

负载均衡 快速入门 产品文档



腾讯云

【版权声明】

©2013-2024 腾讯云版权所有

本文档著作权归腾讯云单独所有，未经腾讯云事先书面许可，任何主体不得以任何形式复制、修改、抄袭、传播全部或部分本文档内容。

【商标声明】

及其它腾讯云服务相关的商标均为腾讯云计算（北京）有限责任公司及其关联公司所有。本文档涉及的第三方主体的商标，依法由权利人所有。

【服务声明】

本文档意在向客户介绍腾讯云全部或部分产品、服务的当时的整体概况，部分产品、服务的内容可能有所调整。您所购买的腾讯云产品、服务的种类、服务标准等应由您与腾讯云之间的商业合同约定，除非双方另有约定，否则，腾讯云对本文档内容不做任何明示或默示的承诺或保证。

文档目录

快速入门

域名化负载均衡快速入门

负载均衡快速入门

IPv6 负载均衡快速入门

传统型负载均衡快速入门

CentOS 下部署 Nginx

CentOS 下部署 Java Web

快速入门

域名化负载均衡快速入门

最近更新时间：2024-01-04 17:08:07

腾讯云负载均衡支持 TCP、UDP、TCP SSL、QUIC、HTTP 和 HTTPS 协议，提供基于域名和 URL 路径的灵活转发能力。本文将引导您如何快速创建一个域名化负载均衡（CLB）实例，并将来自客户端的访问请求转发到后端服务器上（本文中的后端服务器以 CVM 为例）。

前提条件

1. 已创建两个云服务器实例（本文以 `rs-1` 和 `rs-2` 实例为例）。有关如何创建云服务器实例，请参考 [购买并启动云服务器实例](#)。
2. 已在云服务器上搭建后端服务。本文以 HTTP 转发为例，已在两台云服务器上部署 Nginx 服务器，并在 `rs-1` 和 `rs-2` 分别返回一个带有 “Hello nginx! This is rs-1!” 和 “Hello nginx! This is rs-2!” 的 HTML，详情请参考 [CentOS 下部署 Nginx](#)。

说明：

本文以标准账户类型（原“带宽上移账户”）为例。传统账户类型（原“非带宽上移账户”）的 CVM 必须购买公网带宽，因为当前的带宽属性在 CVM 上，而非 CLB 上。若您无法确定账户类型，请参见 [判断账户类型](#)。

示例中后端服务器部署的服务返回值不同，实际场景中，为保持所有用户均有一致体验，后端服务器上一般是部署完全相同的服务。

使用步骤总览

1. [注册自有域名](#)
2. [购买负载均衡实例](#)
3. [配置负载均衡监听器](#)
4. [配置安全组](#)
5. [添加域名解析](#)
6. [验证负载均衡服务](#)

操作步骤

步骤一：注册域名

域名注册是在互联网上建立服务的基础。

如果您已经在其他注册商拥有了自己的域名，您可以将域名转入腾讯云域名服务，详情请参见 [域名转入腾讯云](#)。

如果您还没有域名，您需要进行域名注册。

步骤二：购买负载均衡实例

负载均衡实例成功购买后，系统将自动分配一个**域名**，您可以通过该域名使用负载均衡服务。

1. 登录腾讯云 [负载均衡服务购买页](#)。
2. 在负载均衡 CLB 购买页面，地域选择与云服务器相同的地域，实例类型选择**负载均衡**，网络类型选择**公网**。更多参数详情请参考 [产品属性选择](#)。
3. 单击立即购买，完成付款。
4. 返回实例管理页面，选择对应的地域即可看到新建的实例。

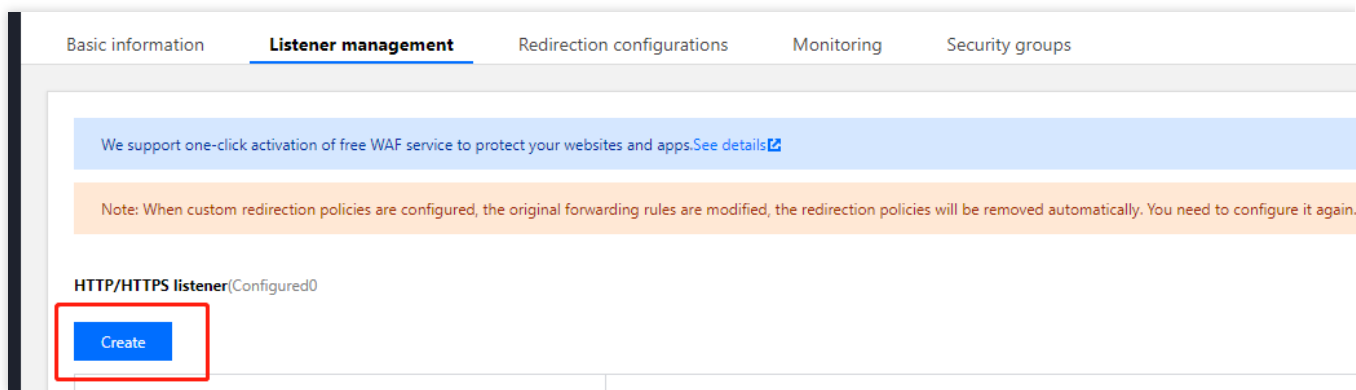
步骤三：配置负载均衡监听器

负载均衡监听器通过指定协议及端口来负责实际转发。本文以负载均衡转发客户端的 HTTP 请求配置为例。更多关于负载均衡监听器的更多内容，请参考 [负载均衡监听器概述](#)。

配置 HTTP 监听协议和端口

当客户端发起请求时，负载均衡会根据监听的前端协议与端口接收请求并向后端服务器转发请求。

1. 登录 [负载均衡控制台](#)。
2. 在“实例管理”页面中，找到目标负载均衡实例，单击**配置监听器**。
3. 在“监听器管理”页签中的“HTTP/HTTPS 监听器”区域下，单击**新建**。



4. 在“创建监听器”对话框中，配置以下内容，配置完成后，单击**提交**。

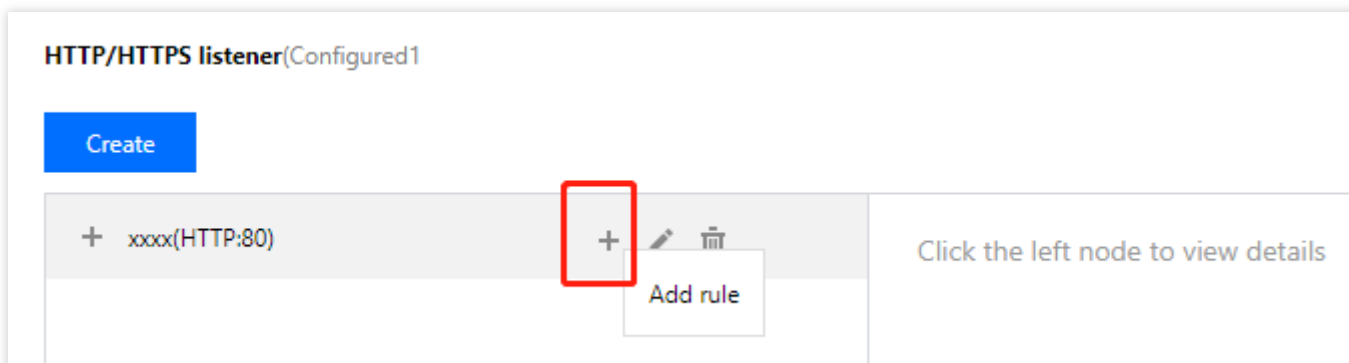
监听器名称，名称可输入60个字符，允许英文字母、数字、中文字符、“-”、“_”、“.”。

监听协议端口。例如 HTTP：80 。

配置监听器的转发规则

当客户端请求时，负载均衡会根据配置的监听器转发规则进行请求转发。

1. 在“监听器管理”页签中，选中刚才新建的监听器，单击+添加规则。



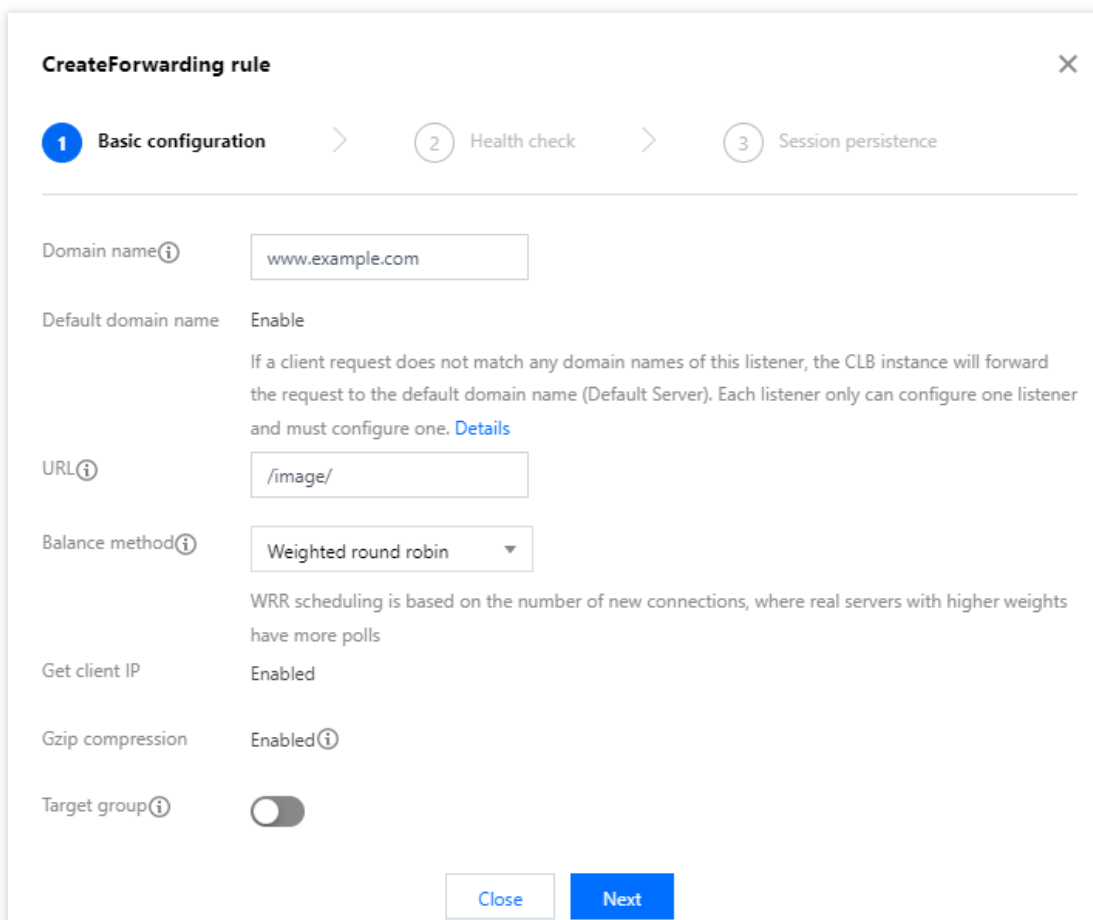
2. 在“创建转发规则”对话框的“基本配置”页签中，配置域名、URL 路径和均衡方式。配置完成后，单击下一步。

域名：您的后端服务所使用的域名，本例使用 `www.example.com`。

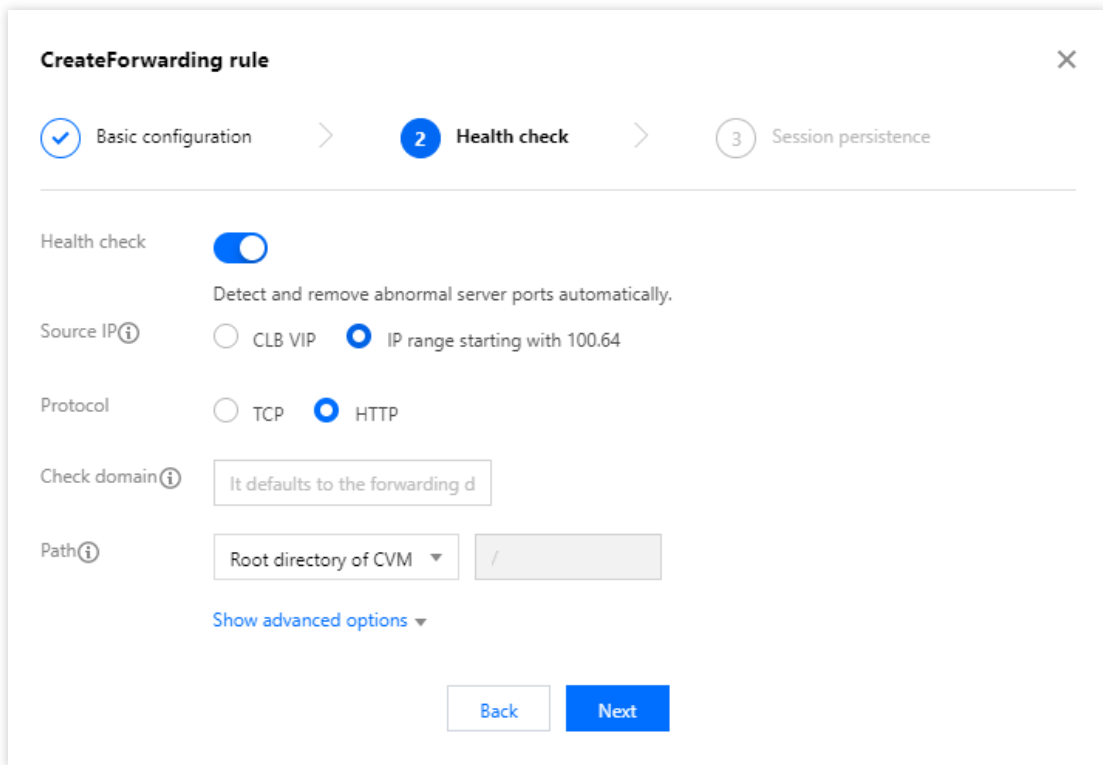
默认域名：当客户端请求没有匹配本监听器的任何域名时，CLB 会将请求转发给默认域名（default server），每个监听器只能配置一个默认域名。若该监听器没有配置默认域名时，CLB 会将请求转发给第一个域名。本例不配置。

URL 路径：您的后端服务的访问路径，本例使用 `/image/`。

均衡方式：选择“加权轮询”。详情请参见 [均衡方式](#)。



3. 在“健康检查”页签，开启健康检查，检查域名和检查路径使用默认的转发域名和转发路径，完成后单击下一步。



Create Forwarding rule

Basic configuration > **2 Health check** > 3 Session persistence

Health check ☒ Detect and remove abnormal server ports automatically.

Source IP ⓘ ☐ CLB VIP ☒ IP range starting with 100.64

Protocol ☐ TCP ☒ HTTP

Check domain ⓘ

Path ⓘ

[Show advanced options](#)

[Back](#) [Next](#)

4. 在“会话保持”页签，关闭会话保持，单击**提交**。

说明：

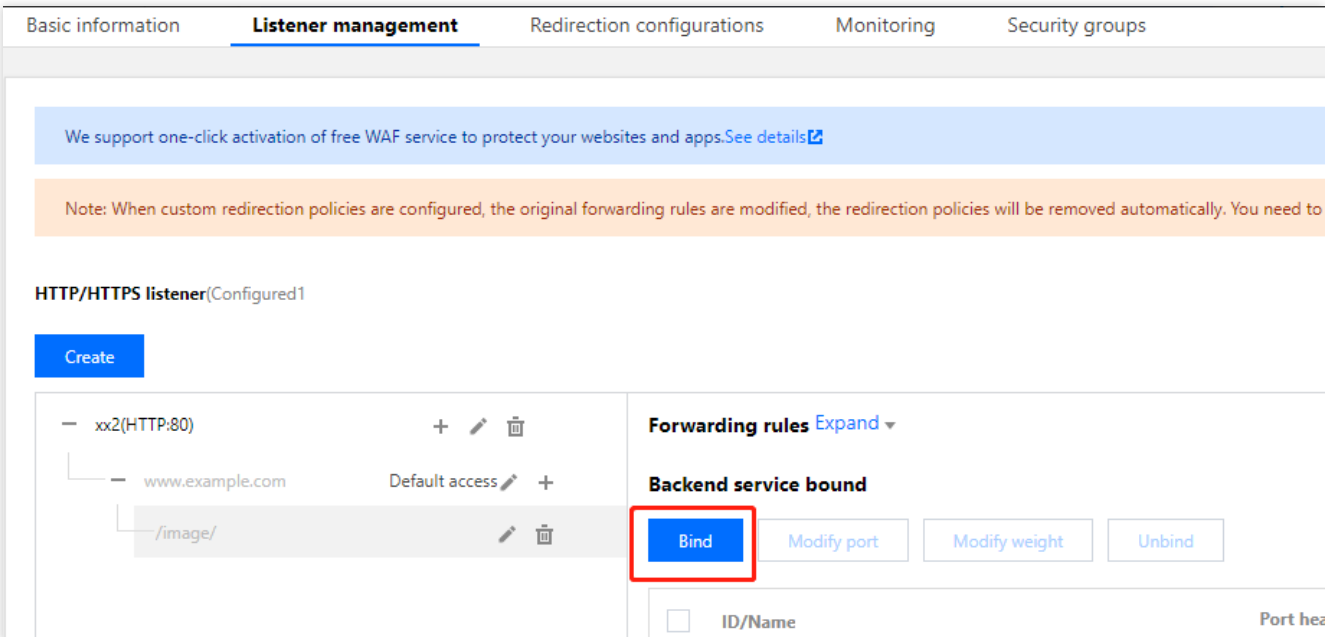
转发规则：一个监听器可配置多个域名，一个域名可配置多条 URL 路径，选中监听器或域名，单击+，即可创建新的规则。

会话保持：若关闭会话保持功能，选择轮询的方式进行调度，则请求会被依次分配到不同后端服务器上；若开启会话保持功能，或关闭会话保持功能但选择 ip_hash 的调度方式，则请求会被持续分配到同一台后端服务器上。

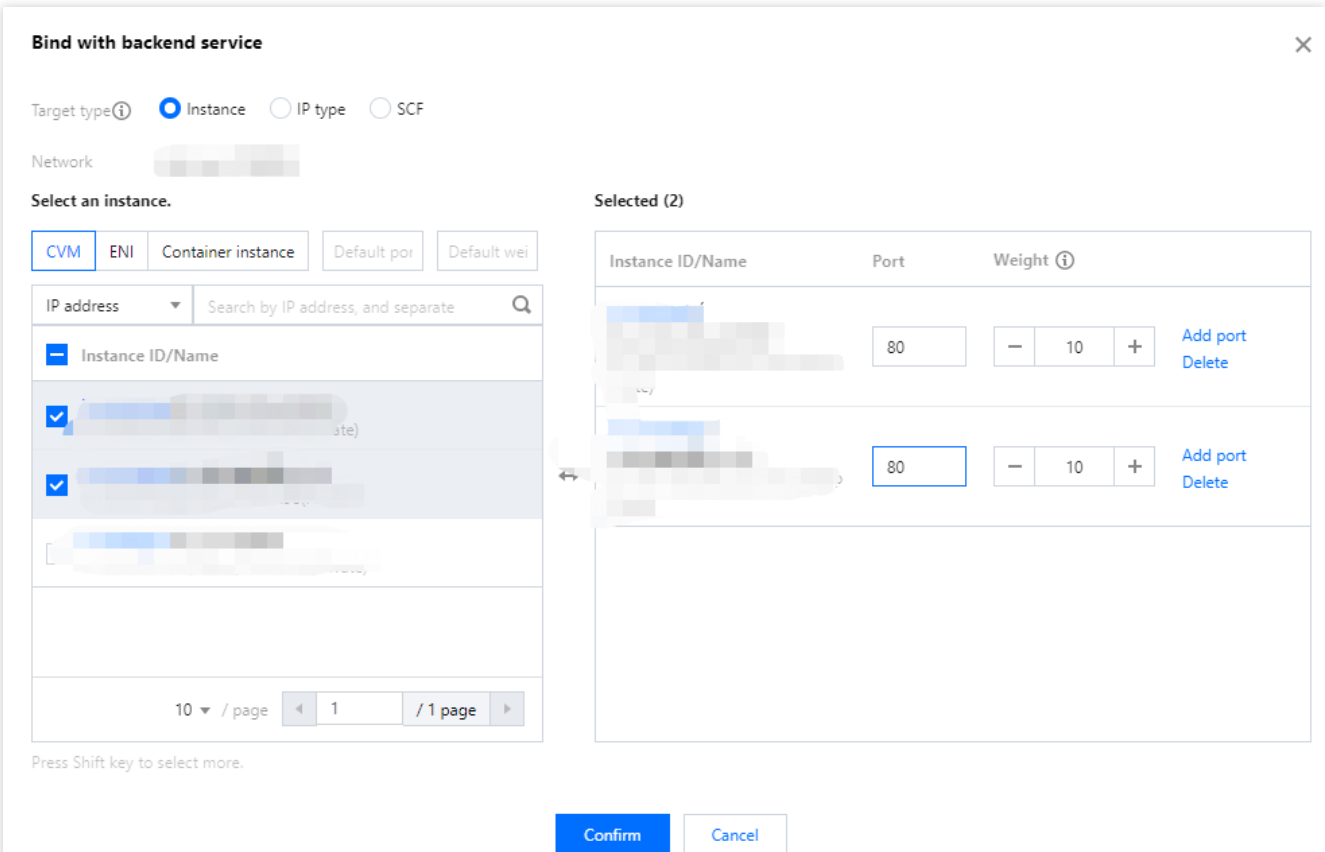
为监听器绑定后端云服务器

当客户端请求时，负载均衡将请求转发到监听器绑定的云服务器上进行处理。

1. 在“监听器管理”页面，单击+展开刚才创建的监听器，选中 URL 路径，在右侧**转发规则详情**区域单击绑定。



2. 在“绑定后端服务”对话框中，选择绑定实例类型为“云服务器”，再选择与 CLB 实例同地域下的云服务器实例 rs-1 和 rs-2 ，设置云服务器端口均为“80”，云服务器权重均为默认值“10”，然后单击**确认**。



3. 返回“转发规则详情”区域，可以查看绑定的云服务器和其健康检查状态，当端口健康状态为“健康”时表示云服务器可以正常处理负载均衡转发的请求。

说明：

一条转发规则（监听协议 + 端口 + 域名 + URL 路径）可以绑定同一台云服务器的多个端口。如用户在 `rs-1` 的 `80` 和 `81` 端口部署了一样的服务，则 CLB 支持示例中的转发规则同时绑定 `rs-1` 的 `80` 和 `81` 端口，两个端口都会接收到 CLB 转发的请求。

步骤四：配置安全组

创建完负载均衡后，您可以配置负载均衡的安全组来隔离公网流量，详情请参考 [配置安全组](#)。
安全组配置完成后，您可以选择开启或关闭安全组默认放通，不同选择配置如下所示。

方法一：开启安全组默认放通

具体操作请参考 [配置安全组默认放通](#)。

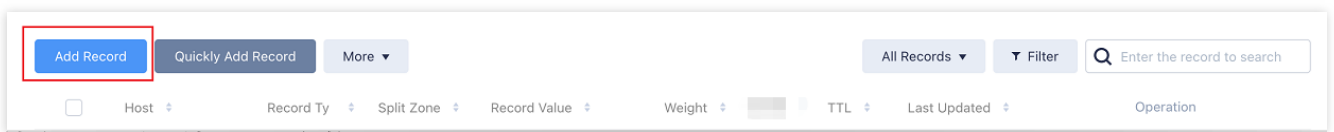
方法二：在 CVM 安全组上放通客户端 IP

具体操作请参考 [配置安全组默认放通](#)。

步骤五：添加域名解析

域名注册成功后，您可为域名添加域名解析，以便通过域名访问网站。

1. 登录 [DNS 解析 DNSPod 控制台](#)，在 **域名解析列表** 页面，单击目标域名右侧 **操作列** 的 **解析**。
2. 在 **记录管理** 页签，单击 **添加记录**。



3. 在添加记录区域，填写以下参数：

3.1 按需填写主机记录，主机记录就是域名前缀，常见用法有：

www：解析后的域名为 `www.example.com`。

@：直接解析主域名 `example.com`。

*****：泛解析，匹配其他所有域名 `*.example.com`。

3.2 选择 **记录类型**，推荐选择 `CNAME`。

Add RecordQuickly Add RecordMore ▼

<input type="checkbox"/>	Host ▼	Record Ty ▼	Split Zone ▼	Record Value ▼	Weight ▼
<input type="checkbox"/>	www	CNAME ▼	Default ▼		

All record types serve for different purposes and A record is generally selected. [View guide](#)

A

The most common record type. It points a domain name to an IPv4 address, e.g. 8.8.8.8.

CNAME

Points a domain name to another domain name, e.g. https://www.dnspod.cn, and finally to the IP address of the target domain.

MX

Used for mail server. Relevant parameters are generally provided by the email service provider.

TXT

Commonly used to verify a domain. You can fill in additional text information.

3.3 线路类型：选择“默认”类型，否则会导致部分用户无法解析。例如，您需要将联通用户指向 2.com，所有非联通用户都指向 1.com。您可以通过添加线路类型为默认、记录值为 1.com 和线路类型为联通、记录值为 2.com 的两条 CNAME 记录来实现。

3.4 填写记录值，CNAME 记录值可填写为负载均衡提供给你的域名。

3.5 其余值可以保持默认值，操作完成后，单击保存。

Add RecordQuickly Add RecordMore ▼

<input type="checkbox"/>	Host ▼	Record Ty ▼	Split Zone ▼	Record Value ▼	Weight ▼
<input type="checkbox"/>	www	CNAME ▼	Default ▼	lb-lpnytest-e4841923	56

4. 添加记录完成后，可以在记录管理页签的记录列表查看刚才添加的记录。

Add RecordQuickly Add RecordMore ▼

All Records ▼Filter

<input type="checkbox"/>	Host ▼	Record Ty ▼	Split Zone ▼	Record Value ▼	Weight ▼	TTL ▼	Last Updated ▼
<input type="checkbox"/>	www	CNAME		lb-lpnytest-e4841923	56	60	2023-01-01 10:00:00

步骤六：验证负载均衡服务

在添加完解析记录十分钟左右后，在浏览器中输入绑定后的 CNAME 域名（如本例中的 www.example.com），若正常显示页面，则表示负载均衡生效。

配置重定向功能（可选）

重定向配置分为手动重定向和自动重定向，详情请参见 [重定向配置](#)：

自动重定向（强制 HTTPS）：PC 端、手机浏览器等以 HTTP 请求访问 Web 服务，在 CLB 代理后，返回 HTTPS 响应。默认强制浏览器以 HTTPS 访问网页。

手动重定向：当出现 Web 业务需要临时下线（如电商售罄、页面维护和更新升级时）需要重定向能力。如果不做重定向，用户的收藏和搜索引擎数据库中的旧地址只能让访客得到一个404或503错误信息页面，降低了用户体验度，导致访问流量白白流失，且该页面积累的搜索引擎评分也会无效。

相关操作

[Linux（CentOS）下部署 Java Web](#)

负载均衡快速入门

最近更新时间：2024-01-04 17:08:07

腾讯云负载均衡支持 TCP、UDP、TCP SSL、HTTP 和 HTTPS 协议，提供基于域名和 URL 路径的灵活转发能力。本文将引导您如何快速创建一个负载均衡（CLB）实例，并将来自客户端的访问请求转发到两台后端云服务器（CVM）上。

前提条件

1. 已创建两个云服务器实例（本文以 `rs-1` 和 `rs-2` 实例为例）。有关如何创建云服务器实例，请参考 [购买并启动云服务器实例](#)。
2. 已在云服务器上搭建后端服务。本文以 HTTP 转发为例，已在两台云服务器上部署 Nginx 服务器，并在 `rs-1` 和 `rs-2` 分别返回一个带有 “Hello nginx! This is rs-1!” 和 “Hello nginx! This is rs-2!” 的 HTML，详情请参考 [CentOS 下部署 Nginx](#)。

注意：

本文以标准账户类型（原“带宽上移账户”）为例。传统账户类型（原“带宽非上移账户”）的 CVM 必须购买公网带宽，因为当前的带宽属性在 CVM 上，而非 CLB 上。若您无法确定账户类型，请参见 [判断账户类型](#)。

示例中后端服务器部署的服务返回值不同，实际情况中，为保持所有用户均有一致体验，后端服务器上一般是部署完全相同的服务。

步骤一：购买负载均衡实例

负载均衡实例成功购买后，系统将自动分配一个 VIP，该 VIP 为负载均衡向客户端提供服务的 IP 地址。

1. 登录腾讯云 [负载均衡服务购买页](#)。
2. 在负载均衡 CLB 购买页面，地域选择与云服务器相同的地域，实例类型选择**负载均衡**，网络类型选择**公网**。更多参数详情请参考 [产品属性选择](#)。

说明：

目前仅广州、上海、南京、济南、杭州、福州、北京、石家庄、武汉、长沙、成都、重庆地域支持静态单线 IP 线路类型，其他地域支持情况请以控制台页面为准。如需体验，请联系商务经理申请。申请通过后，即可在购买页选择中国移动、中国联通或中国电信的运营商类型。

3. 单击**立即购买**，完成付款。
4. 返回**实例管理**页面，选择对应的地域即可看到新建的实例。

<input type="checkbox"/>	ID/Name ↑	Monitor...	Status	VIP	Network...	Network	Health Status	Creation Time
<input type="checkbox"/>	lb- clb-test		Normal	1	Public Network		Health check not enabled (Configuration)	2019-07-05

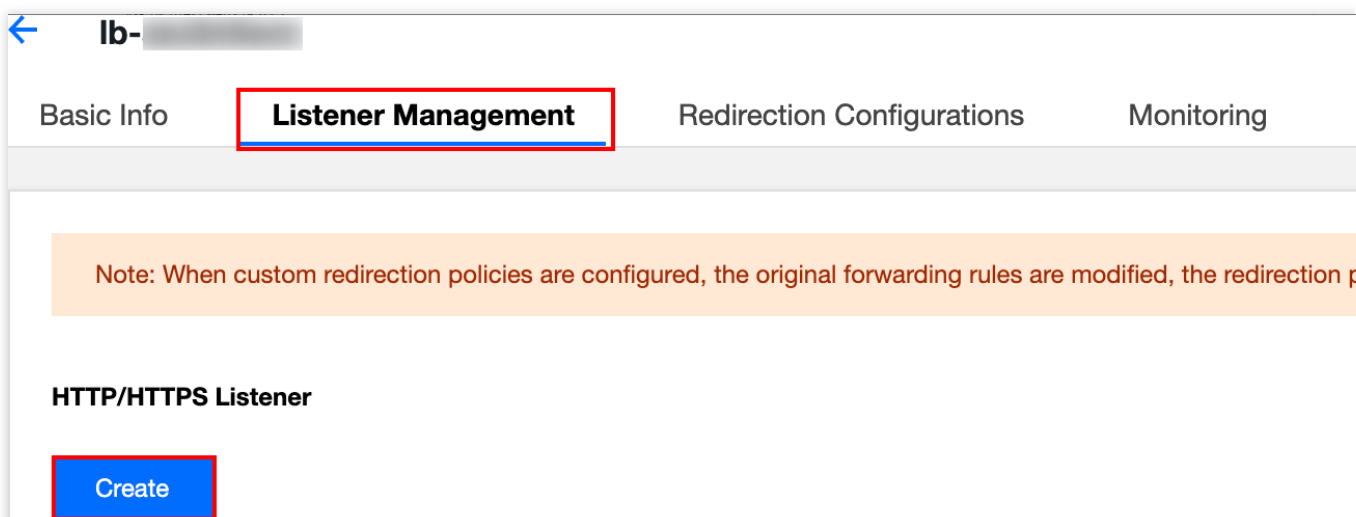
步骤二：配置负载均衡监听器

负载均衡监听器通过指定协议及端口来负责实际转发。本文以负载均衡转发客户端的 HTTP 请求配置为例。

配置 HTTP 监听协议和端口

当客户端发起请求时，负载均衡会根据监听的前端协议与端口接收请求并向后端服务器转发请求。

1. 登录 [负载均衡控制台](#)。
2. 在“实例管理”页面中，找到目标负载均衡实例，单击**配置监听器**。
3. 在“监听器管理”页签中的“HTTP/HTTPS 监听器”区域下，单击**新建**。



4. 在“创建监听器”对话框中，配置以下内容，配置完成后，单击**提交**。

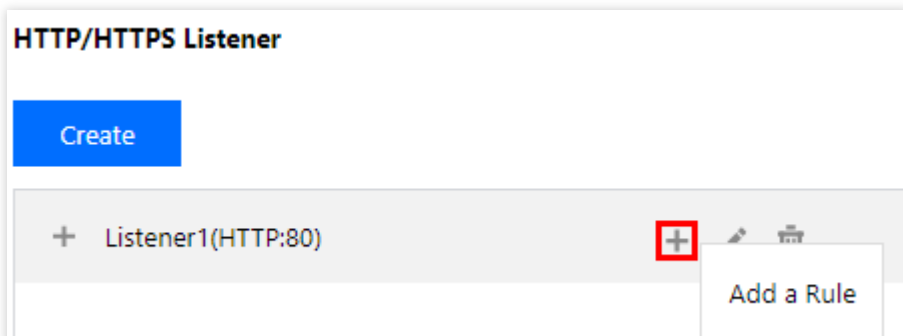
监听器名称，名称可输入60个字符，允许英文字母、数字、中文字符、“-”、“_”、“.”。

监听协议端口。例如 HTTP：80。

配置监听器的转发规则

当客户端请求时，负载均衡会根据配置的监听器转发规则进行请求转发。

1. 在“监听器管理”页签中，选中刚才新建的监听器，单击**+添加规则**。



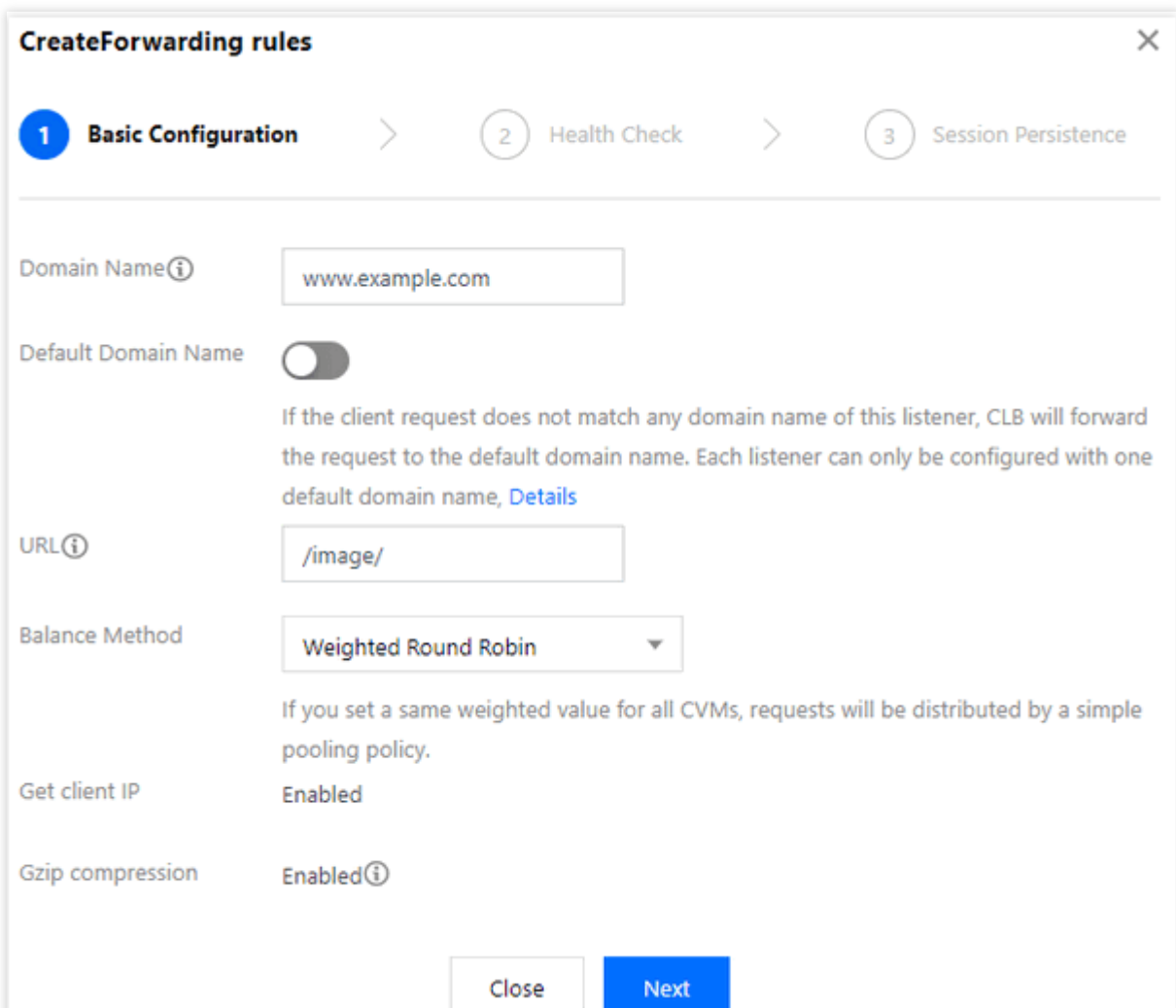
2. 在“创建转发规则”对话框的“基本配置”页签中，配置域名、URL 路径和均衡方式。配置完成后，单击**下一步**。

域名：您的后端服务所使用的域名，本例使用 `www.example.com`。

默认域名：当客户端请求没有匹配本监听器的任何域名时，CLB 会将请求转发给默认域名（default server），每个监听器只能配置一个默认域名。若该监听器没有配置默认域名时，CLB 会将请求转发给第一个域名。本例不配置。

URL 路径：您的后端服务的访问路径，本例使用 `/image/`。

均衡方式选择“加权轮询”。详情请参见 [均衡方式](#)。



3. 在“健康检查”页签，开启健康检查，检查域名和检查路径使用默认的转发域名和转发路径，完成后单击**下一步**。

CreateForwarding rule

Basic configuration

2 Health check

3 Session persistence

Health check

Detect and remove abnormal backend servers

Health check source IP

100.64 range (Recommended)

CLB VIP

You don't need to allow this IP segment in the security group of the backend server. However if the backend server has other security policies (such as iptables), you need to allow the health check source IP. If not, the health check throws an exception.

Check method

TCP

HTTP

Check domain

It defaults to the forwarding d

Path

Root directory of CVM

/

Show advanced options

Back

Next

4. 在“会话保持”页签，关闭会话保持，单击**提交**。

有关负载均衡监听器的更多内容，请参考 [负载均衡监听器概述](#)。

说明：

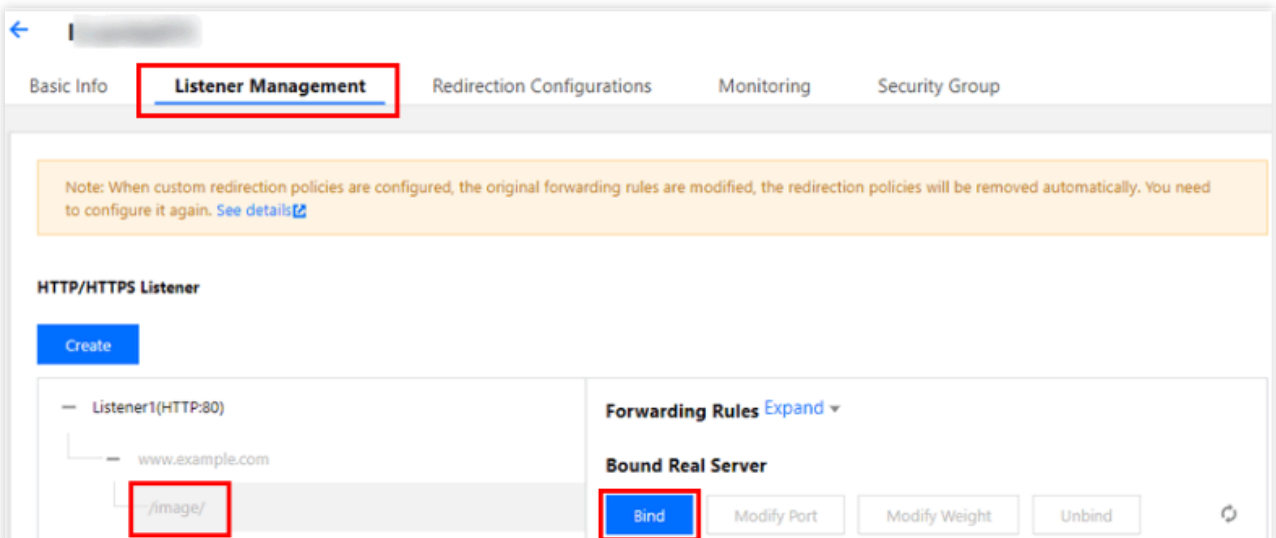
转发规则：一个监听器可配置多个域名，一个域名可配置多条 URL 路径，选中监听器或域名，单击+，即可创建新的规则。

会话保持：若关闭会话保持功能，选择轮询的方式进行调度，则请求会被依次分配到不同后端服务器上；若开启会话保持功能，或关闭会话保持功能但选择 ip_hash 的调度方式，则请求会被持续分配到同一台后端服务器上。

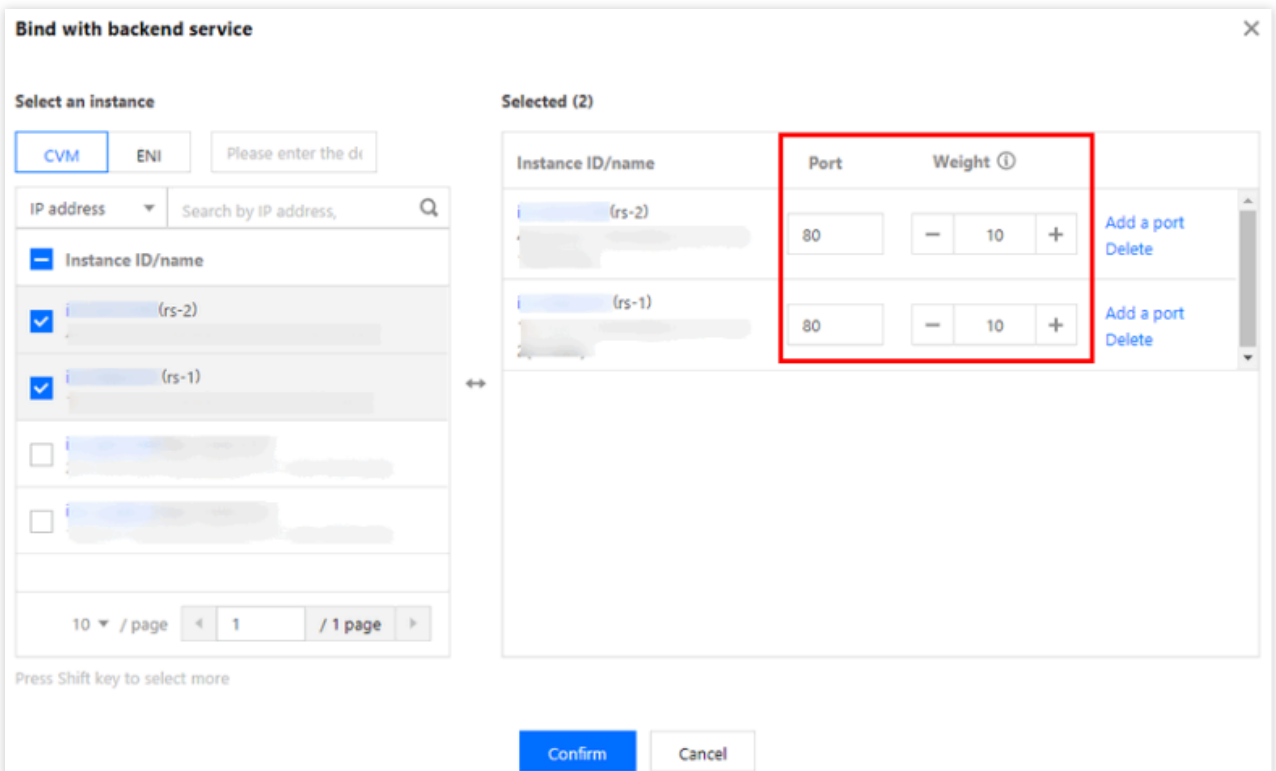
为监听器绑定后端云服务器

当客户端请求时，负载均衡将请求转发到监听器绑定的云服务器上进行处理。

1. 在“监听器管理”页面，单击 + 展开刚才创建的监听器，选中 URL 路径，在右侧“转发规则详情”区域单击**绑定**。



2. 在“绑定后端服务”对话框中，选择绑定实例类型为“云服务器”，再选择与 CLB 实例同地域下的云服务器实例 rs-1 和 rs-2 ，设置云服务器端口均为“80”，云服务器权重均为默认值“10”，然后单击**确认**。



3. 返回“转发规则详情”区域，可以查看绑定的云服务器和其健康检查状态，当端口健康状态为“健康”时表示云服务器可以正常处理负载均衡转发的请求。

说明：
一条转发规则（监听协议 + 端口 + 域名 + URL 路径）可以绑定同一台云服务器的多个端口。如用户在 rs-1 的 80 和 81 端口部署了一样的服务，则 CLB 支持示例中的转发规则同时绑定 rs-1 的 80 和 81 端口，两个端口都会接收到 CLB 转发的请求。

步骤三：配置安全组

创建完负载均衡后，您可以配置负载均衡的安全组来隔离公网流量，详情请参考 [配置安全组](#)。

安全组配置完成后，您可以选择开启或关闭安全组默认放通，不同选择配置如下所示。

方法一：开启安全组默认放通

具体操作请参考 [配置安全组默认放通](#)。

方法二：在 CVM 安全组上放通客户端 IP

具体操作请参考 [配置安全组默认放通](#)。

步骤四：验证负载均衡服务

配置完成负载均衡后，可以验证负载均衡是否生效，即验证通过一个负载均衡实例下不同的**域名+URL** 访问不同的后端云服务器，也可验证**内容路由（content-based routing）** 的功能。

方法一：配置 hosts 将域名指向 CLB

1. 在 Windows 系统中，进入 `C:\Windows\System32\drivers\etc` 目录，修改 hosts 文件，把域名映射到 CLB 实例的 VIP 上。

```
# localhost name resolution is handled within DNS itself.
# 127.0.0.1      localhost
# ::1           localhost
192.168.1.100    www.example.com
```

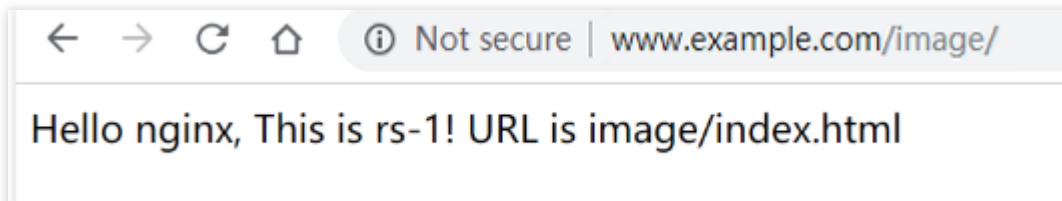
2. 为了验证 hosts 是否配置成功，可以运行 cmd 命令行工具，用 `ping` 命令探测一下该域名是否成功绑定了 VIP，如有数据包，则证明绑定成功。

```
C:\Users\Administrator>ping www.example.com

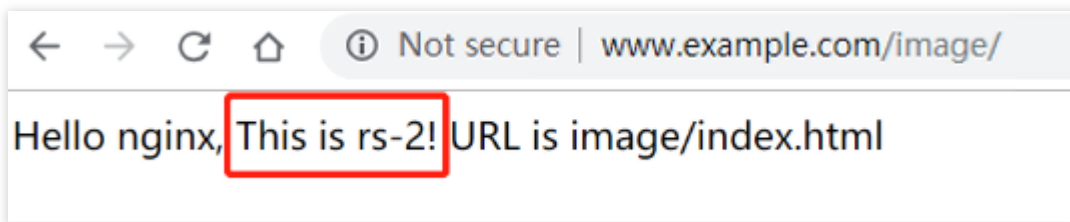
Pinging www.example.com [192.168.1.100] with 32 bytes of data:
Reply from 192.168.1.100: bytes=32 time=159ms TTL=48
Reply from 192.168.1.100: bytes=32 time=149ms TTL=48
Reply from 192.168.1.100: bytes=32 time=158ms TTL=48
Reply from 192.168.1.100: bytes=32 time=150ms TTL=48

Ping statistics for 192.168.1.100:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 149ms, Maximum = 159ms, Average = 154ms
```

3. 在浏览器中输入访问路径 `http://www.example.com/image/`，测试负载均衡服务。若如下图所示，则表示本次请求被 CLB 转发到了 rs-1 这台 CVM 上，CVM 正常处理请求并返回页面。



4. 此监听器的均衡方式是“加权轮询”，且两台 CVM 的权重都是“10”。刷新浏览器，再次发送请求，若如下图所示，则表示本次请求被 CLB 转发到了 rs-2 这台 CVM 上。



注意：

`image/` 后 `/` 必须保留，代表 `image` 是默认的目录，而不是名为 `image` 的文件。

方法二：配置 DNS 解析 DNSPod 将域名指向 CLB

1. 打开 [腾讯云域名注册页面](#) 进行域名查询和注册。本例以 `example.com` 为例。
2. 登录 [DNS 解析 DNSPod 控制台](#)，在“域名解析列表”页面，单击目标域名右侧“操作”列的解析。
3. 在“记录管理”页签，单击添加记录为域名添加 A 记录，在添加记录区域，填写以下参数：
主机记录：即域名前缀。本例以解析所有前缀为例，设为 `*.example.com`。
记录类型：`A 记录`。
线路类型：默认。
记录值：单击[关联云资源](#)，在“关联云资源”对话框勾选刚才创建的 CLB 实例。
TTL：设置为默认值“600s”。
4. 添加完成后，单击保存。
5. 在添加完解析记录十分钟左右后，在浏览器中输入绑定后的 CNAME 域名（如本例中的 `www.example.com`），若正常显示页面，则表示负载均衡生效。

配置重定向功能（可选）

重定向配置分为手动重定向和自动重定向，详情请参见 [重定向配置](#)：

自动重定向（强制 HTTPS）：PC 端、手机浏览器等以 HTTP 请求访问 Web 服务，在 CLB 代理后，返回 HTTPS 响应。默认强制浏览器以 HTTPS 访问网页。

手动重定向：当出现 Web 业务需要临时下线（如电商售罄、页面维护和更新升级时）需要重定向能力。如果不做重定向，用户的收藏和搜索引擎数据库中的旧地址只能让访客得到一个404或503错误信息页面降低了用户体验度，导致访问流量白白流失，且该页面积累的搜索引擎评分也会无效。

相关操作

[Linux（CentOS）下部署 Java Web](#)

[Windows 下安装配置 PHP](#)

IPv6 负载均衡快速入门

最近更新时间：2024-01-04 17:03:35

腾讯云负载均衡支持 IPv4、IPv6 和 IPv6 NAT64 三个 IP 版本，IPv6 负载均衡支持 TCP/UDP/TCP SSL/HTTP/HTTPS 协议，提供基于域名和 URL 路径的灵活转发能力。本文将引导您如何快速使用 IPv6 负载均衡。

说明：

IPv6 负载均衡处于内测阶段，如需使用，请提交 [工单申请](#)。

前提条件

1. 负载均衡只负责转发流量，不具备处理请求的能力。因此，您需要首先搭建处理用户请求的云服务器实例，并完成云服务器的 IPv6 配置。
2. 本文以 HTTP 转发为例，且 Linux 云服务器上必须部署相应的 Web 服务器（如 Apache、Nginx、IIS 等），同时 Web 服务使用的端口需要监听 IPv6。

使用说明

目前仅支持如下地域开通 IPv6 负载均衡：广州、上海、南京、北京、成都、重庆、中国香港、新加坡、弗吉尼亚、圣保罗。

IPv6 负载均衡不支持传统型负载均衡。

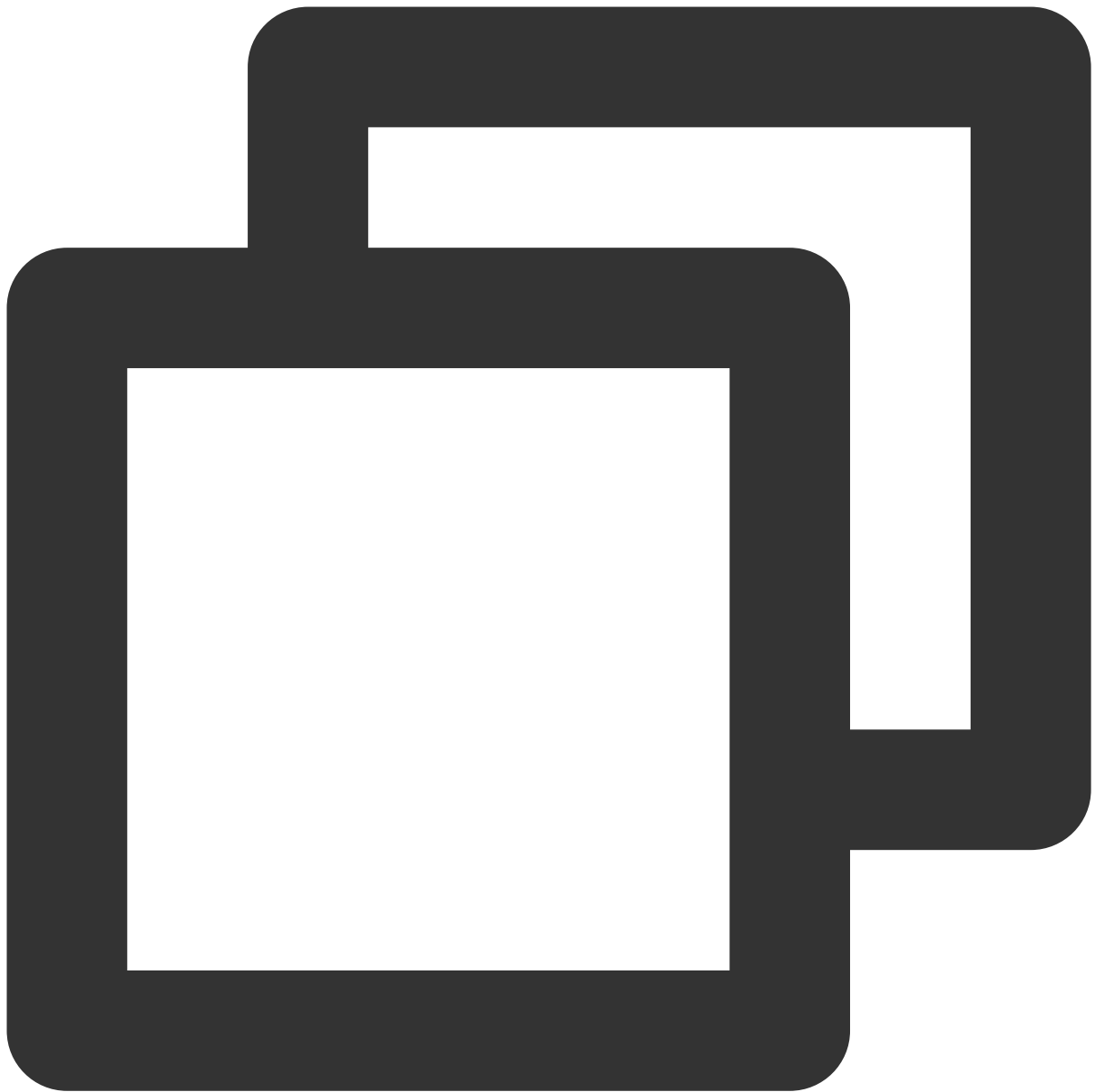
IPv6 负载均衡支持获取客户端 IPv6 源地址。四层 IPv6 负载均衡支持直接获取客户端 IPv6 源地址，七层 IPv6 负载均衡支持通过 HTTP 的 X-Forwarded-For 头域获取客户端 IPv6 源地址。

当前 IPv6 负载均衡是纯公网负载均衡，相同 VPC 的客户端无法通过内网访问该 IPv6 负载均衡。

互联网 IPv6 网络大环境还处于建设初期，如出现线路访问不通的情况，请 [提交工单](#) 反馈，另外在内测期间，不提供 SLA 保障。

步骤1：搭建云服务器并配置 IPv6

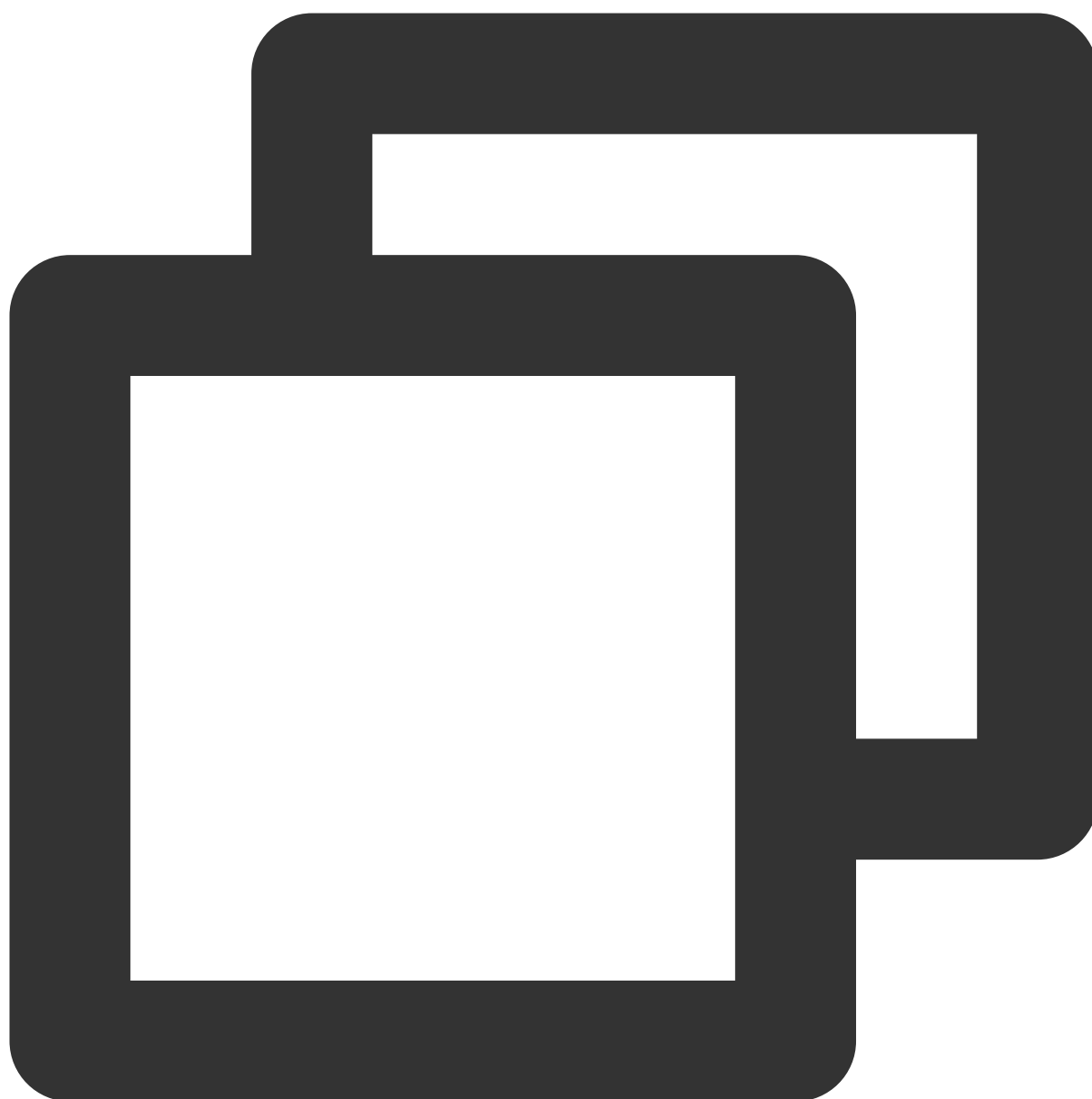
1. 进入 [云服务器控制台](#)，登录云服务器完成 IPv6 的基础配置。
2. 在云服务器中，依次执行如下命令，部署并重启 Nginx 服务。



```
yum install nginx  
service nginx restart
```

3. 查看部署在云服务器上的 Nginx 服务是否已经监听 IPv6。

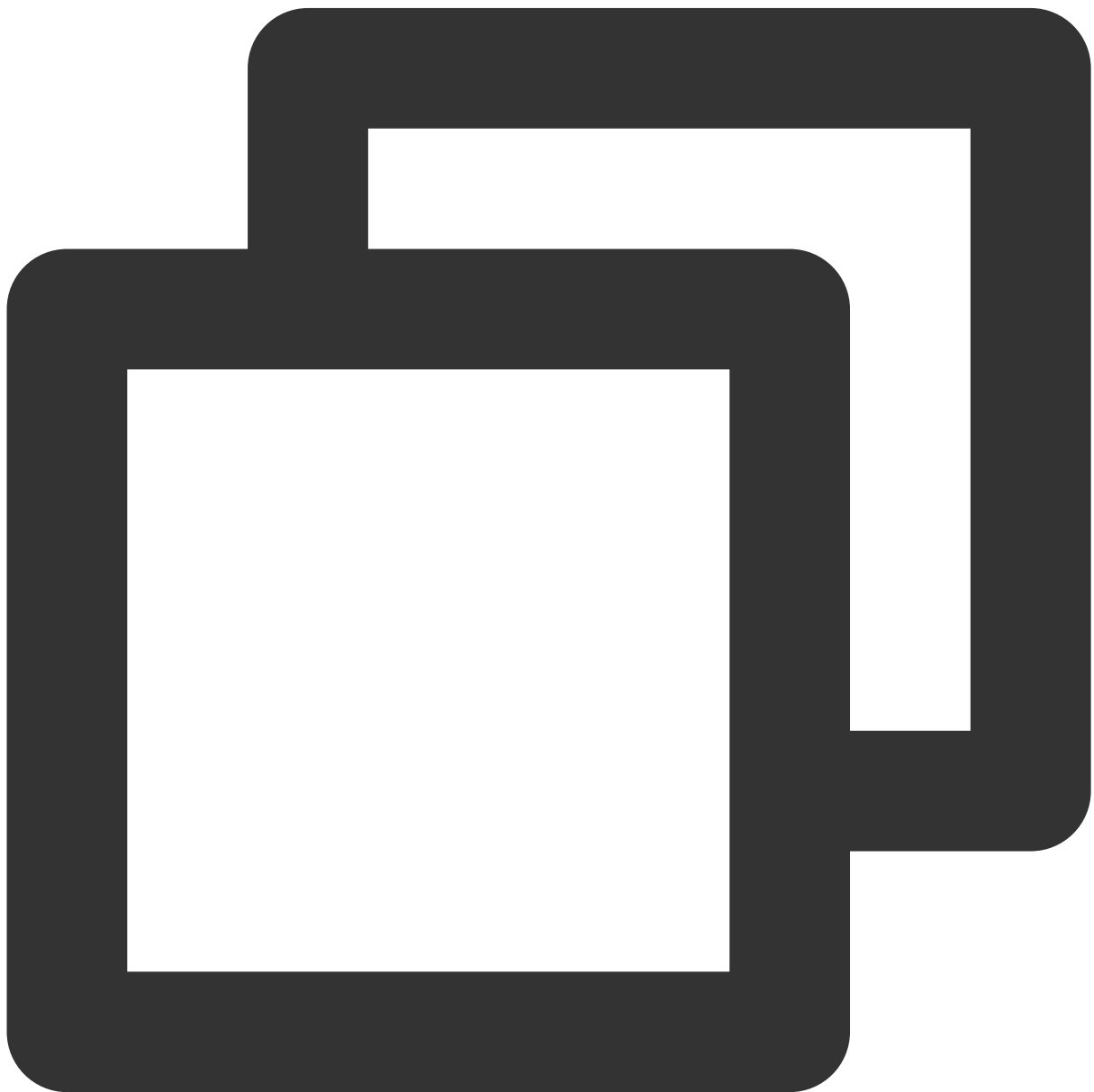
3.1 执行如下命令进行查看。



```
netstat -tupln
```

```
[root@VM_0_14_centos ~]# netstat -tupln
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:80             0.0.0.0:*               LISTEN      4314/nginx: master
tcp        0      0 0.0.0.0:22             0.0.0.0:*               LISTEN      3175/sshd
tcp6       0      0 :::80                  :::*                     LISTEN      4314/nginx: master
udp        0      0 0.0.0.0:68             0.0.0.0:*               *          2890/ncclient
udp        0      0 10.24.0.14:123         0.0.0.0:*               *          3369/ntpd
udp        0      0 127.0.0.1:123          0.0.0.0:*               *          3369/ntpd
udp        0      0 0.0.0.0:56713          0.0.0.0:*               *          4333/ntpd
udp6       0      0 fe80::5054:ff:fe3d::546 :::*                     *          4119/dhclient
udp6       0      0 2402:4e00:1400:1217:123 :::*                     *          3369/ntpd
udp6       0      0 fe80::5054:ff:fe3d::123 :::*                     *          3369/ntpd
udp6       0      0 :::1:123               :::*                     *          3369/ntpd
```

3.2 执行如下命令，打开 Nginx 配置文件进行查看。



```
vim /etc/nginx/nginx.conf
```

```
# Load modular configuration files from the /etc/nginx/conf.d directory.
# See http://nginx.org/en/docs/nginx_core_module.html#include
# for more information.
include /etc/nginx/conf.d/*.conf;

server {
    listen      80 default server;
    listen      [::]:80 default_server;
    server_name _;
    root        /usr/share/nginx/html;

    # Load configuration files for the default server block.
    include /etc/nginx/default.d/*.conf;

    location / {
    }

    error_page 404 /404.html;
        location = /40x.html {
    }

    error_page 500 502 503 504 /50x.html;
        location = /50x.html {
    }
}

# Settings for a TLS enabled server.
#
#
server {
#    listen      443 ssl http2 default server;
#    listen      [::]:443 ssl http2 default_server;
#    server_name _;
#    root        /usr/share/nginx/html;
#
#    ssl_certificate "/etc/pki/nginx/server.crt";
#    ssl_certificate_key "/etc/pki/nginx/private/server.key";
#    ssl_session_cache shared:SSL:1m;
}
```

步骤2：创建 IPv6 负载均衡实例

1. 登录腾讯云官网，进入 [负载均衡购买页](#)。
2. 请正确选择如下参数：
计费模式：仅支持按量计费。
地域：选择目标地域。
IP 版本：IPv6。
运营商类型：BGP（多线）。
网络：请务必选择已获取 IPv6 CIDR 的私有网络和子网。
3. 在购买页选择各项配置后，单击**立即购买**。

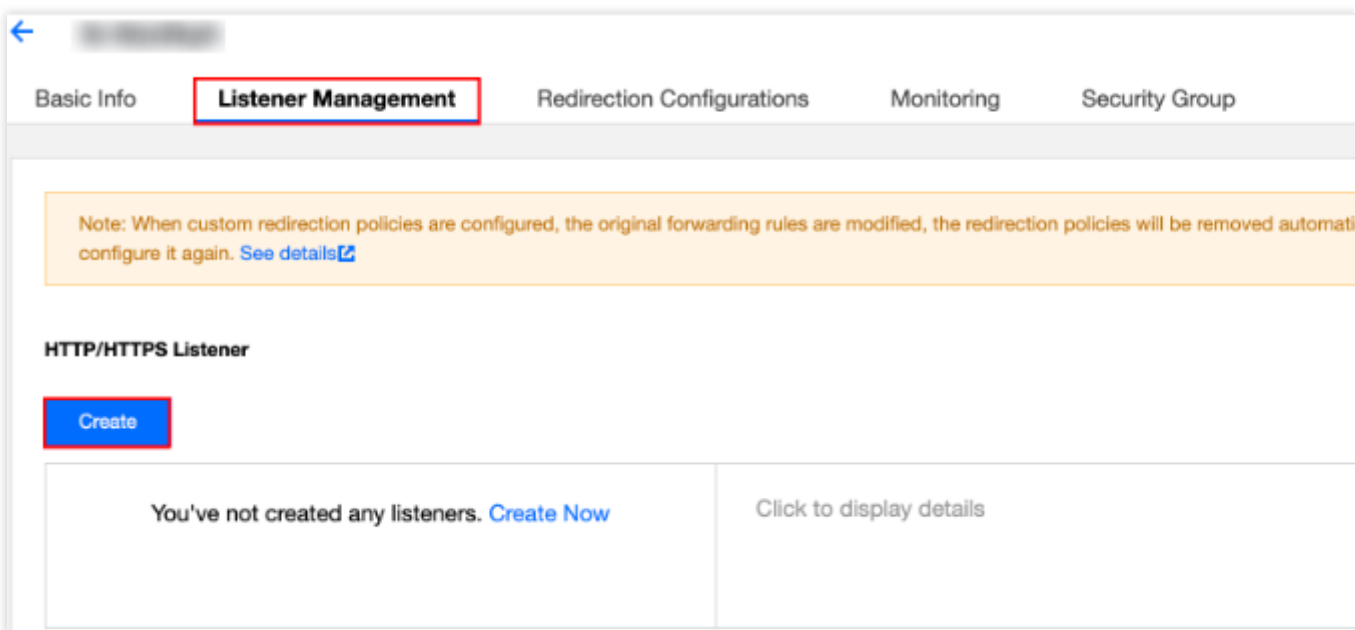
4. 在“CLB 实例列表”页，选择对应的地域即可看到新建的实例。

<input type="checkbox"/>	ID/Name	Monitori...	Status	VIP	Availability ...	Network... Y	Network	ISP	Health Status	Billing Mode	Bandwidth ...	Project Y	Tag
<input type="checkbox"/>			Normal		Qingyuan Zone 1	Public Network		BGP	Health check not enabled (Configuration)	Pay-as-you-go — bandwidth Created at 2020-08-06 11:32	1Mbps	Default Project	-

步骤3：创建 IPv6 负载均衡监听器

配置 HTTP 监听协议和端口

1. 登录 [负载均衡控制台](#)。
2. 在**CLB 实例列表**中，找到已创建的负载均衡实例，单击实例 ID，进入负载均衡详情页。
3. 在**基本信息**模块，可以单击名称后的修改图标修改实例名称。
4. 在**监听器管理**中的 **HTTP/HTTPS 监听器**下，单击**新建**，新建负载均衡监听器。



5. 在弹出框中，配置以下内容：

名称自定义为“IPv6test”。

监听协议端口为 HTTP：80。

6. 单击**提交**，创建负载均衡监听器。

配置监听器的转发规则

1. 在**监听器管理**中，选中刚才新建的监听器 IPv6test，单击 $+$ ，开始添加规则。

2. 在弹出框中，配置域名、URL 路径和均衡方式，单击**下一步**。

域名：您的后端服务所使用的域名，本例使用 `www.xxxxxxxxxxtest.com`。域名支持通配符，详情请参见 [七层转发域名和 URL 规则说明](#)。

URL 路径：您的后端服务的访问路径，本例使用 `/`。

均衡方式选择**加权轮询**。

Create Forwarding rule

1 Basic configuration

2 Health check

3 Session persistence

Domain name ⓘ

www.qcloudipv6test.com

Default domain name

Enable

If a client request does not match any domain names of this listener, the CLB instance will forward the request to the default domain name (Default Server). Each listener only can configure one listener and must configure one. [Details](#)

URL ⓘ

/

Balance method ⓘ

Weighted round robin ▼

WRR scheduling is based on the number of new connections, where real servers with higher weight have more polls

Get client IP

Enabled

Gzip compression

Enabled ⓘ

Target group ⓘ

☐

Close

Next

3. 配置健康检查：开启健康检查，检查域名使用默认的转发域名和转发路径，单击**下一步**。

CreateForwarding rules

Basic Configuration > **2 Health Check** > 3 Session

Health Check ⓘ



Check Domain ⓘ

It defaults to the forwarded d

Path ⓘ

Root Directory of CVM ▼

/

[Show advanced options](#) ▼

Back

Next

4. 会话保持：开启会话保持并配置保持时间，单击提交。

CreateListener

Basic Configuration > Health Check > **3 Session Persistence**

Session Persistence ⓘ



Hold Time ⓘ



30 Seconds

3600 Seconds

Session persistence based on the source IP

54

Back

Submit

有关负载均衡监听器的更多内容，请参见 [负载均衡监听器概述](#)。

说明：

一个监听器（即监听协议：端口）可以配置多个域名，一个域名下可以配置多条 URL 路径，选中监听器或域名，单击+即可创建新的规则。

会话保持：如果用户关闭会话保持功能，选择轮询的方式进行调度，则请求依次分配到不同后端服务器上；如果用户开启会话保持功能，或关闭会话保持功能但选择 ip_hash 的调度方式，则请求持续分配到同一台后端服务器上去。

绑定云服务器

说明：

绑定云服务器前，请确定该云服务器已获取 IPv6 地址。

1. 在**监听器管理**页面，选中并展开刚才创建的监听器，选中域名、选中 URL 路径，在右侧即可看到该 URL 路径绑定的云服务器 IPv6 信息，单击**绑定**。
2. 在弹框中，选择云服务器，并设置云服务器的 Nginx 服务默认端口为80，设置权重（默认值10），单击**确定**。

Bind with backend service

Select an instance

CVM

ENI

Please enter the d

IP address

Search by IP address,

☒ Instance ID/name

☒

☒

10 / page

< 1

/ 1 page >

Press Shift key to select more

Selected (2)

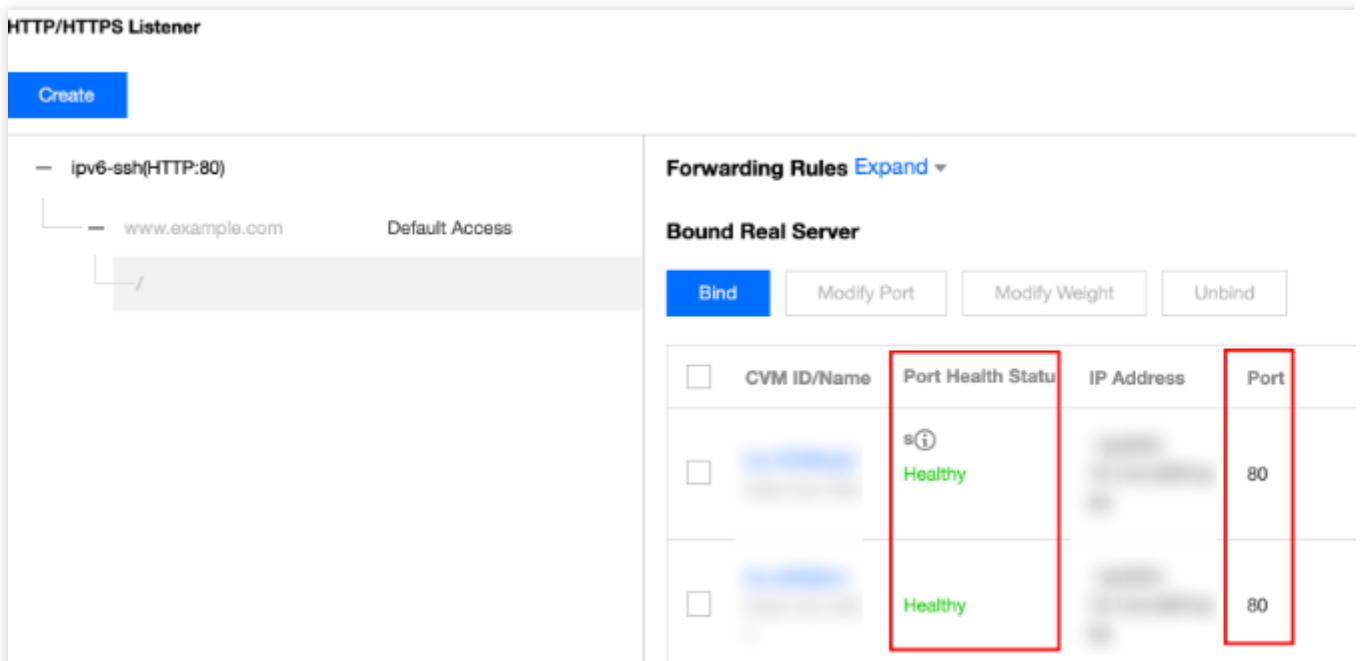
Instance ID/name	Port	Weight ⓘ
	80	- 10 +
	80	- 10 +

Confirm

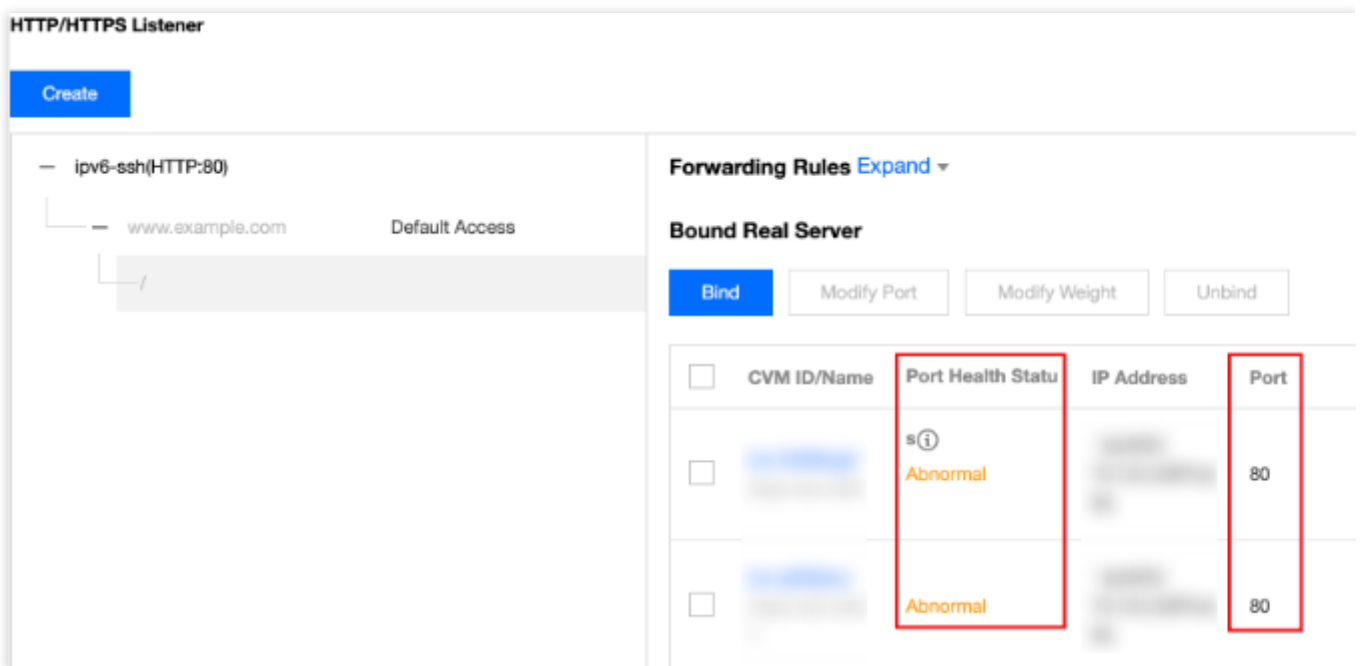
Cancel

3. 成功绑定云服务器后：

请确认端口状态是否为**健康**，如果为**健康**，请进行 [步骤4：测试 IPv6 负载均衡](#)。



如果端口状态为**异常**，请确定监听器是否绑定了正确的云服务器的 Nginx 服务端口，同时登录云服务器检查端口已经正常监听 IPv6，可参见 [步骤1中的第3步](#) 进行查看。



步骤4：测试 IPv6 负载均衡

配置完成 IPv6 负载均衡后，可以验证该架构是否生效，即验证通过一个 CLB 实例下不同的域名 + URL 访问不同的后端云服务器，也即验证内容路由（content-based routing）的功能是否可用。

使用具有 IPv6 公网能力的客户端，访问域名或者负载均衡的 IPv6 地址，如果能够正常访问云服务器的 Web 服务，则表明 IPv6 负载均衡工作正常，示例步骤如下：

1. 打开 [腾讯云域名注册页面](#) 进行域名查询和注册。本例以 `xxxxxxxxxxtest.com` 为例。

2. 登录 [DNS 解析 DNSPod 控制台](#)，单击您所购买的**域名**，在“记录管理”页面单击**添加记录**，为域名添加 AAAA 记录，输入如下内容并保存：

主机记录：即域名前缀，本例设为 `www`。

记录类型：`AAAA`记录。

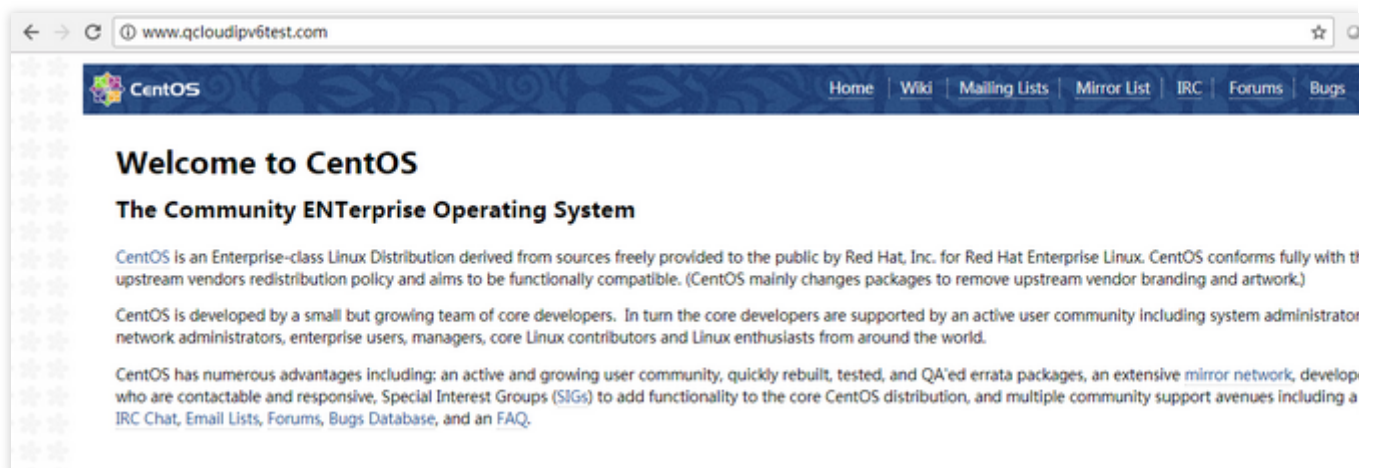
线路类型：默认。

记录值：填写负载均衡的 IPv6 地址。

TTL：设置为默认值“600s”。

3. 添加域名解析后，通过 Ping 域名进行验证。

4. 再通过浏览器访问域名来验证，如下图：



传统型负载均衡快速入门

最近更新时间：2023-03-28 17:08:18

本文将引导您如何初步使用腾讯云传统型负载均衡：创建一个名为 `clb-test` 的传统型公网负载均衡实例，并将来自客户端的请求转发到后端两台云服务器上。

前提条件

1. 负载均衡只负责转发流量，不具备处理请求的能力。因此，您需要有处理用户请求的云服务器实例。在本示例中，只要具有两台云服务器实例即可，您也可以自行规划云服务器数量。本例中已经在广州地域下创建了云服务器实例 `rs-1` 和 `rs-2`。有关如何创建云服务器实例，请参见 [购买并启动云服务器实例](#)。
2. 本文以 HTTP 转发为例，云服务器上必须部署相应的 Web 服务器，如 Apache、Nginx、IIS 等。为了验证结果，示例在 `rs-1` 上部署了 Apache 并返回一个带有“Hello Tomcat! This is rs-1!”的 HTML，在 `rs-2` 上部署了 Apache 并返回一个带有“Hello Tomcat! This is rs-2!”的 HTML。更多云服务器部署内容，请参见 [Linux（CentOS）下部署 Java Web](#) 及 [Windows 下安装配置 PHP](#)。
3. 访问云服务器的公网 IP+路径，若显示结果为您部署好的页面，则表示服务部署成功。

注意：

传统账户类型的云服务器上必须购买公网带宽，因为当前的带宽属性在 CVM 上，而非 CLB 上。若您无法确定账户类型，请参见 [账户类型](#)。

示例中后端服务器部署的服务返回值不同，实际情况下，为保持所有用户均有一致体验，后端服务器上一般是部署完全相同的服务。

购买传统型负载均衡实例

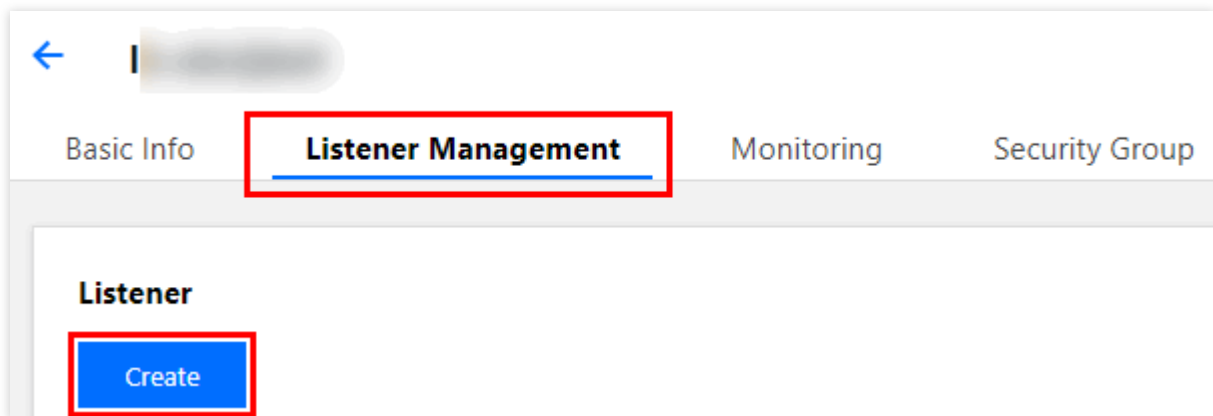
1. 登录腾讯云 [负载均衡服务购买页](#)。
2. 本例地域选择与云服务器相同的广州，实例类型选择**传统型**，网络属性选择**公网**，网络选择 **Default-VPC（默认）**，实例名称填写“clb-test”。
3. 单击**立即购买**，完成付款。有关负载均衡实例的更多内容，请参见 [产品属性选择](#)。
4. 在“CLB 实例列表”页，选择对应的地域即可看到新建的实例。

<input type="checkbox"/>	ID/Name #	Monitor...	Status	Domain Name	VIP	Network...	Network	Health Status	Creatio
<input type="checkbox"/>	clb-test	...	Normal	...	1	Public Network	...	Health check not enabled (Configuration)	2019-01-01

创建负载均衡监听器

负载均衡监听器通过指定协议及端口来负责实际转发。本文以负载均衡转发客户端的 HTTP 请求配置为例。

1. 登录 [负载均衡控制台](#)。
2. 在“CLB 实例列表”中，找到已创建的传统型的负载均衡实例 `clb-test`，单击实例 ID，进入负载均衡详情页。
3. 在“基本信息”部分，可以单击名称后的编辑修改实例名称。
4. 在“监听器管理”的**监听器**下，单击**新建**，新建负载均衡监听器。



5. 在弹出框中配置以下内容：

名称自定义为“Listener1”。

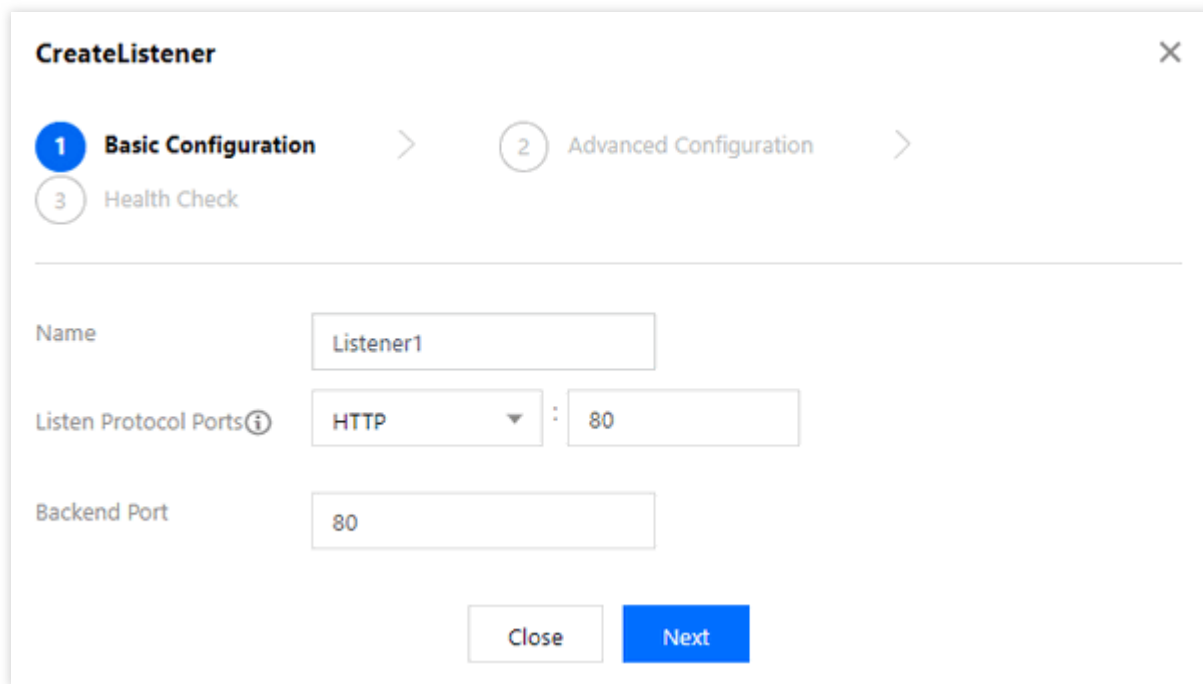
监听协议端口为 `HTTP：80`。

后端端口为 `80`。

均衡方式选择“加权轮询”。

不勾选会话保持。

开启健康检查。



CreateListener

1 Basic Configuration > 2 Advanced Configuration > 3 Health Check

Name: Listener1

Listen Protocol Ports ⓘ: HTTP : 80

Backend Port: 80

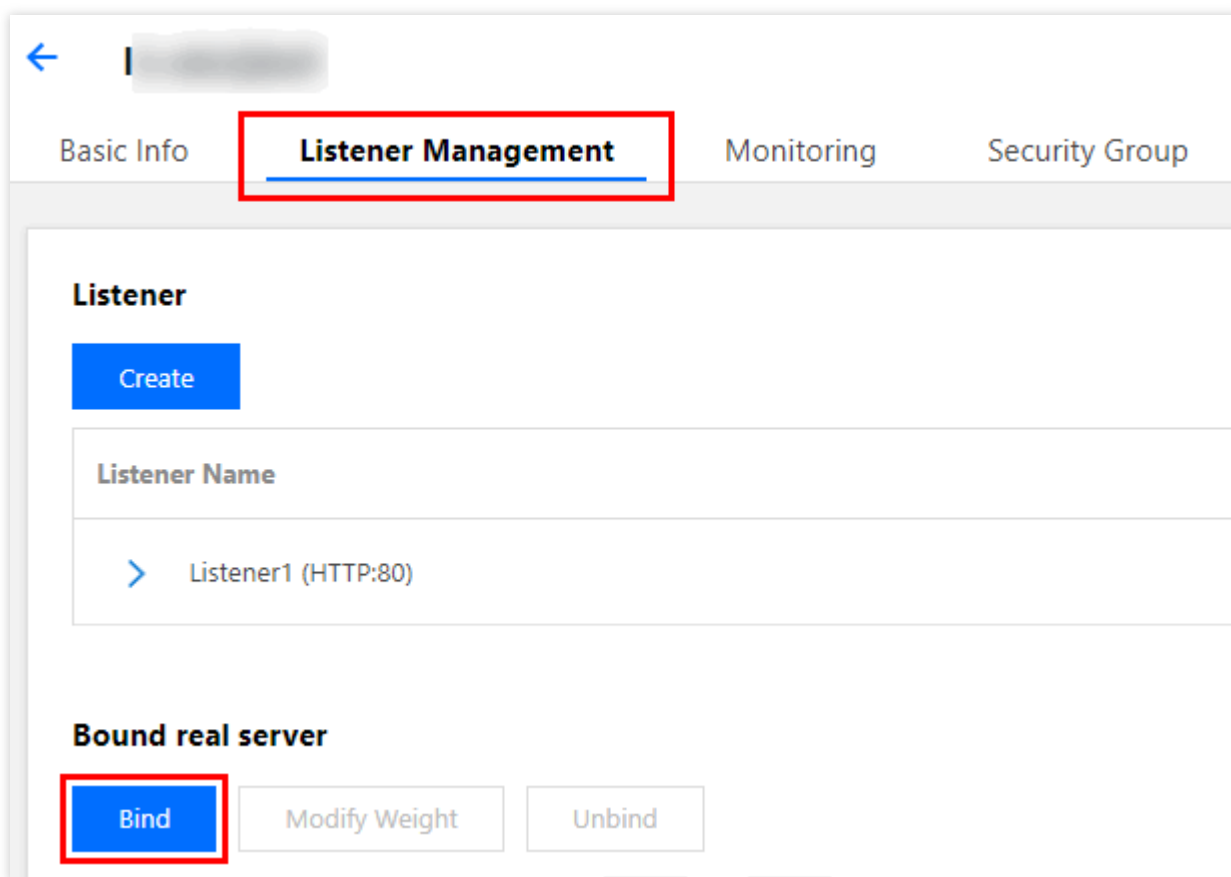
Close Next

6. 单击**完成**，完成负载均衡监听器的创建。

有关负载均衡监听器的更多内容，请参见 [负载均衡监听器概述](#)。

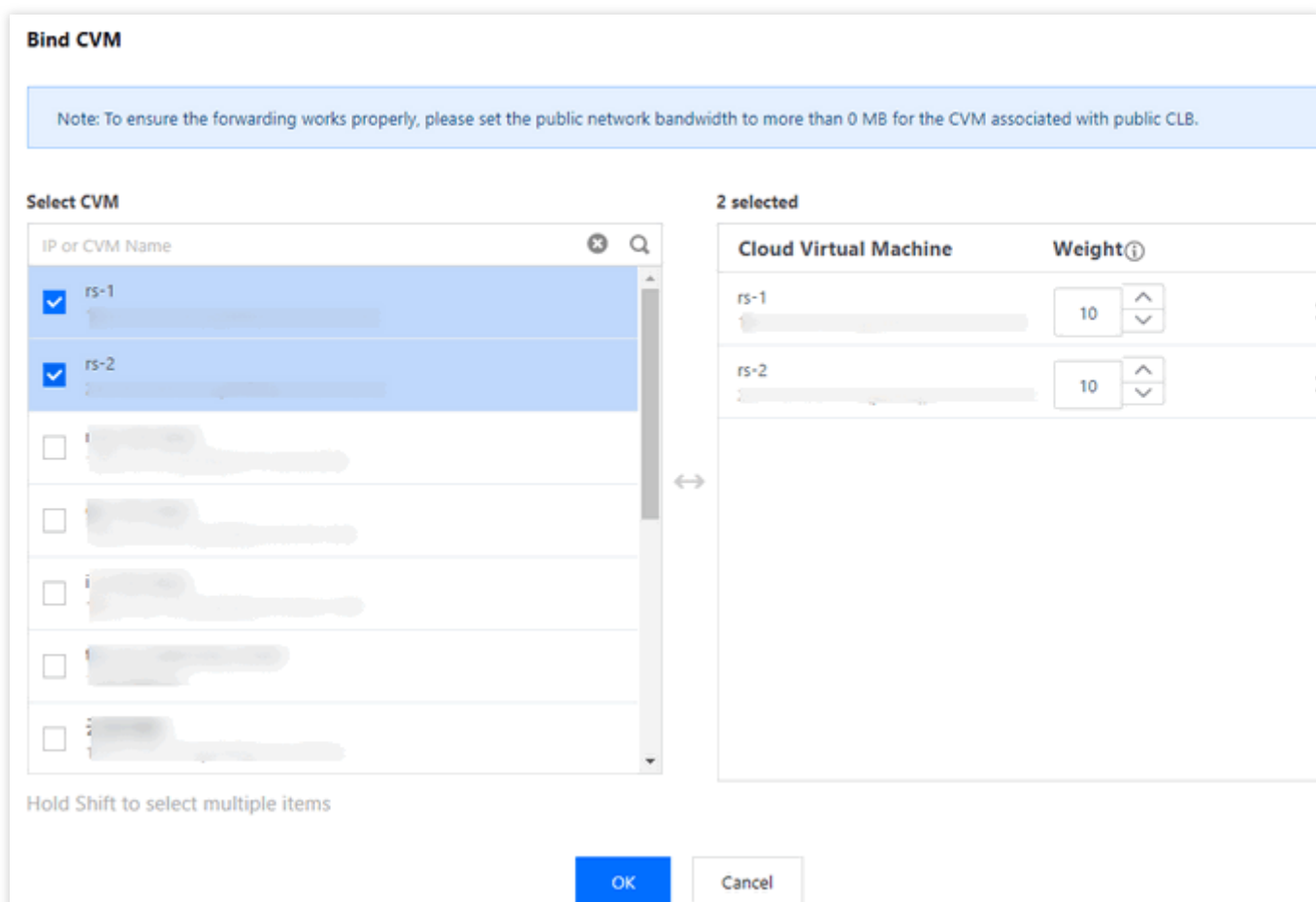
绑定后端云服务器

1. 在“CLB 实例列表”中找到刚才创建的 `clb-test`，单击其 ID，进入负载均衡详情页。
2. 在“监听器管理”的“绑定后端服务”模块中，单击**绑定**。



3. 在弹出框中，选择与 CLB 同地域下的云服务器实例 `rs-1` 和 `rs-2`，权重均设置为默认值“10”。

4. 单击**确定**，完成绑定。



5. 展开监听器 **Listener1**，可以查看后端 CVM 的健康检查状态，当状态为“健康”时表示 CVM 可以正常处理负载均衡转发的请求。

配置安全组

创建完负载均衡后，您可以配置负载均衡的安全组来隔离公网流量，详情请参考 [配置安全组](#)。

安全组配置完成后，您可以选择开启或关闭安全组默认放通，不同选择配置如下所示。

方法一：开启安全组默认放通

说明：

目前该功能处于内测阶段，如果您需要体验该功能，请提交内测申请。传统型内网负载均衡不支持安全组默认放通功能。

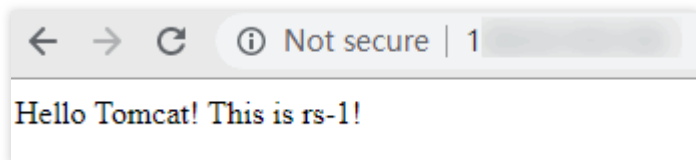
具体操作请参考 [配置安全组默认放通](#)。

方法二：在 CVM 安全组上放通客户端 IP

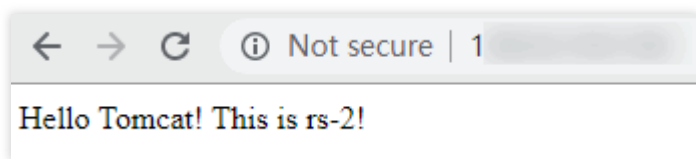
具体操作请参考 [配置安全组默认放通](#)。

验证负载均衡服务

1. 在浏览器中输入负载均衡的服务地址和端口 `http://vip:80`，测试负载均衡服务，如下图所示，表示本次请求被 CLB 转发到了 rs-1 这台 CVM 上，CVM 正常处理请求并返回。



2. 此监听器的轮询算法是“按权重轮询”，且两台 CVM 的权重都是“10”，刷新浏览器，再次发送请求，可以看到本次请求被 CLB 转发到了 rs-2 这台 CVM 上。



注意：

如果用户关闭会话保持功能，选择轮询的方式进行调度，则请求依次分配到不同后端服务器上。

如果用户开启会话保持功能，或关闭会话保持功能但选择 `ip_hash` 的调度方式，则请求持续分配到同一台后端服务器上去。

CentOS 下部署 Nginx

最近更新时间：2024-01-04 17:08:08

本文将为您详细介绍如何在 CentOS 系统下部署 Nginx 项目，适用于刚开始使用腾讯云的個人用户。

软件版本

本文在示例步骤中的软件版本如下，在实际操作时，请您以实际软件版本为准。

- 操作系统：CentOS 7.5
- Nginx 版本：Nginx 1.16.1

安装 Nginx

1. 购买完成后，在云服务器的详情页面，单击**登录**，可以直接登录云服务器，输入自己的用户名密码后，开始搭建 Nginx 环境。有关如何创建云服务器实例，请参见 [创建云服务器实例](#)。

```
# 安装 Nginx :
yum -y install nginx
# 查看 Nginx 版本
nginx -v
# 查看 Nginx 安装目录
rpm -ql nginx
# 启动 Nginx
service nginx start
```

2. 访问该云服务器的公网 IP 地址，出现如下页面则表示 Nginx 部署完成：



3. Nginx 的默认根目录 root 是 `/usr/share/nginx/html`，直接修改 `html` 下的 `index.html` 静态页面，用来标识这个页面的特殊性，相关操作如下：

i. 执行如下命令，进入html 下的 index.html 静态页面：

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

ii. 按“i”进入编辑模式，请在`标签内输入：

```
# 建议直接在<body>下方输入
Hello nginx , This is rs-1!
URL is index.html
```

```
<body>
Hello nginx , This is rs-1!
URL is index.html
```

iii. 按“Esc”，输入 `:wq` 保存编辑。

4. 负载均衡（原“应用型负载均衡”）可以根据后端服务器的路径来进行请求转发，在 `/image` 路径下部署静态页面，相关操作如下：

i. 依次执行如下命令，新建目录 `image` 并进入该目录：

```
mkdir /usr/share/nginx/html/image  
cd /usr/share/nginx/html/image
```

ii. 执行如下命令，在 `image` 目录下创建 `index.html` 静态页面：

```
vim index.html
```

iii. 按“i”进入编辑模式，在页面中输入：

```
Hello nginx , This is rs-1!  
URL is image/index.html
```

iv. 按“Esc”，输入 `:wq` 保存编辑。

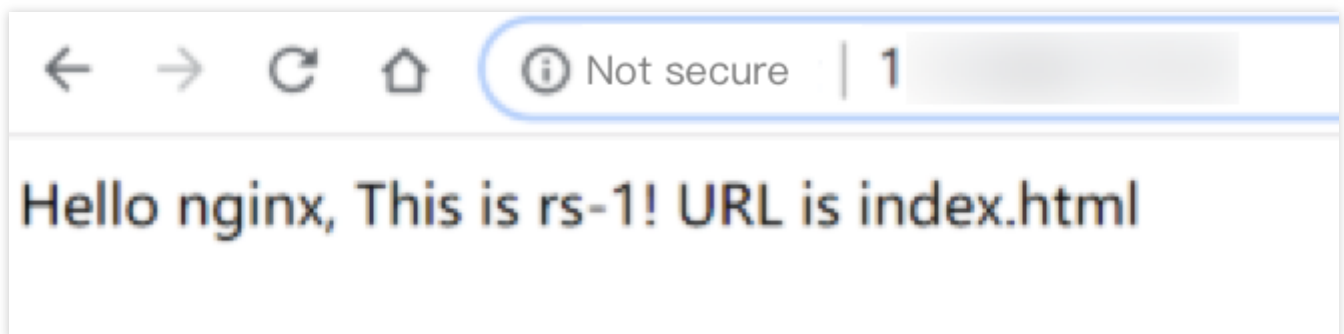
注意：

Nginx 的默认端口是 `80`，如果想修改端口请修改配置文件并重启 Nginx。

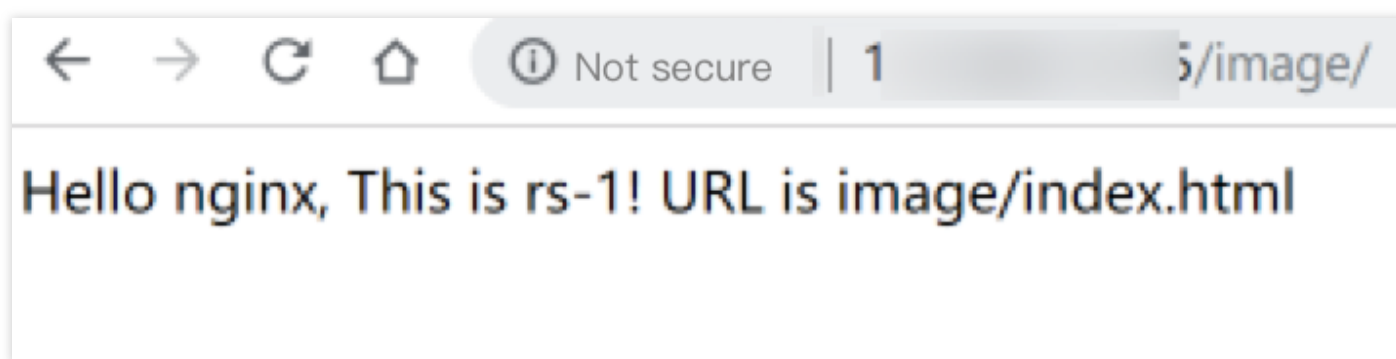
验证 Nginx 服务

访问云服务器的公网 IP + 路径，如果可以显示出已部署好的静态页面，则证明 Nginx 部署成功。

- rs-1 的 `index.html` 页面：



- rs-1 的 /image/index.html 页面：



CentOS 下部署 Java Web

最近更新时间：2024-01-04 17:08:08

本文将为您详细介绍如何在 CentOS 系统下部署 Java Web 项目，适用于刚开始使用腾讯云的个人用户。

软件版本

本文在示例步骤中的软件版本如下，在实际操作时，请您以实际软件版本为准。

操作系统：CentOS 7.5

Tomcat 版本：apache-tomcat-8.5.39

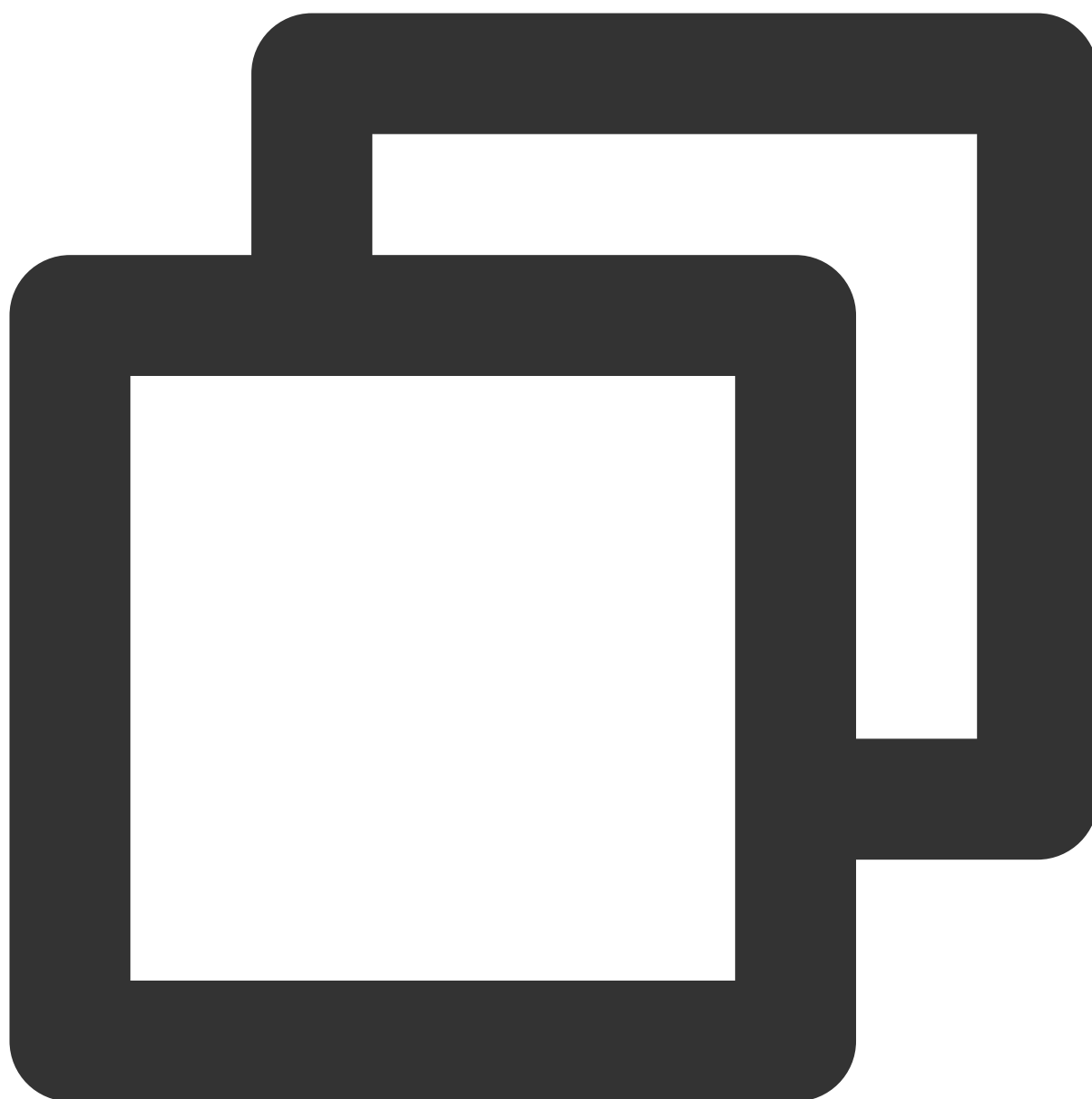
JDK 版本：JDK 1.8.0_201

安装JDK

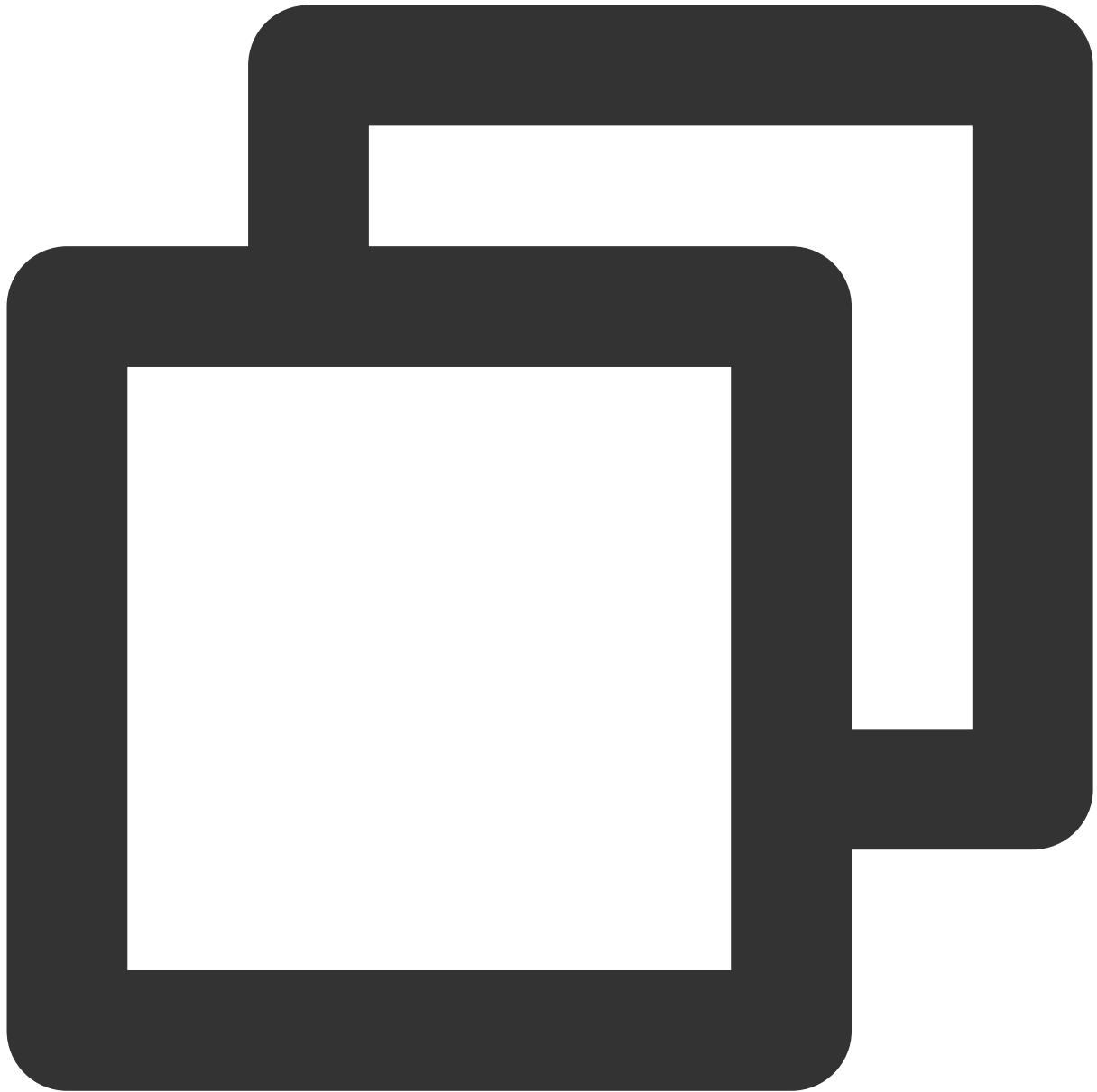
购买负载均衡服务后，在云服务器的详情页面，单击[登录](#)，可以直接登录云服务器，输入自己的用户名密码后，开始搭建 Java Web 环境。有关如何创建云服务器实例，请参见 [云服务器-创建实例](#)。

下载 JDK

输入如下命令：



```
mkdir /usr/java # 创建 java 文件夹  
cd /usr/java   # 进入 java 文件夹
```

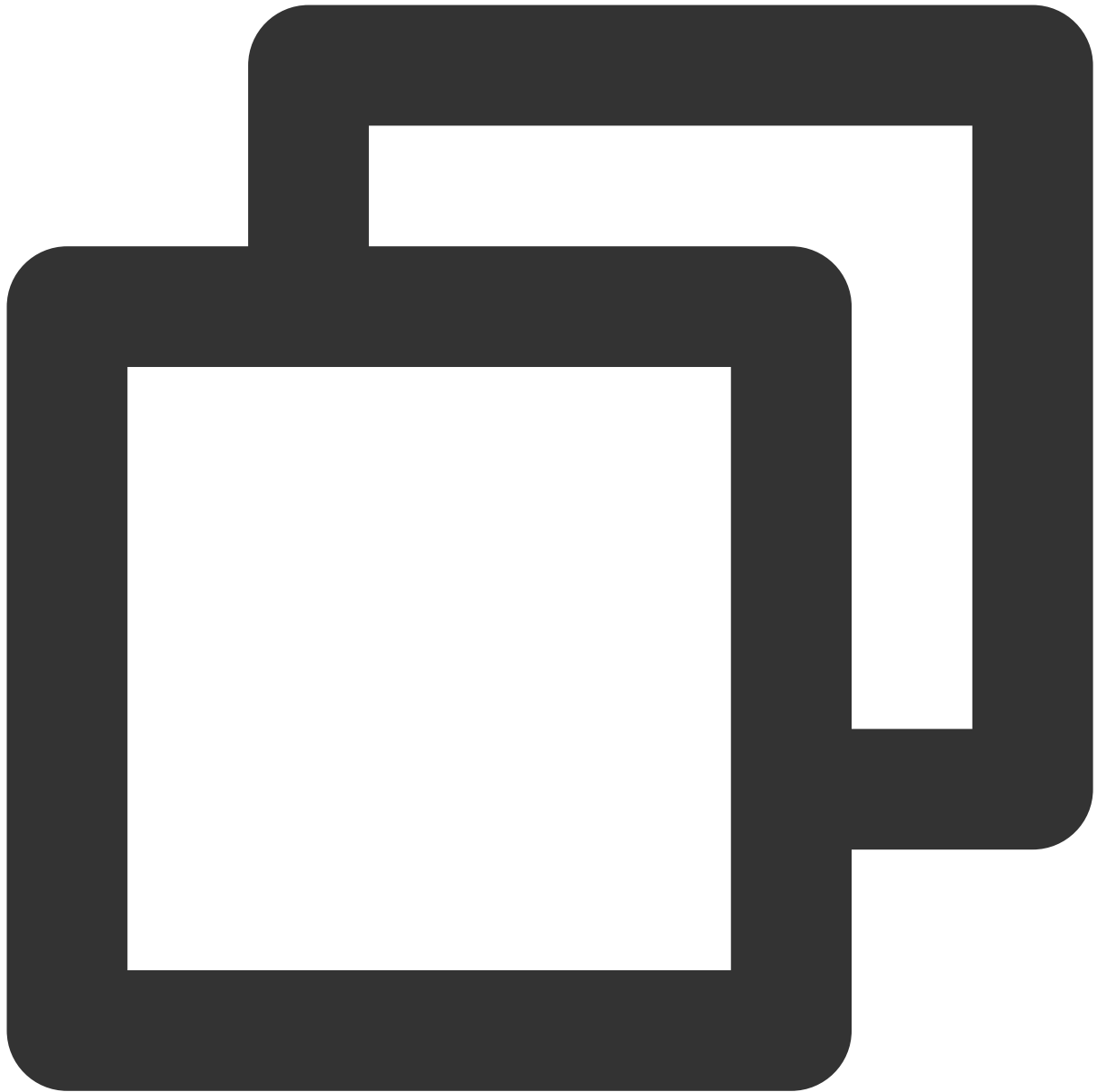


上传 JDK 安装包（推荐）

推荐您使用 WinSCP 或其他工具将 JDK 安装包上传到上述 java 文件夹下，然后解压安装包。

或者

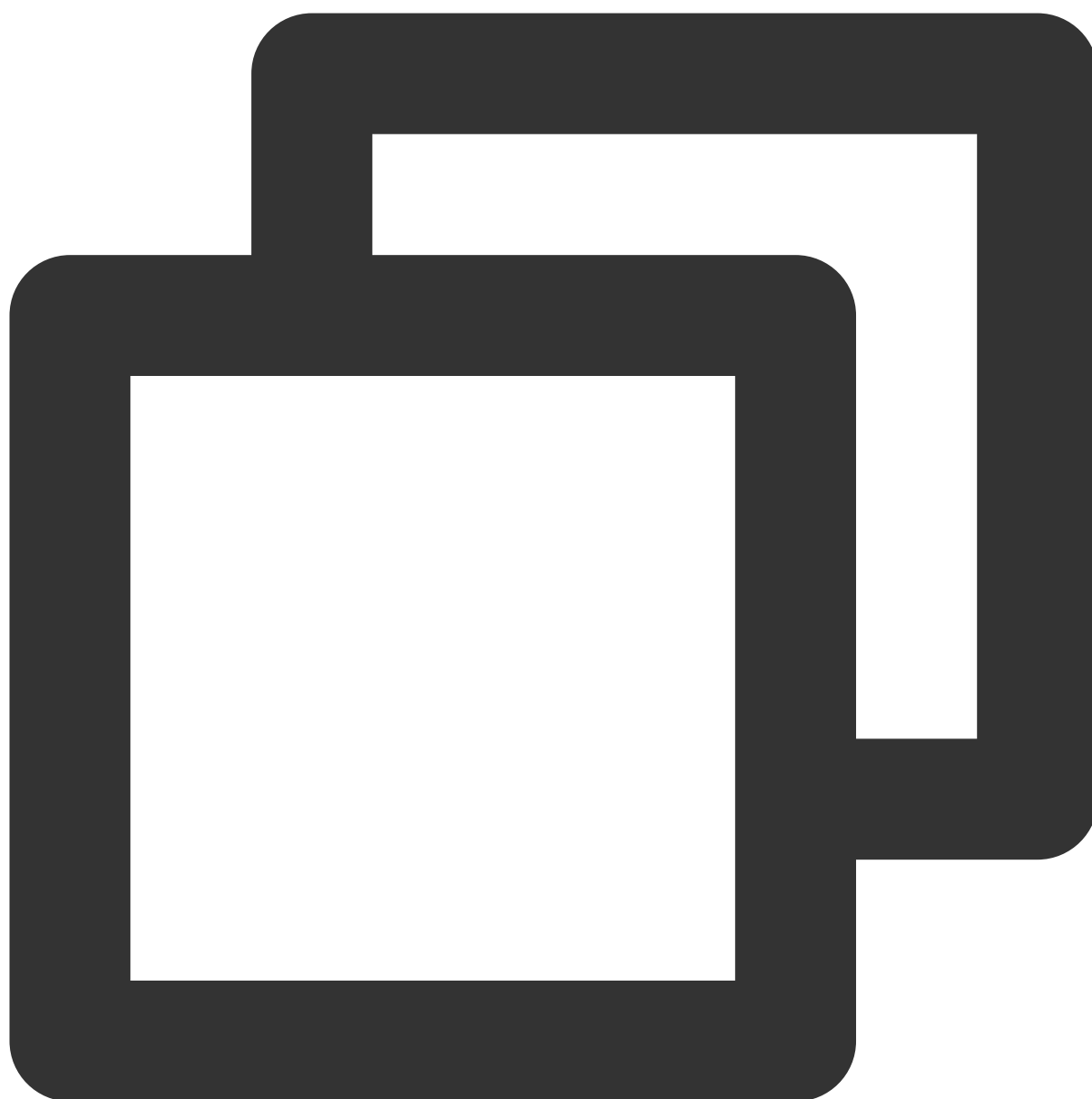
直接使用命令（推荐您使用上传 JDK 安装包的方法）：`wget` 下载链接，下载得到的压缩包无法解压，这是
`wget --no-check-certificate --no-cookies --header "Cookie: oraclelicense=accept-sec`



```
# 解压
chmod +x jdk-8u201-linux-x64.tar.gz
tar -xzf jdk-8u201-linux-x64.tar.gz
```

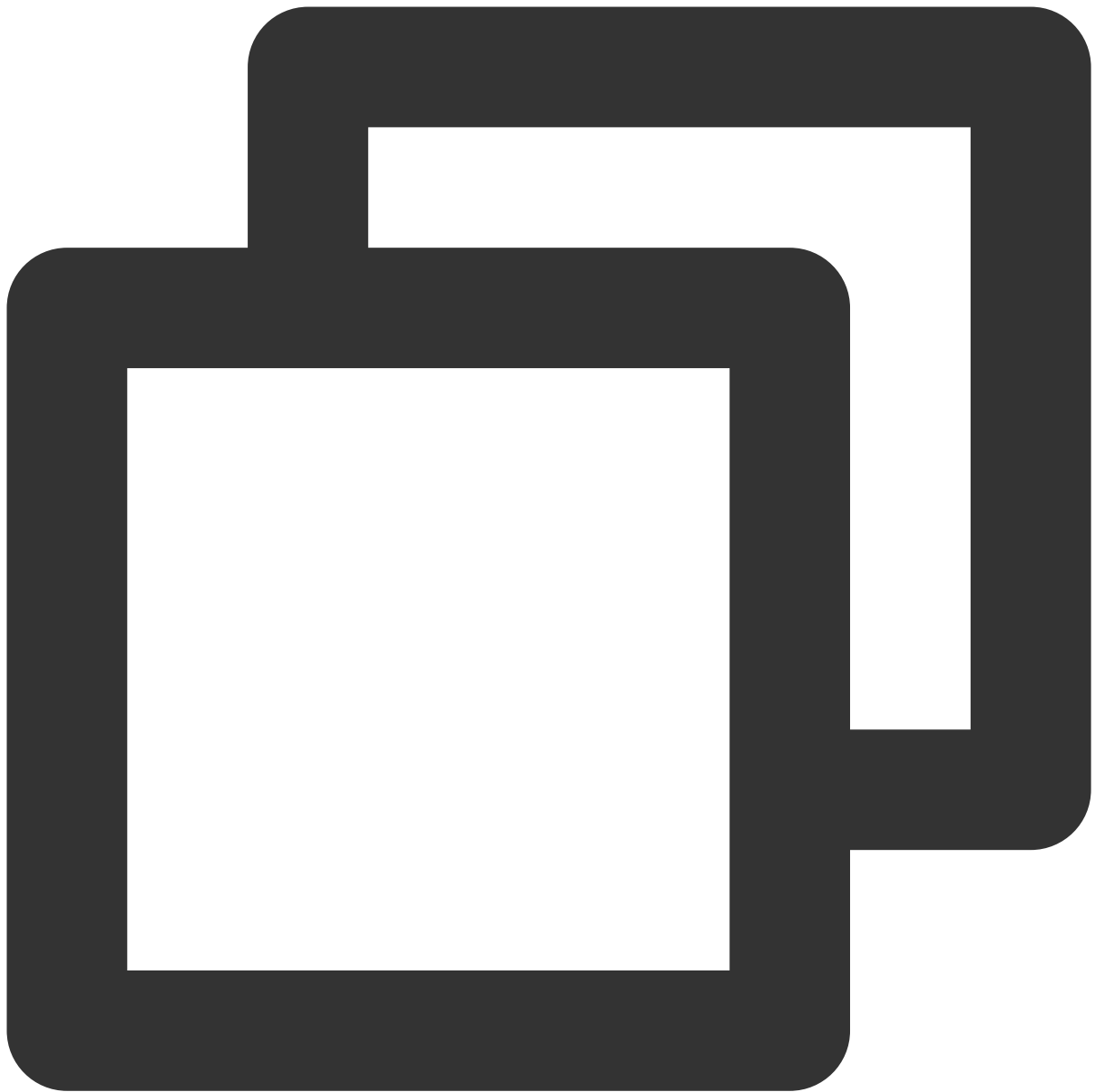
设置环境变量

1. 打开 `/etc/profile` 文件。



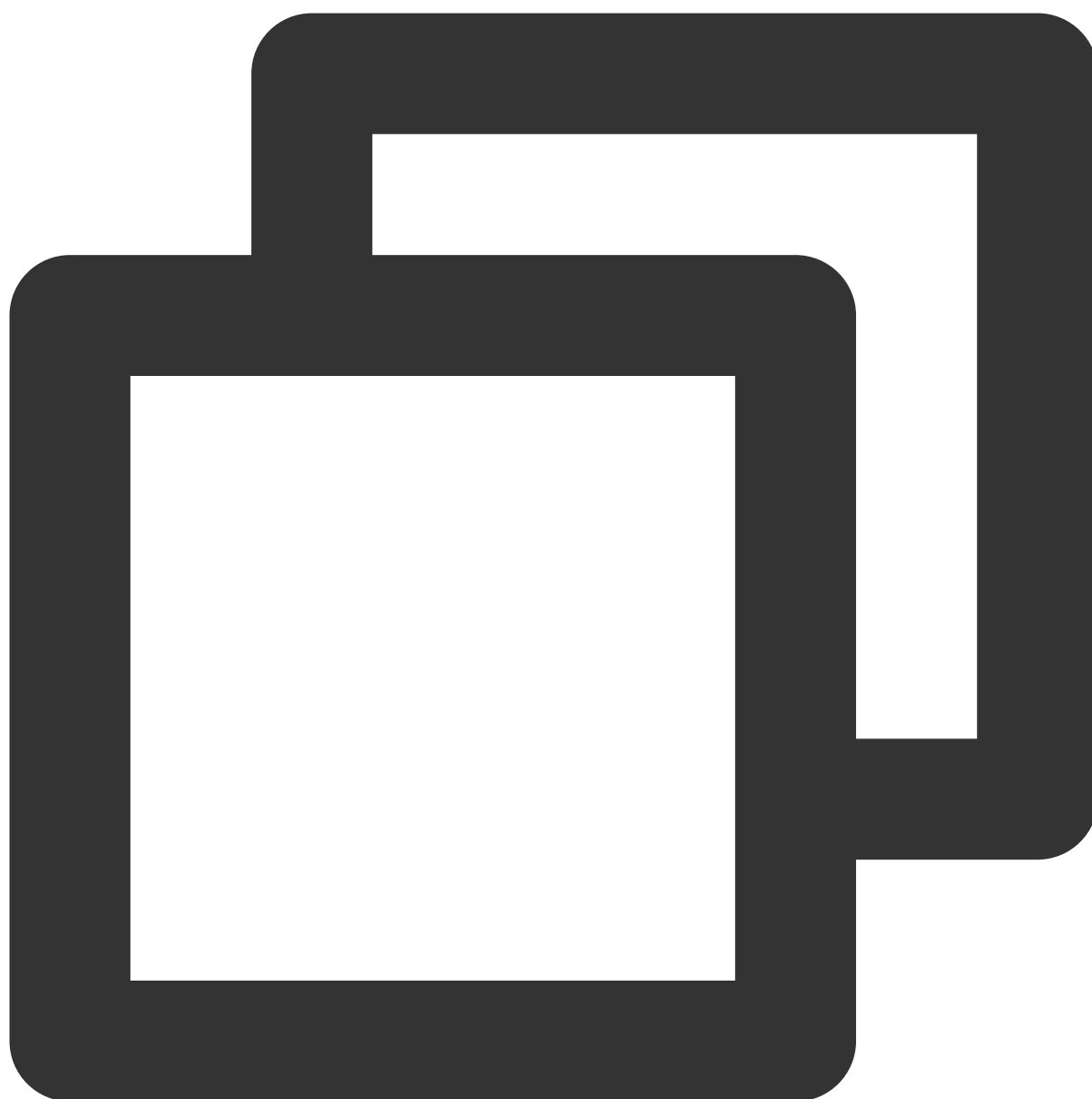
```
vi /etc/profile
```

2. 按下 i 键进入编辑模式，在该文件中添加如下信息。



```
# set java environment
export JAVA_HOME=/usr/java/jdk1.8.0_201
export CLASSPATH=$JAVA_HOME/lib/tools.jar:$JAVA_HOME/lib/dt.jar:$JAVA_HOME/lib
export PATH=$JAVA_HOME/bin:$PATH
```

3. 按下 **Esc** 键退出编辑模式，输入 `:wq` 保存并关闭文件。
4. 加载环境变量。



```
source /etc/profile
```

查看 JDK 是否安装成功

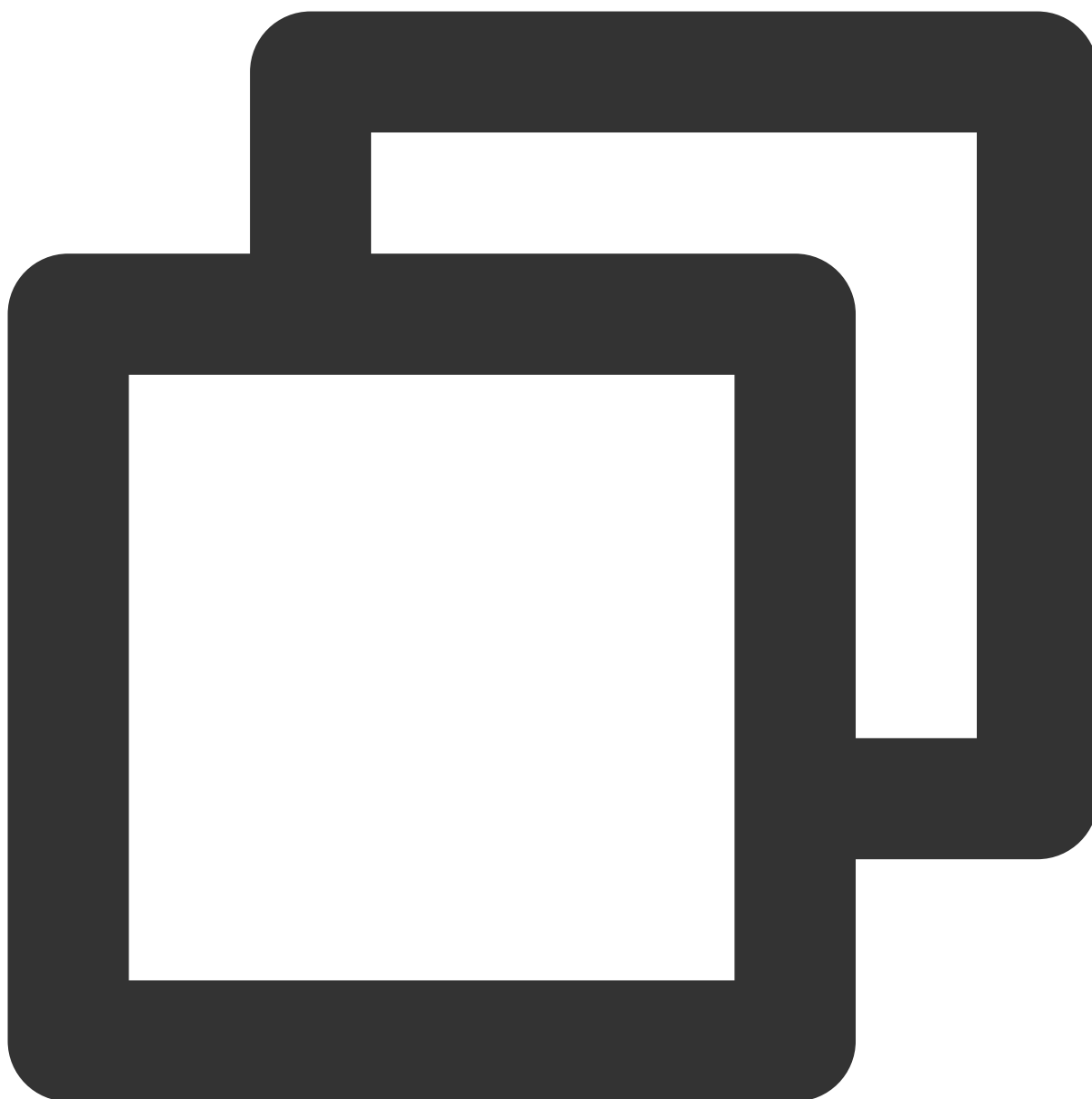
运行 `java -version` 命令，显示 JDK 版本信息时，表示 JDK 已经安装成功。

```
[root@emma /]# java -version
java version "1.8.0_201"
Java(TM) SE Runtime Environment (build 1.8.0_201-b09)
Java HotSpot(TM) 64-Bit Server VM (build 25.201-b09, mixed mode)
```

安装 Tomcat

下载 Tomcat

输入如下命令：



```
# 镜像地址会改变，Tomcat 版本也会不断升级。如果下载链接失效，请您到 [Tomcat 官网] (https://tomcat.apache.org/)  
wget http://mirrors.tuna.tsinghua.edu.cn/apache/tomcat/tomcat-8/v8.5.39/bin/apache-  
tar -xzf apache-tomcat-8.5.39.tar.gz  
mv apache-tomcat-8.5.39 /usr/local/tomcat/
```

在 `/usr/local/tomcat/` 目录中包含如下文件：

bin：脚本文件，包含启动和关闭 Tomcat 服务脚本。

conf：各种全局配置文件，其中最重要的是 **server.xml** 和 **web.xml**。

webapps：Tomcat 的主要 Web 发布目录，默认情况下把 Web 应用文件放于此目录。

logs：存放 Tomcat 执行时的日志文件。

注意：

如果下载链接失效，请替换为 [Tomcat 官网](#) 的最新下载链接。

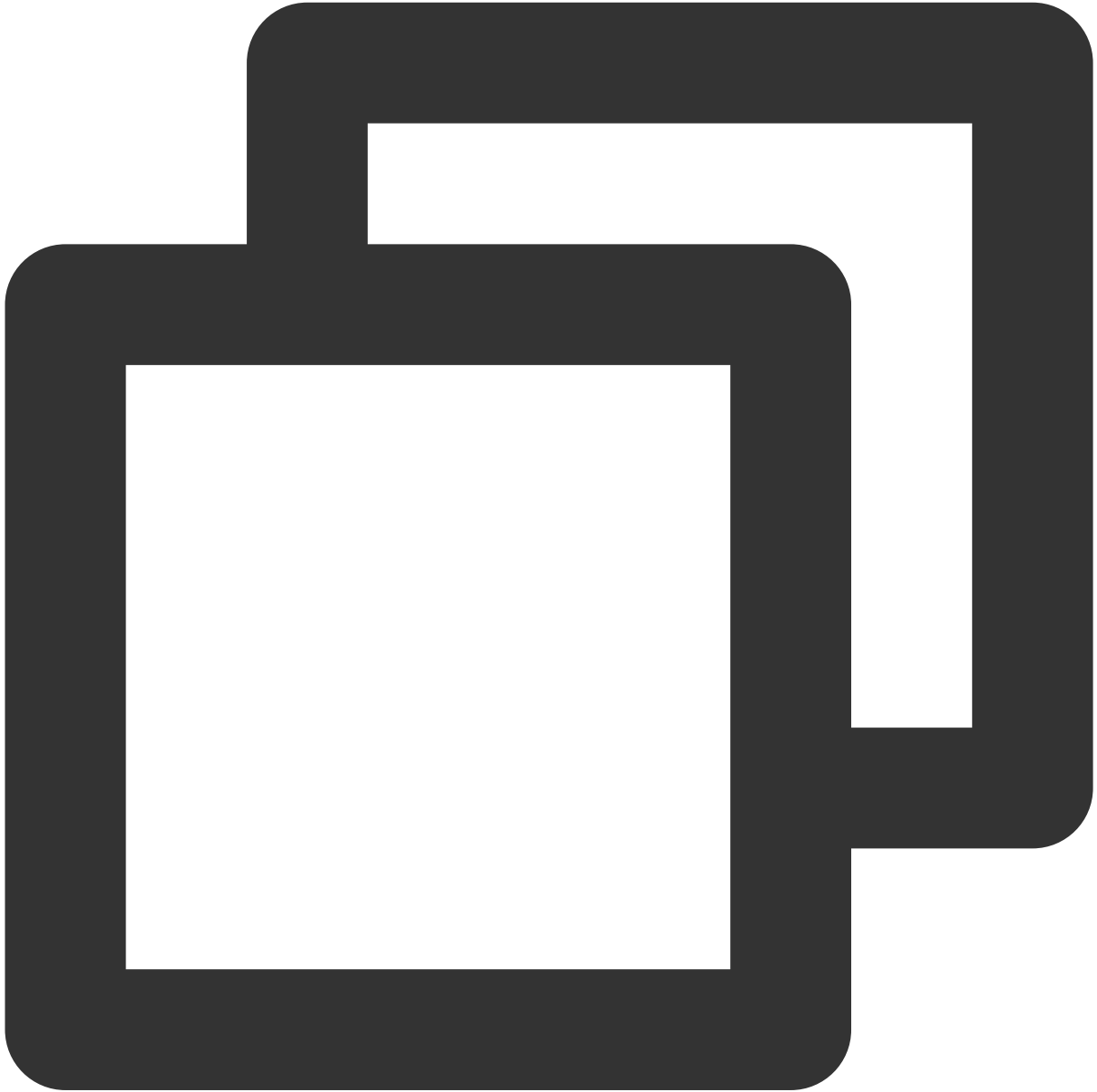
添加用户

```
# 创建一般用户 www来运行Tomcat
useradd www
# 创建网站根目录
mkdir -p /data/wwwroot/default
# 将需要部署的 Java Web 项目文件 WAR 包上传到网站根目录下，然后将网站根目录下文件权限改为 www。
echo Hello Tomcat! > /data/wwwroot/default/index.jsp
```

```
chown -R www.www /data/wwwroot
```

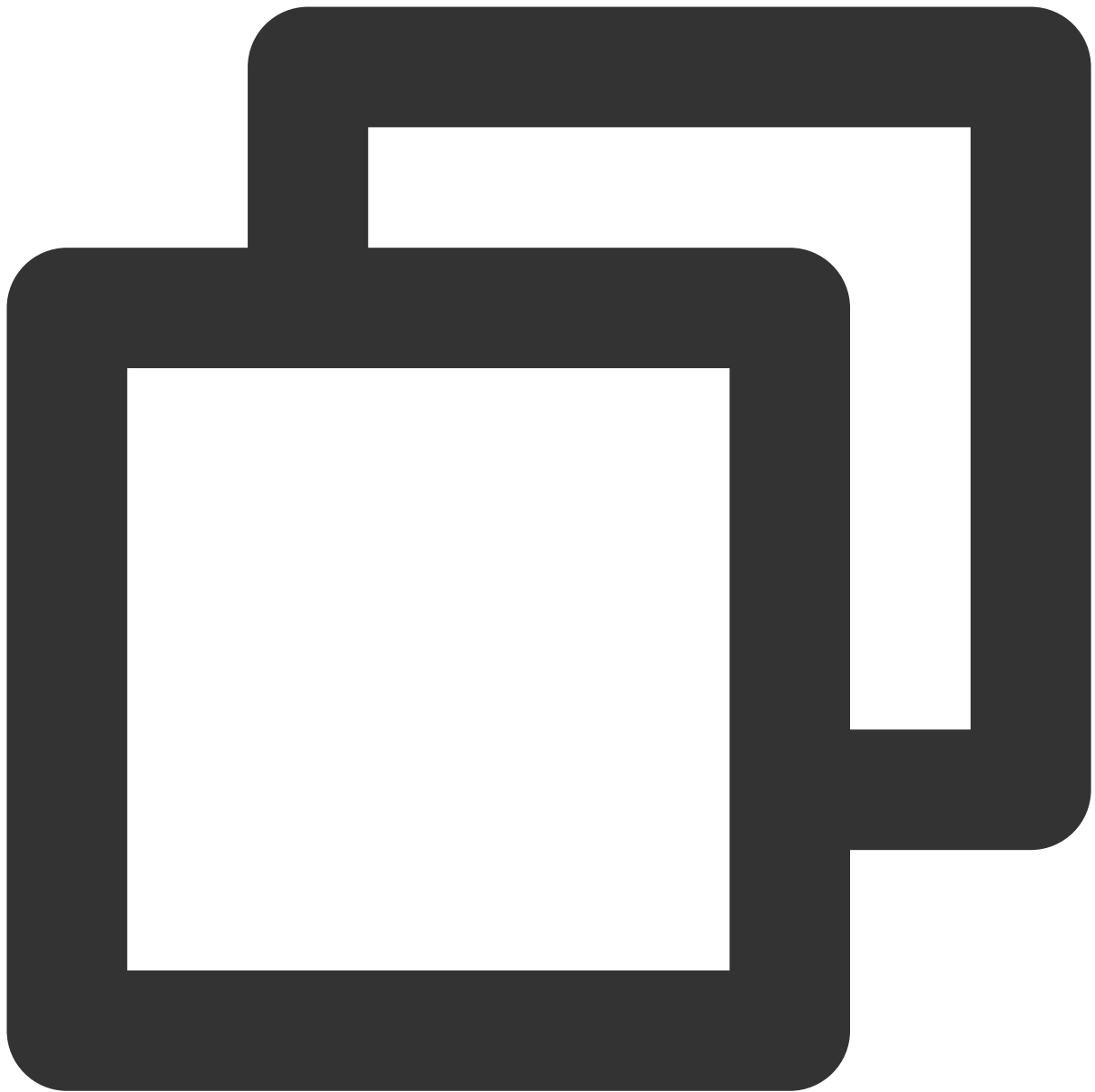
设置 JVM 内存参数

1. 创建一个 `/usr/local/tomcat/bin/setenv.sh` 脚本文件。



```
vi /usr/local/tomcat/bin/setenv.sh
```

2. 按下 `i` 键进入编辑模式，添加如下内容。

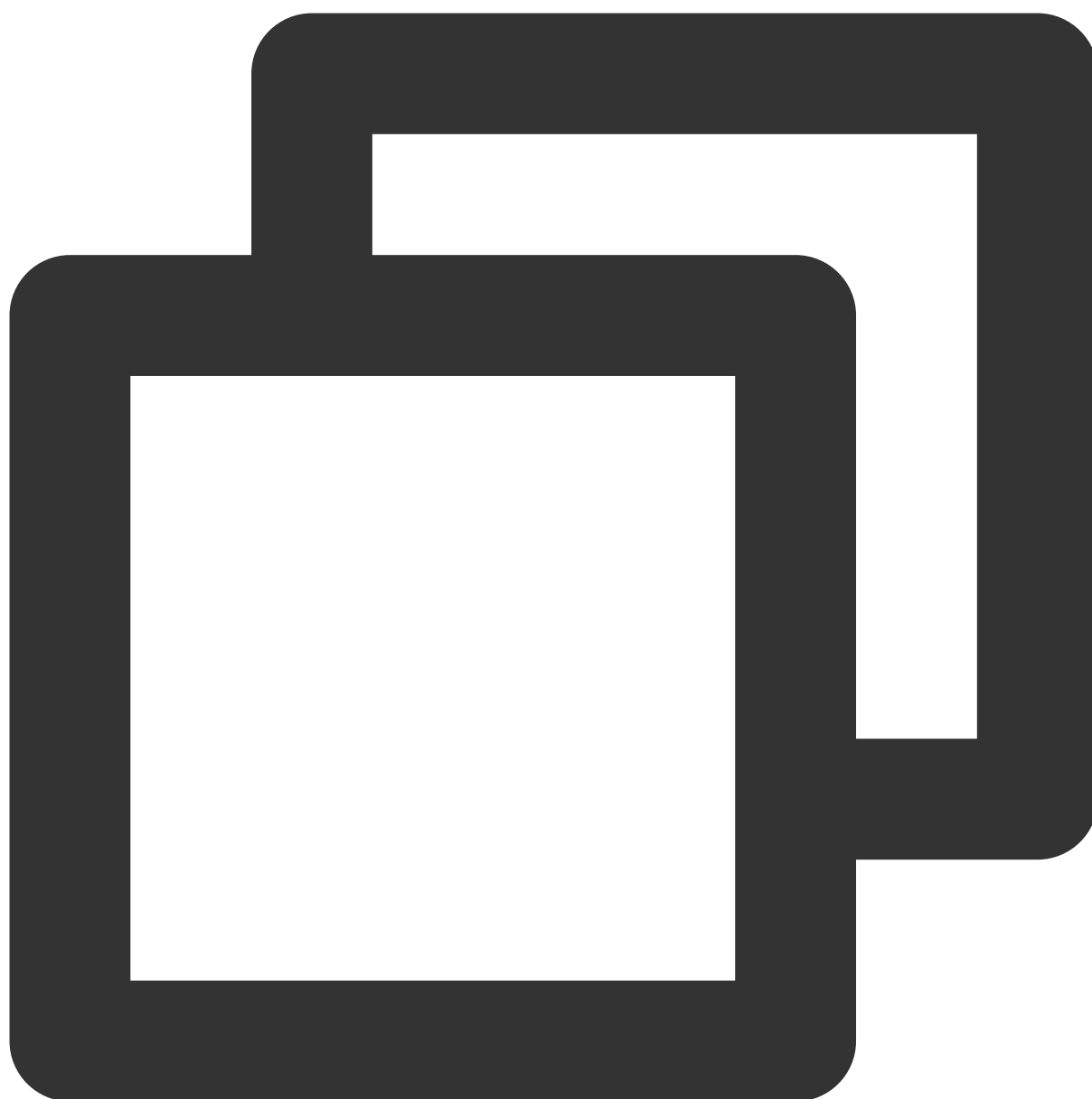


```
JAVA_OPTS='-Djava.security.egd=file:/dev/./urandom -server -Xms256m -Xmx496m -Dfile
```

3. 按 **Esc** 键退出编辑模式，输入 `:wq` 保存并退出编辑。

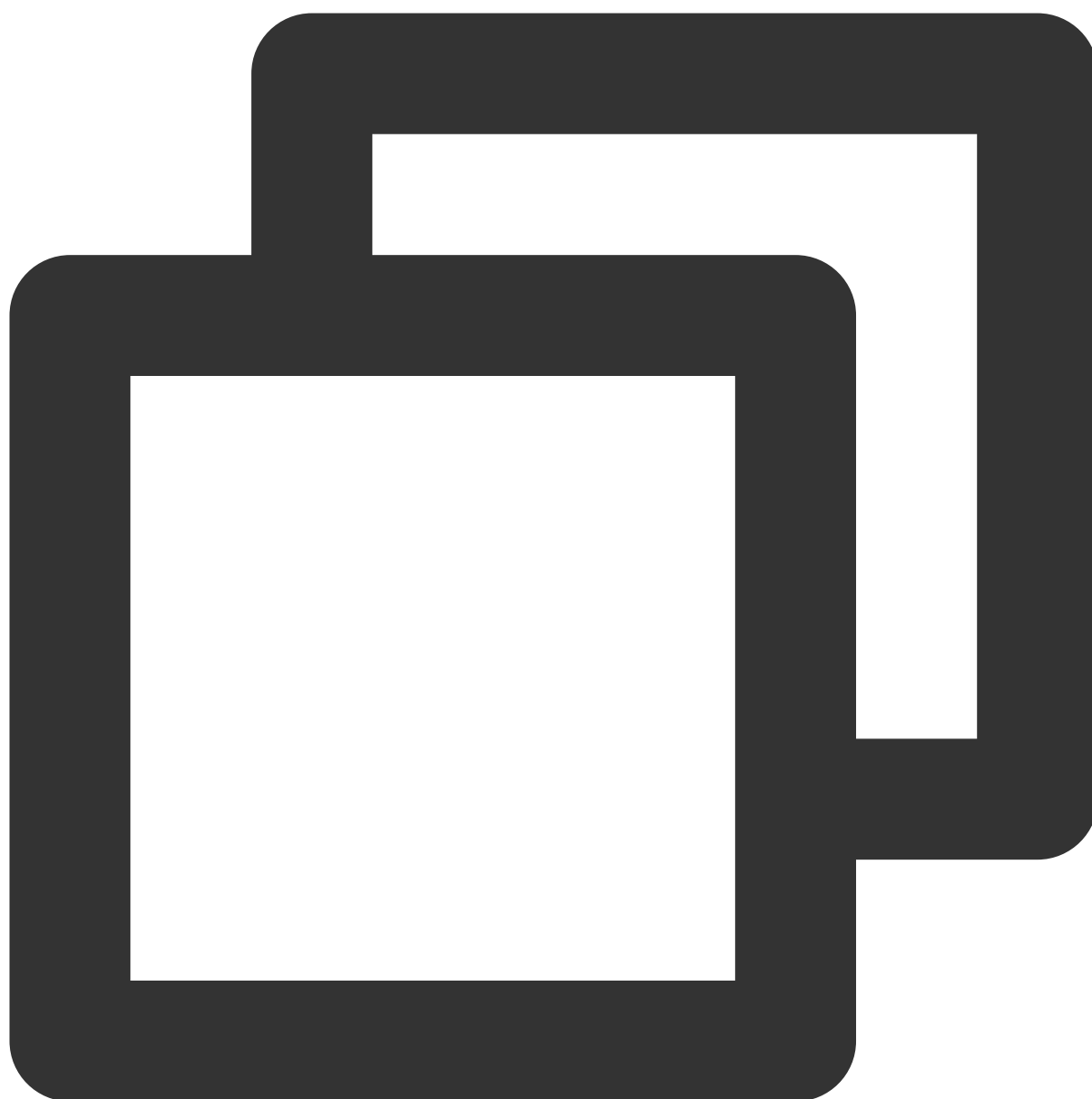
配置 **server.xml**

1. 切换到 `/usr/local/tomcat/conf/` 目录。



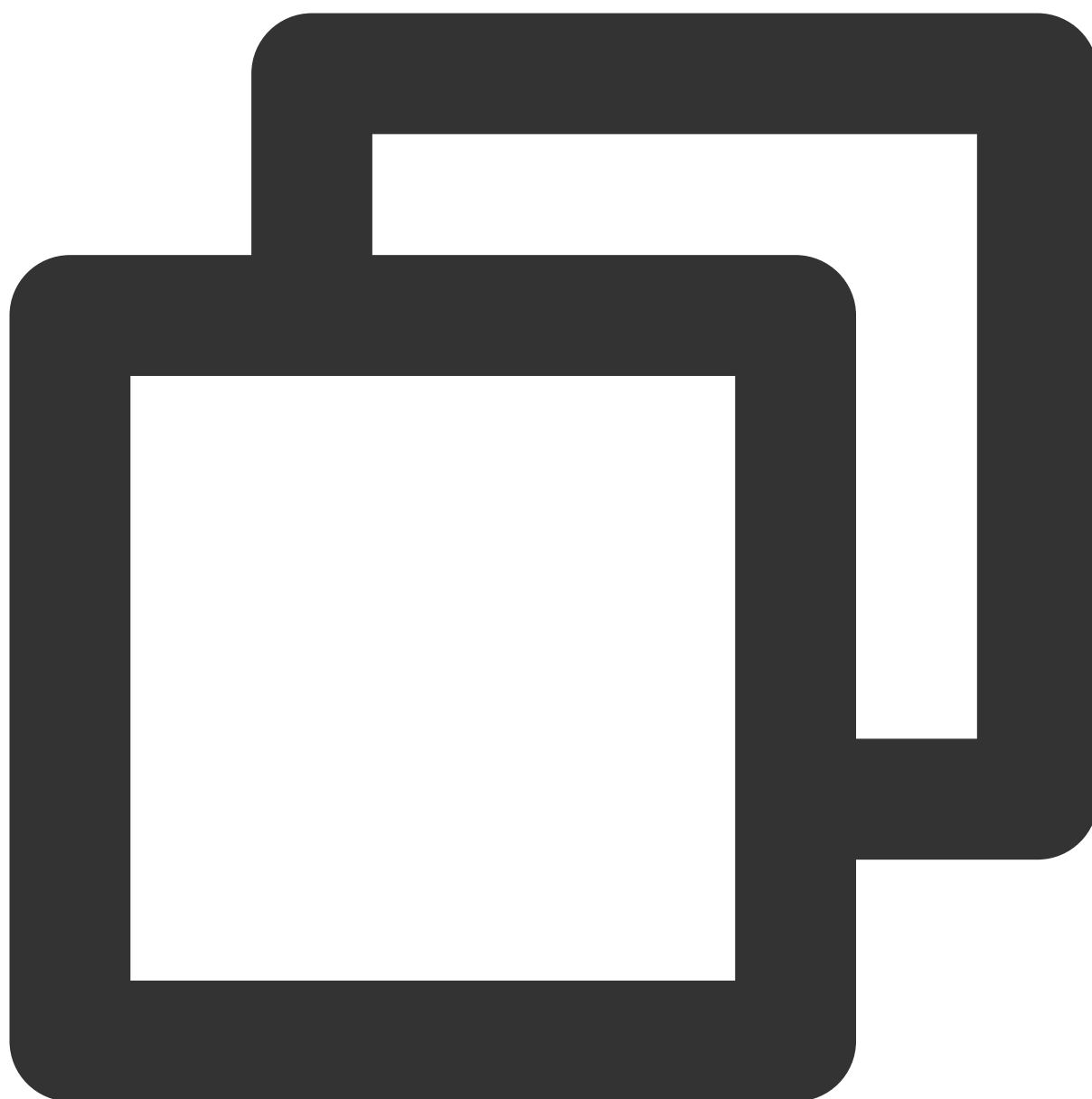
```
cd /usr/local/tomcat/conf/
```

2. 备份 server.xml 文件。



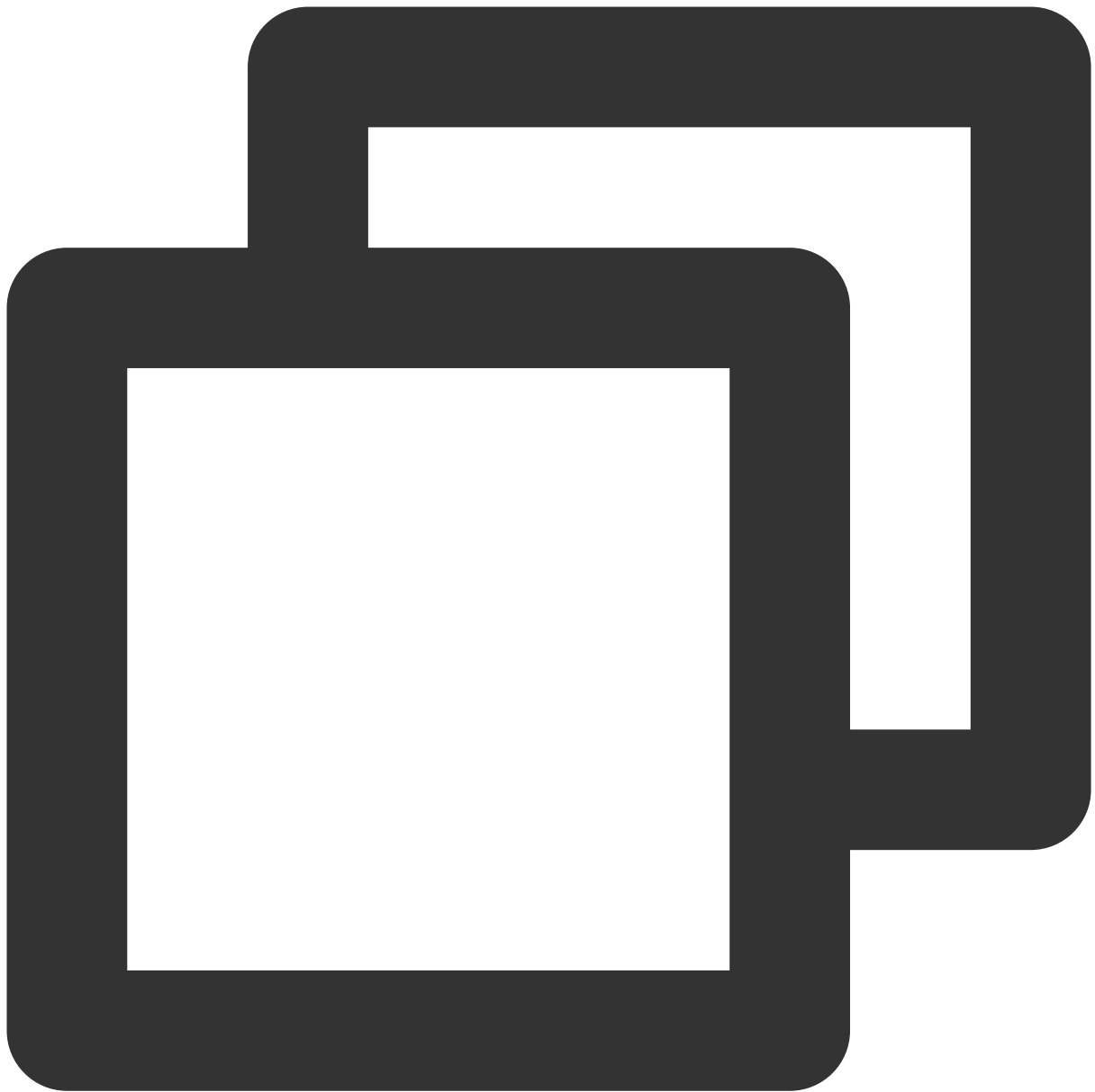
```
mv server.xml server_default.xml
```

3. 创建一个新的 `server.xml` 文件。



```
vi server.xml
```

4. 按下 i 键进入编辑模式，添加如下内容。



```
<?xml version="1.0" encoding="UTF-8"?>
<Server port="8006" shutdown="SHUTDOWN">
<Listener className="org.apache.catalina.core.JreMemoryLeakPreventionListener"/>
<Listener className="org.apache.catalina.mbeans.GlobalResourcesLifecycleListener"/>
<Listener className="org.apache.catalina.core.ThreadLocalLeakPreventionListener"/>
<Listener className="org.apache.catalina.core.AprLifecycleListener"/>
<GlobalNamingResources>
<Resource name="UserDatabase" auth="Container"
  type="org.apache.catalina.UserDatabase"
  description="User database that can be updated and saved"
  factory="org.apache.catalina.users.MemoryUserDatabaseFactory">
```



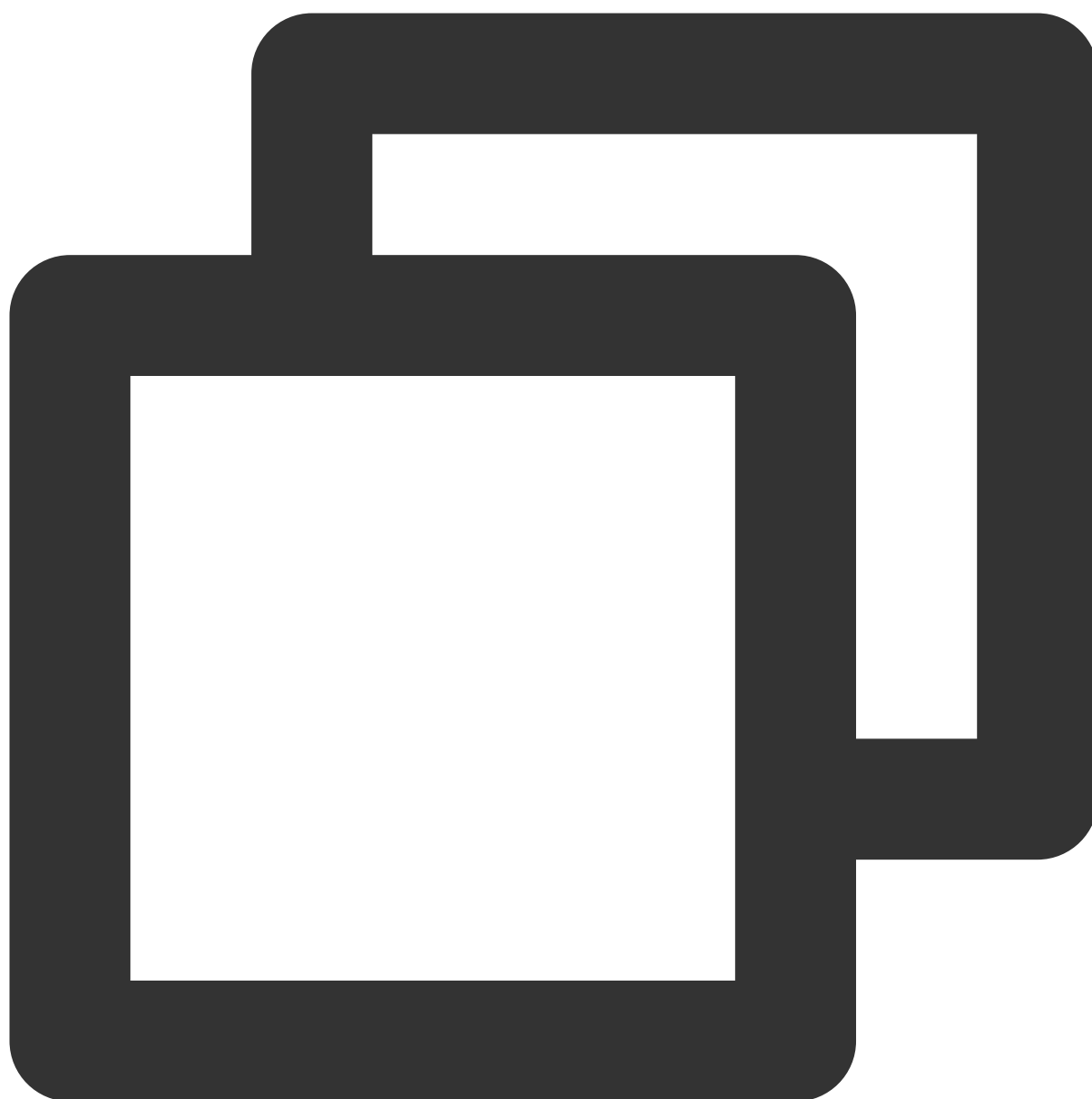
```
    pathname="conf/tomcat-users.xml"/>
</GlobalNamingResources>
<Service name="Catalina">
<Connector port="8080"
    protocol="HTTP/1.1"
    connectionTimeout="20000"
    redirectPort="8443"
    maxThreads="1000"
    minSpareThreads="20"
    acceptCount="1000"
    maxHttpHeaderSize="65536"
    debug="0"
    disableUploadTimeout="true"
    useBodyEncodingForURI="true"
    enableLookups="false"
    URIEncoding="UTF-8"/>
<Engine name="Catalina" defaultHost="localhost">
<Realm className="org.apache.catalina.realm.LockOutRealm">
<Realm className="org.apache.catalina.realm.UserDatabaseRealm"
    resourceName="UserDatabase"/>
</Realm>
<Host name="localhost" appBase="/data/wwwroot/default" unpackWARs="true" autoDeploy
<Context path="" docBase="/data/wwwroot/default" debug="0" reloadable="false" cross
<Valve className="org.apache.catalina.valves.AccessLogValve" directory="logs"
    prefix="localhost_access_log." suffix=".txt" pattern="%h %l %u %t &quot;%r&quot; %s
</Host>
</Engine>
</Service>
</Server>
```

5. 按 Esc 键退出编辑模式，输入 `:wq` 保存并退出编辑。

启动 Tomcat

方法一

进入 Tomcat 服务器的 bin 目录，然后执行 `./startup.sh` 命令启动 Tomcat 服务器。



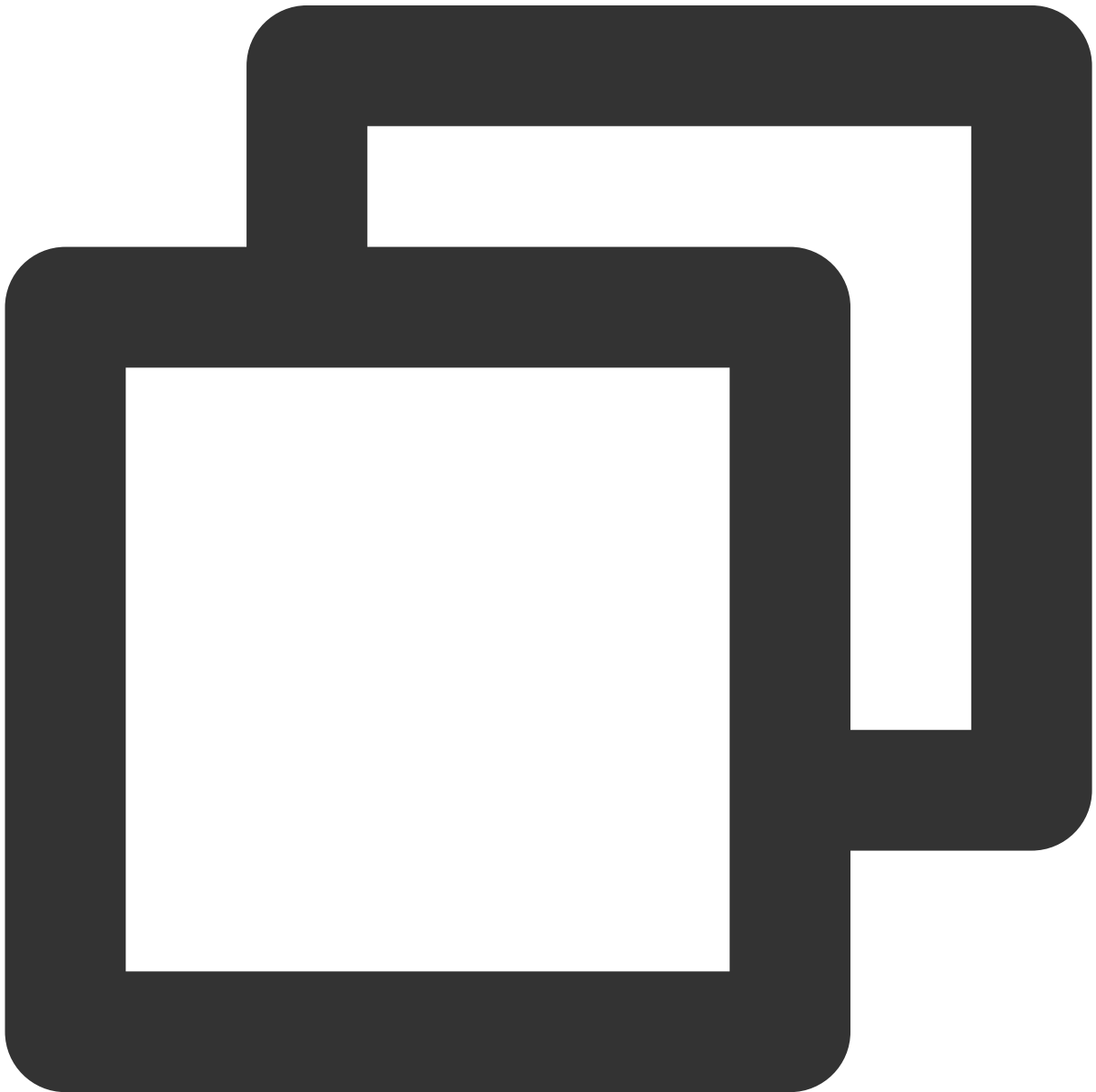
```
cd /usr/local/tomcat/bin  
./startup.sh
```

运行结果如下：

```
[root@emma bin]# ./startup.sh
Using CATALINA_BASE:   /usr/local/tomcat
Using CATALINA_HOME:   /usr/local/tomcat
Using CATALINA_TMPDIR: /usr/local/tomcat/temp
Using JRE_HOME:        /usr/java/jdk1.8.0_201
Using CLASSPATH:       /usr/local/tomcat/bin/bootstrap.jar:/usr/loc
Tomcat started.
```

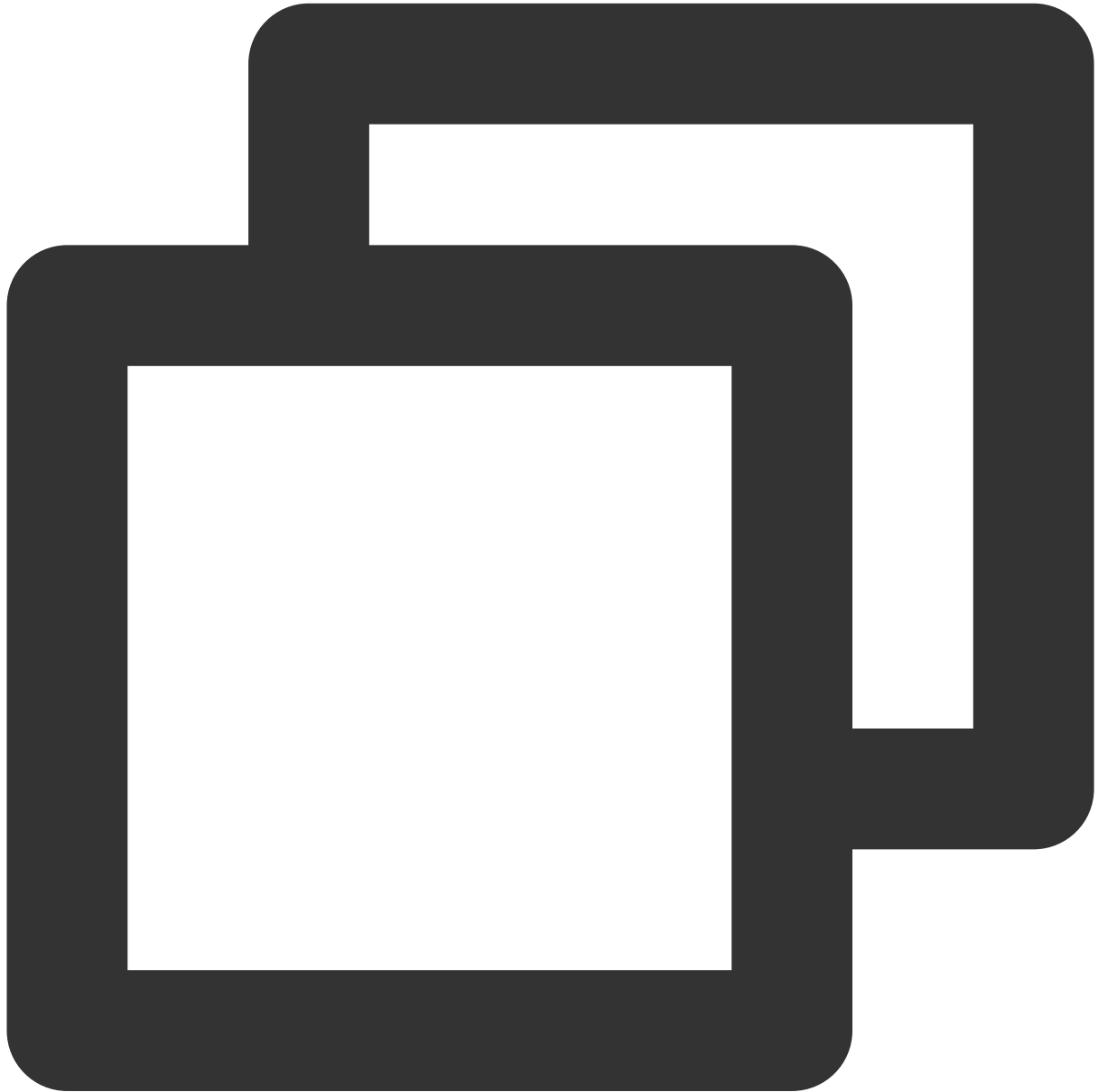
方法二

1. 设置快捷启动，在任何地方都可以通过 `service tomcat start` 来启动 Tomcat。



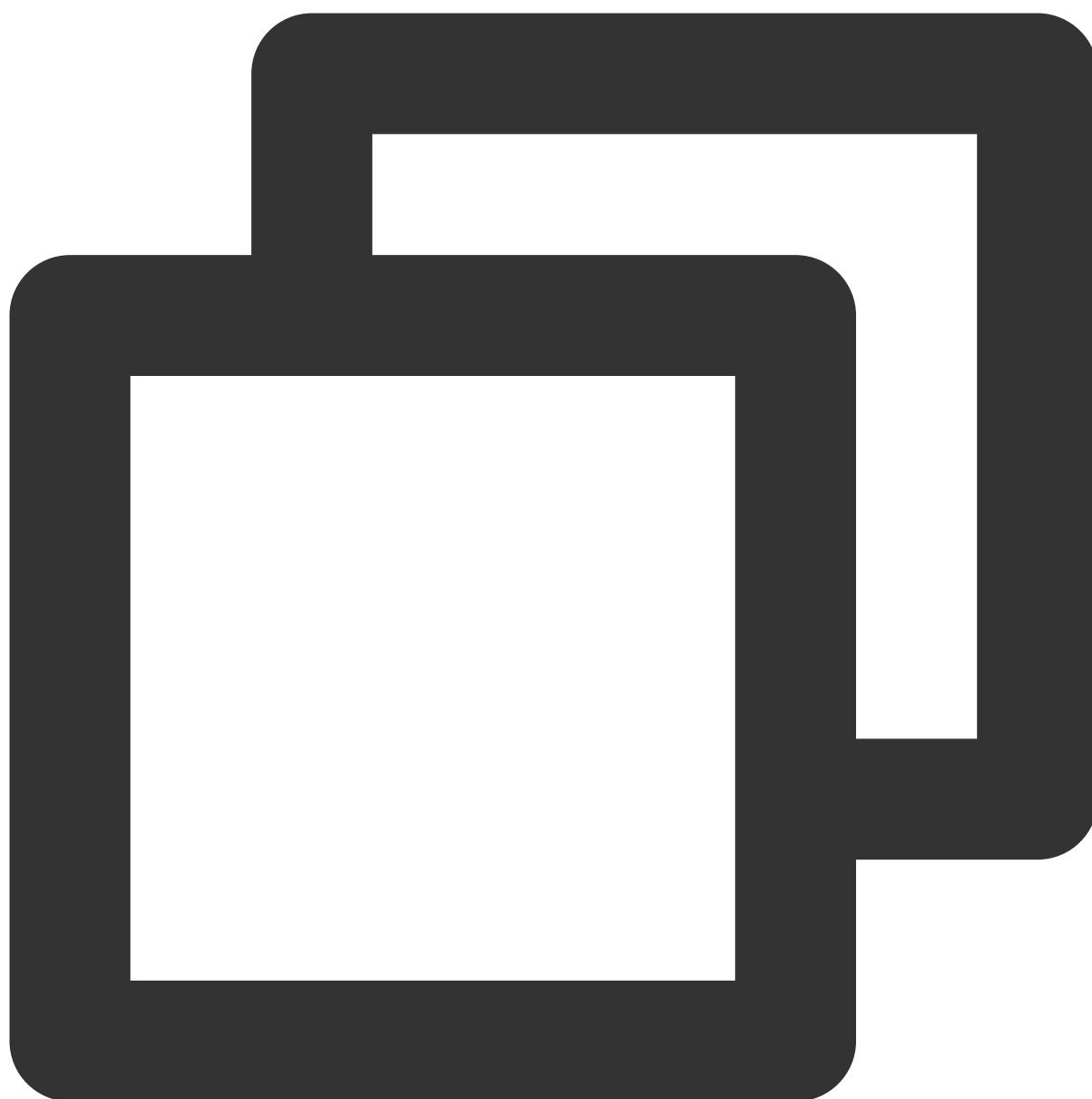
```
wget https://github.com/lj2007331/oneinstack/raw/master/init.d/Tomcat-init
mv Tomcat-init /etc/init.d/tomcat
chmod +x /etc/init.d/tomcat
```

2. 运行以下命令，设置启动脚本 `JAVA_HOME`。



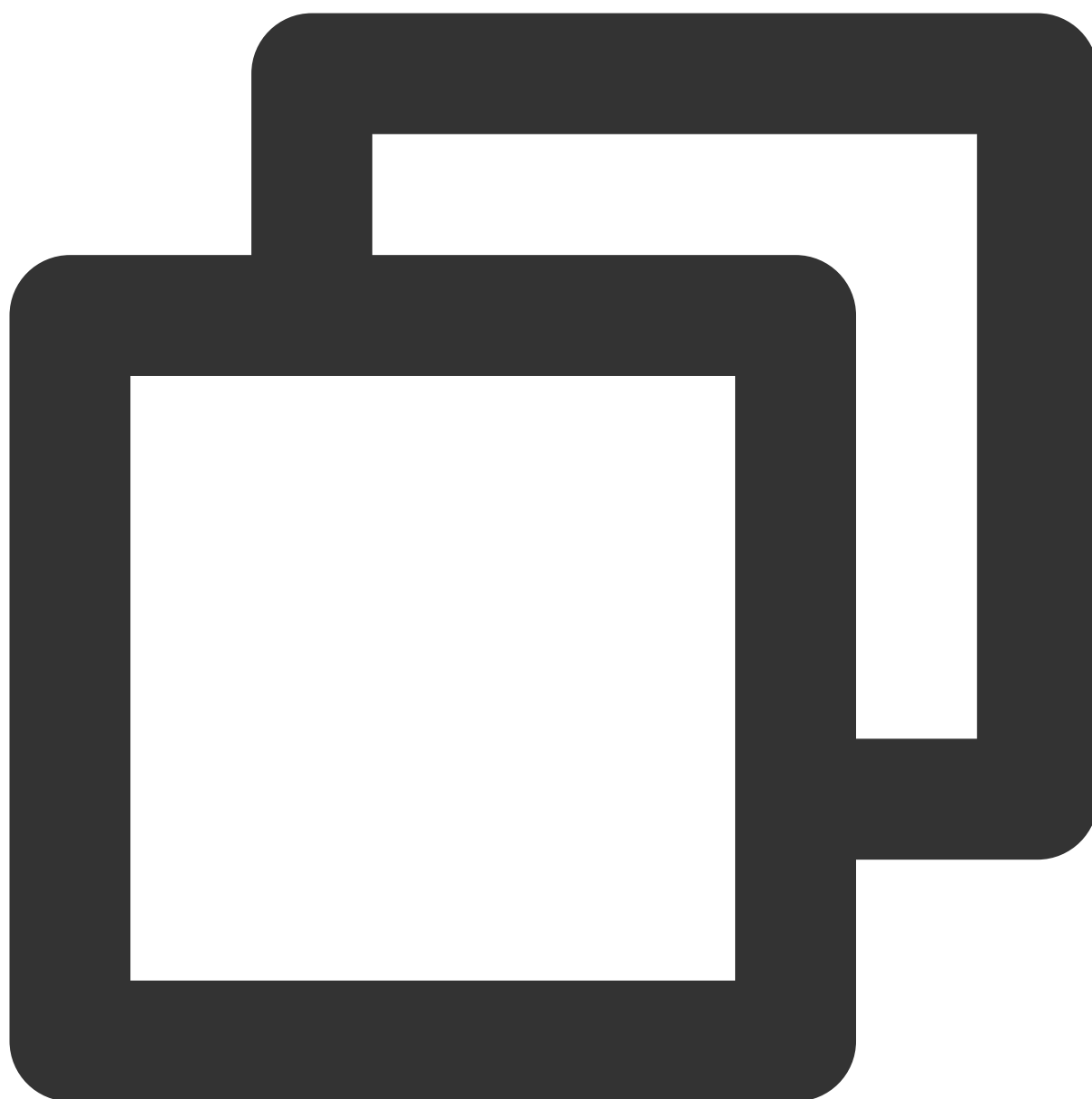
```
sed -i 's@^export JAVA_HOME=.*@export JAVA_HOME=/usr/java/jdk1.8.0_201@' /etc/init.
```

3. 设置自启动。



```
chkconfig --add tomcat  
chkconfig tomcat on
```

4. 启动 Tomcat。

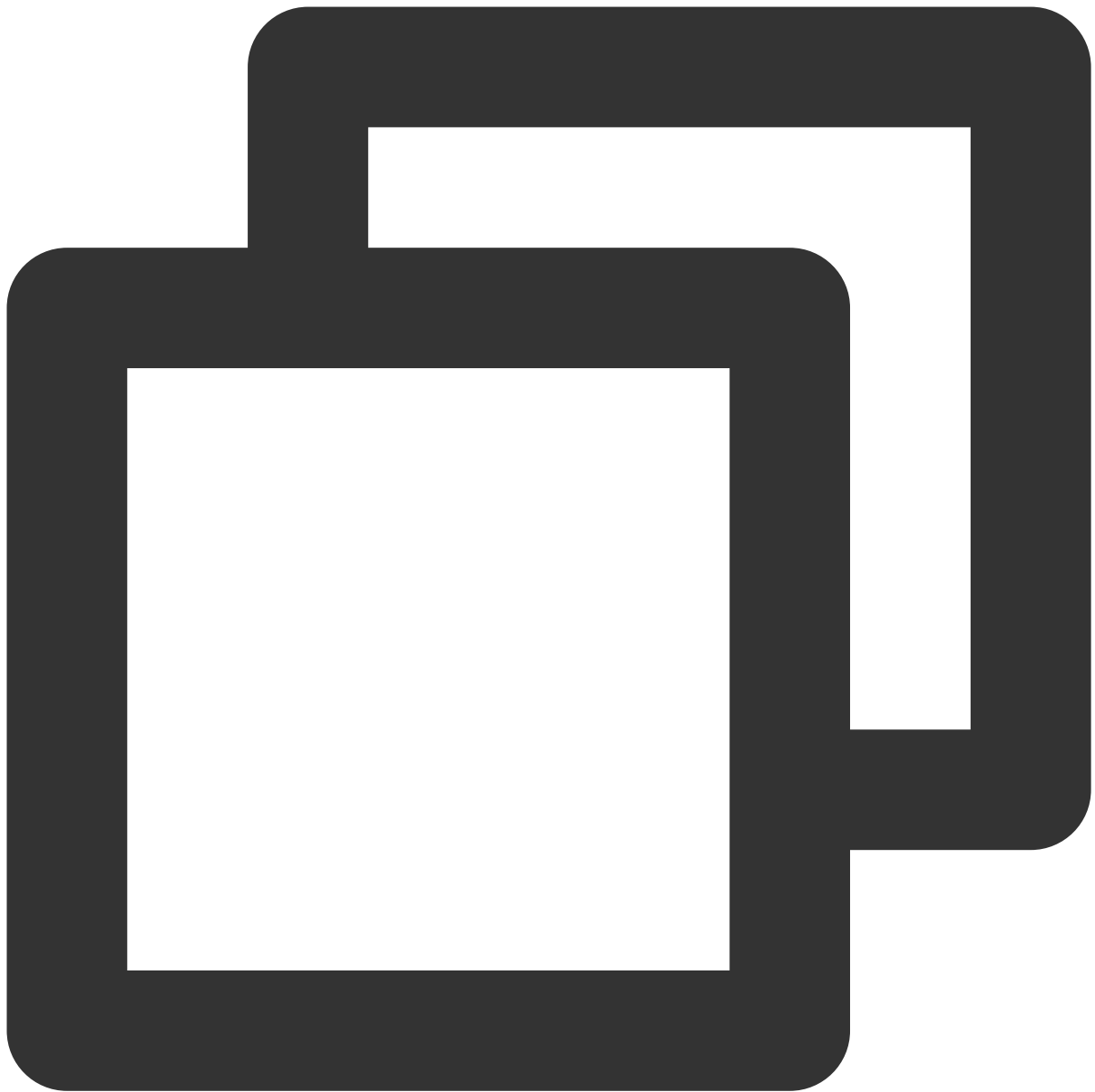


```
# 启动 Tomcat
service tomcat start
# 查看 Tomcat 运行状态
service tomcat status
# 关闭 Tomcat
service tomcat stop
```

运行结果如下：

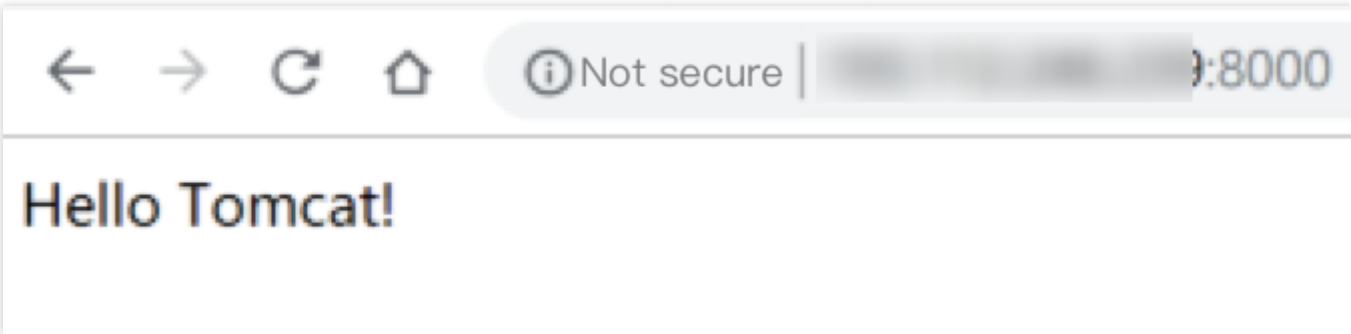
```
[root@emma bin]# service tomcat start
Tomcat is already running (pid: 27293)
[root@emma bin]# service tomcat status
Tomcat is running with pid: 27293
[root@emma bin]# service tomcat stop
Stopping Tomcat
Using CATALINA_BASE:   /usr/local/tomcat
Using CATALINA_HOME:   /usr/local/tomcat
Using CATALINA_TMPDIR: /usr/local/tomcat/temp
Using JRE_HOME:        /usr/java/jdk1.8.0_201
Using CLASSPATH:       /usr/local/tomcat/bin/bootstrap.jar:/usr/loca
waiting for processes to exit
```

5. 若提示没有权限，请切换为 root 用户并修改权限。



```
cd /usr/local  
chmod -R 777 tomcat
```

6. 在浏览器地址栏中输入 `http://公网IP:端口`（端口为 `server.xml` 中设置的 `connector port`）进行访问。出现下图所示页面时表示安装成功。



配置安全组

如果访问不通，请检查安全组。如上示例中 server.xml 中的 connector port 是8080，因此需在对应的云服务器所绑定的安全组上放通 TCP:8080。

Add Inbound rule

Type	Source ⓘ	Protocol port ⓘ	Policy	Notes
Custom ▾	0.0.0.0/0	TCP:8080	Allow ▾	Tomcat
+ New Line				

Completed

Cancel