Alibaba Cloud Server Load Balancer

チュートリアル

Document Version20190524

目次

1 HTTPS リスナーの追加 (一方向認証)

一方向認証で HTTPS リスナーを追加するには、リスナーを設定するときに SLB にサーバー証明書をアップロードするだけです。

ステップ 1 サーバー証明書のアップロード

HTTPS リスナーを設定する (一方向認証) 前に、サーバー証明書を購入し、そのサーバー証明書を SLB の証明書管理システムにアップロードする必要があります。 証明書を SLB にアップロードすれば、バックエンドサーバーに証明書を設定する必要はなくなります。

1. *SLB コンソール*にログインします。
2. 左側のナビゲーションウィンドウで、[証明書] をクリックし、[証明書の作成] をクリックします。
3. サーバー証明書を次のように設定します。
   - リージョン: [中国 （杭州）] を選択します。

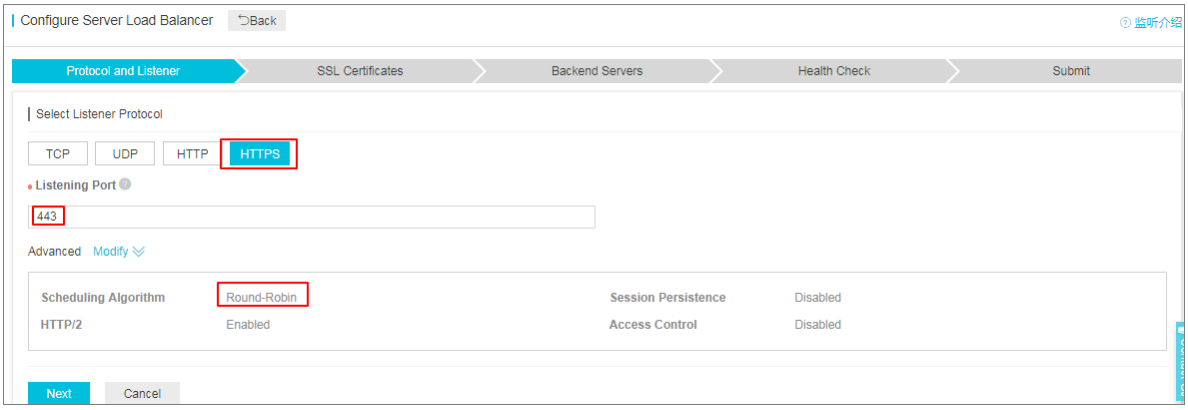
     
     注 :
     証明書を使用するには、証明書のリージョンが SLB インスタンスのリージョンと同じである必要があります。
   - 証明書タイプ: [サーバー証明書] を選択します。
   - 証明書の内容と秘密鍵: サーバー証明書の内容と秘密鍵をコピーします。 有効な証明書の形式を表示するには、[サンプルのインポート] をクリックします。 アップロードする証明書は PEM 形式でなければなりません。 詳しくは、「証明書の形式」をご参照ください。
4. [OK] をクリックします。

ステップ 2 SLB インスタンスの設定

1. *SLBコンソール*にログインします。
2. [Server Load Balancer] ページで、 [SLB インスタンスの 作成] をクリックします。
3. インスタンスを設定し、[今すぐ購入] をクリックします。

   
   注 :
   このチュートリアルでは、インスタンスタイプは [インターネット]、リージョンは [中国 (杭州)] です。詳しくは、「*SLB インスタンスの作成*」をご参照ください。
4. [Server Load Balancer] ページに戻り、[中国 (杭州)] リージョンをクリックします。

5. 作成した SLB インスタンスの ID をクリックするか、[リスナーの設定] をクリックします。
6. [リスナー] タブをクリックし、 [リスナーの追加] をクリックします。
7. [プロトコルとリスナー] タブで、リスナーを設定します。
   - リスナープロトコルの選択: HTTPS
   - リスニングポート: 443
   - スケジューリングアルゴリズム: ラウンドロビン (RR)



8. [次へ] をクリックします。 [SSL 証明書] タブで、アップロードしたサーバー証明書を選択します。
9. [次へ] をクリックします。 表示されたページで、[デフォルトサーバーグループ] をクリックし、[追加] をクリックします。 ECS インスタンスを追加し、バックエンドポートを 80 に設定します。
10. 左側のナビゲーションウィンドウで、[サーバー] > [バックエンドサーバー] をクリックし、[バックエンドサーバーの追加] をクリックして ECS インスタンスを追加します。

ステップ 3 SLBサービスのテスト

1. [Server Load Balancer] ページに戻り、ヘルスチェックステータスを表示します。

   ステータスが [正常] の場合、バックエンドサーバーは SLB リスナーによって転送されたリクエストを受信できます。
2. Web ブラウザーに Server Load Balancer インスタンスのパブリック IP を入力します。

2 HTTPS リスナーの追加 (相互認証)

相互認証を使用して HTTPS リスナーを追加するには、リスナーを設定するときにサーバー証明書と CA 証明書を SLB にアップロードする必要があります。

このチュートリアルでは、自己署名 CA 証明書を使用してクライアント証明書に署名します。 相互認証を使用して HTTPS リスナーを追加するには、以下のステップを実行します。

1. サーバー証明書の準備
2. *Open SSL* を使用して*CA* 証明書を生成
3. クライアント証明書の生成
4. サーバー証明書と*CA* 証明書のアップロード
5. クライアント証明書のインストール
6. *HTTPS* リスナーの設定 *(*相互認証*)*
7. *SLB* サービスのテスト

ステップ 1 サーバー証明書の準備

サーバー証明書は、サーバーから送信された証明書が信頼できるセンターによって署名、発行されているかどうかをクライアントブラウザーで確認するために使用されます。 サーバー証明書は Alibaba Cloud Security の *Certificate Service*、または他のサービスプロバイダーから購入できます。

ステップ 2 Open SSL を使用してCA 証明書を生成

1. 次のコマンドを実行して、*/ root* ディレクトリの下に *ca* フォルダーを作成し、 *ca* フォルダーの下に 4 つのサブフォルダーを作成します。

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

説明：

- *newcerts* フォルダーには、CA 証明書によって署名されたデジタル証明書が保存されます。
- *private* フォルダーには、CA 証明書の秘密鍵が保存されます。
- *conf* には、パラメーターを単純化するために使用される設定ファイルが保存されます。
- *server* フォルダーには、サーバー証明書が保存されます。

2. 次の情報を含む *openssl . conf* ファイルを *conf* ディレクトリに作成します。

```
[ ca ]
 default_ca = foo
[ foo ]
 dir = / root / ca
 database = / root / ca / index . txt
 new_certs_ dir = / root / ca / newcerts
 certificat e = / root / ca / private / ca . crt
 serial = / root / ca / serial
 private_ke y = / root / ca / private / ca . key
 RANDFILE = / root / ca / private /. rand
 default_da ys = 365
 default_cr l_days = 30
 default_md = md5
 Unique_sub ject = no
 Policy = policy_any
[ policy_any ]
 countryNam e = match
 stateOrPro vinceName = match
 organizati onName = match
 organizati onalUnitNa me = match
 localityNa me = optional
 commonName = supplied
 Emailaddre ss = optional
```

3. 次のコマンドを実行して、秘密鍵を生成します。

```
$ CD / root / CA
 $ sudo openssl genrsa - out private / ca . key
```

秘密鍵の生成例を以下に示します。

```
root@iZbp1hfvivcqx1jbwap31iZ:~/ca/conf# cd /root/ca
root@iZbp1hfvivcqx1jbwap31iZ:~/ca# sudo openssl genrsa -out private/ca.key
Generating RSA private key, 2048 bit long modulus
.....+++
..+++
e is 65537 (0x10001)
```

4. 次のコマンドを実行し、プロンプトに従って必要な情報を入力します。 Enter キーを押して、証明書の生成に使用される *csr* ファイルを生成します。

```
$ Sudo OpenSSL req - New - key private / CA . Key - out
 private / CA . CSR
```

注：

コモンネーム は、SLB インスタンスのドメイン名です。

```
root@iZbp1hfvivcqx1jbwap31iZ:~/ca# sudo openssl req -new -key private/ca.key -ou
t private/ca.csr
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:CN
State or Province Name (full name) [Some-State]:ZheJiang
Locality Name (eg, city) []:HangZhou
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Alibaba
Organizational Unit Name (eg, section) []:Test
Common Name (e.g. server FQDN or YOUR name) []:mydomain
Email Address []:a@alibaba.com

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
root@iZbp1hfvivcqx1jbwap31iZ:~/ca#
```

5. 次のコマンドを実行して、 *crt* ファイルを生成します。

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

6. 次のコマンドを実行して、秘密鍵の開始シーケンス番号を設定します。シーケンス番号には、任意の 4 文字を使用できます。

```
$ sudo echo FACE > serial
```

7. 次のコマンドを実行して、CA 鍵ライブラリを作成します。

```
$ sudo touch index.txt
```

8. 次のコマンドを実行して、クライアント証明書を削除するための証明書失効リストを作成します。

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

出力:

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

ステップ 3 クライアント証明書の生成

1. 次のコマンドを実行して、 *ca* ディレクトリの下に、クライアントキーを保存する *users* ディレクトリを生成します。

```
$ Sudo mkdir users
```

2. 次のコマンドを実行して、クライアント証明書の鍵を作成します。

```
$ Sudo OpenSSL FIG /root/CA/users/client. Key 1024
```

注:

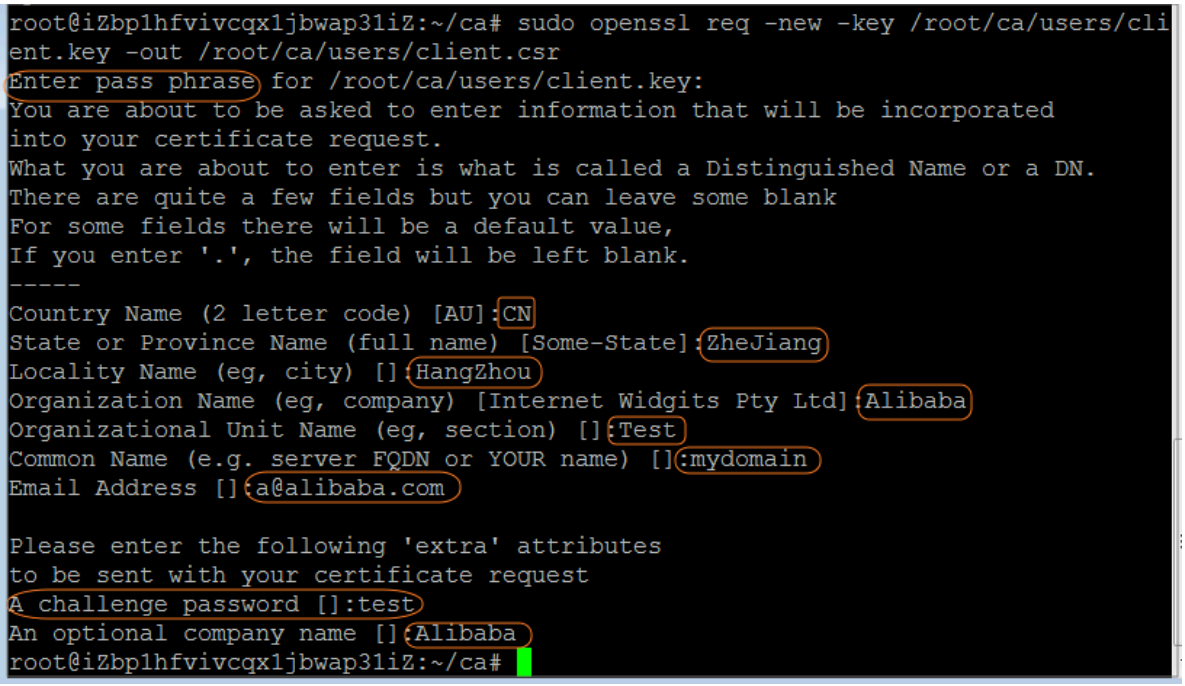
鍵を作成する際、パスフレーズを入力します。 パスフレーズは、不正なアクセスから秘密鍵を保護するためのパスワードです。 同じパスワードを 2 回入力してください。

3. 次のコマンドを実行して、証明書の署名をリクエストするための *csr* ファイルを作成します。

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

指示に従って、前のステップで使用したパスフレーズを入力し、必要な情報を入力します。

> **注:**
> チャレンジパスワードは、クライアント証明書のパスワードです (`client.key` のパスワードとは別にしてください。 このチュートリアルでは、パスワードは test です)。 ルート証明書またはサーバー証明書のパスワードと同じにすることができます。



4. 次のコマンドを実行して、ステップ 2 の CA 鍵を使用してクライアント鍵に署名します。

```
$ 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 "
```

操作確認のプロンプトが表示されたら、 y を 2 回入力します。

```
root@iZbp1hfvivcqx1jbwap31iZ:~/ca# sudo openssl ca -in /root/ca/users/client.csr
 -cert /root/ca/private/ca.crt -keyfile /root/ca/private/ca.key -out /root/ca/us
ers/client.crt -config "/root/ca/conf/openssl.conf"
Using configuration from /root/ca/conf/openssl.conf
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
countryName           :PRINTABLE:'CN'
stateOrProvinceName   :ASN.1 12:'ZheJiang'
localityName          :ASN.1 12:'HangZhou'
organizationName      :ASN.1 12:'Alibaba'
organizationalUnitName:ASN.1 12:'Test'
commonName            :ASN.1 12:'mydomain'
emailAddress          :IA5STRING:'a@alibaba.com'
Certificate is to be certified until Jun  4 15:28:55 2018 GMT (365 days)
Sign the certificate? [y/n]:y


1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
root@iZbp1hfvivcqx1jbwap31iZ:~/ca#
```

5. 次のコマンドを実行して、ほとんどのブラウザーで認識可能な *PKCS12* ファイルに変換します。

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

プロンプトに従ってクライアント鍵のパスフレーズを入力します。

クライアント証明書のエクスポートに使用するパスワードを入力します。 これはクライアント証明書を保護するためのパスワードで、クライアント証明書をインストールするときに必要です。

```
root@iZbp1hfvivcqx1jbwap31iZ:~/ca# sudo openssl pkcs12 -export -clcerts -in /roo
t/ca/users/client.crt -inkey /root/ca/users/client.key -out /root/ca/users/clien
t.p12
Enter pass phrase for /root/ca/users/client.key:
Enter Export Password:
Verifying - Enter Export Password:
root@iZbp1hfvivcqx1jbwap31iZ:~/ca#
```

6. 生成されたクライアント証明書を表示するには、次のコマンドを実行します。

```
cd  users
```

```
ls
```

```
root@iZbp1hfvivcqx1jbwap31iZ:~/ca# cd users
root@iZbp1hfvivcqx1jbwap31iZ:~/ca/users# ls
client.crt  client.csr  client.key  client.p12
root@iZbp1hfvivcqx1jbwap31iZ:~/ca/users#
```

ステップ 4 サーバー証明書と CA 証明書のアップロード

1. *SLB コンソール*にログインします。
2. [Server Load Balancer] ページで、[SLB インスタンスの作成] をクリックします。
3. インスタンスを設定し、[今すぐ購入] をクリックします。

   このチュートリアルでは、インスタンスタイプは [インターネット]、リージョンは [中国 (杭州)] です。 詳しくは「*SLB インスタンスの作成*」をご参照ください。
4. [Server Load Balancer] ページに戻り、マウスをインスタンス名の領域の上に合わせ、表示された鉛筆のアイコンをクリックして、SLB インスタンスの名前を変更します。
5. 左側のナビゲーションウィンドウで、[証明書] タブをクリックします。
6. [証明書のアップロード] をクリックします。
7. [証明書の作成] ページで、次の設定を行い、[OK] をクリックします。
   - リージョン: このチュートリアルでは、[中国 (杭州)] を選択します。

     注 :
     証明書のリージョンは、Server Load Balancer インスタンスのリージョンと同じでなければなりません。
   - 証明書タイプ: [サーバー証明書] を選択します。
   - 証明書の内容と秘密鍵: サーバー証明書の内容と秘密鍵をコピーします。

     注 :
     コンテンツをコピーする前に、[サンプルのインポート] をクリックして有効な証明書形式と秘密鍵形式を表示します。 詳しくは、「証明書の形式」をご参照ください。
8. 左側のナビゲーションウィンドウで、[証明書] をクリックし、[証明書の作成] をクリックして CA 証明書をアップロードします。
9. [証明書の作成] ページで、次の設定を行い、[OK] をクリックします。
   - リージョン: このチュートリアルでは、[中国 (杭州)] を選択します。

     注 :

証明書のリージョンは、Server Load Balancer インスタンスのリージョンと同じでなければなりません。

- 証明書タイプ: [CA 証明書] を選択します。
- 証明書の内容: CA 証明書の内容をコピーします。

注:
コンテンツをコピーする前に、[サンプルのインポート] をクリックして有効な証明書形式と秘密鍵形式を表示します。 詳しくは、「証明書の形式」をご参照ください。

ステップ 5 クライアント証明書のインストール

生成されたクライアント証明書をインストールします。 このチュートリアルでは、Windows オペレーティングシステムと IE Web ブラウザーを例として使用します。

1. Git Bash コマンドラインウィンドウを開き、次のコマンドを実行してステップ 3 で生成したクライアント証明書をエクスポートします。

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

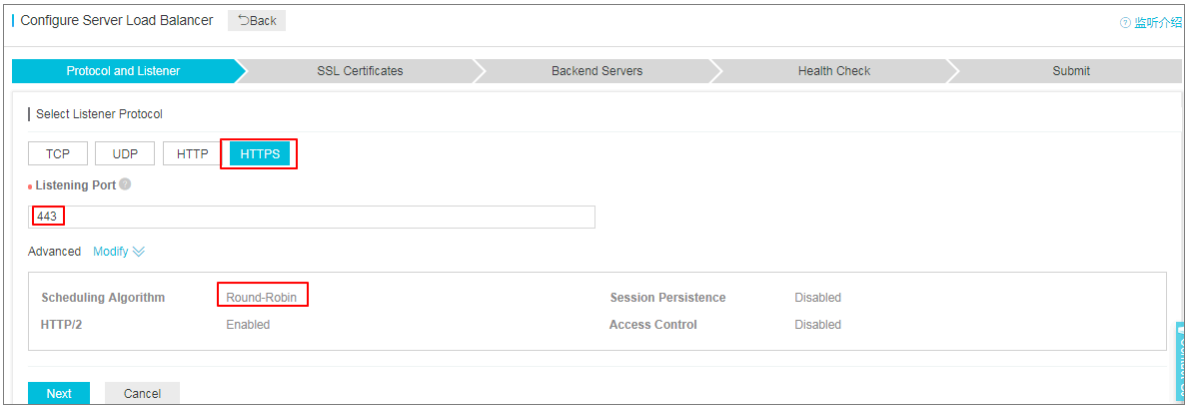
注:
IPaddress は、クライアント証明書が生成されたサーバーの IP です。

2. 証明書を IE Web ブラウザーにインポートします。
   a. IE Web ブラウザーを開き、[インターネットオプション] > [設定] をクリックします。
   b. [コンテンツ] タブをクリックし、[証明書] をクリックして、ダウンロードしたクライアント証明書をインポートします。 証明書をインポートするとき、 *PKCS12* ファイルのパスワードを入力してください。

ステップ 6 HTTPS リスナーの設定 (相互認証)

1. *SLB コンソール*にログインします。
2. [中国 (杭州)] リージョンを選択して、作成した SLB インスタンスの ID をクリックするか、[リスナーの設定] をクリックします。
3. [リスナー] タブをポイントして[リスナーの追加] をクリックします。

4. [プロトコルとリスナー] タブで、リスナーを設定します。
   - リスナープロトコルの選択: HTTPS
   - リスニングポート: 443
   - スケジューリングアルゴリズム: ラウンドロビン (RR)

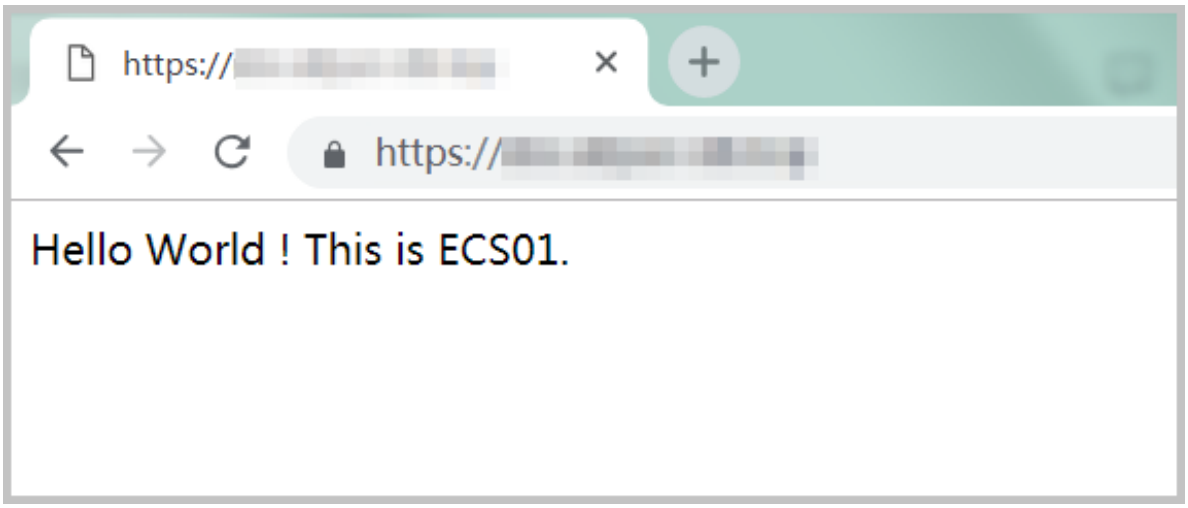


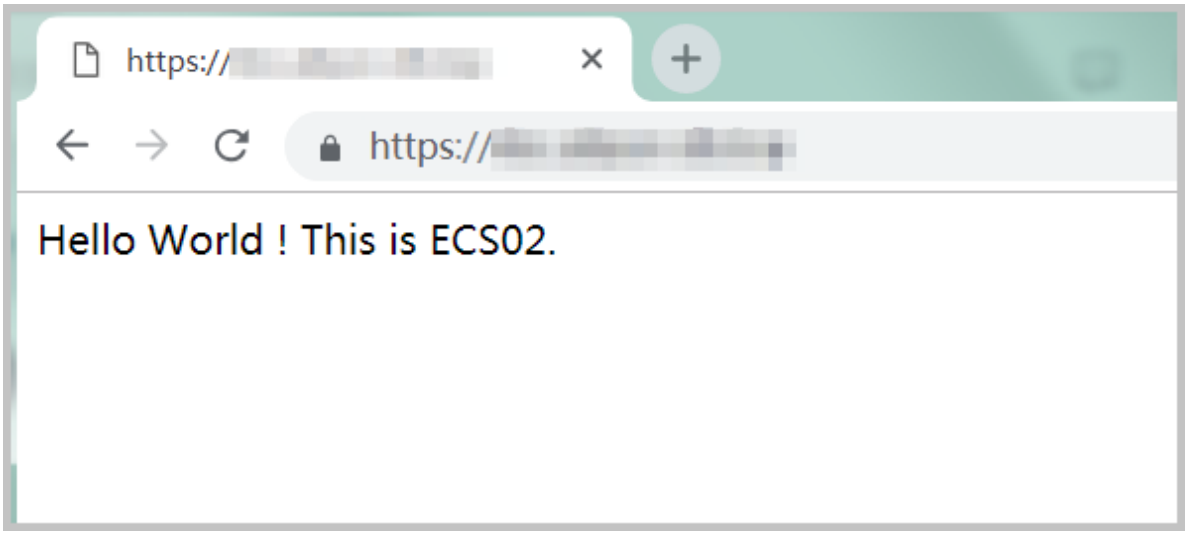
5. [次へ] をクリックします。 [SSL 証明書] タブで、SSL 証明書を設定し、相互認証を有効にします。
   - サーバー証明書: アップロードしたサーバー証明書を選択します。
   - CA 証明書: アップロードした CA 証明書を選択します。
6. [次へ] をクリックします。 表示されたページで、[デフォルトのサーバーグループ] をクリックし、[追加] をクリックします。 ECS インスタンスを追加し、バックエンドポートを 80 に設定します。
7. [次へ] をクリックしてヘルスチェックを有効にします。
8. [次へ] をクリックしてリスナー設定を表示します。
9. [送信] をクリックします。
10. [OK] をクリックします。

ステップ 7 SLB サービスのテスト

1. [Server Load Balancer] ページに戻り、ヘルスチェックステータスを表示します。 ステータスが [正常] の場合、バックエンドサーバーは SLB リスナーによって転送されたリクエストを受信できます。
2. Web ブラウザーに Server Load Balancer インスタンスのパブリック IP アドレスを入力し、クライアント証明書を信頼するかどうかを尋ねるメッセージが表示されたら、[信頼] を選択します。

3. Web ページをリフレッシュすると、リクエストがバックエンドサーバーに均等に分散されていることがわかります。





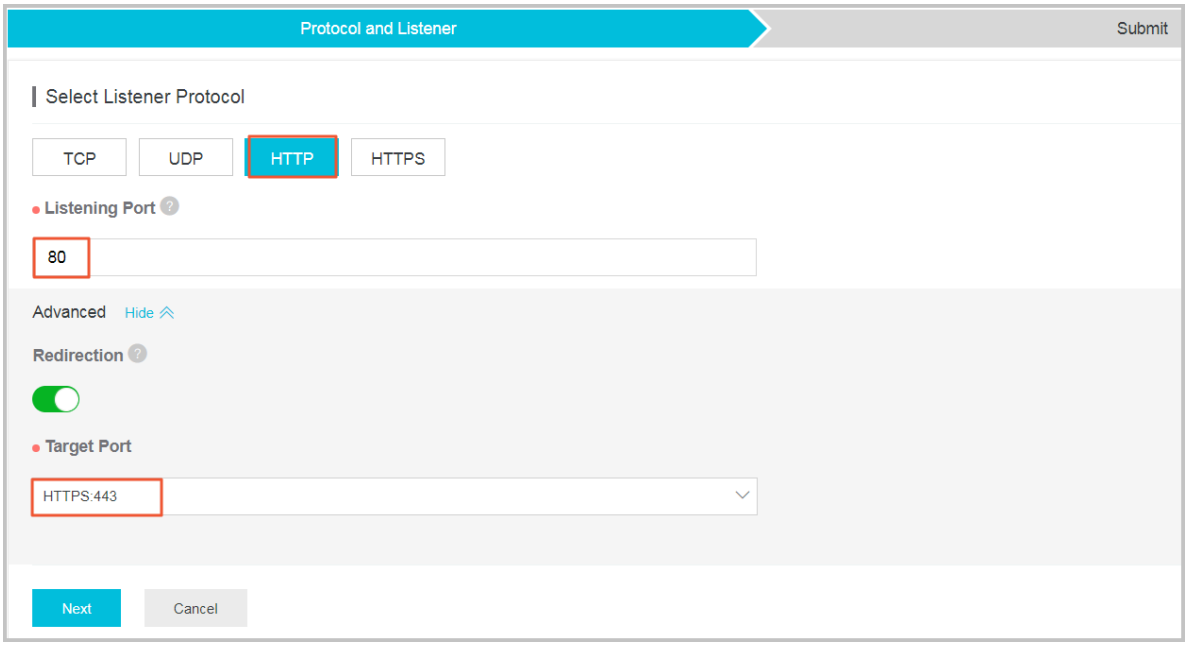
3 HTTP から HTTPS へのリダイレクト

HTTPS は HTTP の安全なバージョンです。 HTTPS では、ブラウザーとサーバー間で送信されるデータは暗号化されています。 SLB (Server Load Balancer) は HTTP リクエストを HTTPS へリダイレクトし、サイト全体の HTTPS 化を促進します。 HTTP リクエストを HTTPS へリダイレクトする機能は、すべてのリージョンでサポートされています。

HTTPS リスナーが作成されていること。 詳しくは、「*HTTPS リスナーの追加*」をご参照ください。

このチュートリアルでは、HTTP 80 リクエストを HTTPS 443 にリダイレクトする例を取り上げます。

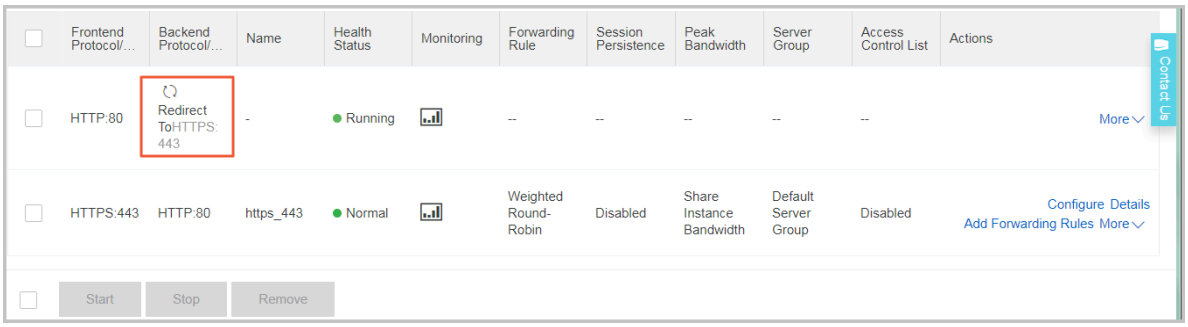
1. *SLBコンソール*にログインします。
2. トップメニューで、SLB インスタンスが配置されているリージョンを選択します。
3. [Server Load Balancer] ページで、対象となる SLB インスタンスの ID をクリックします。
4. [リスナー] タブをクリックし、[リスナーの追加] をクリックします。
5. [Server Load Balancer の設定] ダイアログボックスで、リスナープロトコルとして [HTTP] を選択し、リスニングポートを 80 に設定します。
6. [リダイレクト] を有効にして、対象ポートとして [HTTPS: 443] を選択します。



7. [次へ] をクリックします。

8. 確認して [送信] をクリックします。

   リダイレクト機能を有効にすると、すべての HTTP リクエストが HTTPS リスナーにリダイレクトされ、HTTPS リスナーのリスナー設定に従って送信されます。



4 SLB インスタンスに対する複数ドメイン名 HTTPS Web サイトの設定

このチュートリアルでは、ドメイン名の拡張子の設定方法を説明します。

シナリオ

このチュートリアルでは、例として中国 (杭州) リージョンのパフォーマンス専有型 SLB1 インスタンス (SLB1) を使用します。 一方向認証の HTTPS リスナーを SLB インスタンスに追加します。 ドメイン名 *.example1.com から VServer グループ test1 にリクエストを転送し、ドメイン名 www.example2.com から VServer グループ test2 にリクエストを転送したいとします。

これを実現するには、以下のタスクを実行します。

1. HTTPS リスナーを追加します。
2. 転送ルールを設定します。
3. ドメイン名の拡張子を追加します。

前提条件

- パフォーマンス専有型 SLB1 インスタンスを中国 (杭州) に作成します。 詳しくは、「*SLB インスタンスの作成*」をご参照ください。
- このチュートリアルで必要な証明書をアップロードします。 詳しくは、「証明書の作成」をご参照ください。
  - デフォルトでは、リスナーは default という名前の証明書を使用します。
  - 使用するドメイン名 *.example1.com の証明書 (example1) をアップロードします。
  - 使用するドメイン名 www.example2.com の証明書 (example2) をアップロードします。

ステップ 1 HTTPS リスナーの追加

HTTPS リスナーを追加するには、以下のステップを実行します。

1. 左側のナビゲーションウィンドウで、[インスタンス] > [Server Load Balancer] をクリックします。
2. [Server Load Balancer] ページで、対象となる SLB1 インスタンスを検索して、[操作] 列の [リスナーの設定] をクリックします。

   初めてリスナーを設定する場合は、[ポート/ヘルスチェック/バックエンドサーバー] 列の [設定] をクリックすることもできます。
3. リスナーを設定します。

   このチュートリアルで使用されている設定は次のとおりです。 詳しくは、「*HTTPS リスナーの追加*」をご参照ください。

   - 相互認証: 無効にします。
   - SSL 証明書: アップロードしたサーバー証明書を選択します。
   - バックエンドサーバー: VServer グループ test1 と test2 を作成します。

ステップ 2 転送ルールの設定

転送ルールを設定するには、次のステップを実行します。

1. SLB1 インスタンスの ID をクリックして、[インスタンスの詳細] ページに移動します。
2. [リスナー] タブで、作成済みの HTTPS リスナーを検索して [転送ルールの追加] をクリックします。
3. [転送ルールの追加] ページで、転送ルールを設定します。 詳しくは、「*#unique_10*」をご参照ください。

   このチュートリアルでは、3 つのドメイン名ベースの転送ルールが設定されており、URL は空のままです。

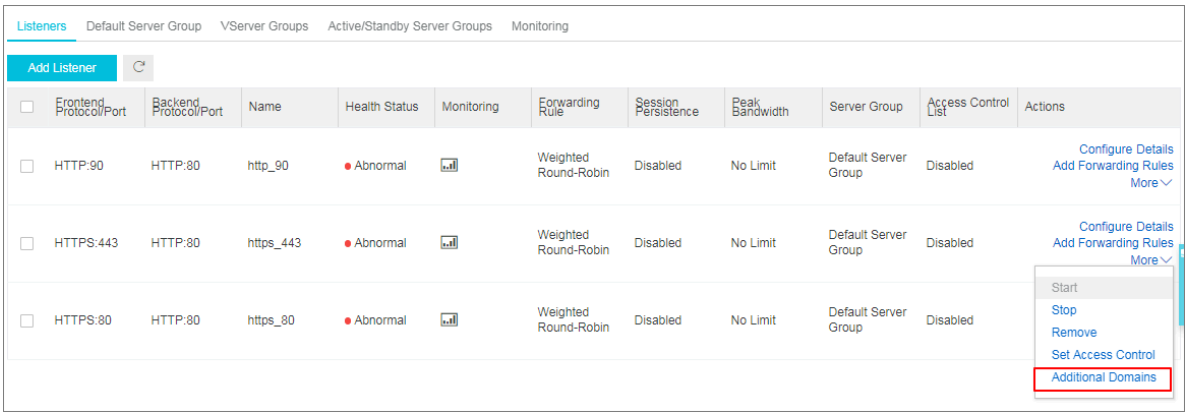
   - ルール名を設定し、[ドメイン名] 列に *.example1.com と入力し、VServer グループ test1 を選択して [転送ルールの追加] をクリックします。
   - ルール名を設定し、[ドメイン名] 列に www.example2.com と入力し、VServer グループ test2を選択して [OK] をクリックします。

   > 注 :
   > 転送ルールで設定されたドメイン名は、*#unique_11/unique_11_Connect_42_section_bk4_ypt_q2b*と証明書で追加されたドメイン名と同じでなければなりません。

ステップ 3 ドメイン名の拡張子の追加

ドメイン名の拡張子を追加するには、次のステップを実行します。

1. SLB1 インスタンスの ID をクリックして、[インスタンスの詳細] ページに移動します。
2. [リスナー] タブで、作成済みの HTTPS リスナーを検索して、[詳細] > [追加ドメイン] を選択します。



3. [追加ドメイン] ページで、[追加ドメインの追加] をクリックしてドメイン名の拡張子を追加します。

   - ドメイン名を入力します。 ドメイン名に使用できるのは英字、数字、ダッシュ、またはドットのみです。

     ドメイン名転送ルールは完全一致とワイルドカードをサポートしています。

     - 完全一致ドメイン名: www.aliyun.com
     - ワイルドカードドメイン名 (汎用ドメイン名): *.aliyun.com、*.market.aliyun.com

     リクエストが複数の転送ルールに一致する場合、完全一致が小範囲ワイルドカードより優先され、小範囲ワイルドカードが大範囲ワイルドカードより優先されます。次の表をご参照ください。

タイプ	リクエスト URL	転送ルールに基づいたドメイン名		
		www.aliyun.com	*.aliyun.com	*.market.aliyun.com
完全一致	www.aliyun.com	√	×	×
ワイルドカード一致	market.aliyun.com	×	√	×

タイプ	リクエスト URL	転送ルールに基づいたドメイン名		
		www.aliyun.com	*.aliyun.com	*.market.aliyun.com
ワイルドカード一致	info.market.aliyun.com	×	×	√

- ドメイン名に関連付けられている証明書を選択します。

> 注：
> 証明書のドメイン名は、追加されたドメイン名の拡張子と同じである必要があります。

> ：
> 設定が完了した後、問題がある場合は、結果に対するキャッシュの影響を避けるためにブラウザを再起動してください。

5 ドメイン名または URL に基づくトラフィック転送

SLB は、ドメイン名ベースまたは URL ベースの転送ルールの設定をサポートしています。 サーバーリソースを適切に割り当てるために転送ルールを追加することで、さまざまなドメイン名または URL のリクエストをさまざまなバックエンドサーバーに転送できます。

> 注 :
> 転送ルールの設定をサポートしているのは、レイヤー 7 リスナー (HTTPS/HTTP プロトコル) だけです。

ドメイン名ベースまたは URL ベースの転送ルールの概要

レイヤー 7 リスナーは、ドメイン名ベースまたは URL ベースの転送ルールを設定して、異なるドメイン名または URL のリクエストを異なる ECS インスタンスに配信することをサポートします。

URL ベースの転送ルールは文字列のマッチングをサポートし、/admin、/bbs、/test などのシーケンシャルマッチングを採用しています。

ドメイン名ベースの転送ルールは完全一致とワイルドカードをサポートしています。

- 完全一致ドメイン名: www.aliyun.com
- ワイルドカードドメイン名 (汎用ドメイン名): *.aliyun.com、*.market.aliyun.com

  リクエストが複数の転送ルールに一致する場合、完全一致が小規模なワイルドカード一致より優先され、小規模なワイルドカードの一致の方が大規模なワイルドカードの一致よりも優先されます。詳しくは次の表を参照してください。

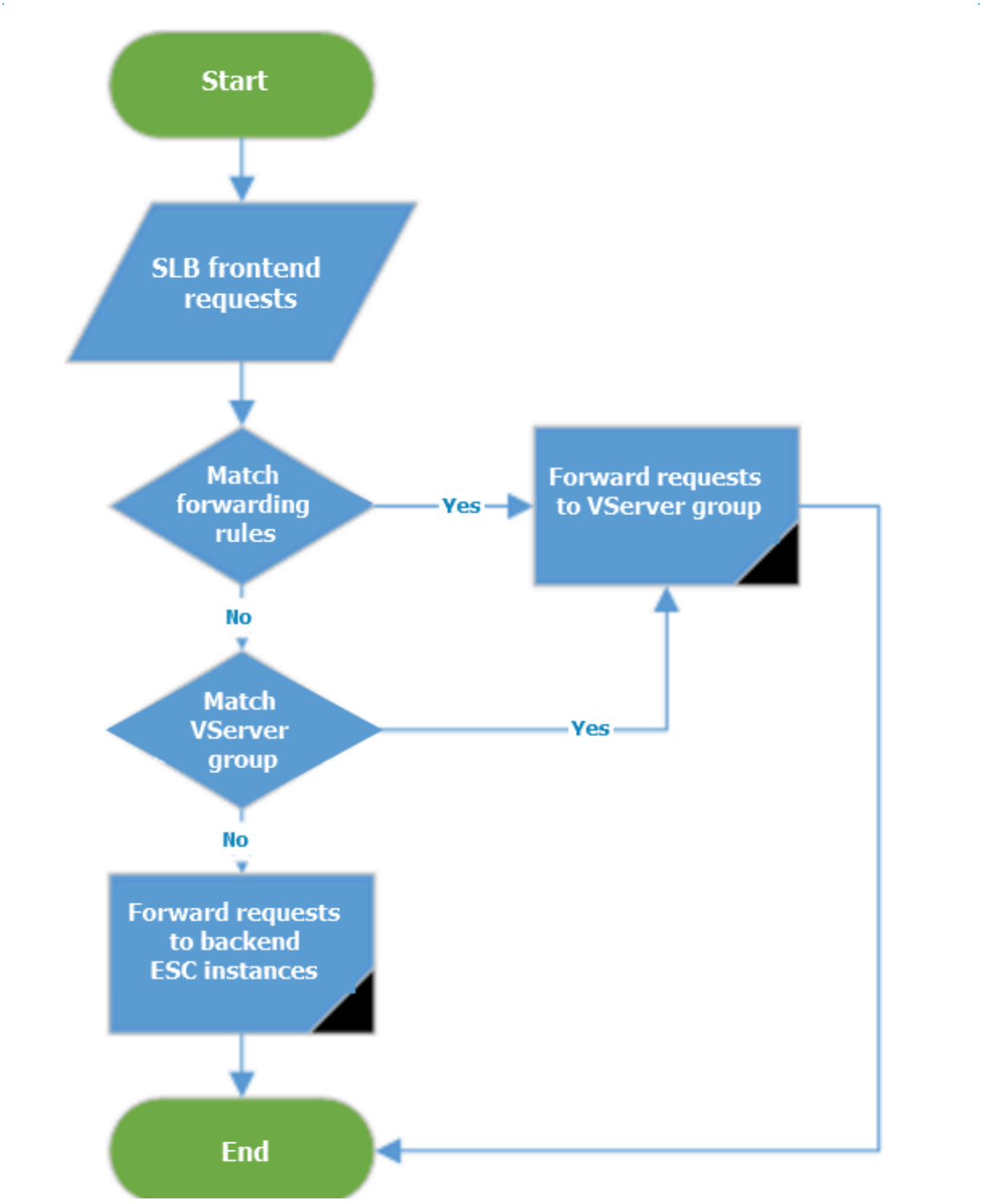
タイプ	リクエスト URL	転送ルールに基づいたドメイン名		
		www.aliyun.com	*.aliyun.com	*.market.aliyun.com
完全一致	www.aliyun.com	√	×	×
ワイルドカード	Market.aliyun.com	×	√	×
ワイルドカード	info.market.aliyun.com	×	×	√

さまざまな VServer グループに関連付けられたさまざまな転送ルールをレイヤー 7 リスナーに追加できます (VServer グループは複数の ECS インスタンスで構成されています)。 たとえば、

リソース使用を最適化するために、すべての読み取りリクエストをバックエンドサーバーのグループに転送し、すべての書き込みリクエストをバックエンドサーバーの別のグループに転送することができます。

転送ルールが設定された後のリクエスト転送のシーケンスは以下のとおりです。

- リクエストが転送ルールと一致する場合、このルールに関連付けられた VServer グループに配信されます。
- そうでない場合、リスナーが VServer グループに関連付けられていると、リクエストはリスナーに設定されている VServer グループに配信されます。
- 上記のいずれの条件も満たされない場合、リクエストはデフォルトのサーバーグループ内の ECS インスタンスに転送されます。



ドメイン名ベースまたは URL ベースの転送ルールの追加

転送ルールを追加する前に、以下の条件が満たされていることを確認してください。

- *HTTP リスナーの追加* または*HTTPS リスナーの追加*
- *VServer グループの管理*。

ドメイン名ベースまたは URL ベースの転送ルールを追加するには、次のステップを実行します。

1. *SLB* コンソールにログインします。
2. リージョンを選択すると、このリージョン内に存在するすべての SLB インスタンスが表示されます。
3. 対象となる SLB インスタンスの ID をクリックします。
4. [リスナー] タブをクリックします。
5. 対象となる HTTP/HTTPS リスナーを検索して、[転送ルールの追加] オプションをクリックします。

6. [転送ルールの追加] ページで、[転送ルールの追加] をクリックします。
7. [転送ルールの追加] ページで、次の情報に従って転送ルールを設定します。
   a. ドメイン名: リクエストのドメイン名を入力します。 文字、数字、ハイフン、ドットのみを使用できます。
   b. URL: リクエストのパスを入力します。 URL はスラッシュ (/) で始める必要があり、文字、数字、および以下の特殊文字のみを含めることができます (-. /%? #&)。

      注 :
      ドメイン名に基づく転送ルールのみを設定したい場合は、URL オプションを空白のままにしてください。

   c. VServer グループ: 関連付けられている VServer グループを選択します。
   d. 説明 (オプション): 説明を入力します。
   e. [確認] をクリックします。

8. [ドメインの追加] または [ルールの追加] をクリックして、別のドメイン名ベースまたは URL ベースの転送ルールを追加します。

   詳しくは、「制限事項」をご参照ください。

転送ルールの編集

転送ルールに関連付けられているバックエンドサーバーを変更できます。

転送ルールを編集するには、次のステップを実行します:

1. *SLB* コンソールにログインします。

2. リージョンを選択すると、このリージョン内に存在するすべての SLB インスタンスが表示されます。
3. 目的の SLB インスタンスの ID をクリックします。
4. [リスナー] タブをクリックします。
5. 対象となるレイヤー 7 リスナーを検索して、[転送ルールの追加] オプションをクリックします。
6. [転送ルール] 領域で対象となるの転送ルールを検索して、[編集] オプションをクリックします。

7. 転送ルールを編集します。 以下の情報に従って、スケジューリングアルゴリズム、セッション維持、ヘルスチェックなどの転送ルールをカスタマイズします。

> 注 :
> 現在、転送ルールの詳細設定のカスタマイズは、次のリージョンでのみサポートされています。
>
> - 中国 (北京)
> - 中国 (杭州)
> - 中国 (上海)
> - 中国 (張家口)
> - 中国 (フフホト)
> - 中国 (香港)
> - シンガポール

・ 日本 (東京)

詳細設定	説明
スケジューリングアルゴリズム	Server Load Balancer は 3 つのスケジューリングアルゴリズムに対応しています。ラウンドロビン、重み付きラウンドロビン (WRR)、重み付け最小接続数 (WLC) です。 ・ [重み付きラウンドロビン (WRR)]: 重みの大きなバックエンドサーバーは、重みの小さなバックエンドサーバーより、多くのリクエストを受信できます。 ・ [ラウンドロビン (RR)]: リクエストは、バックエンドサーバーへ均等かつ順次に配信されます。 ・ [重み付け最小接続数 (WLC)]: 重みの大きいサーバーは、一度に受信できる接続数の割合が高くなります。 重みの値が同じ場合、接続数の少ないバックエンドサーバーの方が、より頻繁に (そして高い確率で) アクセスされます。
セッション維持の有効化	セッション維持を有効にするかしないかを選択します。 セッション維持を有効にした場合、同一のクライアントからのセッションリクエストはすべて、同一のバックエンドサーバーに送信されます。 HTTP セッション維持は Cookie に基づいています。 次の 2 つの方法がサポートされています。 ・ cookie の挿入 : Cookie のタイムアウト時間を指定するだけです。 SLB はバックエンドサーバーからの最初のレスポンスに Cookie を追加します (HTTP/HTTPS レスポンスパケットに SERVERID を挿入します)。 次のリクエストには Cookie が含まれ、リスナーはリクエストを同じバックエンドサーバーに配信します。 ・ cookie の上書き：必要に応じて HTTP/HTTPS レスポンスに挿入される Cookieを設定できます。 バックエンドサーバー上の Cookie のタイムアウト時間とライフサイクルを維持する必要があります。 SLB は新しい Cookie が設定されたことを検出すると元の Cookie を上書きします。 次回クライアントが新しい Cookie で SLB にアクセスすると、リスナーはそのリクエストを前回記録されたバックエンドサーバーに配信します。 詳しくは「セッション維持」 をご参照ください。

詳細設定	説明
ヘルスチェックの有効化	・ ヘルスチェックポート: ヘルスチェックでバックエンドサーバーにアクセスするために使用されるポート。 デフォルトでは、リスナーで設定されたバックエンドポートが使用されます。 ・ ヘルスチェックパス: ヘルスチェックページの URI。 静的ページを確認することを推奨します。 ・ ヘルスチェックドメイン名 (オプション) : バックエンドサーバーのイントラネット IP は、デフォルトでドメイン名として使用されます。 ・ 通常のステータスコード: 正常なサーバーを示す HTTP ステータスコード。 デフォルト値は http_2xx と http_3xx。 ・ レスポンスタイムアウト: ヘルスチェックからのレスポンスを待つ時間。 ECS インスタンスが指定されたタイムアウト期間内にレスポンスしないと、ヘルスチェックは失敗です。 ・ ヘルスチェック間隔: 2 つの連続したヘルスチェック間の時間。 デフォルト値は 2 秒です。 ・ 異常しきい値: ECS インスタンスが異常と判断される前に、同じ ECS インスタンス上の同じ LVS ノードサーバーで (成功から失敗まで) 実行されたヘルスチェックの連続失敗数。 有効値: 2-10。 デフォルト値: 3。 ・ 正常しきい値: ECS インスタンスが正常と判断される前に、同じ ECS インスタンス上の同じ LVS ノードサーバーで (失敗から成功まで) 実行されたヘルスチェックの連続成功数。 有効値: 2-10。 デフォルト値: 3。

8. [確認] をクリックします。

転送ルールの削除

転送ルールを削除するには、以下のステップを実行します。

1. *SLB コンソール*にログインします。
2. リージョンを選択すると、そのリージョン内のすべての SLB インスタンスが表示されます。
3. SLB インスタンスの ID をクリックします。
4. [リスナー] タブをクリックします。

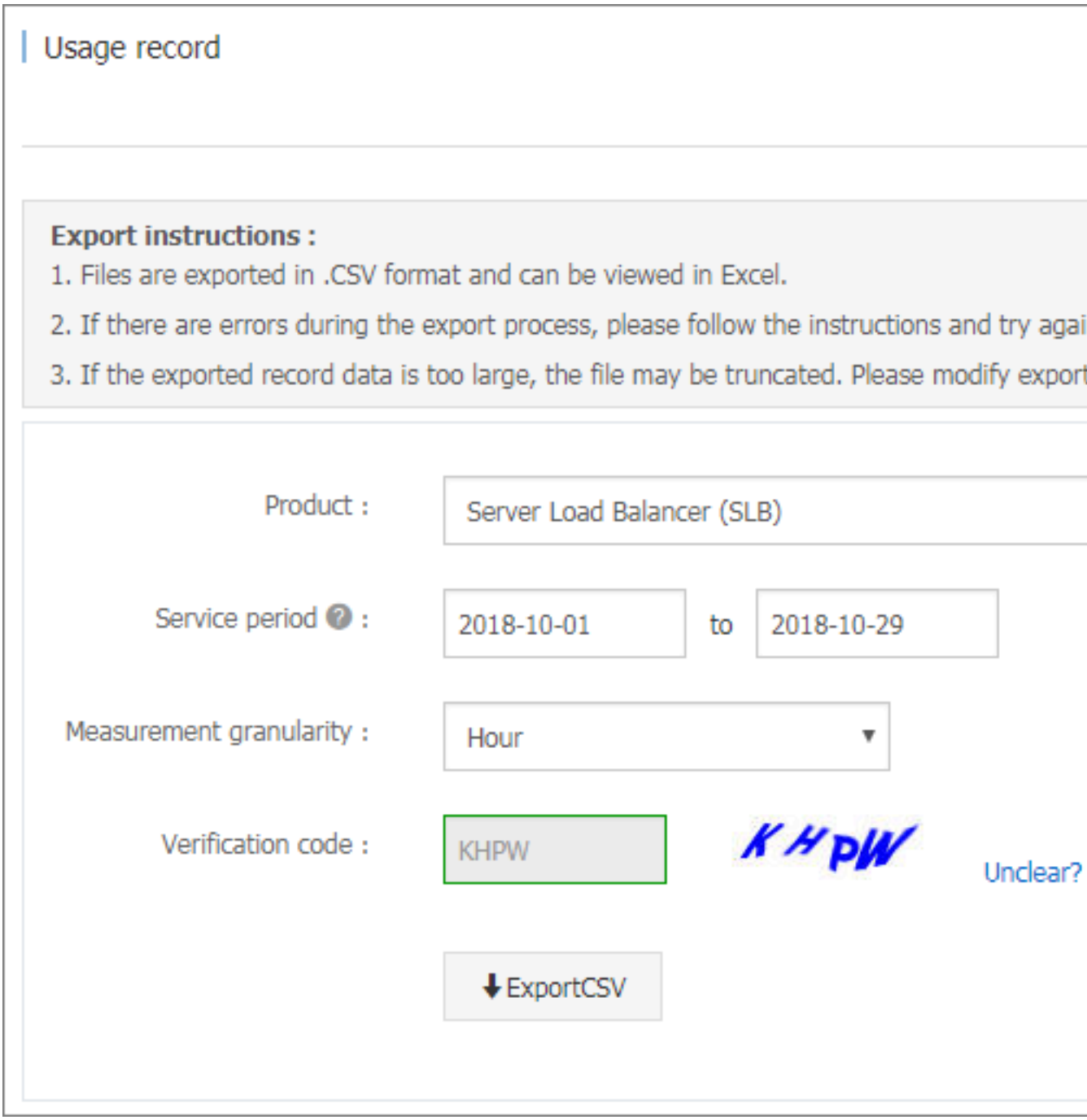
5. 対象となるレイヤー 7 リスナーを検索して、[転送ルールの追加] オプションをクリックします。
6. [転送ルール] 領域で対象となる転送ルールを検索して、[削除] オプションをクリックします。

6 トラフィック使用状況の表示

特定期間内の SLB インスタンスのトラフィック使用状況を表示できます。

1. *SLB* コンソールにログインします。
2. メニューバーの右上隅にある [料金・支払い管理] > [利用状況] を選択します。
3. [アカウントの概要] ページで、[購入レコード] > [使用状況レコード] を選択します。

4. [使用状況レコード] ページで、[Server Load Balancer (SLB)] を選択し、表示するトラフィック使用状況のサービス期間と測定粒度を設定します。



5. [CSV のエクスポート] をクリックして、トラフィック使用状況テーブルを *CSV* 形式で生成します。

   テーブルには以下の情報が含まれています。 特定のインスタンス、リージョン、またはエンドポイントのトラフィック使用状況を表示できます。

A	B	C	D	E	F	G	H	I
Instance ID	Region	Service Address	Service Address T	Bandwidth (bit/s)	Upstream	Downstream	Start Time	End Time
lb- i	cn-beijing-btc-a01	47. 189	internet	0	20480	20480	2018/10/1 0:00	2018/10/1 1:00
lb- i	cn-beijing-btc-a01	47. 189	internet	0	20100	20100	2018/10/1 1:00	2018/10/1 2:00
lb- i	cn-beijing-btc-a01	47. 189	internet	0	20710	20710	2018/10/1 2:00	2018/10/1 3:00
lb- i	cn-beijing-btc-a01	47. 189	internet	0	20354	20354	2018/10/1 3:00	2018/10/1 4:00
lb- i	cn-beijing-btc-a01	47. 189	internet	0	20344	20344	2018/10/1 4:00	2018/10/1 5:00
lb- y	cn-hangzhou-dg-a01	47. 248	internet	0	6988	6988	2018/10/1 0:00	2018/10/1 1:00
lb- y	cn-hangzhou-dg-a01	47. 248	internet	0	6914	6914	2018/10/1 1:00	2018/10/1 2:00
lb- y	cn-hangzhou-dg-a01	47. 248	internet	0	7108	7108	2018/10/1 2:00	2018/10/1 3:00
lb- y	cn-hangzhou-dg-a01	47. 248	internet	0	7094	7094	2018/10/1 3:00	2018/10/1 4:00
lb- y	cn-hangzhou-dg-a01	47. 248	internet	0	7156	7156	2018/10/1 4:00	2018/10/1 5:00
lb- o	cn-hangzhou-dg-a01	11 .62	internet	0	6928	6928	2018/10/1 0:00	2018/10/1 1:00
lb- o	cn-hangzhou-dg-a01	11 .62	internet	0	6914	6914	2018/10/1 1:00	2018/10/1 2:00
lb- o	cn-hangzhou-dg-a01	11 .62	internet	0	6796	6796	2018/10/1 2:00	2018/10/1 3:00
lb- o	cn-hangzhou-dg-a01	11 .62	internet	0	7100	7100	2018/10/1 3:00	2018/10/1 4:00
lb- o	cn-hangzhou-dg-a01	11 .62	internet	0	7110	7110	2018/10/1 4:00	2018/10/1 5:00
lb- x	cn-hangzhou-dg-a01	47. 65	internet	0	6948	6948	2018/10/1 0:00	2018/10/1 1:00
lb- x	cn-hangzhou-dg-a01	47. 65	internet	0	7062	7062	2018/10/1 1:00	2018/10/1 2:00
lb- x	cn-hangzhou-dg-a01	47. 65	internet	0	7122	7122	2018/10/1 2:00	2018/10/1 3:00
lb- x	cn-hangzhou-dg-a01	47. 65	internet	0	6974	6974	2018/10/1 3:00	2018/10/1 4:00
lb- x	cn-hangzhou-dg-a01	47. 65	internet	0	7304	7304	2018/10/1 4:00	2018/10/1 5:00
lb- r	cn-hangzhou-dg-a01	47. 117	internet	0	0	0	2018/10/1 0:00	2018/10/1 1:00
lb- r	cn-hangzhou-dg-a01	47. 117	internet	0	0	0	2018/10/1 1:00	2018/10/1 2:00
lb- r	cn-hangzhou-dg-a01	47. 117	internet	0	0	0	2018/10/1 2:00	2018/10/1 3:00
lb- r	cn-hangzhou-dg-a01	47. 117	internet	0	0	0	2018/10/1 3:00	2018/10/1 4:00
lb-bp13n724mz16d5jl1itar	cn-hangzhou-dg-a01	47.110.20.117	internet	0	0	0	2018/10/1 4:00	2018/10/1 5:00