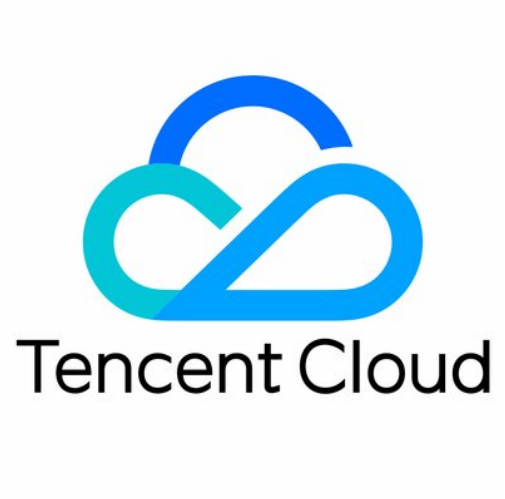
Virtual Private Cloud

Best Practices

Product Documentation



Contents

Best Practices
View the Total Bandwidth for Single-Region Traffic-Based Billing

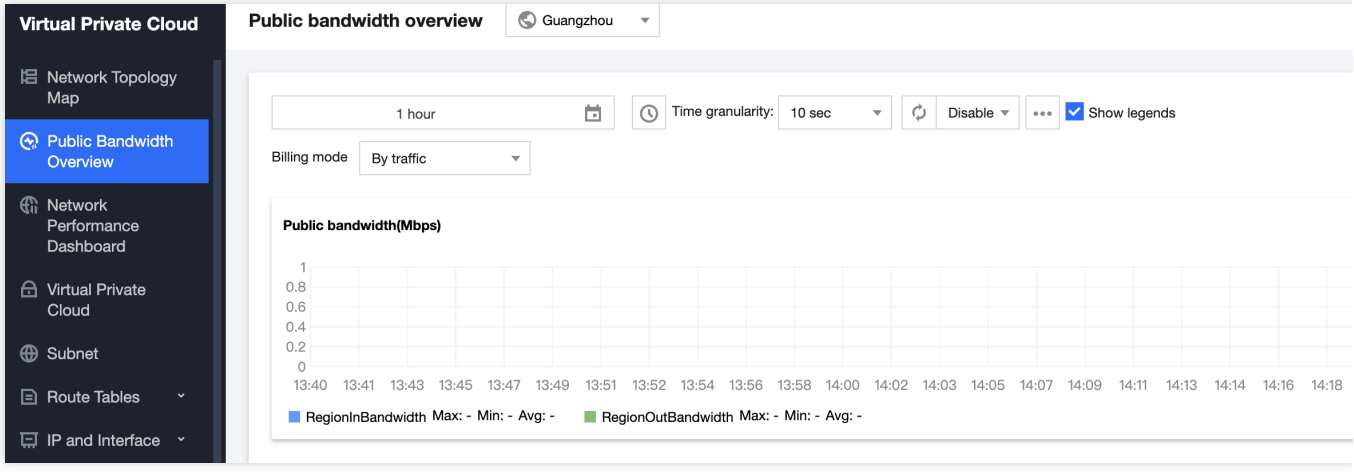
Last updated：2024-01-24 17:22:28

In a single region, the peak of the total bandwidth of all the **instances billed by traffic does not exceed 5Gbps**. The bandwidth peak for billing by hour is only taken as the upper limit peak of bandwidth, and it is not considered as a commitment metric. When there is competition for bandwidth resources, the bandwidth peak might be limited.
This document provides a monitoring view of total outbound and inbound bandwidth for hourly traffic in a single region to facilitate your real-time observation. You can also configure alarms to monitor whether the bandwidth exceeds the limit.

Viewing monitoring data

1. Log in to the internet bandwidth overview console.
2. Select the region, time frame, and other information you desire to view, and you can view the current region. The billing model is based on hourly traffic outbound and inbound bandwidth metrics.

Metric Meanings:
Single-region public network outbound bandwidth: Represents the total outbound bandwidth of resources billed by hourly traffic in the selected region.
Single-region public network inbound bandwidth: Represents the total inbound bandwidth of resources billed by hourly traffic in the selected region.

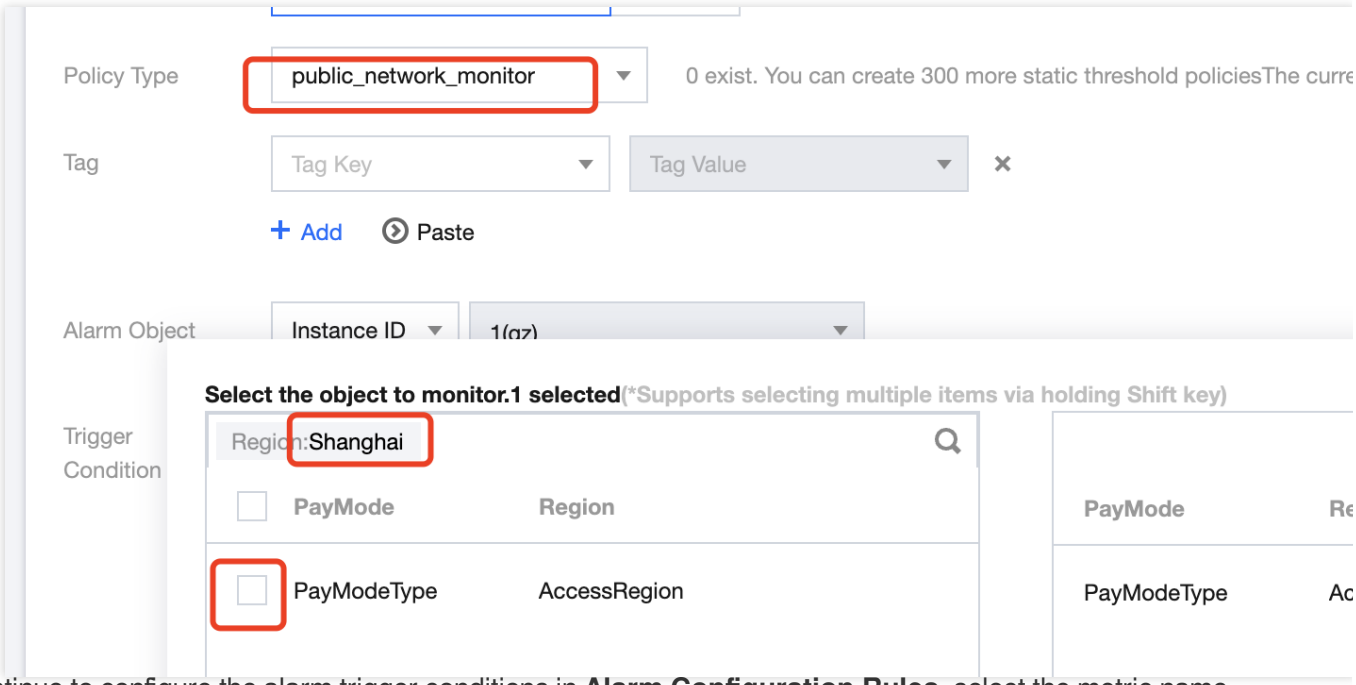


Configure alarms

1. Click the alarm icon

, the interface will skip to the **Create Alarm Policy** page of Tencet Cloud Observability Platform.

2. Set the **Policy Name**, select **Monitor Type** as **Cloud Product Monitoring** and **Policy Type** as **Virtual Private Cloud** / **Public Network Bandwidth Monitoring**, select the region ID that needs to be configured for **Alarm Object,** and check the **Traffic-Based Billing Mode** of the region. Multiple regions can be configured. Click **OK** after completion.

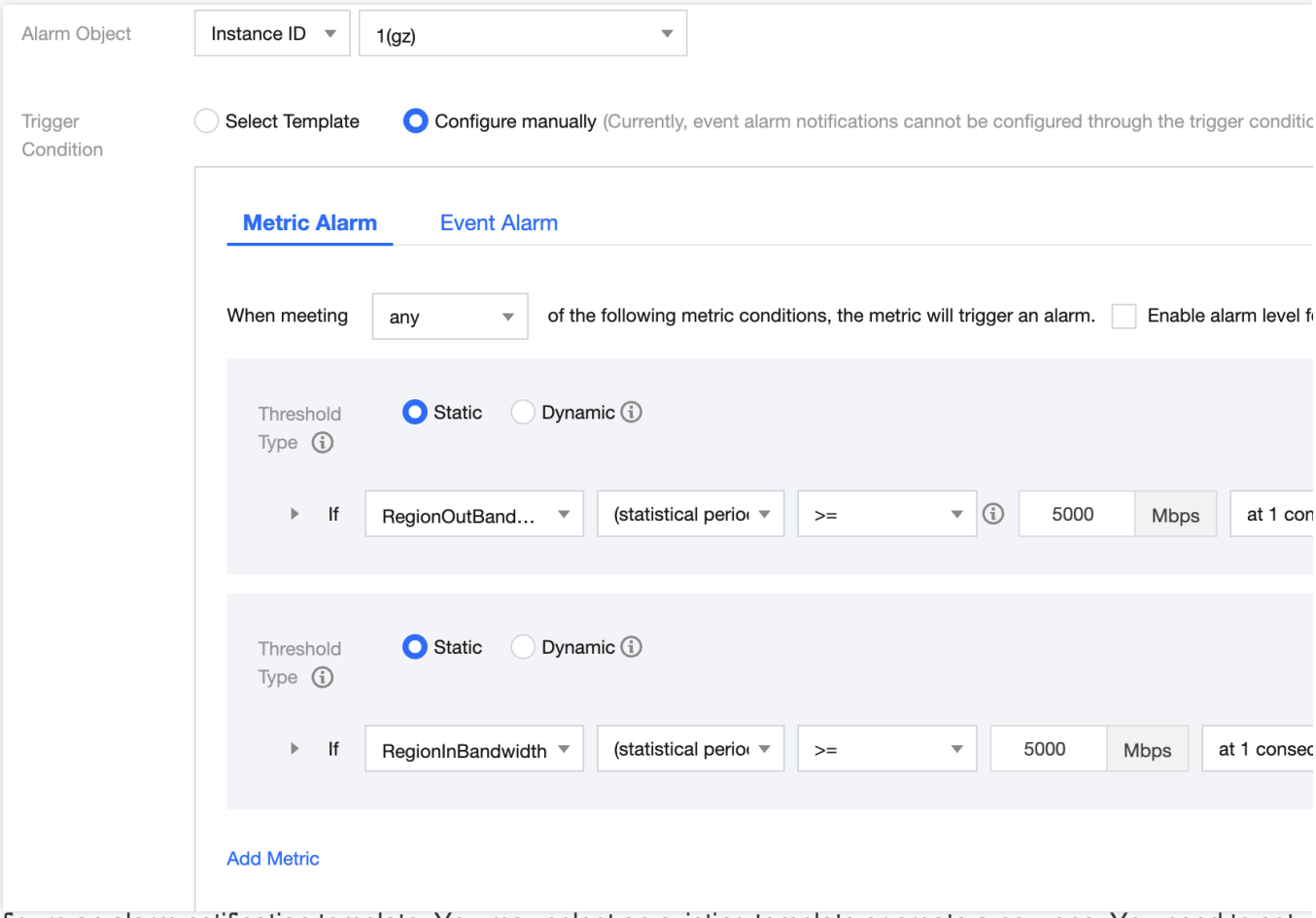


3. Continue to configure the alarm trigger conditions in **Alarm Configuration Rules**, select the metric name.

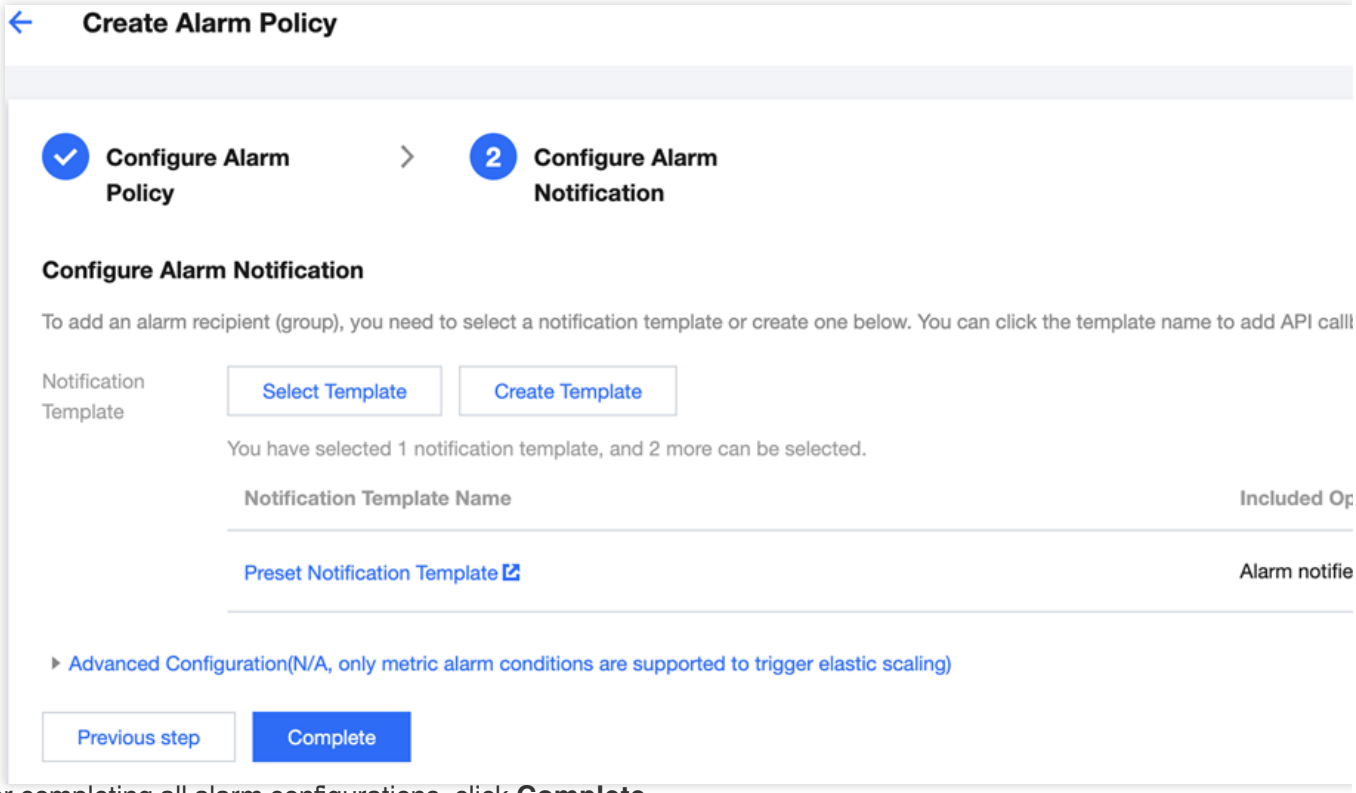
Metric Meanings:

Single-region public network outbound bandwidth: Represents the total outbound bandwidth of resources billed by hourly traffic in the selected region.

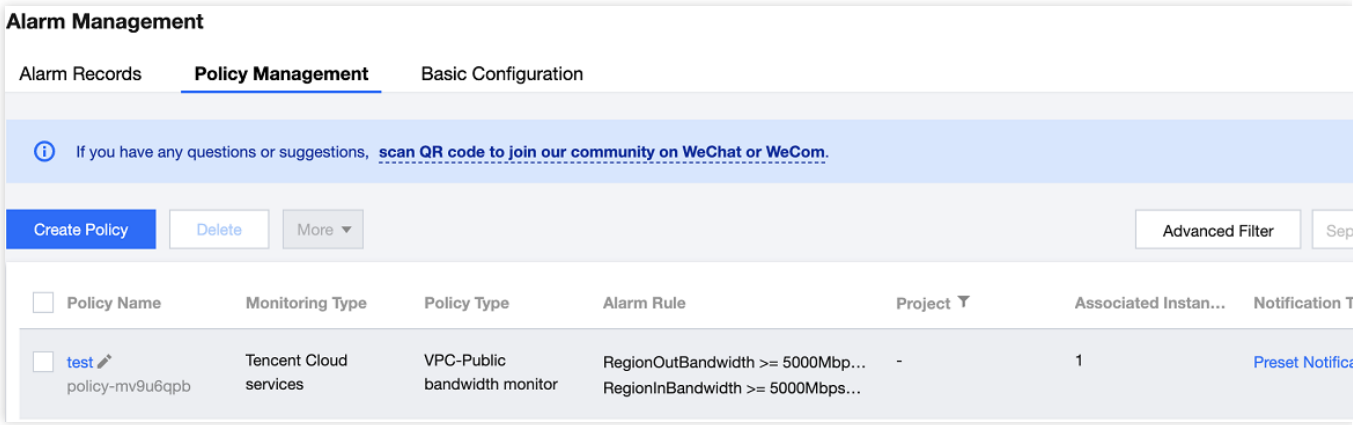
Single-region public network inbound bandwidth: Represents the total inbound bandwidth of resources billed by hourly traffic in the selected region.



4. Configure an alarm notification template: You may select an existing template or create a new one. You need to set template name, recipient objects, notification cycle, receiving channels, etc. in the notification template. Once the settings are completed, click **Confirm**.



5. After completing all alarm configurations, click **Complete**.



Migrating from the Classic Network to VPC
Notes for migration

Last updated：2024-01-24 17:22:28

The classic network is an earlier cloud network provided by Tencent Cloud. As the user scale and services expand, Virtual Private Cloud (VPC) evolves from the classic network to provide independent, controllable, and more secure networks. As a mainstream cloud network, VPC provides better user experience. This document provides answers to questions that you may have when you migrate resources from the classic network to VPC.

Why Do I Need to Migrate Resources to VPC?

A VPC is a logically isolated network space in Tencent Cloud and has the following advantages:
It allows you to customize IP ranges, IP addresses, and routing policies.
It supports more complex scenarios such as Elastic Network Interface (ENI), network access control list (ACL), and cross-region communication.
It improves the disaster recovery capability and availability greatly.
It supports multiple Cloud Virtual Machine (CVM) models.
VPC is more suitable for use cases that require custom configurations compared with the classic network. To provide you with better services, Tencent Cloud will fully upgrade the classic network. From **March 31, 2022**, resources cannot be created on the classic network. Services on the classic network are officially discontinued on **December 31, 2022**. These services are replaced with the corresponding VPC services.

Is My Business Running on the Classic Network Affected After March 31, 2022?

Your existing resources running on the classic network are still available until **December 31, 2022**. We recommend you migrate your resources to VPC as soon as possible.

What Is the Impact If I Do Not Migrate Resources to VPC?

Before the classic network is discontinued, your existing resources will not be affected. After the classic network is discontinued, the services of the classic network will no longer be available. Therefore, we recommend you migrate your resources to VPC before **December 31, 2022**.

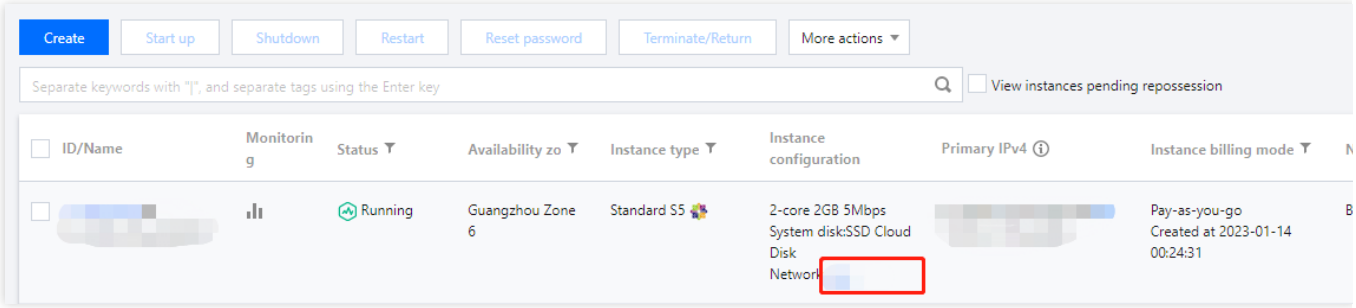
How Do I Determine Whether I Need to Migrate Resources to VPC?

1. Check whether your account is created before **00:00 June 13, 2017**. If your account is created after 00:00 June 13, 2017, the classic network is not supported for your account and the cloud resources that you purchase are already in VPC. In this case, you do not need to migrate resources to VPC. If your account is created before 00:00 June 13, 2017, go to Step 2.
2. Log in to the Tencent Cloud Console, choose **Fees** > **My Orders**, and check whether you have purchased the following resources:
Classic network servers: CVM, GPU Cloud Computing, and FPGA Cloud Computing.
Classic network databases: TencentDB for MySQL, Redis, SQL Server, PostgreSQL, and MongoDB, as well as TDSQL for MySQL.
Classic network load balancers: classic Cloud Load Balancer (CLB) and CLB.
Others: classic network Cloud File Storage (CFS) and classic network Cloud Kafka (CKafka).

Note:

Lighthouse is not a service of the classic network, but a VPC service. Therefore, it does not require migration.

3. If you have purchased any of the preceding resources, log in to the console for each resource and check whether the network attribute of the resource is classic network. If not, ignore the network migration notice.
Take CVM as an example. Log in to the CVM console and check the network attribute of each CVM instance. If the network attribute is **classic network**, you need to perform network migration. If the network attribute is **VPC**, ignore the network migration notice.



Note:

If you have purchased many resources, you can submit a ticket for assistance.

Will the Billing Mode of an Instance Change After the Migration from the Classic Network to VPC?

The billing mode is not changed.

Will the Configuration of an Instance Change After the Migration from the Classic Network to VPC?

The public IP remains unchanged, and the instance can still be accessed by using the original domain name.
The Media Access Control (MAC) address remains unchanged, but the private IP will be changed.
Note:
If the IP of the instance is within the target VPC IP range, you can keep the private IP unchanged by specifying it as the new IP. Otherwise, the private IP will change.

Will the Services Be Interrupted During the Migration from the Classic Network to VPC?

This depends on the specific Tencent Cloud service that you use:
During the migration of a CVM instance, the instance must be restarted, which will interrupt your service for a short while. We recommend you migrate the instance during off-peak hours.
For TencentDB services, your services are not affected as dual-IP accessing is supported during migration.
CLB doesn't support direct migration. You can rebuild instances with the same configuration and gradually migrate the business traffic.

Can I Migrate an Instance Back to the Classic Network?

No, you cannot migrate an instance back to the classic network after it is migrated to VPC.

Can I Migrate a Classic Network Instance in a Region to a VPC Instance in Another Region?

No, instances can be migrated only to the same region and availability zone.

Is Network Migration Implemented by Tencent Cloud?

No, you need to manually perform the migration. If you have any questions, submit a ticket.

Migration Solutions

Last updated：2024-01-24 17:22:28

This document describes how to migrate from your resources from the classic network to VPC.

Note:

Before switching the network, you need to create a VPC in the same region as the classic network instance to be migrated and create a subnet in the same AZ as the instance. For more information, see Creating VPCs.

Tencent Cloud provides the two migration solutions below:

Migrating a ingle instance: Choose this if you only need to migrate instances one by one.

Hybrid access: if your business involves CVM, CLB, and TencentDB instances, you can use this solution to ensure a smooth business migration.

Migrating a Single Instance

You can easily migrate a instance from the classic network to a VPC. See below for details.

Instance	Features
CVM	The instance needs to be restarted The classic network IP is immediately changed to the VPC IP, with no retention time If the CVM instance has a public IP, the public IP will stay unchanged after the network switch, which will not affect the access at domain name
TencentDB for MySQL	Dual-IP access is maintained for a certain period of time. The original classic network IP retention time is as follows: MySQL: 24 hours (1 day) by default and up to 168 hours (7 days) MariaDB: 24 hours (1 day) TDSQL: 24 hours (1 day) Redis: you can choose to expire immediately, release after 1 day, release after 2 days, release after 3 days, or release after 7 days MongoDB: the original IP on v4.0 or above will expire immediately. For other versions, you can choose to expire immediately, release after 1 day, release after 2 days, release after 3 days, or release after 7 days
TencentDB for MariaDB	
TDSQL for MySQL	
TencentDB for Redis	
TencentDB for MongoDB	
TencentDB for PostgreSQL	You can configure up to two networks for each instance, both of which can be used for business access. The IPs of different networks can be the same.

Note:

If you want to keep the resource IP addresses unchanged after the network switch, try to create a VPC that covers the classic network IP.

Create a private DNS service and resolve its domain name. After migrating the resources to a VPC, use Tencent Cloud Private DNS.
Access using the public IP.

Hybrid Access Solution During the Migration

Hybrid access means the services being migrated can access both the classic network and a VPC. Tencent Cloud provides the following hybrid access solutions:
TencentDB: the accessibility of classic network IP and VPC IP ensures the hybrid access at the TencentDB instance level.
COS: access through domain name naturally provides the hybrid access capability.
CVM:
Classiclink: allows the classic network-based CVMs to interconnect with VPC resources such as CVM, TencentDB, and CLB instances.
Peering connection: allows the instances in a VPC to communicate with resources in the classic network (except CVMs).
Note:
To use peering connection, submit a ticket.
To configure Classiclink, see Classiclink.
For migration of CLB instances, see Example: Migrating a Public Network CLB and Example: Configuring Hybrid Access for a Private Network CLB.

Example: Migrating a Public Network CLB

Last updated：2024-01-24 17:22:28

This document describes how to smoothly migrate your public network CLB service from the classic network to a VPC.

Note:

This example is only for reference. In actual migration, please carefully assess the impact and develop the migration plan in advance.

Scenario

Resource configuration of the classic network-based business:

The DNS domain name is resolved to the public network CLB's VIP in the classic network.

The public network CLB is bound with two CVMs (CVM 1 and CVM 2) as the backend servers.

Applications deployed in CVM 1 and CVM 2 can access the backend TencentDB for Redis and TencentDB for MySQL services.

Migration process

1. Create a VPC
2. Migrate TencentDB services
3. Create CVM instances and deploy applications
4. Create a public network CLB and associate it with the CVMs
5. Change the IP address of the DNS domain name
6. Release the classic network resources

Migration directions

1. Create a VPC as instructed in Creating VPCs.
2. Migrate TencentDB for MySQL and TencentDB for Redis instances to the VPC.

Note:

During the migration, the TencentDB instances is still connected. Both the original classic network IP and VPC IP addresses remain valid for a certain period after the migration, thus maintaining your service availability. Please complete the migration of other resources within the period.

3. Create images for the classic network-based CVM 1 and CVM 2 as instructed in Creating Custom Images and use the images to create two CVM instances in the VPC. Then test whether the CVMs can access TencentDB instances.

Note:

If restarting CVM instances during the migration is acceptable to your business, you can directly switch to VPC during off-peak hours. For detailed directions, see Switching to VPC.

4. Create a public network CLB in the VPC and associate it with the two CVMs created in the previous step. For more information, see Getting Started with CLB. Perform a health check to avoid service interruption due to an exception.
5. Resolve the DNS domain name to the public network CLB's VIP in the VPC.
6. Check whether the VPC works well. If yes, release the original public network CLB and CVM resources in the classic network to finish the migration.

Note:

The original classic network IP of a TencentDB instance will be automatically released after expiration.

Example: Configuring Hybrid Access for a Private Network CLB

Last updated：2024-01-24 17:22:28

This document provides a sample configuration for the scenario that both the VPC and classic network are required during the business migration.

Scenario

Resource configuration of the classic network-based business:
The CVM client accesses a private network CLB.
The private network CLB is bound with two CVMs (CVM 1 and CVM 2) as the real servers.
Applications deployed in CVM 1 and CVM 2 can access the backend TencentDB for MySQL services.
Requests:
Migrates resources from the classic network to a VPC
The VPC-based clients has a priority access to the private network CLB service in the classic network.
The classic network access remains available for one month after the migration.

Migration process

1. Create a VPC
2. Migrate TencentDB services
3. Configure a terminal connection
4. Create a private network CLB and configure its backend service
5. Configure a Classiclink
6. Release the classic network resources

Steps

1. Create a VPC as instructed in Creating VPCs.
2. Migrate the TencentDB for MySQL services to the VPC as instructed in Network Switch.

Note:

During the migration, the TencentDB instance still connects. Both the original classic network IP and VPC IP addresses remain valid after the migration, thus maintaining your service availability.

3. Configure a terminal connection service to allow the CVM client in the VPC to access the public network CLB service in the classic network.

Note:

A terminal connection does not support cross-region or cross-account communication. If you want to establish a terminal connection, please submit a ticket.

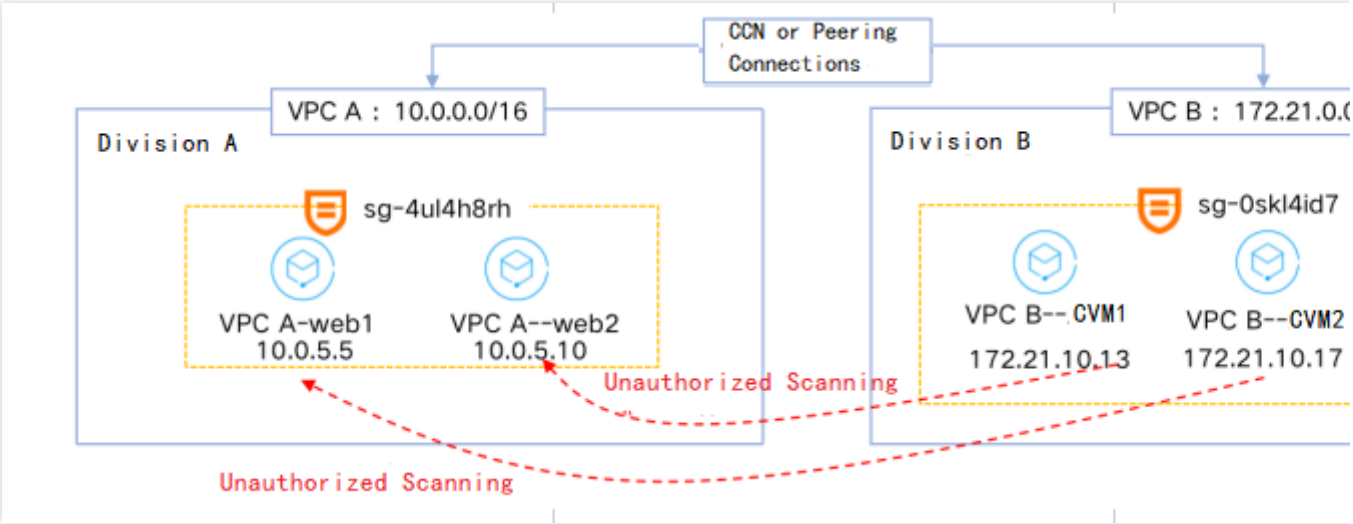
4. Create a private network CLB instance and its real server in the VPC, and configure the related services.
5. Configure a Classiclink to allow the classic network-based CVM to access the private network CLB instance in the VPC. Test whether the VPC provides services normally.
6. After the VPC service is normal and VPC-based CVM starts accessing the private network CLB in the VPC, delete the terminal connection, maintain Classiclink, and release the resources in the classic network.

Best Practices of Security Group Change
Sample of Security Group Change

Last updated：2024-01-24 17:22:28

Use Case

VPC A and VPC B are connected through a CCN connection or peering connection, allowing CVMs on the VPC B to access CVMs on VPC A through port 80.
It's detected that two IPs from VPC B, `172.21.10.13` and `172.21.10.17` , keep scan the non-open ports of VPC A.
Need to add a policy in the security group to block the IP `172.21.10.13` and `172.21.10.17` .

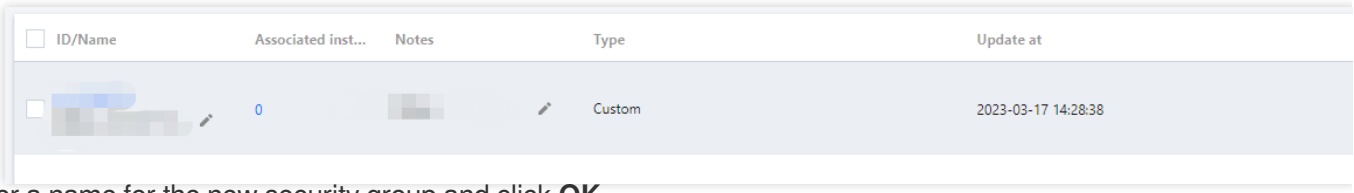


Directions

Note:
It is recommended to operate during downtime or off-peak hours of your business.

Step 1. Clone the security group

1. Log in to the VPC console and select **Security** > **Security Groups** in the left sidebar.
2. Locate the security group that is associated with the attacked instance (`sg-4ul4h8rh` in this example). Click **More** > **Clone**.



3. Enter a name for the new security group and click **OK**.

Step 2. Modify the cloned security group

Note:

In this example, the security group is associated with multiple instances. Instance `web1` in VPC A is used as a test instance, to which the cloned security group will be bound after the modification.

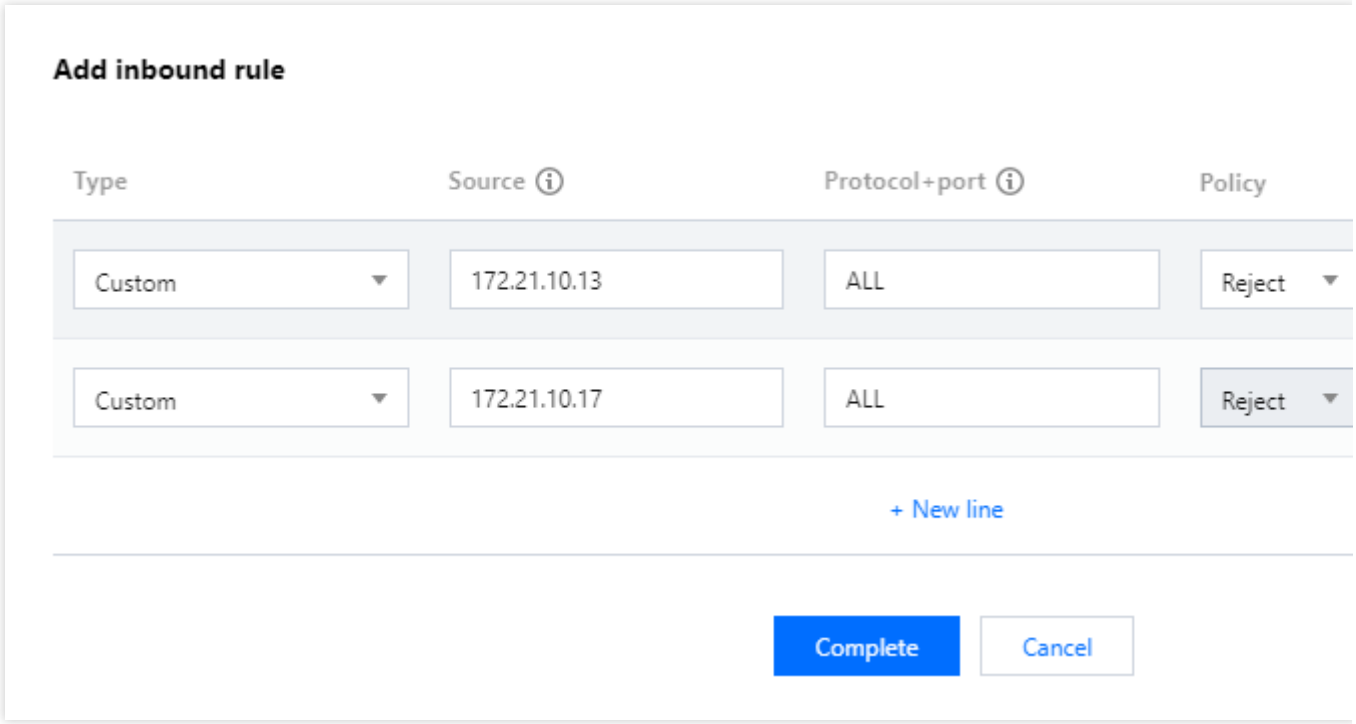
1. Click **Clone security group ID**.
2. Click **Add rule** on the tab of inbound rule.
3. In the pop-up dialog box, select **Refuse** for source IP addresses (in this sample, `172.21.10.13` and `172.21.10.17`). Click **OK**.

Note:

Common errors:

Incorrect policy: in this sample, you should select "Refuse" to block the traffic from source IP addresses. If you select "Allow", the traffic will not be blocked as desired.

IP range too wide: in this sample, only the two CVMs (CVM1 and CVM2) in VPC B need to be blocked, and CVM3 still needs to communicate with CVMs in VPC A. If `172.21.10.0/24` is also entered as the source IP address, CVM3 in VPC B will unable to access the CVMs in VPC A. Therefore, you should enter the specific IP addresses or narrow the IP range.

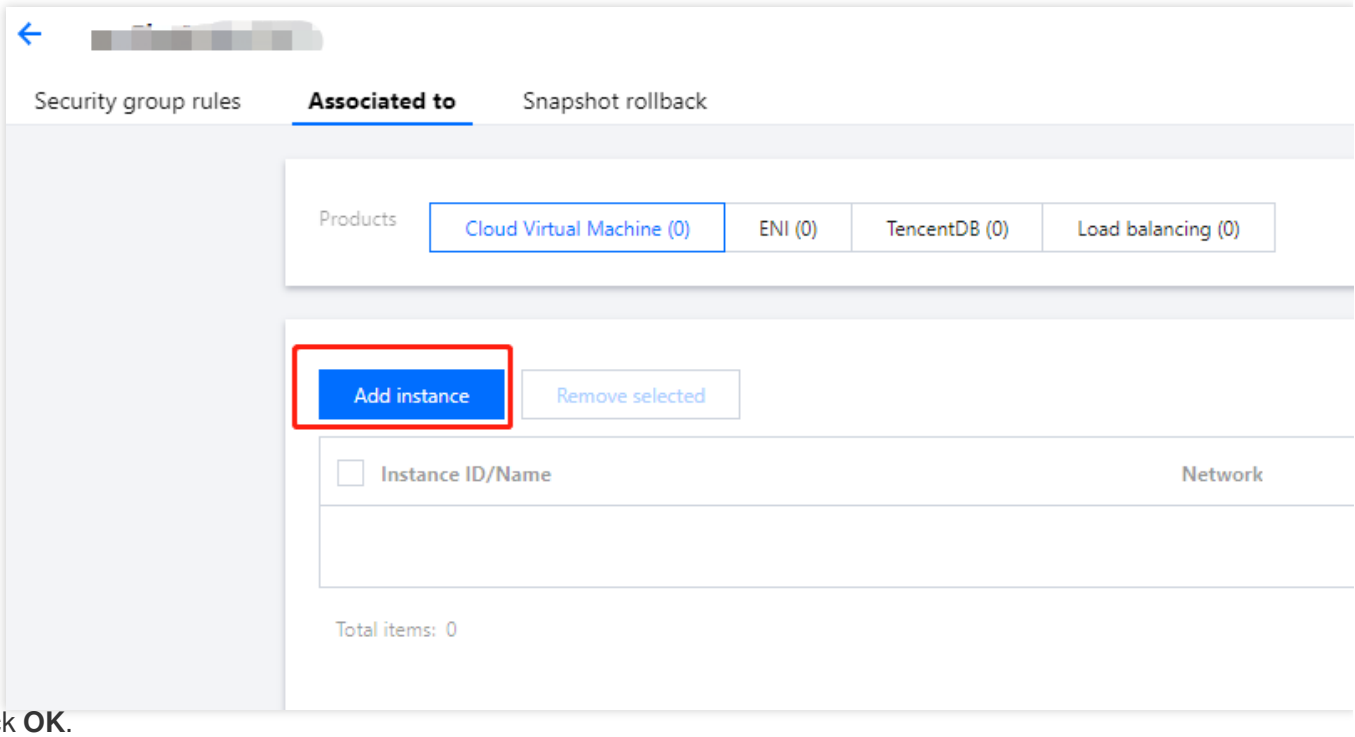


Step 3. Bind the modified cloned security group to the test instance

Note:

When multiple security groups are bound, they will be matched from top to bottom. In this sample, we move the cloned security group to the top to ensure that it is matched first.

1. Click **Manage instances** on the right of the cloned security group to go to the **Associate with instance** page.
2. Click **Add instance**, and select the desired instance. In this sample, it is the test instance web1.



3. Click **OK**.
4. Observe if the test instance continues to run properly.

If it runs properly, proceed to the next step.
If there is any exception, roll back the change and backtrack the issue. If the issue is identified in the change process, assess whether continue to modify the cloned security group and proceed the change. If yes, repeat step 2. If no, end the process.

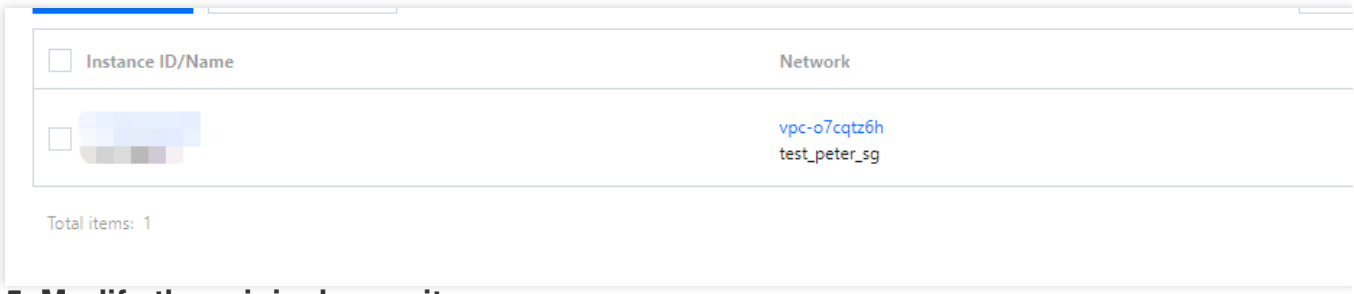
Note:

It is observed that the test instance runs properly, which means the modification for the cloned security group meets expectation.

Step 4. Unbind the cloned security group from the test instance

Note:

Unbind the cloned security group from the test instance and modify the original security group.
Note that if only one security group is associated with an instance, you cannot unbind the security group.
1. Click **Manage instances** on the right of the cloned security group to go to the **Associate with instance** page.
2. Click **Remove from the security group** on the right of the instance, from which the security group needs to be unbound (in this sample, it is the test instance web1).



Step 5. Modify the original security group

1. Click **Original security group ID**. In this sample, it is the ID of the security group bound with web1 and web2 in VPC A.
2. Click **Add rule** on the tab of inbound rule.
3. In the pop-up dialog box, select "Refuse" for source IP addresses (in this sample, `172.21.10.13` and `172.21.10.17`). Click **OK**.

Step 6. Roll back the change unconditionally after two minutes

Note:

The purpose is to find out in time whether there is a temporary impact on associated CVMs caused by the change of the security group, so as to make response decisions and reduce the impact.
1. Wait about two minutes after the original security group rule is modified, then delete the modification made to the security group in step 5.
2. Observe whether the associated CVMs run properly for at least 30 minutes.
If all CVMs run properly, proceed to the next step.

If there is any exception, it is suggested that you end the change immediately and restart it after backtracking the issue and assessing the impact.

Note:

In this sample, all CVMs run properly during the 30 minutes.

Step 7. Modify the original security group again

1. Click **Original security group ID**. In this sample, it is the ID of the security group bound with web1 and web2 in VPC A.
2. Click **Add rule** on the tab of inbound rule.
3. In the pop-up dialog box, select "Refuse" for source IP addresses (in this sample, `172.21.10.13` and `172.21.10.17`). Click **OK**.
4. Observe whether associated CVMs run properly for at least 30 minutes.

If all CVMs run properly, it indicates the change is successful.

If there is any exception, it is suggested that you end the change immediately and restart it after backtracking the issue and assessing the impact.

Note:

In this sample, all CVMs run properly. The change process is completed.

Configuring CVM Instance as Public Gateway

Last updated：2024-01-24 17:22:28

Warning:

Using a single CVM instance as the public gateway has the risk of single point of failure. We recommend you use NAT Gateway in the production environment.

As of December 6, 2019, Tencent Cloud will no longer support configuring a CVM instance as the public gateway on the CVM purchase page. If you need to configure a gateway, follow the instructions below.

Overview

If some of your CVM instances in a Tencent Cloud VPC do not have common public IPs but you need to access the public network, you can use a CVM instance with a common public IP or EIP. The public gateway CVM instance translates the source IP for outbound traffic. When other CVM instances access the public network through the public gateway CVM instance, the source IPs will be translated into the public IP of the public gateway CVM instance.

Prerequisites

You have logged in to the CVM console.

As a public gateway CVM instance can forward route forwarding requests only from subnets other than the one it resides, it must be in different subnets from the CVM instances that need to access the public network through it.

A public gateway CVM instance must be a Linux CVM instance, as a Windows CVM instance cannot be used as a public gateway.

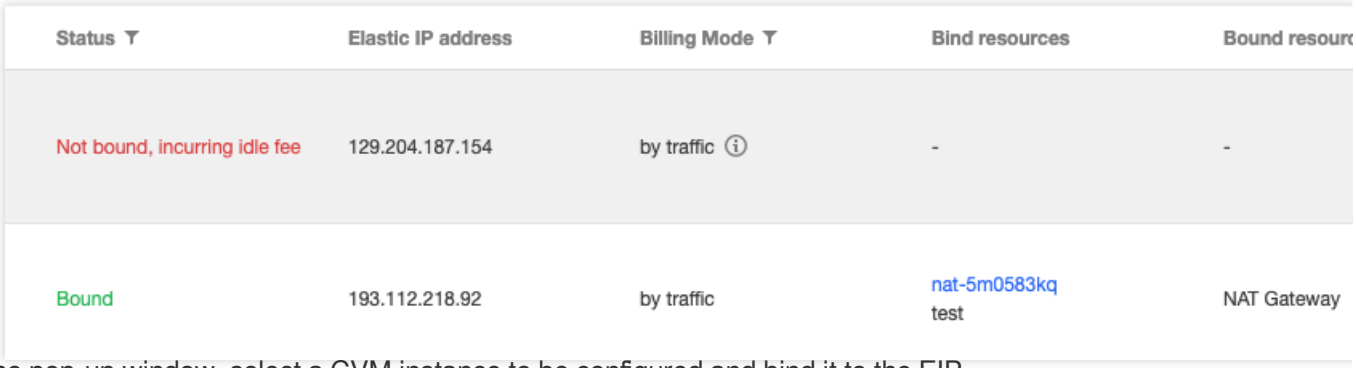
Directions

Step 1. Bind an EIP (optional)

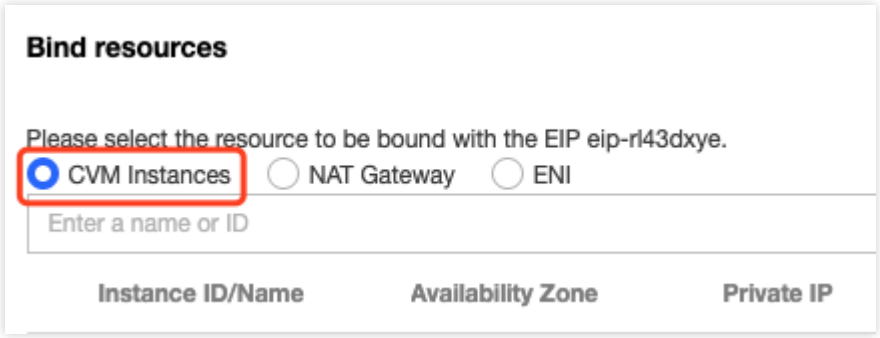
Note:

Skip this step if the public gateway CVM already has a public IP.

1. Log in to the CVM console and select **EIP** on the left sidebar.
2. Locate the target EIP and select **More** > **Bind** in the **Operation** column.



3. In the pop-up window, select a CVM instance to be configured and bind it to the EIP.

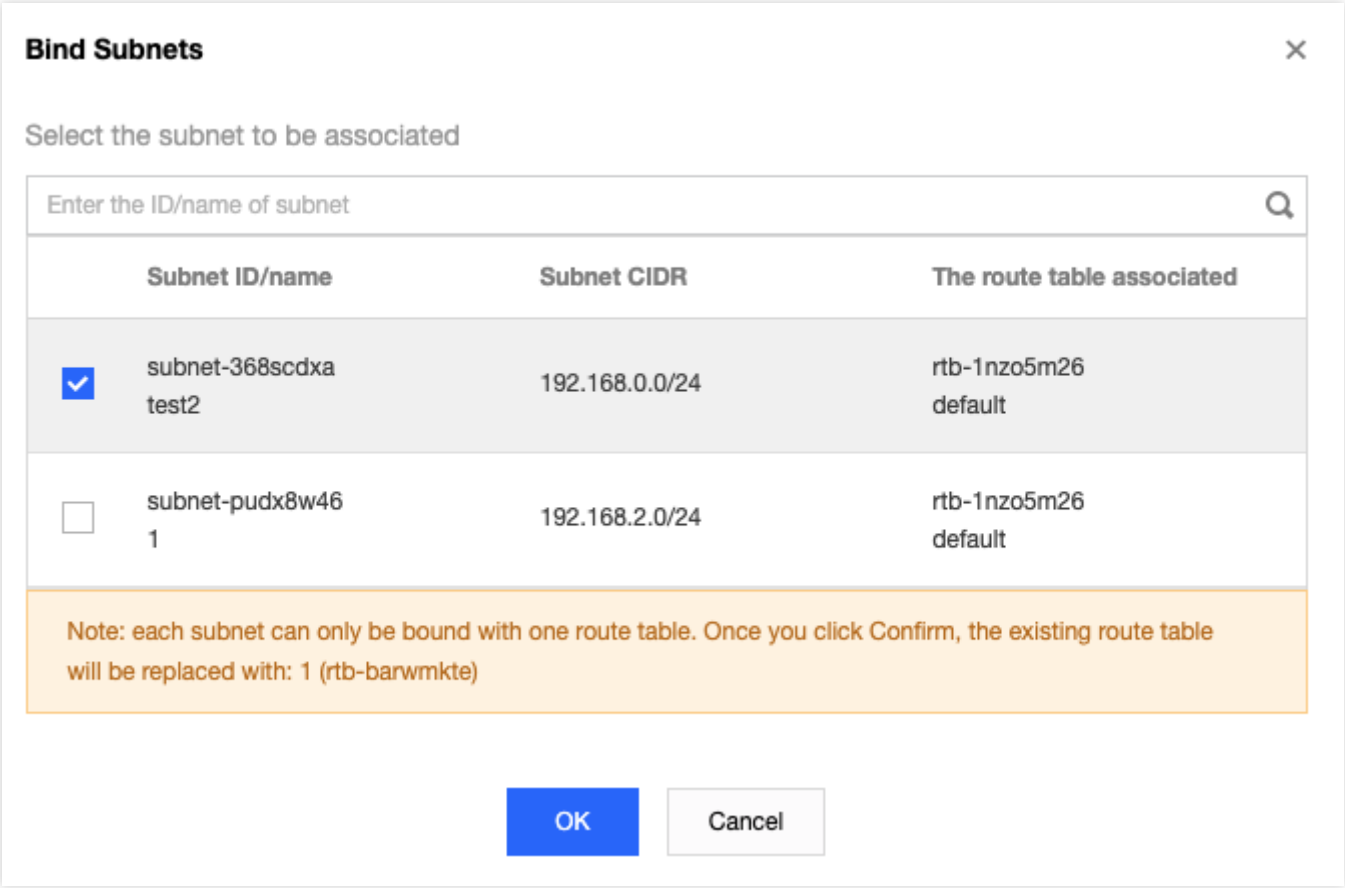


Step 2. Configure a route table for the gateway subnet

Note:

The gateway subnet and other subnets cannot share the same route table. You need to create a separate route table for the gateway subnet and associate them.

1. Create a custom route table.
2. Associate the route table with the subnet where the public gateway CVM resides.



Step 3. Configure a route table for other subnets

Configure a route table for other subnets and a default route through the public gateway CVM instance, so that the CVM instances within these subnets can access the public network through the route forwarding capability of the public gateway.

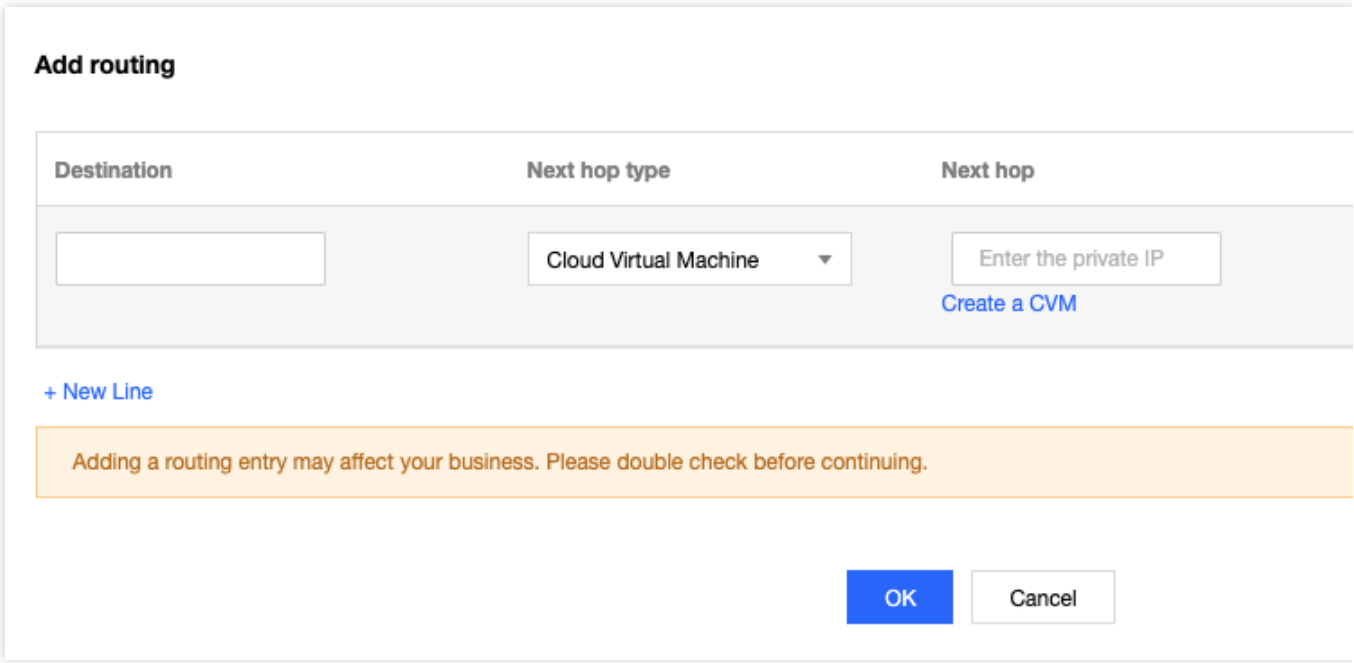
Add the following routing policies to the route table:

Destination: The public IP you want to access.

Next hop type: CVM.

Next hop: Private IP of the CVM instance to which the EIP is bound in step 1.

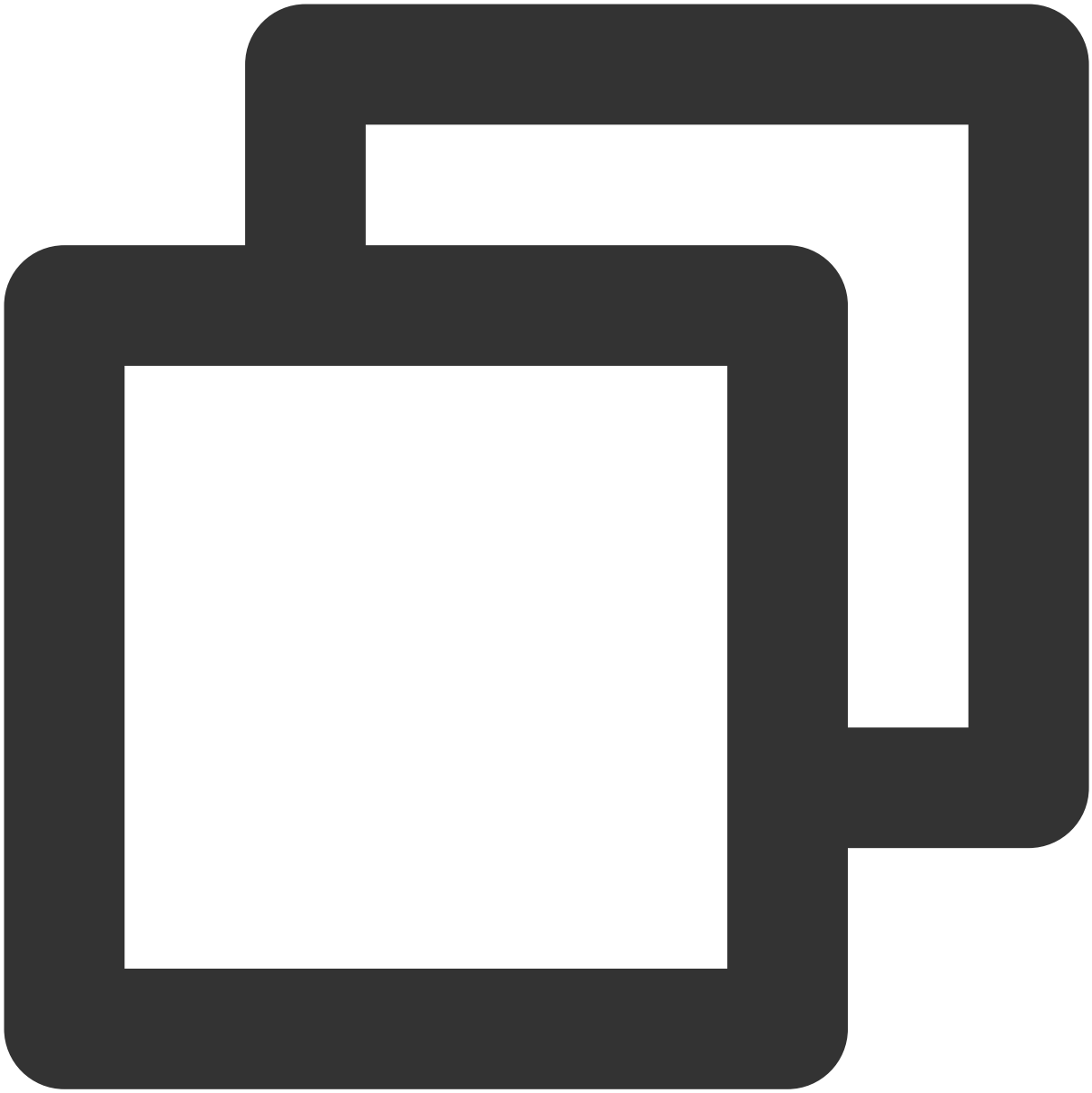
For more information, see Managing Routing Policies.



Step 4. Configure a public gateway

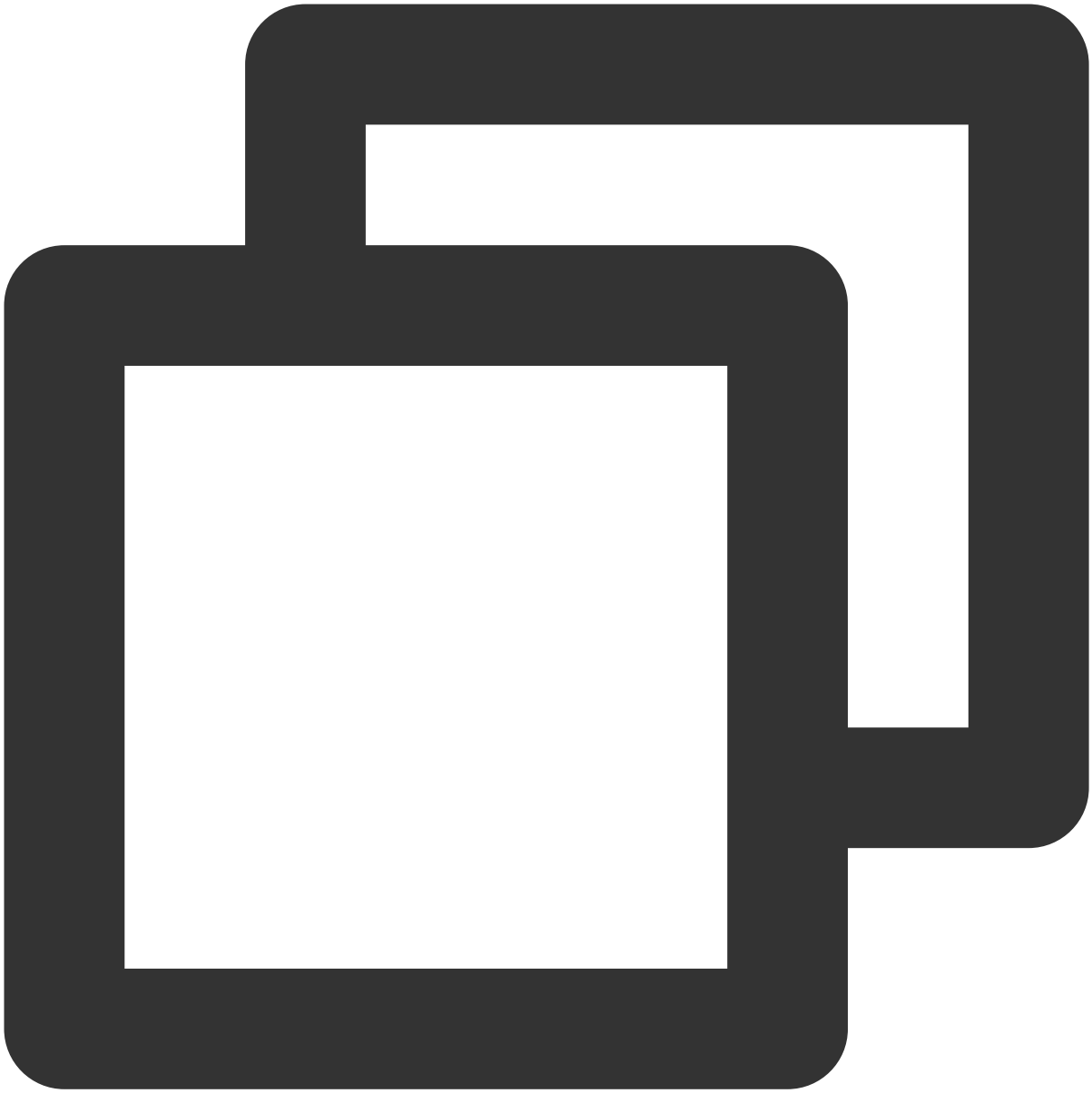
1. Log in to the public gateway CVM instance and perform the following operations to enable the network forwarding and NAT proxy features:

1.1 Run the following command to create the `vpcGateway.sh` script in `usr/local/sbin` .



```
vim /usr/local/sbin/vpcGateway.sh
```

1.2 Press **i** to switch to the edit mode and add the following code to the script.

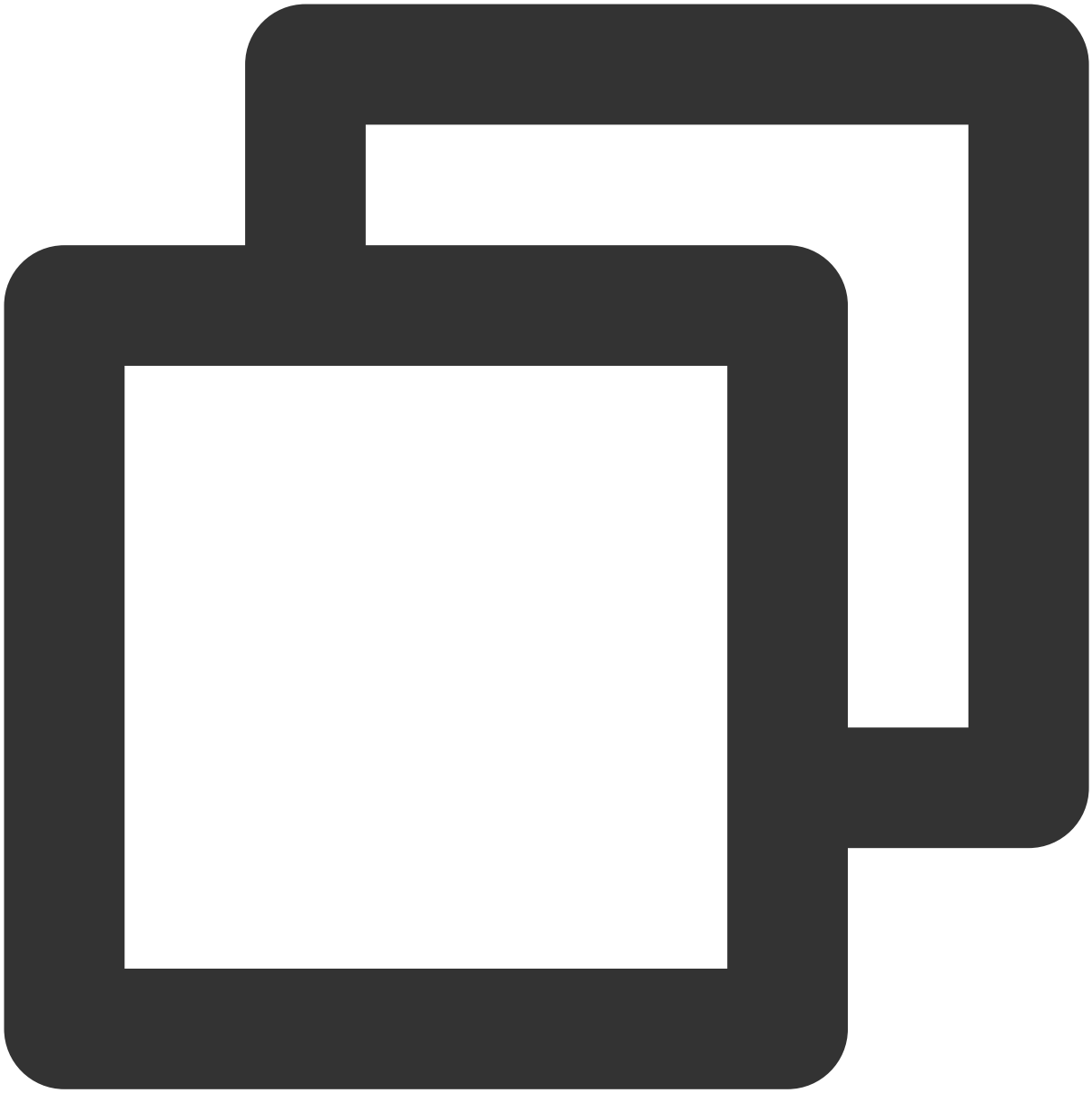


```
#!/bin/bash
echo "------------------------------------------------"
echo " `date`"
echo "(1)ip_forward config......"
file="/etc/sysctl.conf"
grep -i "^net\\.ipv4\\.ip_forward.*" $file &>/dev/null && sed -i \\
's/net\\.ipv4\\.ip_forward.*/net\\.ipv4\\.ip_forward = 1/' $file || \\
echo "net.ipv4.ip_forward = 1" >> $file
echo 1 >/proc/sys/net/ipv4/ip_forward
[ `cat /proc/sys/net/ipv4/ip_forward` -eq 1 ] && echo "-->ip_forward:Success" || \\
echo "-->ip_forward:Fail"
```

```
echo "(2)Iptables set......"
iptables -t nat -A POSTROUTING -j MASQUERADE && echo "-->nat:Success" || echo "-->n
iptables -t mangle -A POSTROUTING -p tcp -j TCPOPTSTRIP --strip-options timestamp &
echo "-->mangle:Success" || echo "-->mangle:Fail"
echo "(3)nf_conntrack config......"
echo 262144 > /sys/module/nf_conntrack/parameters/hashsize
[ `cat /sys/module/nf_conntrack/parameters/hashsize` -eq 262144 ] && \\
echo "-->hashsize:Success" || echo "-->hashsize:Fail"
echo 1048576 > /proc/sys/net/netfilter/nf_conntrack_max
[ `cat /proc/sys/net/netfilter/nf_conntrack_max` -eq 1048576 ] && \\
echo "-->nf_conntrack_max:Success" || echo "-->nf_conntrack_max:Fail"
echo 10800 >/proc/sys/net/netfilter/nf_conntrack_tcp_timeout_established \\
[ `cat /proc/sys/net/netfilter/nf_conntrack_tcp_timeout_established` -eq 10800 ] \\
&& echo "-->nf_conntrack_tcp_timeout_established:Success" || \\
echo "-->nf_conntrack_tcp_timeout_established:Fail"
```

1.3 Click **Esc** and enter **:wq** to save and close the file.

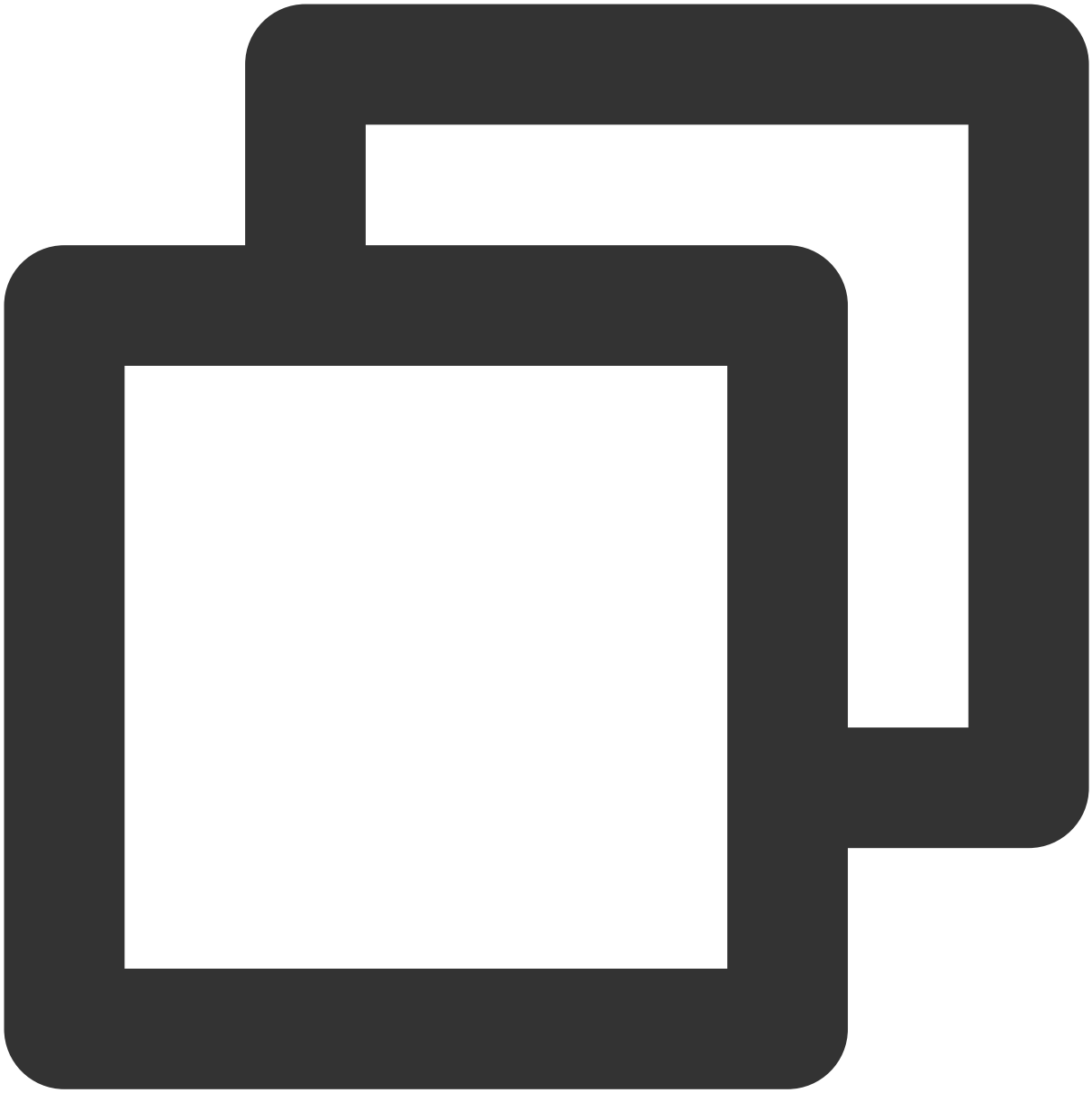
1.4 Run the following command to set the script permission.



```
chmod +x /usr/local/sbin/vpcGateway.sh
echo "/usr/local/sbin/vpcGateway.sh >/tmp/vpcGateway.log 2>&1" >> /etc/rc.local
```

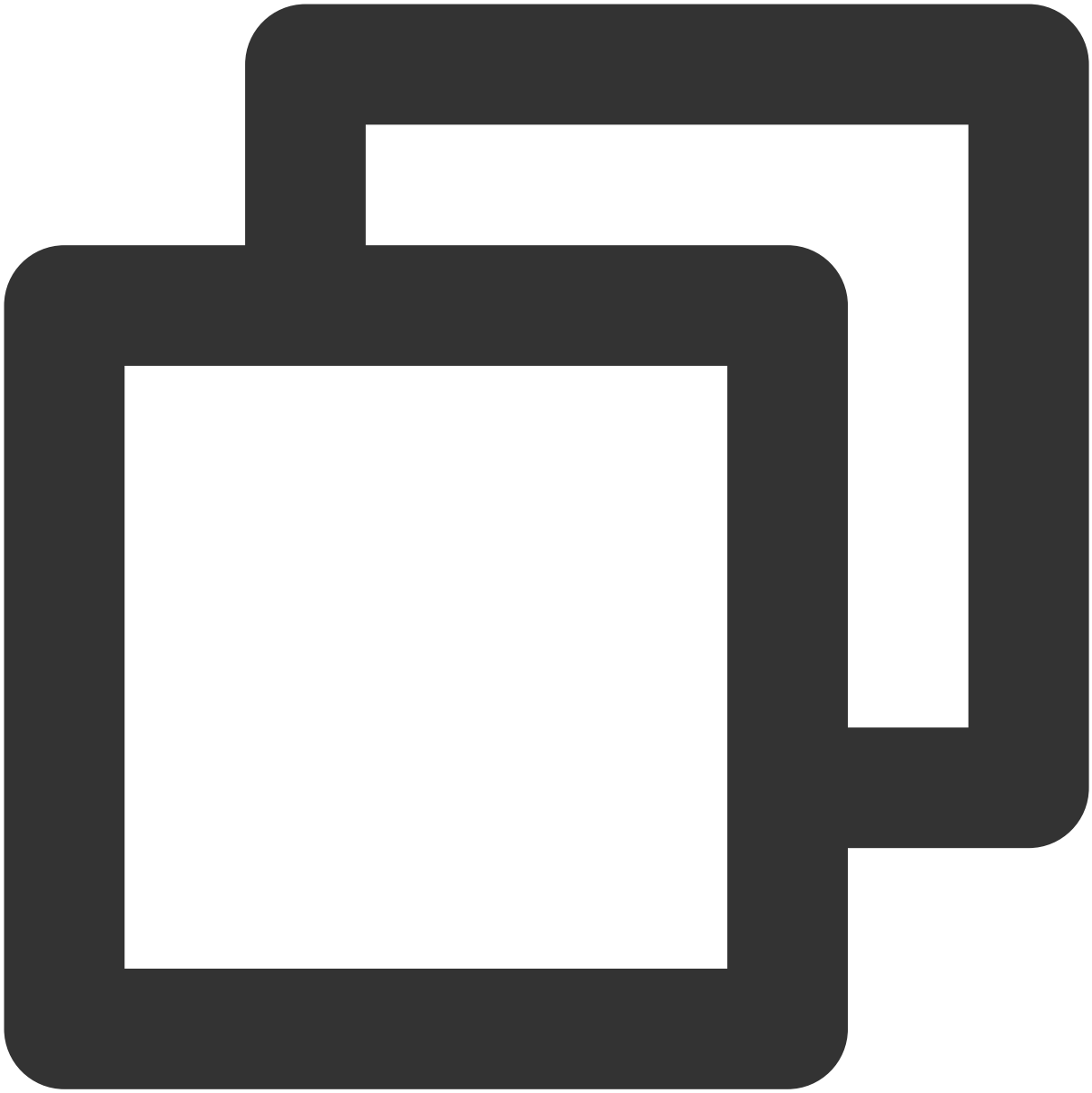
2. Set the RPS of the public gateway.

2.1 Run the following command to create the `set_rps.sh` script in `usr/local/sbin` .



```
vim /usr/local/sbin/set_rps.sh
```

2.2 Press **i** to switch to the edit mode and add the following code to the script.



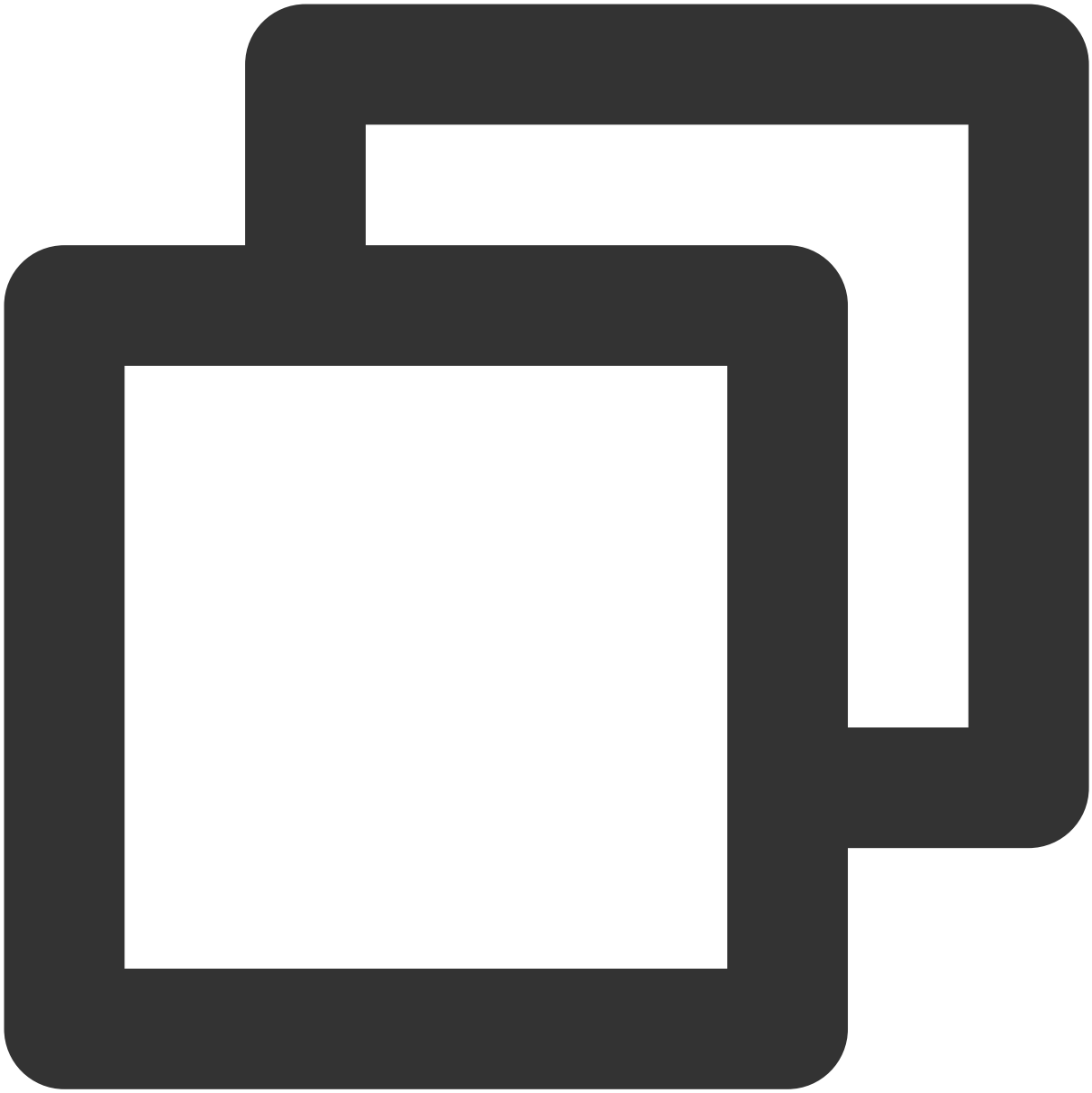
```
# !/bin/bash
echo "------------------------------------------"
date
mask=0
i=0
total_nic_queues=0
get_all_mask() {
local cpu_nums=$1
if [ $cpu_nums -gt 32 ]; then
mask_tail=""
mask_low32="ffffffff"
```

```
idx=$((cpu_nums / 32))
cpu_reset=$((cpu_nums - idx * 32))
if [ $cpu_reset -eq 0 ]; then
mask=$mask_low32
for ((i = 2; i <= idx; i++)); do
mask="$mask,$mask_low32"
done
else
for ((i = 1; i <= idx; i++)); do
mask_tail="$mask_tail,$mask_low32"
done
mask_head_num=$((2 ** cpu_reset - 1))
mask=$(printf "%x%s" $mask_head_num $mask_tail)
fi
else
mask_num=$((2 ** cpu_nums - 1))
mask=$(printf "%x" $mask_num)
fi
echo $mask
}
set_rps() {
if ! command -v ethtool &>/dev/null; then
source /etc/profile
fi
ethtool=$(which ethtool)
cpu_nums=$(cat /proc/cpuinfo | grep processor | wc -l)
if [ $cpu_nums -eq 0 ]; then
exit 0
fi
mask=$(get_all_mask $cpu_nums)
echo "cpu number:$cpu_nums mask:0x$mask"
ethSet=$(ls -d /sys/class/net/eth*)
for entry in $ethSet; do
eth=$(basename $entry)
nic_queues=$(ls -l /sys/class/net/$eth/queues/ | grep rx- | wc -l)
if (($nic_queues == 0)); then
continue
fi
cat /proc/interrupts | grep "LiquidIO.*rxtx" &>/dev/null
if [ $? -ne 0 ]; then # not smartnic
#multi queue don't set rps
max_combined=$(
$ethtool -l $eth 2>/dev/null | grep -i "combined" | head -n 1 | awk '{print $2}'
)
#if ethtool -l $eth goes wrong.
[[ ! "$max_combined" =~ ^[0-9]+$ ]] && max_combined=1
if [ ${max_combined} -ge ${cpu_nums} ]; then
```

```
echo "$eth has equally nic queue as cpu, don't set rps for it..."
continue
fi
else
echo "$eth is smartnic, set rps for it..."
fi
echo "eth:$eth queues:$nic_queues"
total_nic_queues=$(($total_nic_queues + $nic_queues))
i=0
while (($i < $nic_queues)); do
echo $mask >/sys/class/net/$eth/queues/rx-$i/rps_cpus
echo 4096 >/sys/class/net/$eth/queues/rx-$i/rps_flow_cnt
i=$(($i + 1))
done
done
flow_entries=$((total_nic_queues * 4096))
echo "total_nic_queues:$total_nic_queues flow_entries:$flow_entries"
echo $flow_entries >/proc/sys/net/core/rps_sock_flow_entries
}
set_rps
```

2.3 Click **Esc** and enter **:wq** to save and close the file.

2.4 Run the following command to set the script permission.



```
chmod +x /usr/local/sbin/set_rps.sh
echo "/usr/local/sbin/set_rps.sh >/tmp/setRps.log 2>&1" >> /etc/rc.local
chmod +x /etc/rc.d/rc.local
```

3. Restart the public gateway CVM instance to apply the configuration. Then, test whether a CVM instance without a public IP can access the public network.

Building HA Primary/Secondary Cluster with HAVIP + Keepalived

Last updated：2024-01-24 17:22:28

This document describes how to use Keepalived with HAVIP to build a high availability primary/secondary cluster in the Tencent Cloud VPC.

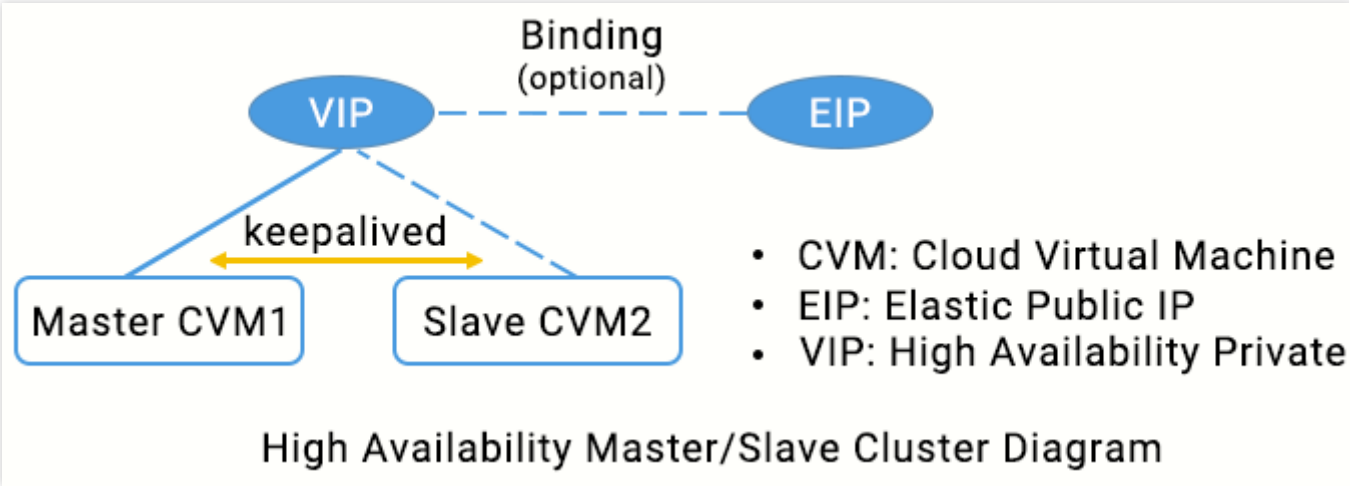
Note:

HAVIP is currently in beta, and switching between primary/secondary servers may take 10 seconds. To try it out, please submit a ticket.

Basic Principle

Typically, a high availability primary/secondary cluster consists of two servers: an active primary server and a standby secondary server. The two servers share the same VIP (virtual IP) which is only valid for the primary server. When the primary server fails, the secondary server will take over the VIP to continue providing services. This mode is widely used in MySQL source/replica switch and Ngnix web access.

Keepalived is a VRRP-based high availability software that can be used to build a high availability primary/secondary cluster among VPC-based CVMs. To use Keepalived, first complete its configuration in the `keepalived.conf` file.



In traditional physical networks, the primary/secondary status can be negotiated with Keepalived's VRRP protocol. The primary device periodically sends free-of-charge ARP messages to purge the MAC table or terminal ARP table of the uplink exchange to trigger the VIP migration to the primary device.

In a Tencent Cloud VPC, a high availability primary/secondary cluster can also be implemented by deploying Keepalived on CVMs, with the following differences:

The VIP must be a HAVIP applied for from Tencent Cloud.

HAVIP is subnet-sensitive and can only be bound to a server under the same subnet through announcement.

Note

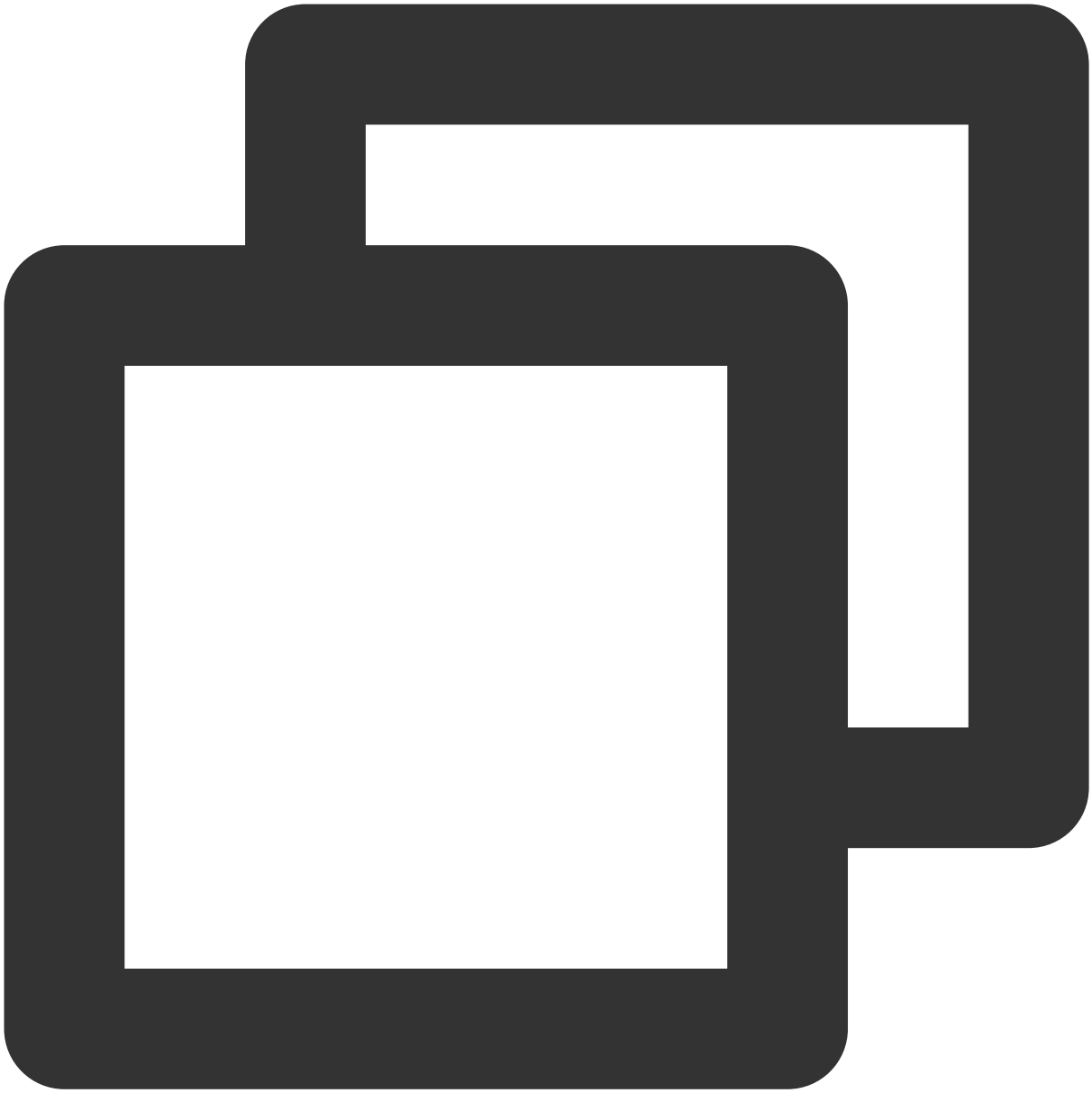
We recommend VRRP communications in unicast mode.

Note:

In this document, we use unicast mode for VRRP communications. If you want to use multicast mode, please submit a ticket. After the application is approved, enable the VPC multicast feature. For more information, see Enabling or Disabling Multicast. You **do not need to configure** the "unicast_peer" parameter in the keepalived configuration file.

We recommend that you use Keepalived **1.2.24 or later versions**.

Ensure that the `garp` parameters have been configured. Because Keepalived relies on ARP messages to update the IP address, these configurations ensure that the primary device always sends ARP messages for the communication.



```
garp_master_delay 1
garp_master_refresh 5
```

Configure a unique VRRP router ID for each primary/secondary cluster in the VPC.

Do not use the strict mode. Ensure the “vrrp_strict” configurations have been deleted.

Control the number of HAVIPs bound to a single ENI to be no more than 5. If you need to use multiple VIPs, add or modify `vrrp_garp_master_repeat 1` in the “global_defs” section of the Keepalived configuration file.

Specify the `adver_int` parameter properly to balance anti-network jitter and disaster recovery speed. If the `advert_int` parameter is set too small, frequent switchover and temporary **active-active (split brain)** may

occur in case of network jitter. If the `advert_int` parameter is set too large, it takes a long time for primary-secondary switching to take place after the primary server fails, which cause long service interruption. **Please fully assess the impact of the active-active (split brain) status on your businesses.**

Set the `interval` parameter in the specific execution item of `track_script` script (such as `checkhaproxy`) to a larger value, avoiding the `FAULT` status caused by script execution timeout.

Optional: be aware of increased disk usage due to log printing. This can be solved using logrotate or other tools.

Operation Directions

Note:

This document uses the following environments as an example. Please replace with your actual configurations.

Primary CVM: HAVIP-01, 172.16.16.5

Secondary CVM: HAVIP-02, 172.16.16.6

HAVIP: 172.16.16.12

EIP: 81.71.14.118

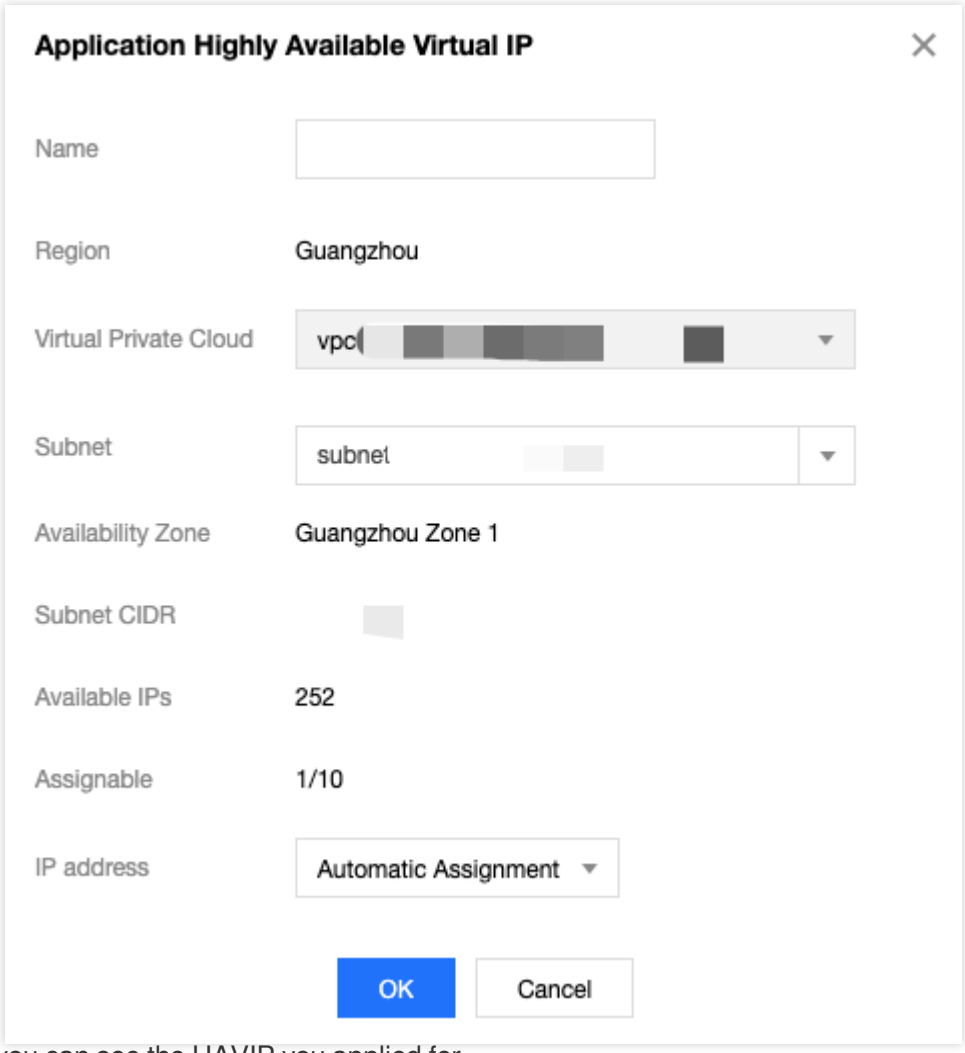
Image: CentOS 7.6 64-bit

Step 1: apply for a VIP

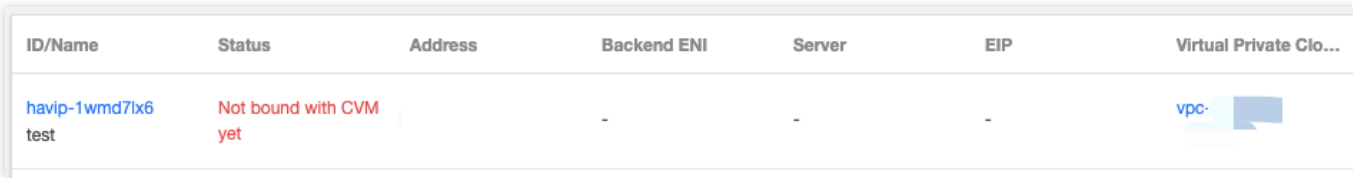
1. Log in to the VPC console.
2. Select **IP and ENI** > **HAVIP** in the left sidebar to enter the HAVIP management page.
3. Select the target region on the HAVIP management page and click **Apply**.
4. In the pop-up dialog box, enter the name, select a VPC and a subnet for the HAVIP, and click **OK**.

Note:

The IP address of the HAVIP can be automatically assigned or manually specified. If you choose to enter an IP address, make sure that the entered private IP address is within the subnet IP range and is not a reserved IP address of the system. For example, if the subnet IP range is `10.0.0.0/24`, the entered private IP address should be within `10.0.0.2 - 10.0.0.254`.



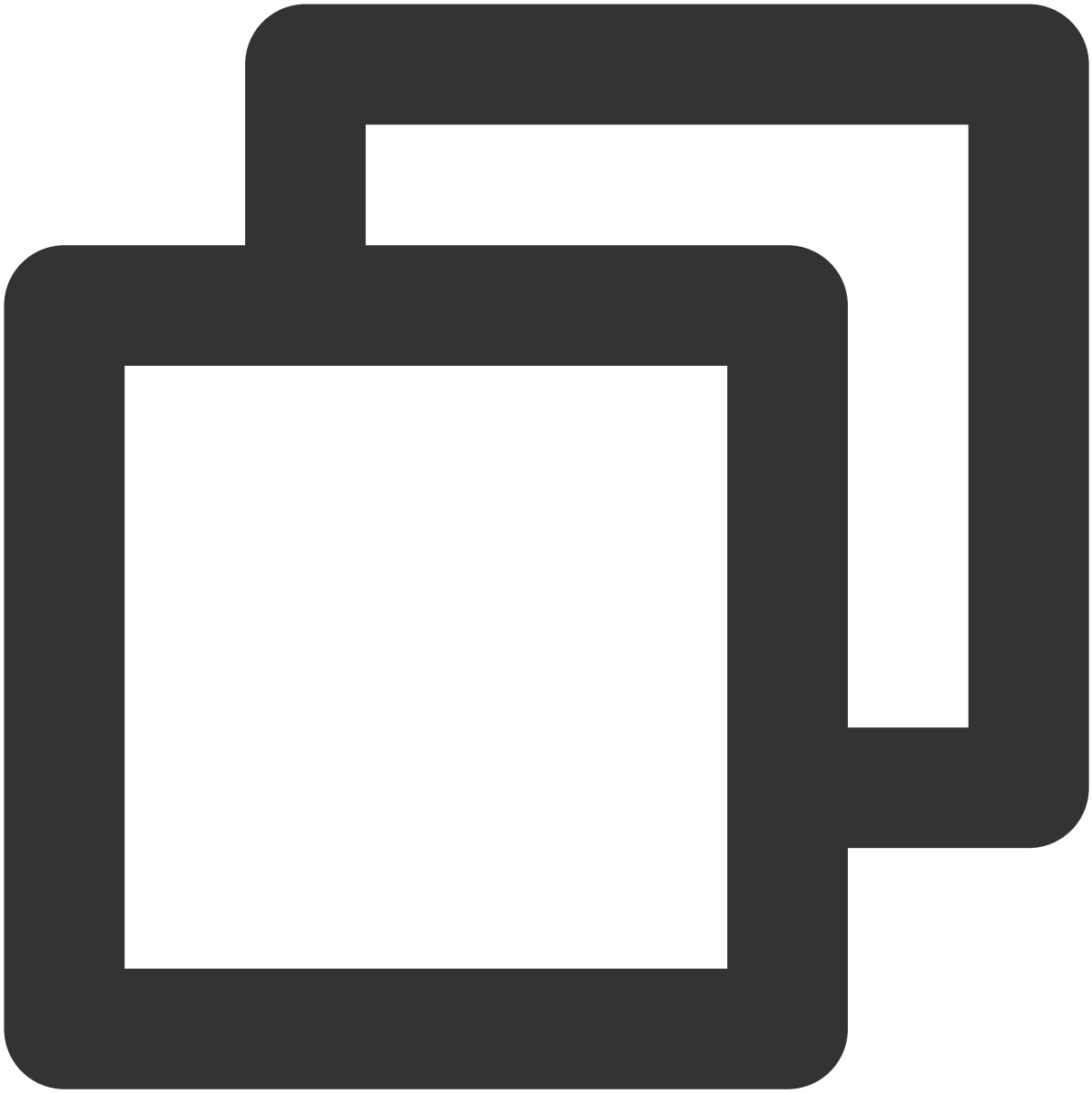
Then you can see the HAVIP you applied for.



Step 2: install Keepalived (version 1.2.24 or later) on primary and secondary CVMs

This document uses CentOS 7.6 as an example to install Keepalived.

1. Run the following command to verify whether the Keepalived version meets the requirements.



```
yum list keepalived
```

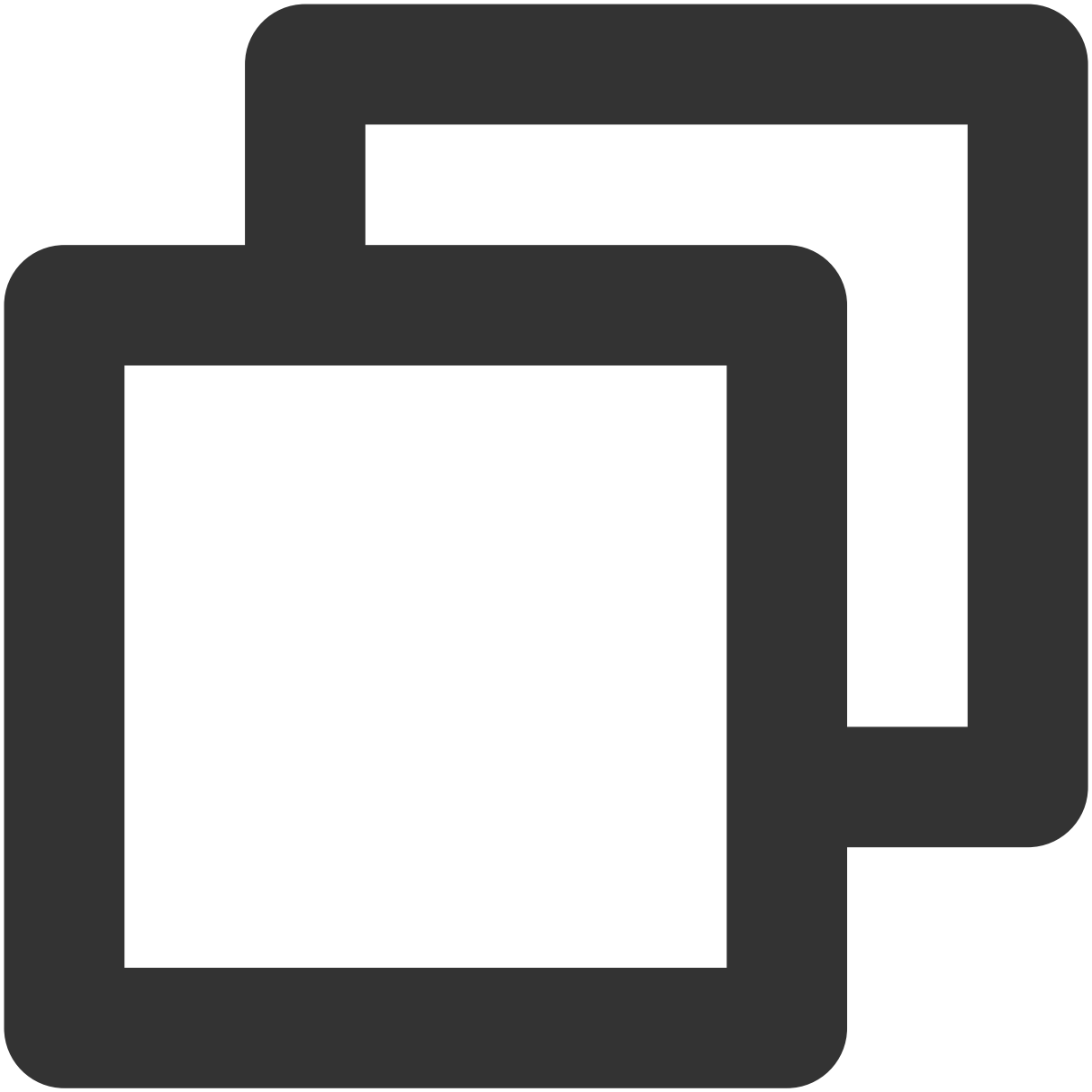
If yes, proceed to step 2

If no, proceed to step 3

2. In

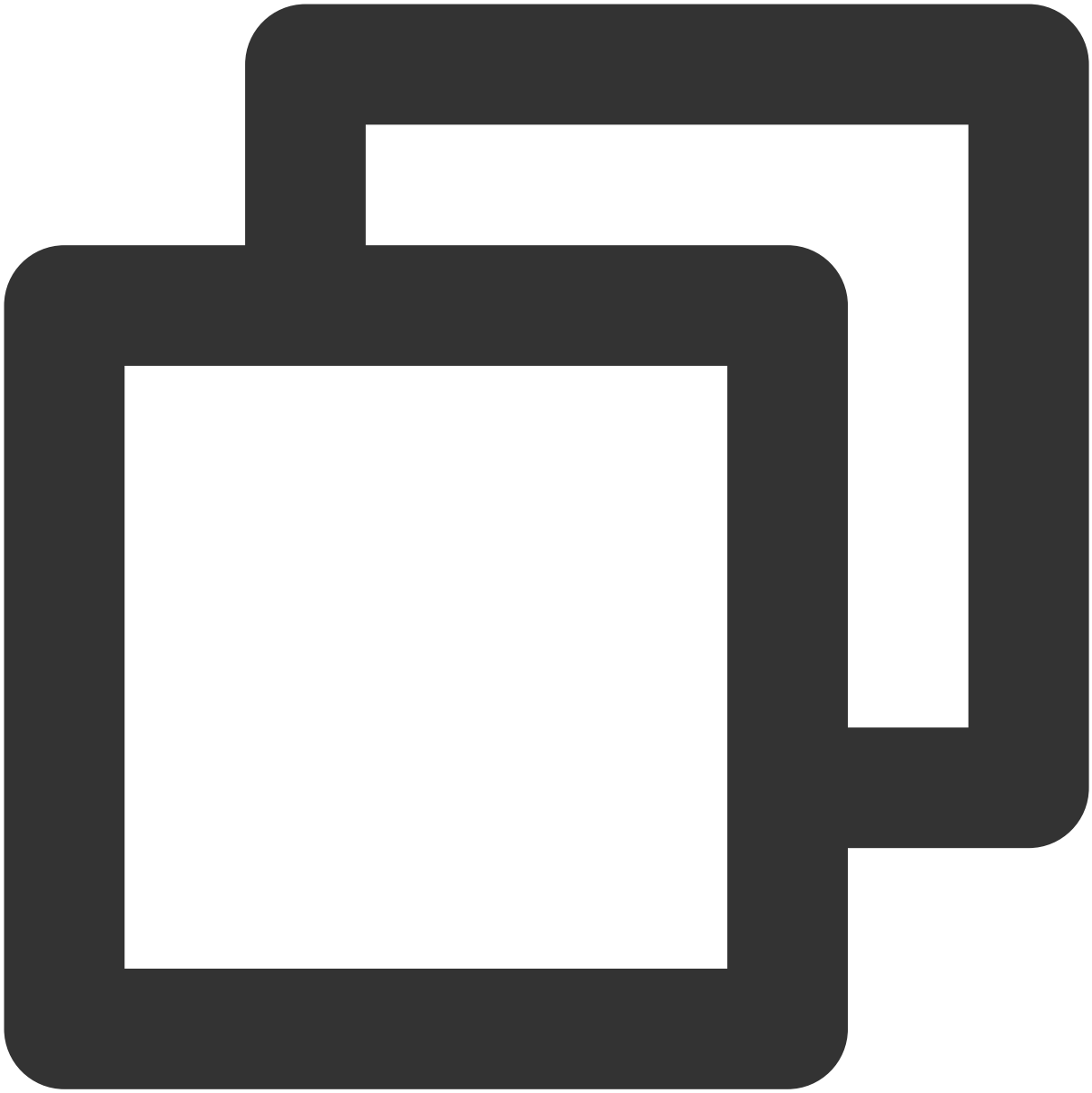
stall th

e software package using the `yum` command.



```
yum install -y keepalived
```

3. Install the software package using the source code.



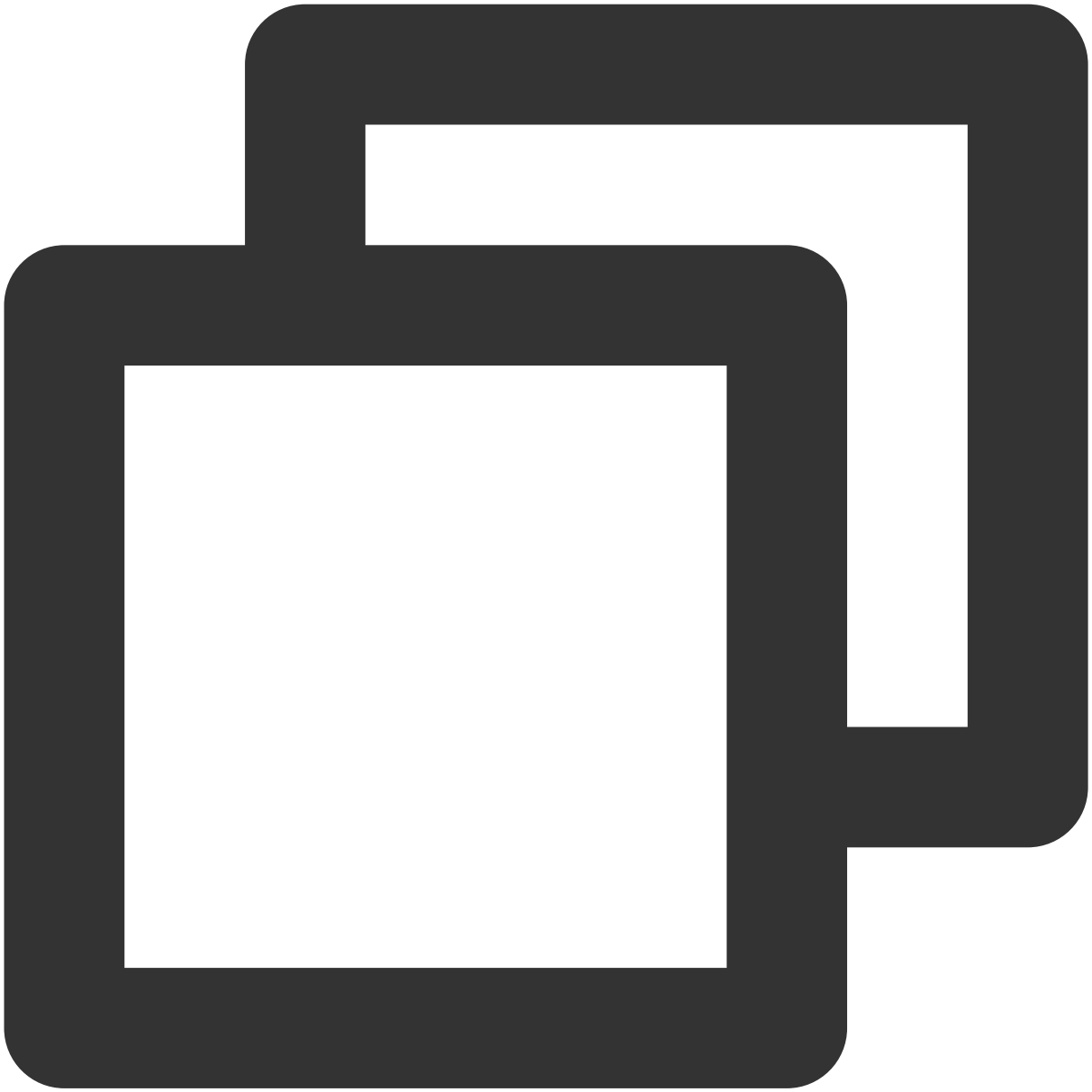
```
tar zxvf keepalived-1.2.24.tar.gz
cd keepalived-1.2.24
./configure --prefix=/
make; make install
chmod +x /etc/init.d/keepalived   // Prevent occurrence of env: /etc/init.d/keepali
```

Step 3: configure Keepalived, and bind HAVIP to the primary and secondary CVMs.

1. Log in to the primary CVM HAVIP-01 and run `vim /etc/keepalived/keepalived.conf` to modify its configurations.

Note:

In this example, HAVIP-01 and HAVIP-02 are configured with the same weight. Both are in the **BACKUP** status, with a priority of 100. This will reduce the number of switchovers caused by network jitter.



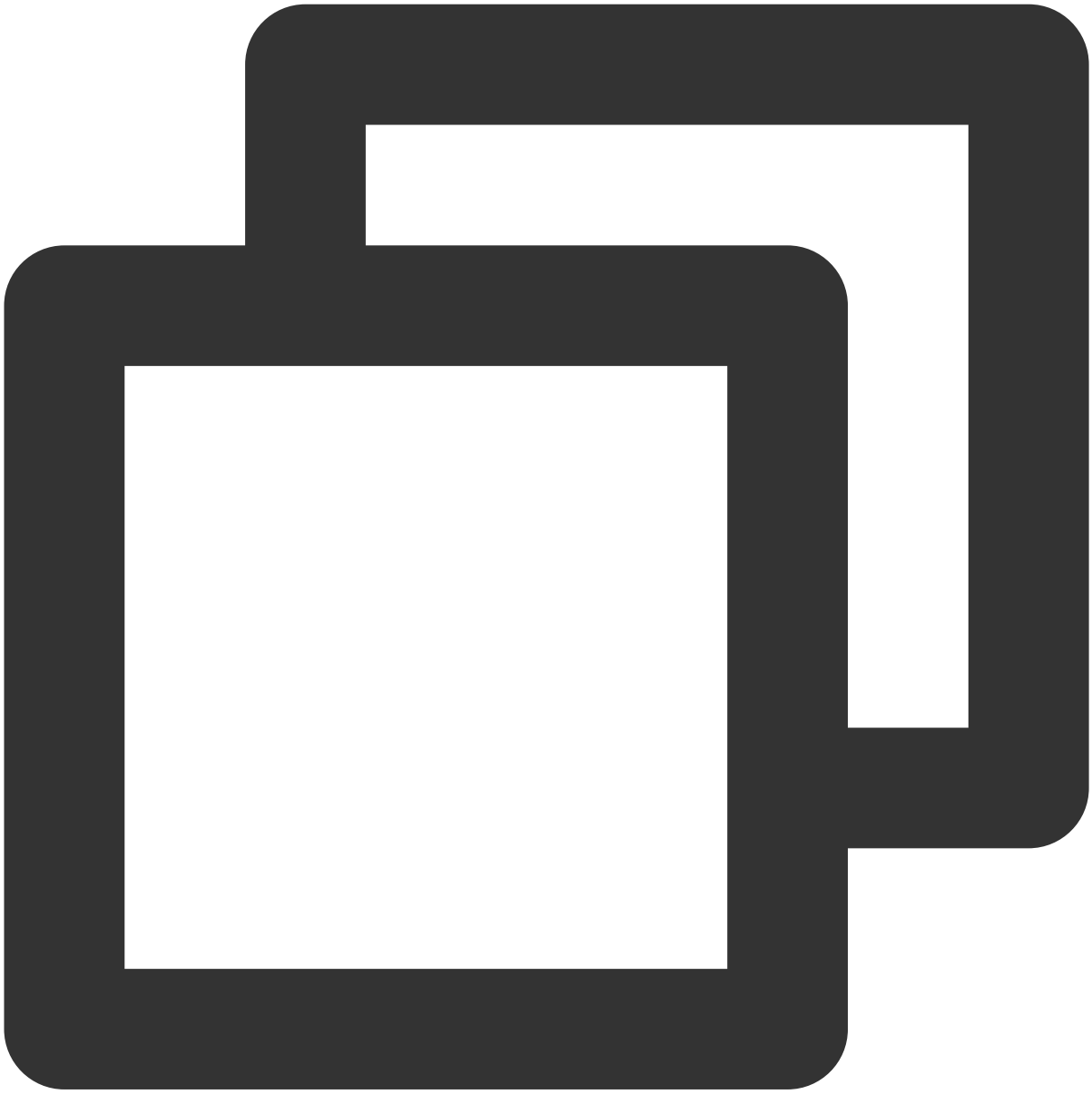
```
! Configuration File for keepalived
global_defs {
   notification_email {
     acassen@firewall.loc
     failover@firewall.loc
     sysadmin@firewall.loc
   }
```

```
    notification_email_from Alexandre.Cassen@firewall.loc
    smtp_server 192.168.200.1
    smtp_connect_timeout 30
    router_id LVS_DEVEL
    vrrp_skip_check_adv_addr
    vrrp_garp_interval 0
    vrrp_gna_interval 0
}
vrrp_script checkhaproxy
{
     script "/etc/keepalived/do_sth.sh"   # Check whether the service process ru
     interval 5
}
vrrp_instance VI_1 {
# Select proper parameters for the primary and secondary CVMs.
state BACKUP              # Set the initial status to `Backup`
    interface eth0          # The ENI such as `eth0` used to bind a VIP
    virtual_router_id 51        # The`virtual_router_id` value for the cluster
    nopreempt                   # Non-preempt mode
    # preempt_delay 10      # Effective only when `state` is `MASTER`
    priority 100            # Configure the same weight for the two devices
    advert_int 5
    authentication {
        auth_type PASS
        auth_pass 1111
    }
    unicast_src_ip 172.16.16.5 # Private IP address of the local device
    unicast_peer {
        172.16.16.6               # IP address of the peer device
    }
    virtual_ipaddress {
        172.16.16.12              # HAVIP
    }
    notify_master "/etc/keepalived/notify_action.sh MASTER"
    notify_backup "/etc/keepalived/notify_action.sh BACKUP"
    notify_fault "/etc/keepalived/notify_action.sh FAULT"
    notify_stop "/etc/keepalived/notify_action.sh STOP"
    garp_master_delay 1    # How long it will take before the ARP cache can be u
    garp_master_refresh 5   # Time interval between which the primary node sends

    track_interface {
                eth0               # ENI that bound with VIP, such as `eth0`
        }
    track_script {
       checkhaproxy
    }
}
```

2. Press **Esc** to exit the edit mode and enter **:wq!** to save and close the file.

3. Log in to the secondary CVM HAVIP-02 and run `vim /etc/keepalived/keepalived.conf` to modify its configurations.

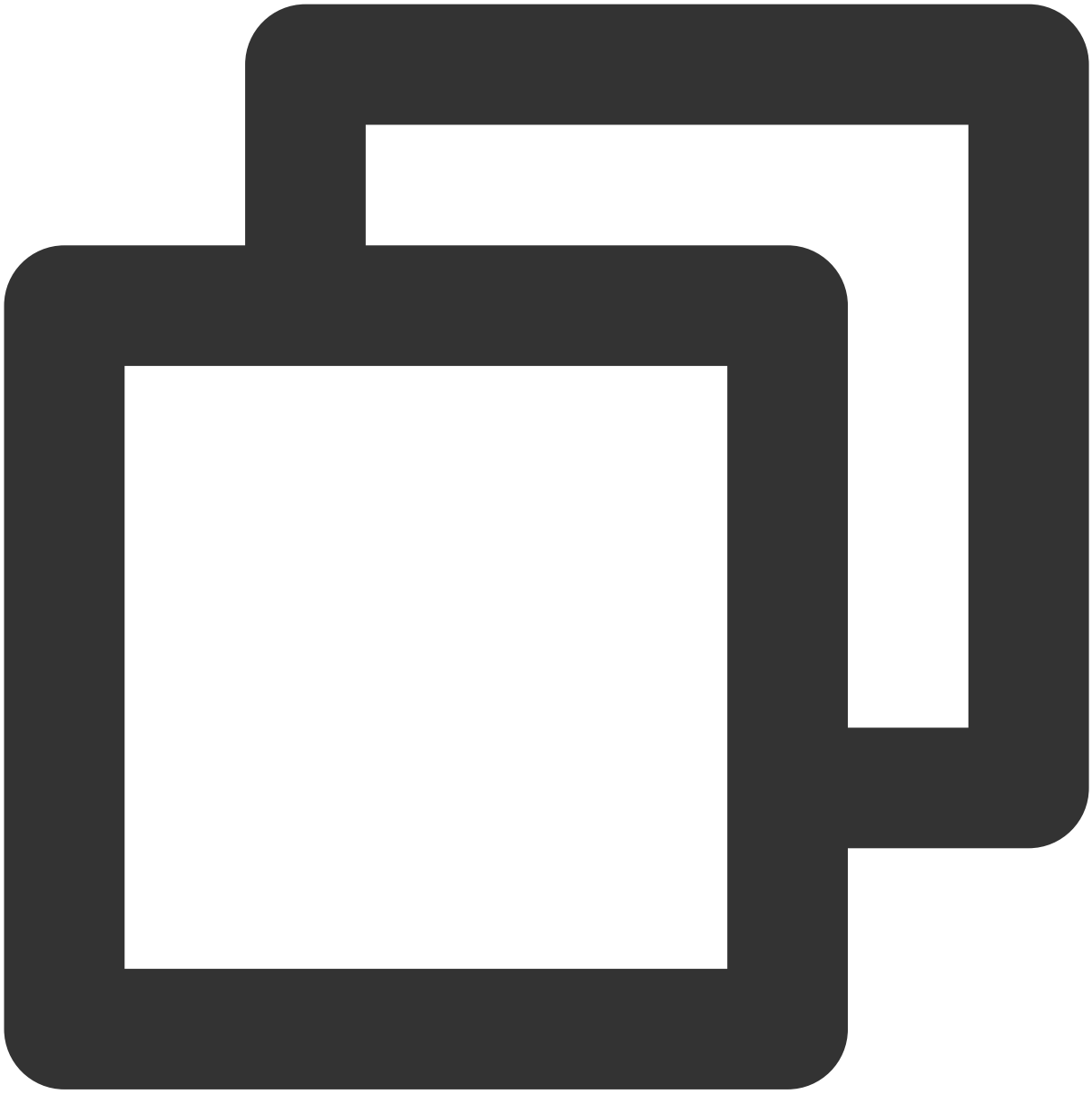


```
! Configuration File for keepalived
global_defs {
   notification_email {
     acassen@firewall.loc
     failover@firewall.loc
     sysadmin@firewall.loc
```

```
    }
    notification_email_from Alexandre.Cassen@firewall.loc
    smtp_server 192.168.200.1
    smtp_connect_timeout 30
    router_id LVS_DEVEL
    vrrp_skip_check_adv_addr
    vrrp_garp_interval 0
    vrrp_gna_interval 0
}
vrrp_script checkhaproxy
{
    script "/etc/keepalived/do_sth.sh"
     interval 5
}
vrrp_instance VI_1 {
# Select proper parameters for the primary and secondary CVMs.
state BACKUP            #Set the initial status to `Backup`
    interface eth0          # The ENI such as `eth0` used to bind a VIP
    virtual_router_id 51        # The`virtual_router_id` value for the cluster
    nopreempt                   #Non-preempt mode
    # preempt_delay 10      # Effective only when `state` is `MASTER`
    priority 100            # Configure the same weight for the two devices
    advert_int 5
    authentication {
        auth_type PASS
        auth_pass 1111
    }
    unicast_src_ip 172.16.16.6   #Private IP of the local device
    unicast_peer {
        172.16.16.5               #IP address of the peer device
    }
    virtual_ipaddress {
        172.16.16.12              # HAVIP
    }
    notify_master "/etc/keepalived/notify_action.sh MASTER"
    notify_backup "/etc/keepalived/notify_action.sh BACKUP"
    notify_fault "/etc/keepalived/notify_action.sh FAULT"
    notify_stop "/etc/keepalived/notify_action.sh STOP"
    garp_master_delay 1    # How long it will take before the ARP cache can be upda
    garp_master_refresh 5   #Time interval between which the primary node sends ARP
    track_interface {
                eth0               # ENI that bound with VIP, such as `eth0`
        }
    track_script {
       checkhaproxy
    }
}
```

4. Press **Esc** to exit the edit mode and enter **:wq!** to save and close the file.

5. Restart Keepalived for the configuration to take effect.



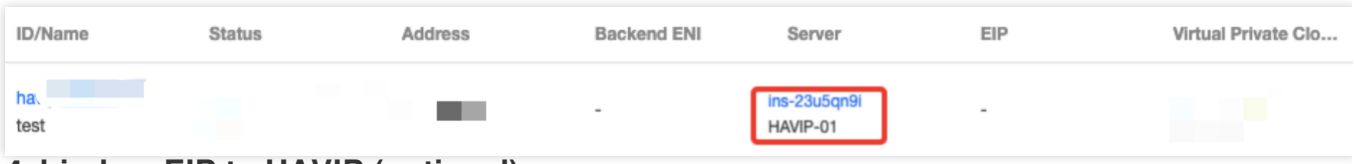
```
systemctl start keepalived
```

6. Check the primary/secondary status of the two CVMs, and confirm that both have HAVIP correctly bound.

Note:

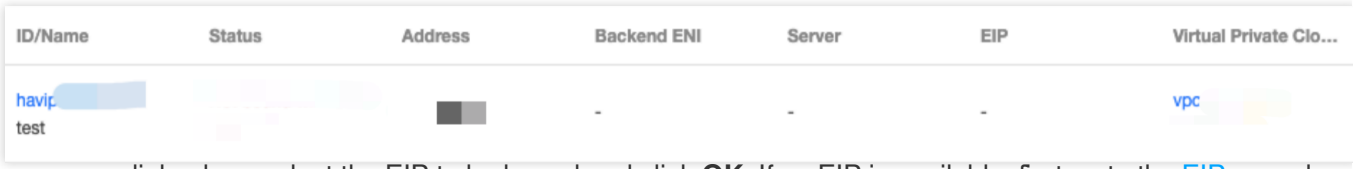
In this example, HAVIP-01 starts the Keepalived first and will normally serve as the primary node.

Log in to the HAVIP console. You will see that HAVIP is bound to the primary CVM HAVIP-01, as shown below.

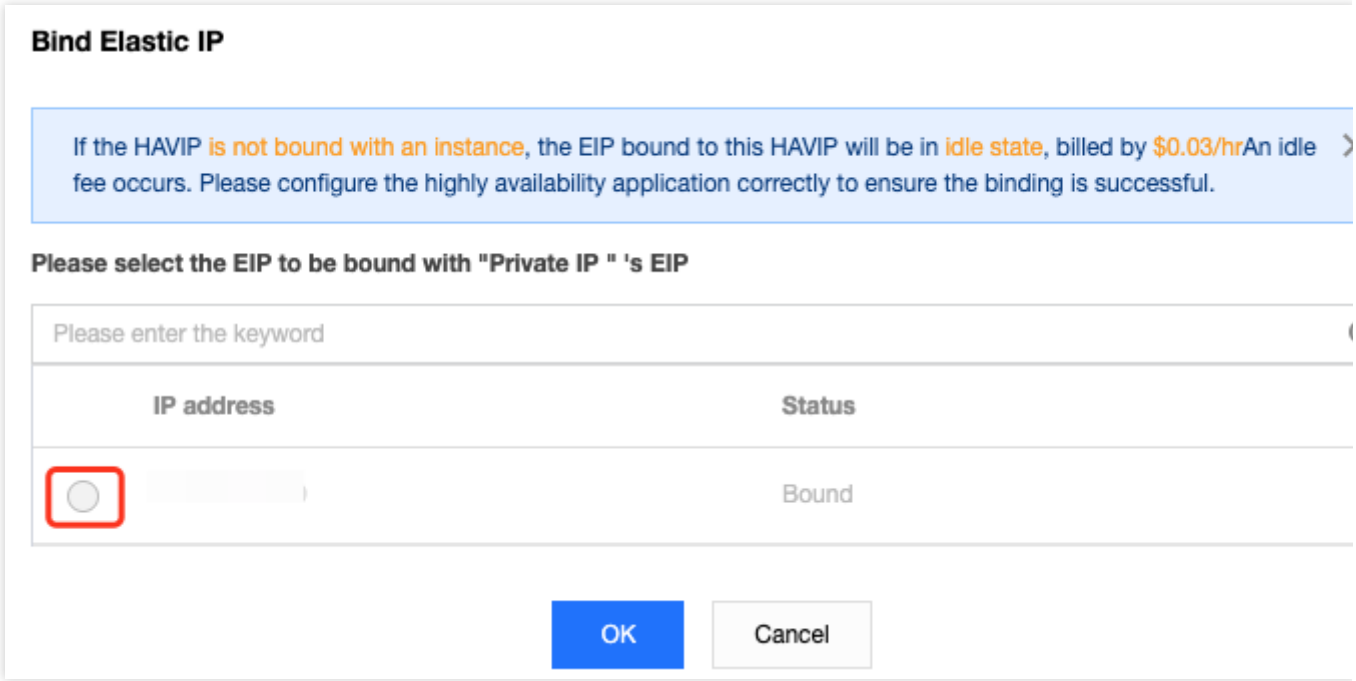


Step 4: bind an EIP to HAVIP (optional)

1. Log in to the HAVIP console, locate the HAVIP you have applied for in Step 1, and click **Bind**.



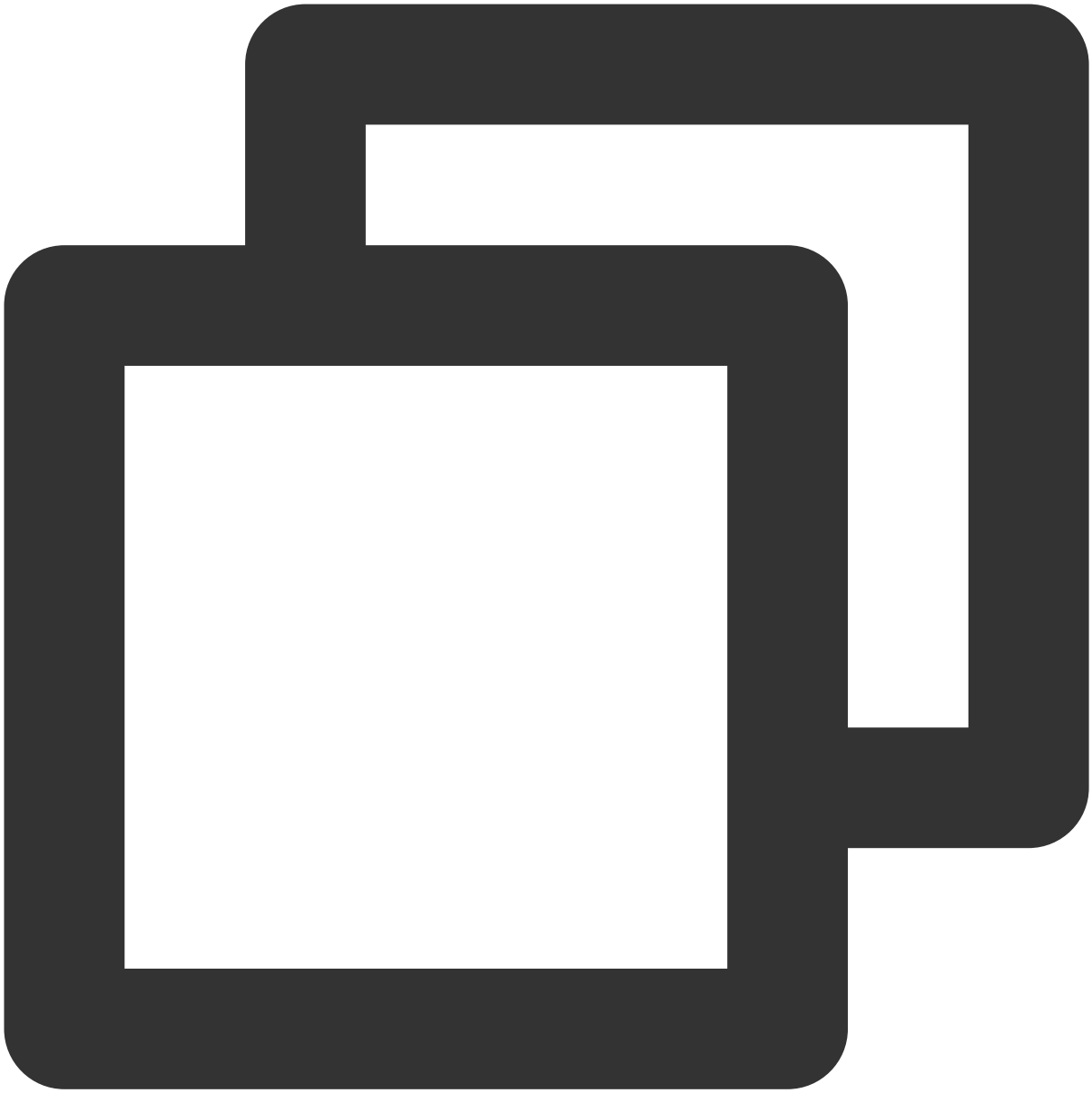
2. In the pop-up dialog box, select the EIP to be bound and click **OK**. If no EIP is available, first go to the EIP console to apply.



Step 5: use notify_action.sh for simple logging (optional)

The Keepalived's main logs are still recorded in "/var/log/message", and you can add the "notify" script for simple logging.

1. Log in to the CVM and run the `vim /etc/keepalived/notify_action.sh` command to add the following "notify_action.sh" script.



```
#!/bin/bash
#/etc/keepalived/notify_action.sh
log_file=/var/log/keepalived.log
log_write()
{
    echo "[`date '+%Y-%m-%d %T'`] $1" >> $log_file
}
[ ! -d /var/keepalived/ ] && mkdir -p /var/keepalived/

case "$1" in
    "MASTER" )
```

```
        echo -n "$1" > /var/keepalived/state
        log_write " notify_master"
        echo -n "0" /var/keepalived/vip_check_failed_count
        ;;
    "BACKUP" )
        echo -n "$1" > /var/keepalived/state
        log_write " notify_backup"
        ;;
    "FAULT" )
        echo -n "$1" > /var/keepalived/state
        log_write " notify_fault"
        ;;
    "STOP" )
        echo -n "$1" > /var/keepalived/state
        log_write " notify_stop"
        ;;
    *)
        log_write "notify_action.sh: STATE ERROR!!!"
        ;;
esac
```

2. Run the `chmod a+x /etc/keepalived/notify_action.sh` command to modify the script permission.

Step 6: verify whether VIP and public IP are switched normally during primary/secondary switch

Simulate the CVM failure by restarting the Keepalived process or restarting the CVM to check whether the VIP can be migrated.

If the primary/secondary switch succeeds, the secondary CVM will become the server bound with the HAVIP in the console.

You can also ping a VIP from within the VPC to check the time lapse from network interruption to recovery. Each switch may cause an interruption for about 4 seconds. If you ping the EIP bound to HAVIP over a public network, the result will be the same.

Run the `ip addr show` command to check whether the HAVIP is bound to the primary ENI.

Creating a High-availability Database by Using HAVIP + Windows Server Failover Cluster

Last updated：2024-01-24 17:22:28

1. **Creating HAVIPs**

Log in to the VPC console and create a HAVIP. For detailed directions, see Creating HAVIPs.

2. **Binding and configuration**

The configuration is the same as that in the traditional mode. The backend server declares and negotiates on the device that will be bound with the created HAVIP. You simply need to specify the virtual IP address in the configuration file as HAVIP.

In the cluster manager, add the HAVIP that was just created.

3. **Verification**

After the configuration is completed, directly switch nodes for testing.

In normal situations, you will see that the network recovers after a short interruption (no interruption will be noticed at all if the switching is fast enough), and online services will not be affected.

Hybrid Cloud Primary/Secondary Communication (DC and VPN)

Last updated：2024-01-24 17:22:28

If your business is deployed in both a local IDC and a Tencent Cloud VPC, you can connect them via Direct Connect or VPN. To improve the business availability, you set up both DC and VPN connections and configure them as the primary and secondary linkage for redundant communication. This document guides you through how to configure the DC and VPN connection as primary/secondary linkages to connect your IDC to the cloud.

Note:

The route priority feature is currently in beta test. To try it out, please submit a ticket.

The next hop type determines the route priority in the VPC route table. By default, the route priority from high to low is CCN, direct connect gateway, VPN gateway, and others.

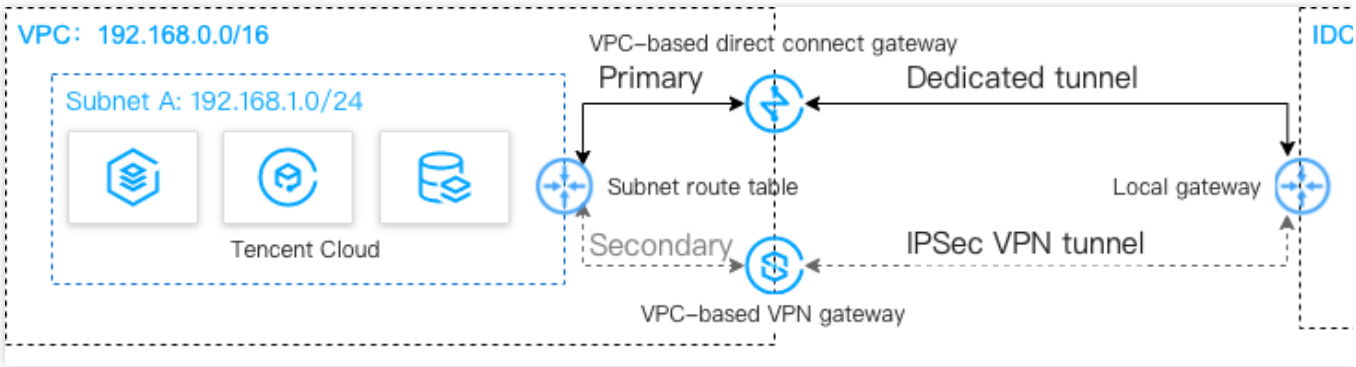
Currently, you cannot adjust the route priority in the console. If needed, please submit a ticket.

Scenarios

You have deployed businesses in a Tencent Cloud VPC and an IDC. To interconnect them, you need to configure network connection services for high-availability communications as follows:

Direct Connect (primary): connects the local IDC to a VPC-based direct connect gateway through a connection. When the connection linkage is normal, all data traffic between the IDC and the VPC is forwarded through the connection.

VPN connection (secondary): establishes an IPsec VPN tunnel to interconnect the local IDC and the Tencent Cloud VPC. When the connection linkage fails, traffic will be forwarded using this linkage to ensure the business availability.



Prerequisites

Your local IDC gateway device should support the IPsec VPN feature and can act as a customer gateway to create a VPN tunnel with the VPN gateway.

The IDC gateway device has configured with a static IP address.

Sample data and configuration:

Configuration item			Sample value
Network	VPC information	Subnet CIDR block	192.168.1.0/24
		Public IP of the VPN gateway	203.xx.xx.82
	IDC information	Subnet CIDR block	10.0.1.0/24
		Public IP of the gateway	202.xx.xx.5

Steps

1. Connect IDC to VPC through Direct Connect
2. Connect IDC to VPC through a VPN connection
3. Configure network probes
4. Configure an alarm policy
5. Switch between the primary and secondary routes

Directions

Step 1: connect IDC to VPC through Direct Connect

1. Log in to the Direct Connect console and open Dedicated Tunnels and click **Connections** on the left sidebar to create a connection.
2. Log in to the VPC console and click **Direct Connect Gateway** on the left sidebar. Click **+New** to create a standard direct connect gateway for which the **Associate Network** is **VPC**. If the IDC IP range conflicts with the VPC IP range, select the **NAT Type**.
3. Go to the **Dedicated Tunnels** page and click **+New** to create a dedicated tunnel. Enter the tunnel name, select the connection type and the direct connect gateway instance just created. Configure the IP addresses on both the Tencent Cloud and IDC sides, select the static route, and enter CPE IP range. After the configuration is complete, click **Download configuration guide** and complete the IDC device configurations as instructed in the guide.
4. In the route table associated with the VPC subnet for communication, configure a routing policy with the direct connect gateway as the next hop and IDC IP range as the destination.

Note:

For detailed configurations, see Getting Started.

Step 2: connect IDC to VPC through a VPN connection

1. Log in to the VPN Gateway console and click **+New** to create a VPN gateway for which the **Associate Network** is **Virtual Private Cloud**.
2. Click **Customer Gateway** on the left sidebar and click **+New** to configure a customer gateway (a logical object of the VPN gateway on the IDC side). Enter the public IP address of the VPN gateway on the IDC side, such as `202.xx.xx.5` .
3. Click **VPN Tunnel** on the left sidebar and click **+New** to complete configurations such as SPD policy, IKE, and IPsec.
4. Configure the same VPN tunnel as the step 3 on the local gateway device of the IDC to ensure a normal connection.
5. In the route table associated with the VPC subnet for communication, configure a routing policy with the VPN gateway as the next hop and IDC IP range as the destination.

Note:

For detailed directions, see Connecting VPC to IDC (Route Table).

Step 3: configure network probes

Note:

After the first two steps, there are two VPC routes to IDC. That is, both direct connect gateway and VPN gateway act as the next hop. By default, the direct connect gateway route has a higher priority, making it the primary path and the VPN gateway the secondary path.

To stay on top of the primary/secondary connection quality, configure two network probes separately to monitor the key metrics such as latency and packet loss rate and check the availability of primary/secondary routes.

1. Log in to the VPC console.
2. Click **+New** to create a network probe. Enter a name and destination IP, select a VPC and subnet, and set the **Source Next Hop** to direct connect gateway.
3. Repeat the step 2 and set the **Source Next Hop** to VPN gateway. After the configuration is complete, you can check the probed network latency and packet loss rate of the direct connect gateway and VPN connection.

Note:

For detailed configurations, see Network Probe.

Step 4: configure an alarm policy

You can configure an alarm policy for linkages. When a linkage has an exception, alarm notifications are sent to you automatically via emails and SMS message, alerting you of the risks in advance.

1. Log in to the CM console and go to the **Alarm Policy** page.
2. Click **Create**. Enter the policy name, select VPC/Network Probe for the policy type, specify the network probe instances as the alarm object, and configure trigger conditions, alarm notifications, and other information. Then click **Complete**.

Step 5: switch between primary and secondary routes

After receiving the exception alarms about the direct connect gateway, you need to manually disable the primary route, and forward traffic to the secondary route VPN gateway.

1. Log in to the VPC console and go to the **Route Tables** page.
2. Locate the route table associated with the VPC subnet for communication, click the **ID/Name** to enter its details page. Click

to disable the primary route with the CCN as the next hop. Then the VPC traffic destined to IDC will be forwarded to the VPN gateway, instead of the direct connect gateway.

Hybrid Cloud Primary/Secondary Communication (CCN and VPN)

Last updated：2024-01-24 17:22:28

If your business is deployed in both a local IDC and a Tencent Cloud VPC, you can connect them via Cloud Connect Network (CCN) or VPN. To improve the business availability, you set up both CCN and VPN connections and configure them as the primary and secondary linkage for redundant communication. This document guides you through how to configure the CCN and VPN connection as primary/secondary linkages to connect your IDC to the cloud.

Note:

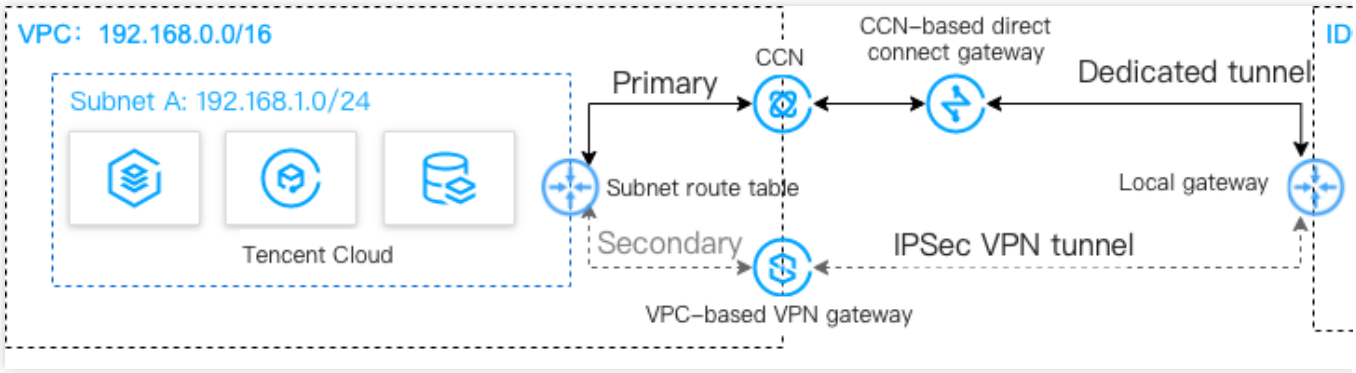
The route priority feature is currently in beta test. To try it out, please submit a ticket.

Scenarios

Suppose you have deployed your business in both Tencent Cloud VPC and an IDC. To interconnect them, you need to configure network connection services for high-availability communications as follows:

CCN (primary): connects the local IDC to a CCN-based direct connect gateway through a physical connection, and adds both the direct connect gateway and the VPC to a CCN to enable interconnection. When the connection linkage is normal, all data traffic between the IDC and the VPC are forwarded over CCN through the physical connection.

VPN connection (secondary): establishes an IPsec VPN tunnel to interconnect the local IDC and the Tencent Cloud VPC. When the connection linkage fails, traffic will be forwarded using this linkage to ensure the business availability.



Prerequisites

Your local IDC gateway device should support the IPsec VPN feature and can act as a customer gateway to create a VPN tunnel with the VPN gateway.

The IDC gateway device has configured with a static IP address.

Sample data and configuration:

Configuration item			Sample value
Network	VPC information	Subnet CIDR block	192.168.1.0/24
		Public IP of the VPN gateway	203.xx.xx.82
	IDC information	Subnet CIDR block	10.0.1.0/24
		Public IP of the gateway	202.xx.xx.5

Steps

1. Configure a Direct Connect instance
2. Configure a VPN connection
3. Configure network probes
4. Configure an alarm policy
5. Switch between the primary and secondary routes

Directions

Step 1: connect IDC to VPC through CCN

1. Log in to the Direct Connect console and click **Connections** on the left sidebar to create a connection.
2. Log in to the VPC console and click **Direct Connect Gateway** on the left sidebar. Click **+New** to create a direct connect gateway for which the **Associate Network** is **CCN**.
3. Click the **ID/Name** of the direct connect gateway just created to enter its details page. Select the **IDC IP Range** tab to enter the IDC IP range, such as `10.0.1.0/24` .
4. Go to the CCN page and click **+New** to create a CCN instance.
5. Go to the Dedicated Tunnels page and click **+New** to create a dedicated tunnel to connect the CCN-based direct connect gateway. Enter the tunnel name, select **CCN** for the **Access Network**, and then select the CCN-based direct connect gateway instance created earlier. Configure the IP addresses on both the Tencent Cloud and IDC sides, and select the BGP route. After the configuration is complete, click **Download configuration guide** and complete the IDC device configurations as instructed in the guide.
6. Associate the VPC and the CCN-based direct connect gateway with the CCN instance to interconnect the VPC and the IDC.

Note:

For detailed directions, see Migrating IDC to the Cloud Through CCN.

Step 2: connect IDC to VPC through a VPN connection

1. Log in to the VPN Gateway console and click **+New** to create a VPN gateway for which the **Associate Network** is **Virtual Private Cloud**.
2. Click **Customer Gateway** on the left sidebar and click **+New** to configure a customer gateway (a logical object of the VPN gateway on the IDC side). Enter the public IP address of the VPN gateway on the IDC side, such as `202.xx.xx.5`.
3. Click **VPN Tunnel** on the left sidebar and click **+New** to complete configurations such as SPD policy, IKE, and IPsec.
4. Configure the same VPN tunnel as the step 3 on the local gateway device of the IDC to ensure a normal connection.
5. In the route table associated with the VPC subnet for communication, configure a routing policy with the VPN gateway as the next hop and IDC IP range as the destination.

Note:

For detailed configurations of VPN gateways in different versions,
For a VPN gateway v1.0 and v2.0, see Connecting VPC to IDC (SPD Policy).
For a VPN gateway v3.0, see Connecting VPC to IDC (Route Table).

Step 3: configure network probes

Note:

After the first two steps, there are two VPC routes to IDC. That is, both CCN and VPN gateway act as the next hop. The CCN route has a higher priority, making it the primary path and the VPN gateway the secondary path.
To stay on top of the primary/secondary connection quality, configure two network probes separately to monitor the key metrics such as latency and packet loss rate and check the availability of primary/secondary routes.

1. Go to the Network Probe page on VPC console.
2. Click **+New** to create a network probe. Enter a name and destination IP, select a VPC and subnet, and set the **Source Next Hop** to CCN.
3. Repeat the step 2 and set the **Source Next Hop** to VPN gateway. After the configuration is complete, you can check the probed network latency and packet loss rate of the CCN and VPN connection.

Note:

For detailed configurations, see Network Probe.

Step 4: configure an alarm policy

You can configure an alarm policy for linkages. When a linkage has an exception, alarm notifications are sent to you automatically via emails and SMS message, alerting you of the risks in advance.

1. Log in to the CM console and go to the **Alarm Policy** page.
2. Click **Create**. Enter the policy name, select VPC/Network Probe for the policy type, specify the network probe instances as the alarm object, and configure trigger conditions, alarm notifications, and other information. Then click **Complete**.

Step 5: switch between primary and secondary routes

After receiving a CCN network exception alarm, you need to manually disable the primary route, and forward traffic to the secondary route VPN gateway.

1. Log in to the VPC console and go to the **Route Tables** page.
2. Locate the route table associated with the VPC subnet for communication, click the **ID/Name** to enter its details page. Click

to disable the primary route with the CCN as the next hop. Then the VPC traffic destined to IDC will be forwarded to the VPN gateway, instead of the CCN.

CVM Access to Internet Through EIP

Last updated：2024-01-24 17:22:29

An EIP is a region-level static public IP. It can connect a VPC-based CVM instance to the public network. This document describes how to bind an EIP to a CVM instance for public network access.

Overview

A VPC-base CVM instance, if you do not allocate a public IP when purchasing it, it cannot access to the public network.
However you can bind an EIP to the CVM instance to access the public network.

Directions

Step 1. Apply for an EIP

Note:
If you already have an idle EIP, you can skip this step and proceed to Step 2.
1. Log in to the VPC console.
2. Click **IP and ENI** > **Public IP/EIP** to enter the public IP page.
3. At the top of the **Public IP/EIP** page, select the same region as the CVM instance and click **Apply**.
4. In the **Apply for EIP** pop-up window, configure the parameters as needed and click **OK**.

Step 2. Bind an EIP to the CVM instance

1. On the **Public IP/EIP** page, select **More** > **Bind** on the right of the EIP.
2. In the **Bind resources** pop-up window, select **CVM instance**, select your CVM instance ID, and click **OK**.

Step 3. Verify the public network access through the EIP

1. Go to the CVM console, click **Login** on the right of the CVM instance, and enter the password to access the CVM UI.
2. Run `ping www.qq.com` to test the data connectivity. If data is returned, the CVM instance can access the public network.