo openssl ca -in /root/ca/users/client.csr -cert /root/ca/private/ca.crt -keyfile /root/ca/private/ca.key -out /root/ca/users/client.crt -config "/root/ca/conf/openssl.conf"
```

   Enter *y* when you are prompted to confirm the following two operations.

5. Run the following command to convert the certificate to a *PKCS12* file that can be verified by most browsers:

```
sudo openssl pkcs12 -export -clcerts -in /root/ca/users/client.crt -inkey /root/ca/users/client.key -out /root/ca/users/client.p12
```

   Enter the passphrase for the client key and press Enter.

   Then, enter the password used to export the client certificate. This password is used to protect the client certificate and is required when the client certificate is installed.

6. Run the following commands to view the generated client certificate:

```
cd users
 ls
```

Step 4: Upload the server certificate and the CA certificate

1. Log on to the CLB console.
2. On the **Instances** page, click **Create CLB**.
3. Set the parameters of the CLB instance, click **Buy Now**, and then complete the payment.

   Set Instance Type to **Internet** and Region to **China (Hangzhou)**. For more information, see Create a CLB instance.
4. On the **Instances** page, find the CLB instance that you created and click the icon next to the instance name to modify the name.
5. In the left-side navigation pane, click **Certificates**.
6. Click **Create Certificate**.
7. On the **Create Certificate** page, set the parameters of the certificate and click **OK**.
   - **Region**: Select the region. In this example, **China (Hangzhou)** is selected.

     > **Note** The region of the certificate must be the same as that of the CLB instance.

- **Certificate Type**: Select the type of certificate that you want to upload. In this example, **Server Certificate** is selected.
- **Public Key Certificate and Private Key**: Paste the public key and private key to the field.

> **Note** Before you paste the keys, you can click **Example** to view the valid certificate format. For more information, see Certificate requirements.

8. In the left-side navigation pane, click **Certificates**. Then click **Create Certificate** to upload the CA certificate.
9. On the **Create Certificate** page, set the parameters of the certificate and click **OK**.
   - **Region**: Select the region. In this example, **China (Hangzhou)** is selected.

> **Note** The region of the certificate must be the same as that of the CLB instance.

   - **Certificate Type**: Select the type of certificate that you want to upload. In this example, **CA Certificate** is selected.
   - **Client CA Certificate**: Paste the CA certificate to the field.

> **Note** Before you paste the certificate, you can click **Example** to view the valid certificate format. For more information, see Certificate requirements.

Step 5: Install the client certificate

Install the generated client certificate on the client. In this example, the Windows operating system and Internet Explorer browser are used.

1. Open Git Bash and run the following command to export the client certificate from Step 3.

```
scp root@IPaddress:/root/ca/users/client.p12 ./
```

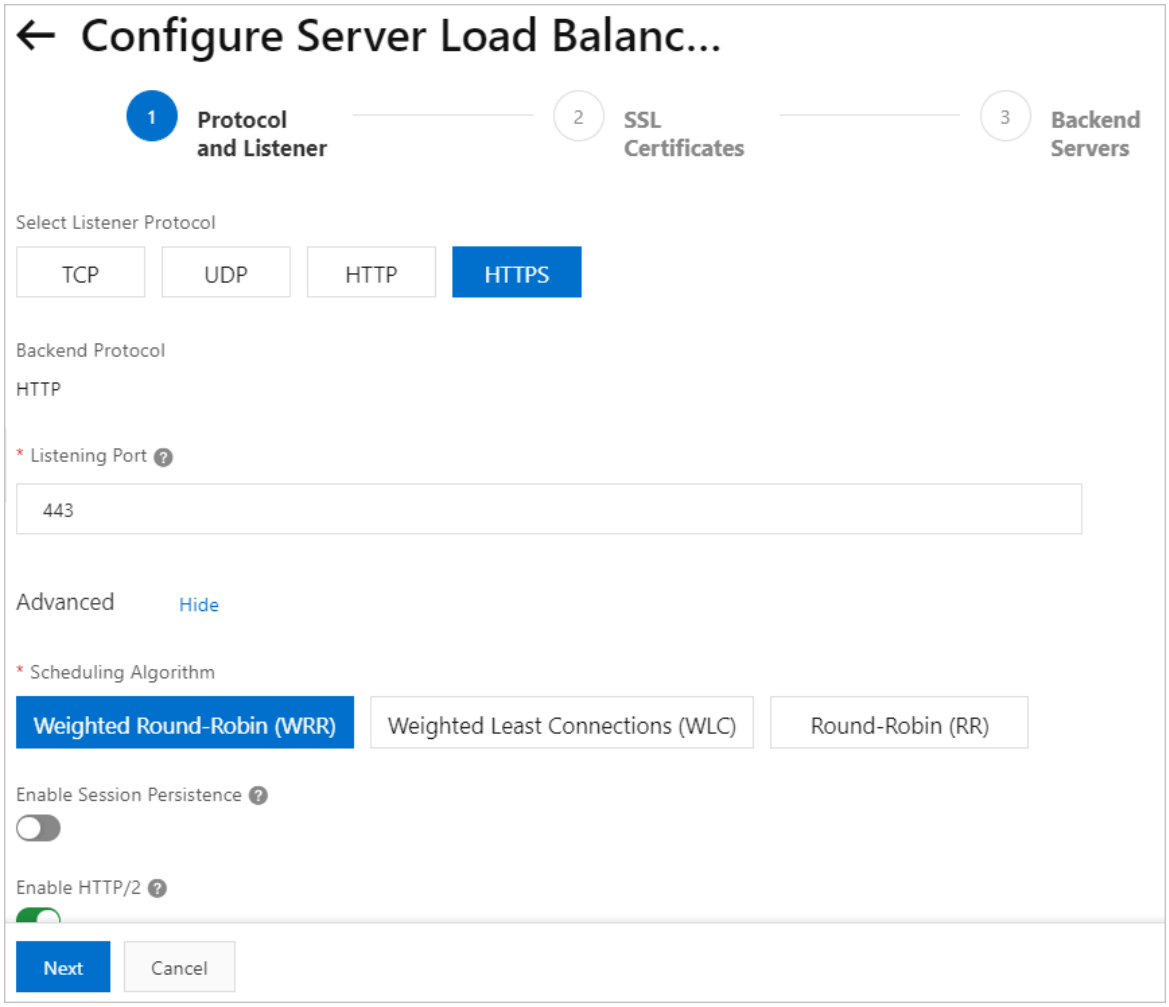
> **Note** IPaddress is the IP address of the server from which the client certificate is generated.

2. Import the certificate to the Internet Explorer browser.
   i. Open Internet Explorer and choose **Settings > Internet Options**.
   ii. Click the **Content** tab, and then click **Certificates** to import the downloaded client certificate. When you import the certificate, enter the password of the *PKCS12* file from Step 3.

Step 6: Configure an HTTPS listener for mutual authentication

1. Log on to the CLB console.
2. On the Instances page, select the **China (Hangzhou)** region. Find the CLB instance that you want to manage and click its ID or click **Configure Listener** in the Actions column.
3. On the **Listener** tab, click **Add Listener**.
4. On the **Protocol and Listener** wizard page, set the parameters of the listener.
   - **Select Listener Protocol**: HTTPS

- **Listening Port**: 443
- **Scheduling Algorithm**: Round Robin (RR)

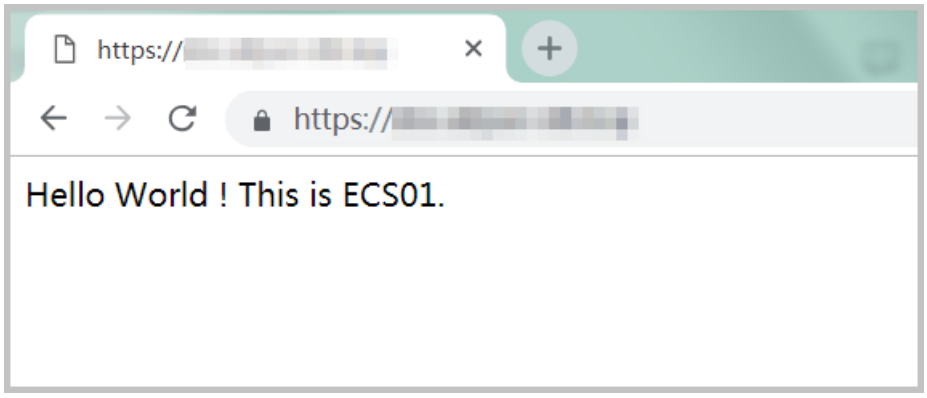


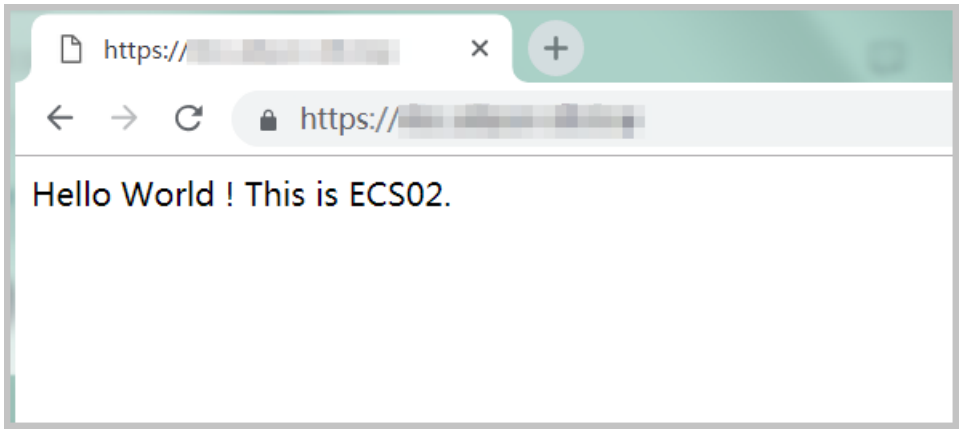
5. Click **Next**. On the **SSL Certificates** wizard page, set the parameters of the certificates and turn on the Enable Mutual Authentication switch.
   - **Select Server Certificate**: Select the server certificate that you uploaded.
   - **Select CA Certificate**: Select the CA certificate that you uploaded.
6. Click **Next**. On the Backend Servers wizard page, click **Default Server Group** and then click **Add More**. Add ECS instances and set the backend port to 80.
7. Click **Next** and turn on the Enable Health Check switch.
8. Click **Next** and check the HTTPS listener configurations.
9. Click **Submit**.
10. Click **OK**.

Step 7: Test the CLB service

1. On the **Instances** page, view the health check status. If **Normal** is displayed, the backend servers can receive requests forwarded by CLB listeners.
2. Enter the publicly accessible domain name of the CLB instance in the browser. Select Trust when you are prompted whether to trust the client certificate.

3. Refresh the web page. You can find that the requests are evenly distributed to the backend servers.





2.3. Redirect requests from HTTP to HTTPS

HTTPS is a protocol used to secure data transmission. Classic Load Balancer (CLB) supports HTTP-to-HTTPS redirection to achieve end-to-end data transfer over HTTPS. This secures data transmission. HTTP-to-HTTPS redirection is available in all regions.

Prerequisites

A CLB instance is created. For more information, see **Create a CLB instance**.

Context

In this example, HTTP-to-HTTPS redirection is enabled for CLB to redirect HTTP requests received on port 80 to port 443 that is used by an HTTPS listener.

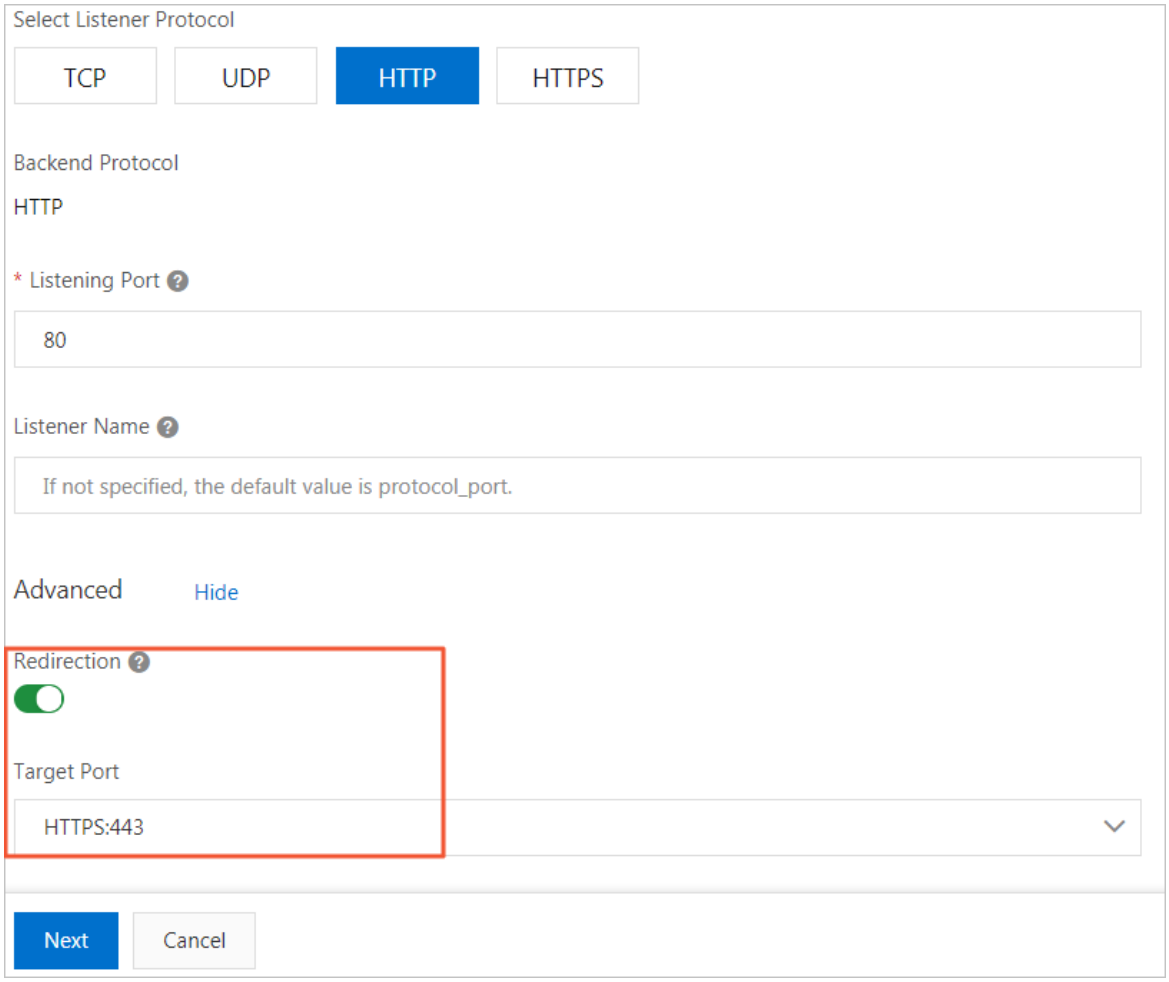
Step 1: Create an HTTPS listener

- Log on to the CLB console.
- In the top navigation bar, select the region where the CLB instance is deployed.
- Use one of the following methods to open the listener configuration wizard:
  - On the **Instances** page, find the CLB instance that you want to manage and click **Configure Listener** in the **Actions** column.
  - On the **Instances** page, click the ID of the CLB instance that you want to manage. On the **Listener** tab, click **Add Listener**.

- In the **Protocol and Listener** step, set the following parameters and click **Next**: For more information about the other parameters and how to create an HTTPS listener, see **Create an HTTPS listener**.
  - **Select Listener Protocol**: **HTTPS**.
  - **Listening Port**: **443**.

Step 2: Create an HTTP listener and enable HTTP-to-HTTPS redirection for the listener

1. Log on to the CLB console.
2. In the top navigation bar, select the region where the CLB instance is deployed.
3. Use one of the following methods to open the listener configuration wizard:
   - On the **Instances** page, find the CLB instance that you want to manage and click **Configure Listener** in the **Actions** column.
   - On the **Instances** page, click the ID of the CLB instance that you want to manage. On the **Listener** tab, click **Add Listener**.
4. In the **Protocol and Listener** step, set Select Listener Protocol to **HTTP** and Listening Port to **80**.
5. Click **Modify** next to **Advanced**.
6. Enable **Redirection**, set Target Port to **HTTPS:443**, and then click **Next**.



7. Confirm the configurations and click **Submit**. In the message that appears, click **OK**.

2.4. Configure a multi-domain HTTPS website on an SLB instance

This topic describes how to add domain name extensions.

Background information

A guaranteed-performance CLB instance CLB1 in the China (Hangzhou) region is used as an example. An HTTPS listener is added to the SLB instance. One-way authentication is enabled for the listener. You want to forward requests from the domain name *.example.com to the VServer group test1 and forward requests from the domain name www.aliyundoc.com to the VServer group test2.

Perform the following steps:

1. Add an HTTPS listener.
2. Configure forwarding rules.
3. Add domain name extensions.

Prerequisites

- A guaranteed-performance CLB instance CLB1 is created in the China (Hangzhou) region. For more information, see Create a CLB instance.
- An SSL certificate is uploaded. For more information, see Certificate overview.
  - The default certificate used by the listener is default.
  - The certificate example1 is used by the domain *.example.com.
  - The certificate example2 is used by the domain www.aliyundoc.com.

Step 1: Add an HTTPS listener

Perform the following operations:

1. In the left-side navigation pane, choose **Instances > Instances**.
2. On the **Instances** page, find the CLB1 instance and click **Configure Listener** in the **Actions** column.

   If it is the first time you configure a listener for the instance, you can also click **Configure** in the **Port/Health Check/Backend Server** column.
3. Configure a listener.

   The following configurations are used for this example. For more information, see Add an HTTPS listener.

   - Mutual authentication: Disabled.
   - SSL certificate: Select the uploaded server certificate that is named default.
   - Backend servers: Create VServer groups test1 and test2.

Step 2: Configure forwarding rules

Perform the following operations:

1. Click the ID of the CLB1 instance to go to the **Instance Details** page.
2. On the **Listener** tab, find the HTTPS listener and then **Set Forwarding Rule**.
3. On the **Add Forwarding Rule** page, configure a forwarding rule. For more information, see Forward requests based on domain names or URLs.

   For this example, domain name-based forwarding rules are configured and URLs are left empty.

   - Set a rule name. Enter *.example.com in the **Domain Name** column, select the VServer group test1, and then click **Add Rule**.
   - Set a rule name. Enter www.aliyundoc.com in the **Domain Name** column, select the VServer group test2, and then click **OK**.

   > **Note** The domain names configured in the forwarding rules must be the same as the domain names added in the certificate and Step 3: Add domain name extensions.

Step 3: Add domain name extensions

Perform the following operations:

1. Click the ID of the CLB1 instance to go to the **Instance Details** page.
2. On the **Listener** tab, find the HTTPS listener and then choose > **Additional Domains**.
3. On the **Additional Domains** page, click **Add Additional Domain** to add domain name extensions.
   - Enter domain names. The domain name can contain only letters, digits, hyphens (-), and periods (.).

     Domain name-based forwarding rules include exact matching and wildcard matching.

     - Exact domain name: www.aliyun.com
     - Wildcard domain name: *.aliyun.com and *.market.aliyun.com

       When a request matches multiple forwarding rules, exact matching prevails over exact wildcard matching, and exact wildcard matching prevails over less exact wildcard matching. The following table describes the priority of domain name-based forwarding rules.

Mode	Request URL	Domain name-based forwarding rule		
		www.aliyun.com	*.aliyun.com	*.market.aliyun.com
Exact matching	www.aliyun.com	✓	×	×
Exact wildcard matching	market.aliyun.com	×	✓	×
Less exact wildcard matching	info.market.aliyun.com	×	×	✓

   - Select the certificate associated with the domain name.

> **Note** The domain name in the certificate must be the same as the added domain name extension.

> **Notice** If a problem occurs after the configuration is complete, restart the browser to avoid the impact of the cache on the results.

2.5. Forward requests based on domain names or URLs

Classic Load Balancer (CLB)

supports domain-based and URL-based forwarding rules. You can forward requests from different domain names or URLs to different backend servers. This allows you to optimize load balancing among your server resources.

> **Note** You can configure forwarding rules only for Layer 7 listeners that use HTTPS or HTTP.

Introduction

You can configure domain-based or URL-based forwarding rules for Layer 7 Classic Load Balancer (CLB) instances. After you configure the rules, requests are distributed to Elastic Compute Service (ECS) instances based on domain names or URLs.

URL-based forwarding rules support string matching. Longest prefix matching is applied. For example, if two forwarding rules */abc* and */abcd* are configured and a request with the prefix of */abcde* is received, the rule */abcd* is applied.

Domain name-based forwarding rules support exact matching and wildcard matching.

- You can enter a specific domain name such as `www.aliyun.com`.
- Wildcard domain name: `*.aliyun.com` and `*.market.aliyun.com`.

  If a request matches multiple forwarding rules, exact matching has a higher priority than wildcard matching, and exact wildcard matching has a higher priority than less exact wildcard matching. The following table describes the priorities.

  > **Note** In the following table, "√" indicates that the request matches the rule and "×" indicates that the request does not match the rule.

Type	Request URL	Domain name-based forwarding rule		
		www.aliyun.com	*.aliyun.com	*.market.aliyun.com
Exact matching	www.aliyun.com	√	×	×
Wildcard matching	market.aliyun.com	×	√	×
	info.market.aliyun.com	×	×	√

You can add multiple forwarding rules to a listener. Each forwarding rule is associated with a vServer group. A vServer group contains one or more ECS instances. You can configure a listener to forward read requests to one vServer group and forward write requests to another vServer group. This allows you to optimize load balancing among your server resources.

- Model 1: If a request contains a domain name, the system matches the domain name of the request against domain-based forwarding rules.
  - If the domain name of the request matches the domain name specified in a forwarding rule, the system continues to match the URL.

    In this case, if the URL of the request matches the URL specified in a forwarding rule, the request is forwarded to the vServer group specified in this forwarding rule. If no forwarding rule matches the URL, the request is forwarded based on a root domain-based forwarding rule. Root domain-based forwarding rules contain only domain names. URLs are not specified in root domain-based forwarding rules.

    If root domain-based forwarding rules are not configured for the domain name, the error message 404 is returned to the client.
  - If no forwarding rule matches the domain name of the request, the request is forwarded in Model 2.
- Model 2: If a request does not contain a domain name or no forwarding rule matches the domain name, the request is forwarded based on URL-based forwarding rules. URL-based forwarding rules contain only URLs. Domain names are not specified in URL-based forwarding rules.

  In this case, if the URL of the request matches the URL specified in a URL-based forwarding rule, the request is forwarded to the vServer group specified in the forwarding rule. If no URL-based forwarding rule matches the URL, the request is forwarded to the default server group.

Configure domain-based or URL-based forwarding rules

Before you configure forwarding rules for a listener, make sure that the following requirements are met:

- For more information, see Add an HTTP listener or Add an HTTPS listener.
- Create a vServer group

To configure domain-based and URL-based forwarding rules, perform the following steps:

1. Log on to the CLB console.
2. Select the region where the CLB instance is deployed.
3. On the **Instances** page, click the ID of the CLB instance that you want to manage.
4. On the **Listener** tab, find the Layer 7 listener that you want to manage and click **Set Forwarding Rule** in the **Actions** column.
5. In the **Add Forwarding Rules** panel, set the following parameters:
   - **Domain Name**: Enter the domain name from which requests are forwarded. The domain name can contain only letters, digits, hyphens (-), and periods (.).
   - **URL**: Enter the URL to which requests are forwarded. The URL must start with a forward slash (/). The URL can contain only letters, digits, hyphens (-), periods (.), forward slashes (/), percent signs (%), question marks (?), number signs (#), and ampersands (&).

> **Note** If the URL of a request contains special characters, you must use URL escape codes to convert the special characters. For example, if the URL of a request contains a number sign (#), the number sign must be converted to the escape code (% 23). This way, the system can match the URL of the request against the URL-based forwarding rule.

- **VServer Group**: Select the vServer group that you want to associate with the forwarding rule.
- **Description**: Enter a description.
- **Add Forwarding Rules**: Specify whether to add another forwarding rule.

6. Click **Add Domain** to add a domain-based forwarding rule, or click **Add Rule** to add a URL-based forwarding rule.

   For more information about the maximum number of forwarding rules that can be configured for an HTTP or HTTPS listener, see Limits.

Modify forwarding rules

You can change the backend servers specified in forwarding rules.

1. Log on to the CLB console.
2. Select the region where the CLB instance is deployed.
3. On the **Instances** page, click the ID of the CLB instance that you want to manage.
4. On the **Listener** tab, find the Layer 7 listener that you want to manage and click **Set Forwarding Rule** in the **Actions** column.
5. In the **Forwarding Rules** section of the **Add Forwarding Rules** panel, click **Edit** in the **Actions** column.
6. You can configure the forwarding rule based on the information in the following table and click **OK**. For example, you can specify a scheduling algorithm and configure session persistence and health checks.

Advanced settings	Description
Scheduling Algorithm	Select a scheduling algorithm. ○ **Weighted Round-Robin**: Backend servers that have higher weights receive more requests than backend servers that have lower weights. ○ **Round Robin**: Requests are distributed to backend servers in sequence.

Advanced settings	Description
Enable Session Persistence	Specify whether to enable session persistence. After session persistence is enabled, CLB forwards all requests from a client to the same backend server. CLB persists HTTP sessions based on cookies. CLB allows you to use the following methods to process cookies: ○ **Insert cookie**: If you select this option, you need only to specify the timeout period of cookies. CLB inserts a cookie (SERVERID) into the first HTTP or HTTPS response that is sent to a client. The next request from the client contains this cookie and the listener forwards this request to the recorded backend server. ○ **Rewrite cookie**: If you select this option, you can specify the cookie to be inserted into an HTTP or HTTPS response. You must specify the timeout period and lifetime of cookies on backend servers. When CLB detects a user-defined cookie, CLB rewrites the original cookie with the user-defined cookie. The next request from the client contains the user-defined cookie, and the listener forwards the request to the recorded backend server. For more information, see Configure session persistence.

Advanced settings	Description
Enable Health Check	◦ **Health Check Method**: In this example, the default value **HEAD** is used. ◦ **Health Check Port**: Set the port that is used by health checks to access backend servers. By default, the backend port configured for the listener is used for health checks. ◦ **Health Check Path**: The URL that is used for health checks. We recommend that you perform health checks on static web pages. ◦ **Health Check Domain Name (Optional)**: The private IP addresses of backend servers are converted into the domain names that are used for health checks. ◦ **Normal Status Code**: Select the HTTP status code that indicates a successful health check. Default values: http_2xx and http_3xx. ◦ **Response Timeout**: Specify the timeout period for a health check response. If a backend ECS instance does not send an expected response within the specified response timeout period, the health check fails. ◦ **Health Check Interval**: Specify the time interval between two consecutive health checks. Default value: 2. Unit: seconds. ◦ **Unhealthy Threshold**: Specify the number of failed health checks that must be consecutively performed before an ECS instance can be declared unhealthy. Valid values: 2 to 10. Default value: 3. ◦ **Healthy Threshold**: Specify the number of successful health checks that must be consecutively performed before an ECS instance can be declared healthy. Valid values: 2 to 10. Default value: 3.

Delete a forwarding rule

To delete a forwarding rule, perform the following operations:

1. Log on to the CLB console.
2. Select the region where the CLB instance is deployed.
3. On the **Instances** page, click the ID of the CLB instance that you want to manage.
4. On the **Listener** tab, find the Layer 7 listener that you want to manage and click **Set Forwarding Rule** in the **Actions** column.
5. In the **Forwarding Rules** section of the **Add Forwarding Rules** panel, click **Delete** in the **Actions** column.

References

- CreateDomainExtension: adds an additional domain name.

2.6. Forward requests from the same domain name but different URLs

Classic Load Balancer (CLB)

supports domain-based and URL-based forwarding rules. You can forward requests from the same domain name but different URLs to separate backend servers. This allows you to optimize load balancing among your server resources.

Context

> **Note** You can configure forwarding rules only for Layer 7 listeners that use HTTPS or HTTP.

This topic uses four Elastic Compute Services (ECS) instances that are deployed with NGINX servers to demonstrate how to configure domain-based and URL-based forwarding rules. The following table lists the forwarding requirements.

Frontend request	Forwarded to
www.example.com/tom	Backend servers SLB_tom1 and SLB_tom2, which belong to the vServer group TOM.
www.example.com/jerry	The backend servers SLB_jerry1 and SLB_jerry2 belong to the vServer group JERRY.

Prerequisites

1. An Internet-facing

   CLB

   instance is created. For more information, see Create a CLB instance.
2. A Layer 7 listener is created. **Round Robin (RR)** is selected as the scheduling algorithm. For more information, see Add an HTTP listener or Add an HTTPS listener.
3. Two vServer groups, TOM and JERRY, are created. For more information, see Create a vServer group.
   - Servers SLB_tom1 and SLB_tom2 are added to the vServer group TOM. The port is set to 80 and the default weight value 100 is used.
   - Servers SLB_jerry1 and SLB_jerry2 are added to the vServer group JERRY. The port is set to 80 and the default weight value 100 is used.

Configure forwarding rules

Perform the following operations:

1. In the top navigation bar, select the region where the

   CLB

   instance is deployed.
2. On the **Instances** page, click the ID of the CLB instance.
3. On the **Listener** tab, find the Layer 7 listener that you want to manage and click **Set Forwarding**

Rule in the **Actions** column.

4. Configure two forwarding rules: forward requests from *www.example.com/tom* to the vServer group TOM and requests from *www.example.com/jerry* to the vServer group JERRY.

   Configure the following parameters:

   - **Domain Name**: Enter the domain name from which requests are forwarded. The domain name can contain only letters, digits, hyphens (-), and periods (.).
   - **URL**: Enter the URL from which requests are forwarded. The URL must start with a forward slash (/). The URL can contain only letters, digits, and the following special characters: `-./%?#&` .

     > **Note** If you only want to configure only a domain-based forwarding rule, leave URL empty.

   - **VServer Group**: Select the vServer group that you want to associate with the forwarding rule.
   - **Description**: Enter a description.

     > **Note** For more information about the maximum number of forwarding rules that can be configured for an HTTP or HTTPS listener, see Limits.

5. Click **Add Forwarding Rule** and click **OK**.
6. Check whether the forwarding rules work as expected.
   - Enter *www.example.com/jerry* in the browser and the following result will be returned: `This is Jerry2!` .
   - Enter *www.example.com/tom* in the browser and the following result will be returned: `This is Tom1.` .
   - Enter *www.example.com* in the browser and the following result will be returned: `Welcome to nginx!` .

Related information

- Introduction

2.7. Preserve client IP addresses when Layer 7 listeners are used

This topic describes how to preserve client IP addresses when Layer 7 listeners of Server Load Balancer (SLB) are used.

Context

When Layer 7 listeners (HTTP and HTTPS listeners) are used, you must configure the corresponding application servers to obtain client IP addresses carried in the `X-Forwarded-For` header. Client IP addresses carried in the X-Forwarded-For HTTP header use the following format:

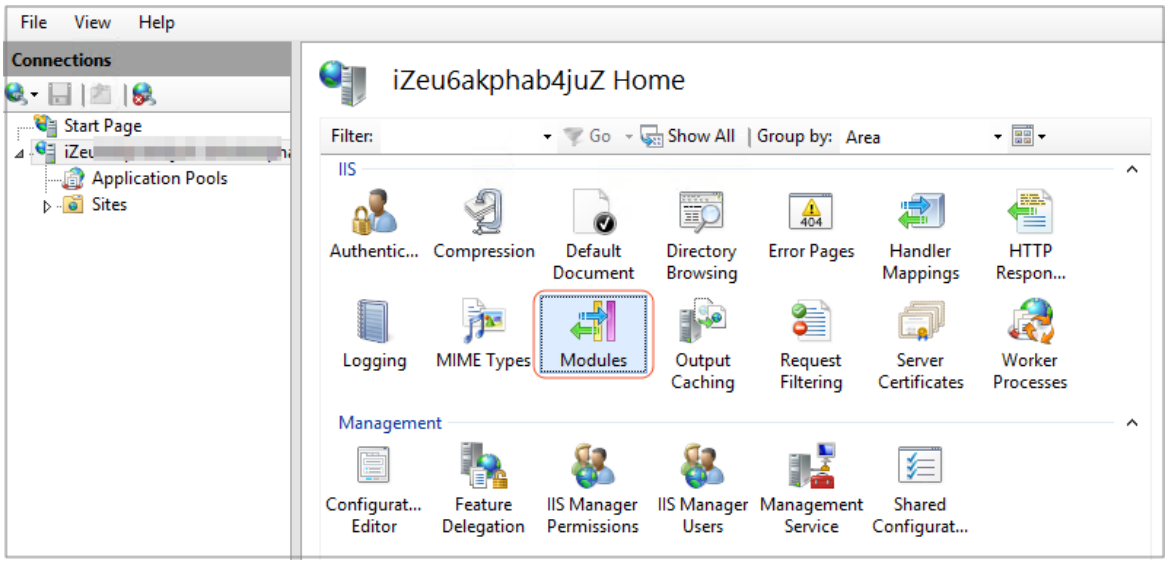
```
X-Forwarded-For: Client IP address, Proxy Server 1 IP address, Proxy Server 2 IP address,...
```

Therefore, the first IP address carried in the X-Forwarded-For header is the client IP address that you want to obtain.

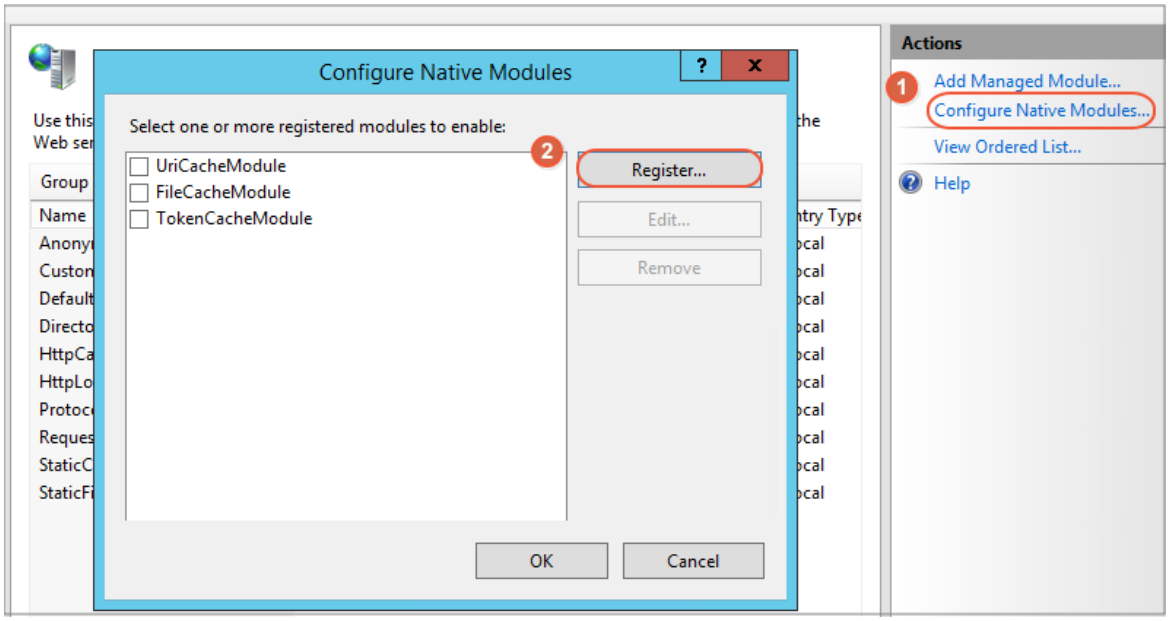
> **Note** HTTPS listeners offload the work of encryption and decryption from backend application servers to SLB instances. The application servers still use the HTTP protocol. Therefore, application servers cannot tell the differences between HTTP and HTTPS listeners.

Configure an IIS7 or IIS8 server

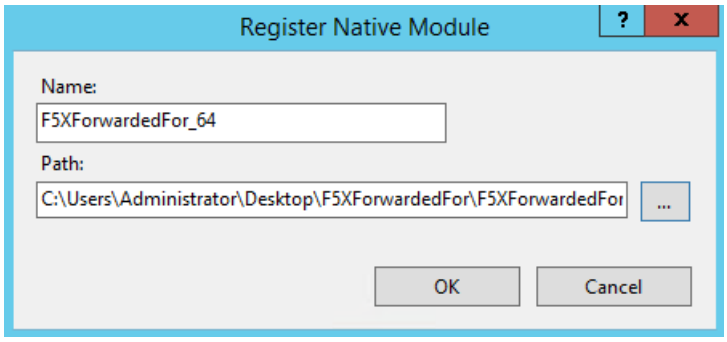
1. Download and decompress the F5XForwardedFor file.
2. Copy the *F5XFFHttpModule.dll* and *F5XFFHttpModule.ini* files from the *x86*\ or *x64*\ directory on your server to another directory, such as *C:\F5XForwardedFor*\. The directory in which the files are located varies based on the operating system version. Make sure that the IIS process has read permissions on the directory.
3. Open **Internet Information Services (IIS) Manager** and double-click **Modules**.



4. Click **Configure Native Modules**, and then click **Register** in the dialog box that appears.

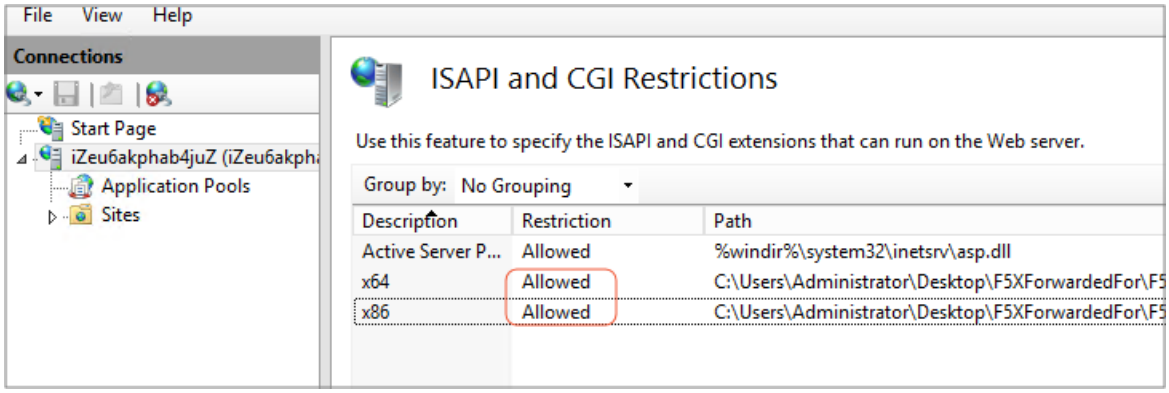


5. Add the downloaded *.dll* file.



6. Add the ISAPI and CGI restrictions for the added files and set the restrictions to Allowed.

> **Note** Make sure that the ISAPI and CGI applications are installed.



7. Restart the IIS server and wait until the configurations take effect.

Configure an Apache server

In this example, the configuration files are stored in */alidata/*. You can change the directory path based on the actual value in the following commands.

1. Run the following command to install the mod_rpaf module:

```
wget https://github.com/gnif/mod_rpaf/archive/v0.6.0.tar.gz
tar zxvf v0.6.0.tar.gz
sudo apt-get install apache2-dev
whereis apxs2
cd mod_rpaf-0.6.0/alidata/server/httpd/bin/apxs -i -c -n mod_rpaf-2.0.so mod_rpaf-2.0.
c
```

2. Append the following content to the end of the Apache configuration file */alidata/server/httpd/conf/httpd.conf*:

```
LoadModule rpaf_module modules/mod_rpaf-2.0.so
RPAFenable On
RPAFsethostname On
RPAFproxy_ips  <IP_address>
RPAFheader X-Forwarded-For
```

> **Note** To obtain the proxy server IP address, add the CIDR block of the proxy server to `RPAFproxy_ips <IP_address>`, such as 100.64.0.0/10 (100.64.0.0/10 is reserved by Alibaba Cloud. It is not used by any user and therefore causes no security risks) of SLB and the CIDR blocks of Anti-DDoS. Separate multiple CIDR blocks with commas (,).

3. Restart Apache.

```
/alidata/server/httpd/bin/apachectl restart
```

Configure a NGINX server

In this example, the configuration files are stored in *alidata/*. You can change the directory path based on the actual value in the following commands.

1. Run the following command to install http_realip_module:

```
 wget http://nginx.org/download/nginx-1.0.12.tar.gz
 tar zxvf nginx-1.0.12.tar.gz
 cd nginx-1.0.12
 ./configure --user=www --group=www --prefix=/alidata/server/nginx --with-http_stub_sta
tus_module --without-http-cache --with-http_ssl_module --with-http_realip_module
 make
 make install
 kill -USR2 `cat /alidata/server/nginx/logs/nginx.pid`
 kill -QUIT `cat /alidata/server/nginx/logs/ nginx.pid.oldbin`
```

2. Run the following command to open the *nginx.conf* file:

```
vi /alidata/server/nginx/conf/nginx.conf
```

3. Append new fields to the end of the following content:

```
 fastcgi connect_timeout 300;
 fastcgi send_timeout 300;
 fastcgi read_timeout 300;
 fastcgi buffer_size 64k;
 fastcgi buffers 4 64k;
 fastcgi busy_buffers_size 128k;
 fastcgi temp_file_write_size 128k;
```

The fields that need to be appended:

```
 set_real_ip_from IP_address;
 real_ip_header X-Forwarded-For;
```

> **Note** To obtain the proxy server IP address, add the CIDR block of the proxy server to `set_real_ip_from <IP_address>`, such as 100.64.0.0/10 (100.64.0.0/10 is reserved by Alibaba Cloud. It is not used by any user and therefore causes no security risks) of SLB and the CIDR blocks of Anti-DDoS. Separate multiple CIDR blocks with commas (,).

4. Run the following command to restart NGINX:

```
/alidata/server/nginx/sbin/nginx -s reload
```

2.8. Use SLB in Auto Scaling

You can associate Server Load Balancer (SLB) instances with a scaling group. The SLB instances distribute traffic to multiple ECS instances in the scaling group. This improves the performance of the scaling group.

Prerequisites

- You have at least one SLB instance in the **Active** state. For more information, see Create a CLB instance.
- The SLB instance and the scaling group are in the same region.
- The SLB instance and the scaling group are in the same VPC if their network type is VPC.
- If the network type of the SLB instance is classic network, the network type of the scaling group is VPC, and the backend server group of the SLB instance contains VPC-type ECS instances, the ECS instances and the scaling group must be in the same VPC.
- At least one listener is configured on the SLB instance. For more information, see Listener overview.
- Health check is enabled on the SLB instance. For more information, see Configure health checks.

Context

SLB allows multiple ECS instances in a region to share the service load by using the IP address of an SLB instance. These ECS instances act as a high-performance and high-availability application service pool. SLB distributes and controls traffic by using SLB instances, listeners, and backend servers. For more information, see What is CLB?

After an SLB instance is associated with a scaling group, all ECS instances including those automatically and manually created in the scaling group are added to the backend server group of the SLB instance. The SLB instance distributes traffic to the ECS instances based on traffic distribution policies and health check policies. This improves resource availability.

> **Note** Each ECS instance in the backend server group of an SLB instance has a default load balancing weight of 50. You can adjust the weight of an ECS instance. For more information, see Change the weight of a backend server.

Procedure

The following section describes the procedure to associate an SLB instance with a scaling group in the Auto Scaling console. For more information about other configurations of a scaling group, see Create a scaling group.

1. Log on to the Auto Scaling console.
2. In the left-side navigation pane, click **Scaling Groups**.
3. In the top navigation bar, select a region.
4. Go to the page for associating SLB instances with a scaling group.
   - To create a scaling group to associate with SLB instances, click **Create**.
   - To modify a scaling group that is not associated with SLB instances, find the scaling group and click **Edit** in the **Actions** column.
5. Optional. Configure the **Network Type** parameter.

The network type of a scaling group cannot be changed after the scaling group is created.

6. Configure the **Associate SLB Instance** parameter.
   i. Select the SLB instances to be associated with the scaling group.

      You can only associate a limited number of SLB instances with a scaling group. For more information, see 使用限制. If your SLB instance does not appear in the drop-down list, check whether your SLB instance meets the prerequisites.

   ii. Select backend server groups for the SLB instance.

      You can select the default server group and vServer groups. For more information, see Backend server overview.

      - The default server group contains the ECS instances that receive requests forwarded by a listener. If no vServer group or primary and secondary server group is specified for the listener, the listener forwards all requests to the ECS instances in the default server group.
      - You can select vServer groups if you want to forward requests to different backend servers, or forward requests based on domain names or URLs.
7. Configure the remaining settings.

Related information

References

- CreateScalingGroup
- AttachLoadBalancers
- DetachLoadBalancers
- AttachVServerGroups
- DetachVServerGroups

2.9. Enable Proxy Protocol for a Layer 4 listener to retrieve client IP addresses

This topic describes how to configure a Layer 4

Classic Load Balancer (CLB)

listener to retrieve client IP addresses.

Context

In most cases, backend servers of Layer 4 CLB can retrieve client IP addresses. However, if the client IP address of a request is translated into another IP address before CLB forwards the request to a backend server, the backend server cannot retrieve the client IP address. In this case, you can enable Proxy Protocol to pass the client IP address to the backend server. After you enable Proxy Protocol for a Layer 4

CLB

listener, CLB adds a TCP header to the request without modifying the existing headers. The TCP header carries information such as the source IP address, destination IP address, source port, and destination port.

CLB

supports only Proxy Protocol v2. For more information, see The PROXY protocol.

Scenarios

You can enable Proxy Protocol for listeners of an IPv6

CLB

instance to which IPv4 backend servers are added.

Prerequisites

- Before you enable Proxy Protocol, make sure that your backend servers support Proxy Protocol v2.

  NGINX Plus R16 and later versions and open source NGINX 1.13.11 and later versions support Proxy Protocol v2.
- If a server group is associated with multiple

  CLB

  listeners, you must enable Proxy Protocol for all listeners.

Step 1: Create a TCP or UDP listener

1. Log on to the CLB console.
2. In the top navigation bar, select the region where the

   CLB

   instance is deployed.
3. On the **Instances** page, find the CLB instance and click **Configure Listener** in the **Actions** column.
4. Configure the listener as prompted.
   - **Select Listener Protocol**: Select **TCP** or **UDP**.
   - **Proxy Protocol**: Click **Modify** next to **Advanced** and select **Use the proxy protocol to pass client IP addresses to backend servers**.

   For more information, see Add a TCP listener and Add a UDP listener.

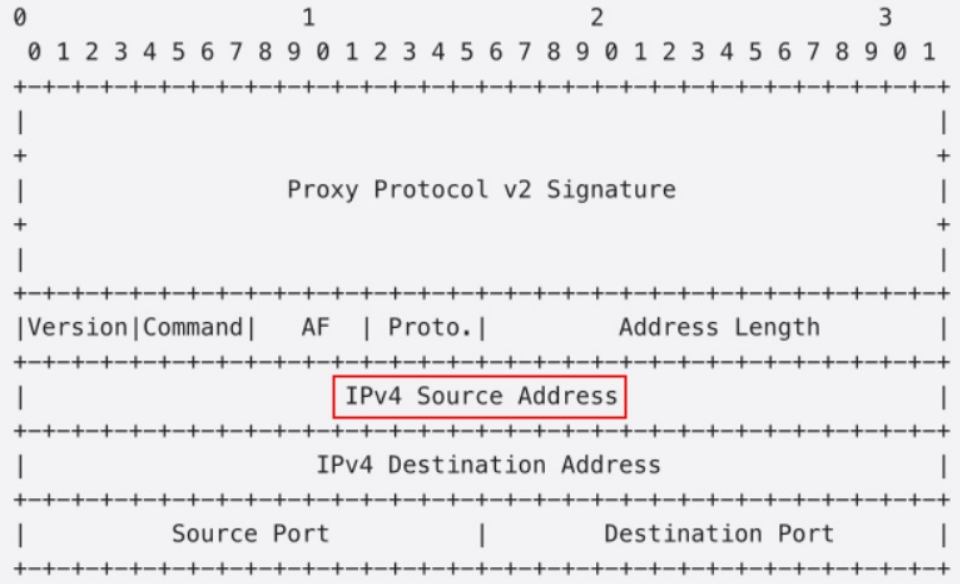
Step 2: Configure NGINX

Run the following command to enable Proxy Protocol to retrieve client IP addresses:

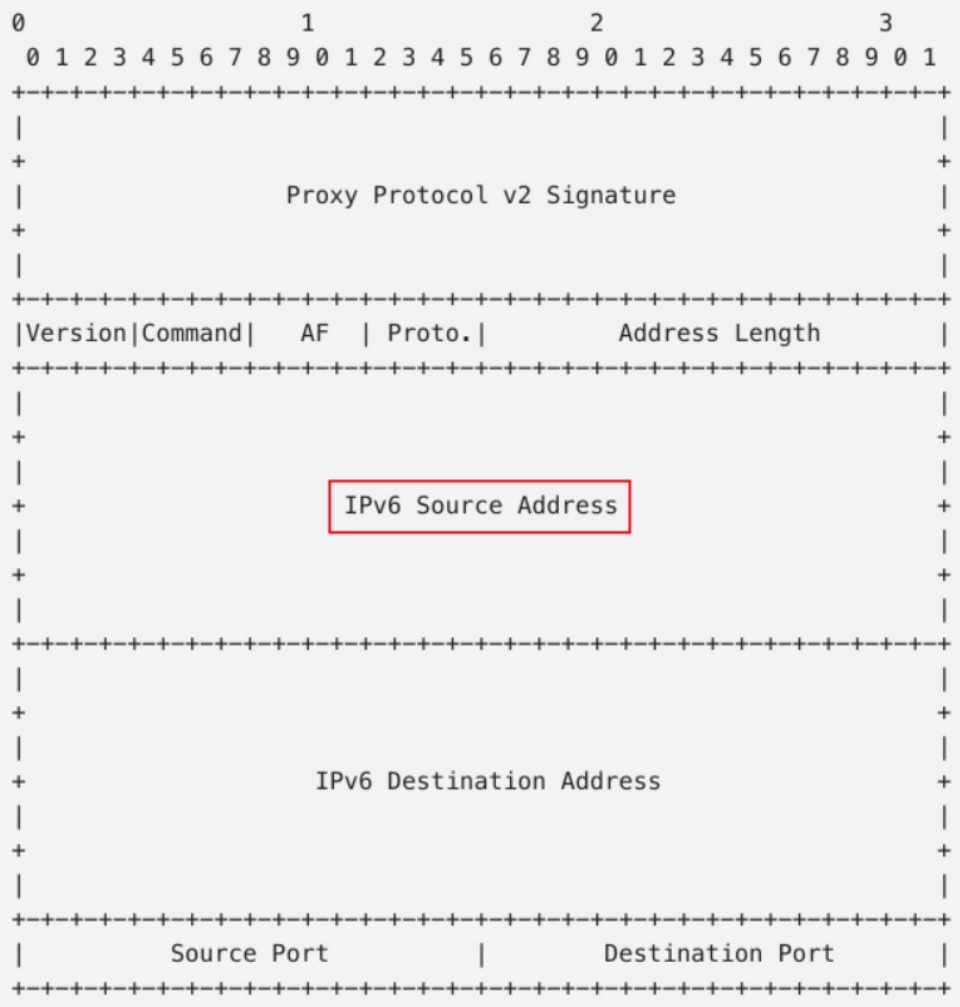
```
http {
    #...
    server {
        listen 80   proxy_protocol;
        listen 443  ssl proxy_protocol;
        #...
    }
}
stream {
    #...
    server {
        listen 12345 proxy_protocol;
        #...
    }
}
```

Step 3: Retrieve client IP addresses

- The following example shows how an IPv4 client IP address is preserved in the Proxy Protocol v2 header in the binary format.



- The following example shows how an IPv6 client IP address is preserved in the Proxy Protocol v2 header in the binary format.



2.10. Perform a stress test

Layer-4 Server Load Balancer (SLB) uses Linux Virtual Server (LVS) and Keepalived to provide the load balancing service, whereas Layer-7 SLB uses Tengine.

Overview

In a Layer-4 listener, requests are directly sent to backend servers after being passed through LVS. However, in a Layer-7 listener, requests are sent to Tengine before they are sent to backend servers. Due to this additional step, the performance of a Layer-7 listener is inadequate when compared with a Layer-4 listener.

A Layer-7 listener may show poor performance during a stress test. A Layer-7 listener connecting to two ECS instances is surprisingly inferior to a Layer-4 listener connecting to one ECS instance. Except the preceding reason regarding the process, the following situations may also cause poor performance of a Layer-7 listener during a stress test:

- Insufficient client ports

  During a stress test, insufficient client ports cause connection failures. The timestamp attribute of TCP connections is erased by SLB by default. As a result, tw_reuse of Linux protocol stack (reuse of ports in time_wait state) does not work, and connections in the time_wait state accumulate.

Solution: We recommend that you enable persistent connections on clients and use RST (set the SO_LINGER attribute for sockets) to close connections, instead of using FIN packets.

- Full accept queues on the backend server

  If accept queues are full on the backend server, the backend server does not respond with syn_ack packets and the client times out.

  Solution: Run the `sysctl -w net.core.somaxconn=1024` command to change the value of net.core.somaxconn and restart the application on the backend server. The default value of net.core.somaxconn is 128.

- Excessive connections to the backend server

  When you use a Layer-7 SLB instance, persistent connections are changed to short-lived connections after the connections pass through Tengine due to the design of the network architecture. As a result, too many connections are sent to the backend server, which leads to poor performance of a Layer-7 SLB instance during a stress test.

- Limits from dependencies of the backend server

  If the backend server is normal after requests are sent to the server and the system still exhibits poor performance, it may be caused by the dependencies of the backend server (such as inadequate system support from the databases).

- Unhealthy status of the backend server

  If a backend server is declared as unhealthy after health check results are received, or the health status of the server is unstable, an SLB instance may has poor performance during a stress test.

Usage notes

Before you perform a stress test, take note of the following points:

- Use short-lived connections to test the forwarding performance of an SLB instance.

  A stress test is intended to measure the forwarding capability of SLB, in addition to session persistence and balancing capability. Therefore, we recommend that you use short-lived connections to test SLB and backend server processing capabilities. You must check the problem of insufficient client ports.

- Enable persistent connections to test the throughput of an SLB instance.

  Set a small timeout value (such as 5 seconds) for the test tool. If the timeout time is too long, the test result may show an increased average response time. This makes it difficult to determine whether the proper stress test level has been reached. If the timeout time is reduced, the test result indicates the success rate, which makes it easy to determine the stress test level.

- Build a static web page on the backend server to avoid loss caused by application logic.
- We also recommend the following listener configurations:
  - Disable session persistence to avoid the case where most loads are sent to a backend server.
  - Disable health check to reduce access requests sent to the backend server.
  - Use five or more clients to test the performance of 5,000 concurrent connections.

Recommended test tools

We recommend that you do not use Apache Bench (ab).

Apache Bench applies 3s, 6s, and 9s interruptions in a progressive manner when a large number concurrent connections exist. Apache Bench uses the content length to determine whether requests are successful. SLB returns inconsistent content lengths when multiple backend servers are tested. This will result in inaccurate stress test results.

PTS allows a great number of concurrent connections. PTS allocates public IP addresses from all over the country, so that the pressure sources are scattered. PTS also integrates Cloud Monitor to view all end-to-end performance data in real time.

An stress test example

In this example, an SLB instance is created and two ECS instances are added as backend servers. A TCP listener and an HTTP listener are created. The backend port is 80. The ECS instances use 1 vCPU, 512 MiB memory, and CentOS 6.3 (64-bit) operating system. Perform the following operations:

1. Install Apache Web Server to provide web services.

   ```
   yum install -y httpd
   ```

2. Initialize the default homepage index.html.

   ```
   echo "testvm" > /var/www/html/index.html
   ```

3. Start the HTTP service.

   ```
   service httpd start
   ```

4. Access the local port 80 to confirm that the web services are available.

   ```
   curl localhost
   ```

5. Create a test script in PTS and start the stress test.

2.11. Use access logs to find unhealthy backend servers

When the response is delayed, you can view the response time of the SLB instance in the dashboard provided by Log Service. Then you can find unhealthy backend servers.

This topic describes how to use access logs to rapidly find unhealthy backend servers. For more information, see Configure access logs.

Step 1: Configure SLB access logs

Before you configure LB access logs, make sure that:

1. A Layer-7 listener is added.
2. Log Service is activated.

Perform the following operations:

1. Log on to the Server Load Balancer console.
2. In the left-side navigation pane, choose **Logs > Access Logs**.
3. Select the region for the SLB instance.
4. Click **Authorize**, and then click **Confirm Authorization Policy** to allow SLB to write logs to Log

Service.

If you log on to the console as a RAM user, you must first use your Alibaba Cloud account to authorize the RAM user. For more information, see Authorize a RAM user to use the access logs feature.

> **Note** If you have authorized SLB, skip this step.

5. On the **Access Logs** page, find the SLB instance and click **Configure Logging** in the Actions column.
6. Configure Project and Logstore and then click **OK**.

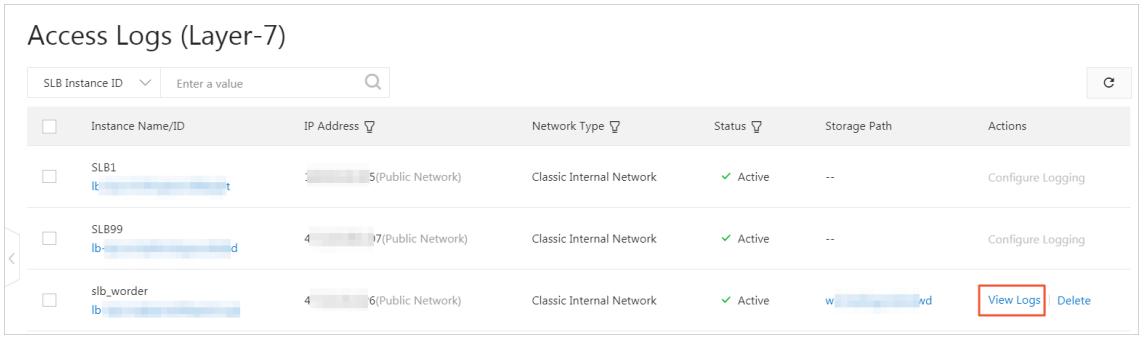
> **Note** Make sure that the name of the Project is globally unique and the region of the Project is the same as that of the SLB instance.

Step 2: View access logs

Perform the following operations:

1. Go to the log search page. You can go to the search page from the SLB or Log Service console.
   - From the SLB console

     On the **Access Logs** page, click **View Logs**.

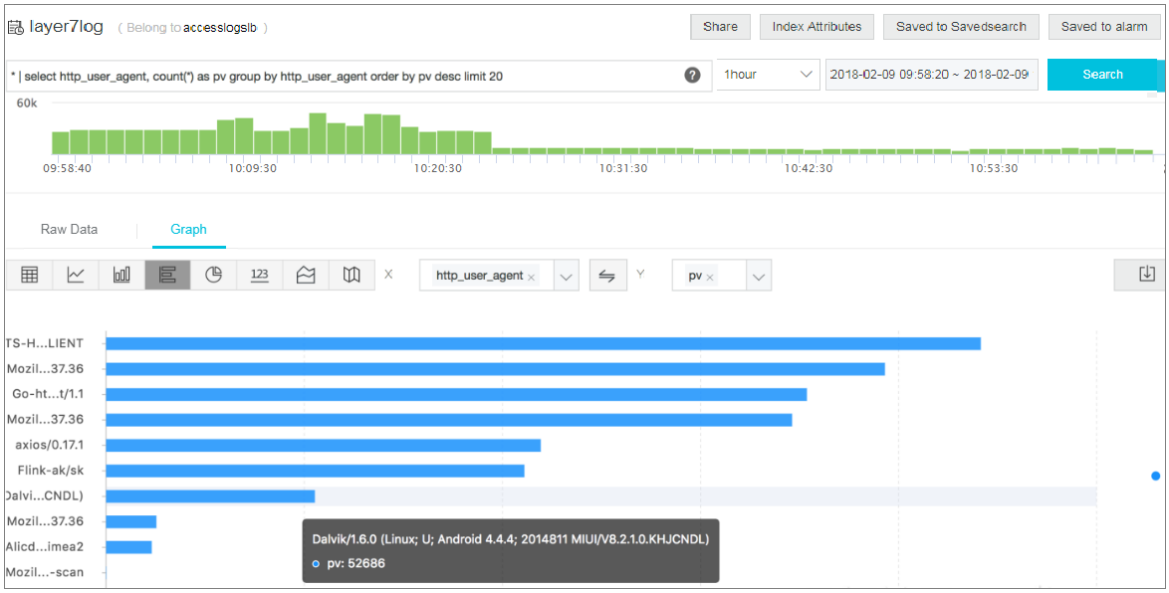
     

   - From the Log Service console

     On the **Logstores** page, click **Search** of the Logstore.
2. Click the log field to view detailed information.
3. Enter an SQL statement to query specific access logs.

   For example, you can enter the following SQL statement to query the Top20 clients, which is used to analyze the request source and assist business decision-making.

```
* | select ip_to_province(client_ip) as client_ip_province, count(*) as pv group by
    client_ip_province order by pv desc limit 50
```



Step 3: Find unhealthy backend servers

You can find unhealthy backend servers by checking the dashboard of Log Service.

1. Log on to the Log Service console and click the project link of the SLB instance.
2. In the left-side navigation pane, click

   .
3. Click the link of the SLB access logs.
4. In the dashboard, view the value in the **top upstream response time** tab. You can choose to display the **Average upstream response time (s)** in descending order to check if the response time of a backend server is more than one second.

   If the response time is within one second, run the **ssh** command to log on to the backend server. Check if the CPU utilization is high and handle the issue if yes.

2.12. Specify an IP address for an SLB instance by using OpenAPI Explorer

This topic describes how to specify an internal IP address when you create a Server Load Balancer (SLB) instance by using OpenAPI Explorer. You can specify the IP address used by the CIDR block of the vSwitch to which the SLB instance to be created belongs as the internal IP address of the SLB instance.

Procedure

1. Log on to the OpenAPI Explorer console.
2. Search for the CreateLoadBalancer API operation.
3. Configure the parameters.

   Only some parameters are listed here. For more information, see Create a CLB instance.

   - RegionId: the region ID of the SLB instance. For this example, it is *cn-hangzhou*.

- VpcId: the ID of the VPC to which the SLB instance belongs.

  You can log on to the Virtual Private Cloud console and select China (Hangzhou) to view the VPC ID.

- VSwitchId: the ID of the vSwitch to which the SLB instance belongs. This parameter is required when you specify an SLB IP address.

  You can log on to the Virtual Private Cloud console and click the ID of the VPC to which the SLB instance belongs. Click the number of vSwitches on the **Resources** tab to view the ID of the vSwitch.

  Click the ID of the vSwitch to view the CIDR block of the vSwitch. Example: 192.168.0.0/24.

- Address: the internal IP address of the SLB instance. This IP address must belong to the CIDR block of the vSwitch. For this example, it is 192.168.0.3.

4. Click **Submit Request**.

   The following response parameters are returned:

   - XML format

```
<? xml version="1.0" encoding="UTF-8" ? >
    <NetworkType>vpc</NetworkType>
    <LoadBalancerName>auto_named_slb</LoadBalancerName>
    <Address>192.168.0.3</Address>
    <ResourceGroupId>rg-acfmxazb4ph*****</ResourceGroupId>
    <RequestId>09197EEB-7013-4F56-A5CE-A756FFE5B75D</RequestId>
    <AddressIPVersion>ipv4</AddressIPVersion>
    <LoadBalancerId>lb-bp1h66tp5uat84kh******</LoadBalancerId>
    <VSwitchId>vsw-bp14cagpfysr29fe*****</VSwitchId>
    <VpcId>vpc-bp18sth14qii3pn*****</VpcId>
```

   - JSON format

```
{
    "NetworkType": "vpc",
    "LoadBalancerName": "auto_named_slb",
    "Address": "192.168.0.3",
    "ResourceGroupId": "rg-acfmxazb4******",
    "RequestId": "09197EEB-7013-4F56-A5CE-A756FFE5B75D",
    "AddressIPVersion": "ipv4",
    "LoadBalancerId": "lb-bp1h66tp5uat84******",
    "VSwitchId": "vsw-bp14cagpfysr29*******",
    "VpcId": "vpc-bp18sth14qii3*********"
}
```

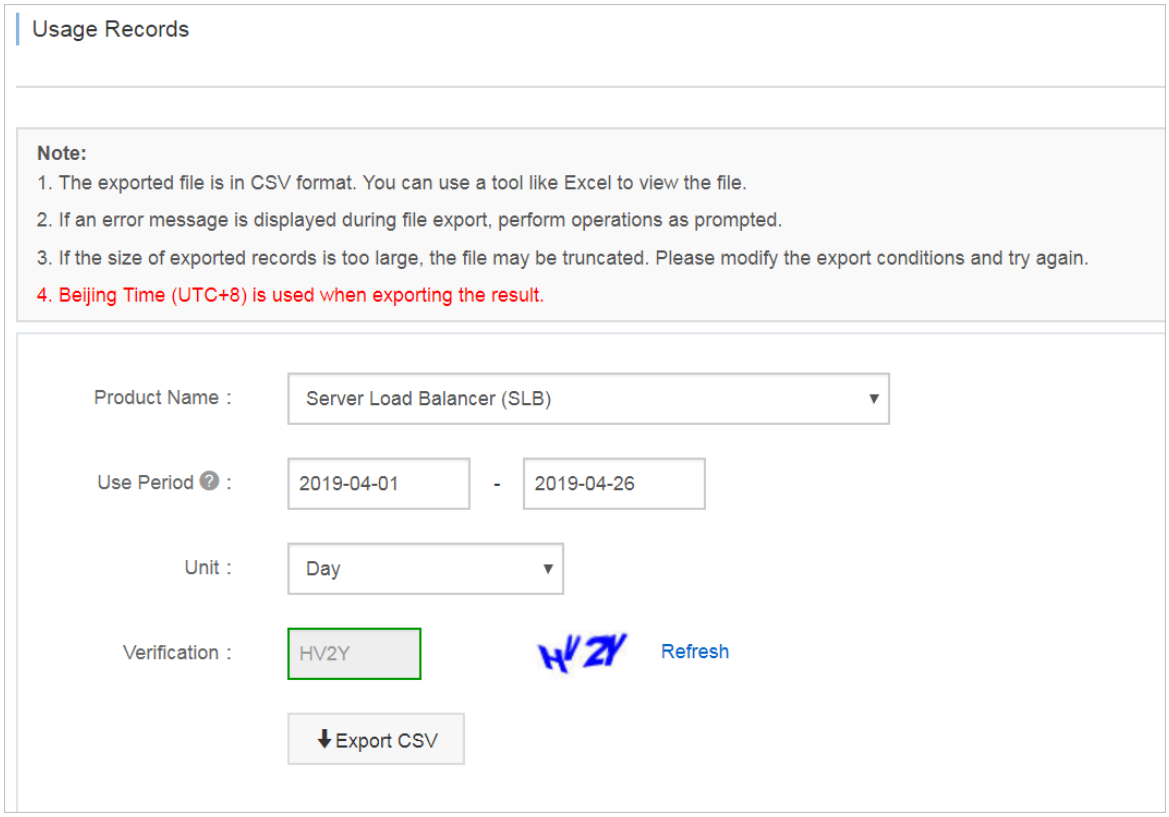
5. Log on to the Server Load Balancer console and select the China (Hangzhou) region. Check whether the SLB instance to which the 192.168.0.3 IP address is assigned is created.

2.13. View traffic usage

This topic describes how to view the traffic usage of a Server Load Balancer (SLB) instance. You can view the traffic usage of an SLB instance in a certain period through the SLB console.

Procedure

1. Log on to the CLB console.
2. In the upper-right corner of the top navigation bar, choose **Billing Management > Billing Management**.
3. In the left-side navigation pane, select **Usage Records**.
4. On the **Usage Records** page, select **Server Load Balancer (SLB)** from the Product Name drop-down list, set a period and measurement cycle, and enter the verification code.



5. Click **Export CSV** to generate a traffic usage table in the *.CSV* format.

   The table includes the following information. You can view the traffic usage by instance, region, and endpoint.

	A	B	C	D	E	F	G	H	I
1	Instance ID	Region	Service Address	Service Address	Bandwidth	Upstream traffic (Byte)	Downstream traffic (Byte)	Start Time	End Time
2	lb-	cn-hangzhou-dg-a01	1 66	internet	0	0	0	2019/4/1 0:00	2019/4/2 0:00
3	lb-	cn-hangzhou-dg-a01	1 66	internet	0	0	0	2019/4/2 0:00	2019/4/3 0:00
4	lb-	cn-hangzhou-dg-a01	1	internet	0	0	0	2019/4/1 0:00	2019/4/2 0:00
5	lb-	cn-hangzhou-dg-a01	1	internet	0	0	0	2019/4/2 0:00	2019/4/3 0:00
6	lb-	cn-hangzhou-dg-a01	1 42	internet	0	0	0	2019/4/1 0:00	2019/4/2 0:00
7	lb-	cn-hangzhou-dg-a01	1 42	internet	0	0	0	2019/4/2 0:00	2019/4/3 0:00
8	lb-	cn-hangzhou-dg-a01	1 4	internet	0	0	0	2019/4/1 0:00	2019/4/2 0:00
9	lb-	cn-hangzhou-dg-a01	1 4	internet	0	0	0	2019/4/2 0:00	2019/4/3 0:00
10	lb-	cn-hangzhou-dg-a01	1 35	internet	0	0	0	2019/4/1 0:00	2019/4/2 0:00
11	lb-	cn-hangzhou-dg-a01	1 35	internet	0	0	0	2019/4/2 0:00	2019/4/3 0:00
12	lb-	cn-hangzhou-dg-a01	1	internet	0	262302	169494	2019/4/1 0:00	2019/4/2 0:00
13	lb-	cn-hangzhou-dg-a01	1	internet	0	231549	144793	2019/4/2 0:00	2019/4/3 0:00
14	lb-	cn-hangzhou-dg-a01	4	internet	0	0	0	2019/4/1 0:00	2019/4/2 0:00
15	lb-	cn-hangzhou-dg-a01	4	internet	0	0	0	2019/4/2 0:00	2019/4/3 0:00