

Global Accelerator

FAQ

Issue 01
Date 2024-01-22



Copyright © Huawei Technologies Co., Ltd. 2024. All rights reserved.

No part of this document may be reproduced or transmitted in any form or by any means without prior written consent of Huawei Technologies Co., Ltd.

Trademarks and Permissions



HUAWEI and other Huawei trademarks are trademarks of Huawei Technologies Co., Ltd.

All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

The purchased products, services and features are stipulated by the contract made between Huawei and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

Security Declaration

Vulnerability

Huawei's regulations on product vulnerability management are subject to the *Vul. Response Process*. For details about this process, visit the following web page:

<https://www.huawei.com/en/psirt/vul-response-process>

For vulnerability information, enterprise customers can visit the following web page:

<https://securitybulletin.huawei.com/enterprise/en/security-advisory>

Contents

1 Can I Use the Ping Command to Test Latency?.....	1
2 How Will I Be Billed for Global Accelerator?.....	2
3 How Can I Increase the Global Accelerator Quotas?.....	4
4 How Can I Use Traffic Dial to Distribute Traffic?.....	5
5 What Types of Endpoints Can Be Added to a Global Accelerator?.....	7
6 Can I Use Global Accelerator in an Area That Is Not Listed Among the Acceleration Areas?.....	8
7 What Are the Statuses and Health Check Results of Endpoints?.....	9
8 What Should I Do If an Endpoint Is Unhealthy?.....	10
9 Most Frequently Asked Questions.....	12
10 Configuring the TOA Module.....	17

1 Can I Use the Ping Command to Test Latency?

No.

You can use the **ping** command to test the Internet connectivity between clients and access points but not to test latency.

For example, run the following command:

```
C:\Users\*****>ping 10.108.172.1
Pinging 10.108.172.1 with 32 bytes of data:
Reply from 10.108.172.1: bytes=32 time<1ms TTL=254
Reply from 10.108.172.1: bytes=32 time<1ms TTL=254
Reply from 10.108.172.1: bytes=32 time<1ms TTL=254
Reply from 10.108.172.1: bytes=32 time<1ms TTL=254

Ping statistics for 10.108.172.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip time in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

2 How Will I Be Billed for Global Accelerator?

Billing Mode

Only pay-per-use is available. The bill for each month is generated at the beginning of the next calendar month.

Billing Items

Table 2-1 Global Accelerator pricing details

Billing Item	Description	Price
Global accelerators	<p>You are charged based on how long each global accelerator is retained in your account.</p> <p>The smallest billing unit is one hour. Partial hours are counted as full hours.</p> <p>Global accelerator price = Unit price x Required duration</p>	\$0.356 USD/hour
Data transfer	<p>You are charged for either the inbound or outbound traffic, whichever direction has more traffic.</p> <p>Data transfer price = Unit price x Traffic used</p>	<p>The actual price is subject to what is displayed on the Global Accelerator console.</p> <p>NOTE</p> <ul style="list-style-type: none"> • See Acceleration Area for available acceleration areas. • The regions where an endpoint group can be deployed are those you can select on the Global Accelerator console.

Billing Examples

Suppose you have an application deployed in Guangzhou, if you want end users in Türkiye to be able to access your application faster, you need a global accelerator.

If end users in Türkiye access your application, inbound traffic to your application is 1 GB and outbound traffic from your application is 20 GB, you are only charged for the 20 GB of outbound traffic but not the 1 GB of inbound traffic.

The total price for using this global accelerator for an hour can be calculated using the following formula: Global accelerator price + Data transfer price = $\$0.356 \text{ USD/hour} \times 1 \text{ hour} + 1.098 \times 20 \text{ GB} = \22.316 USD

Changes Between Billing Modes

The billing mode cannot be changed.

Renewal

For details, see [Renewal Management](#).

Expiration and Overdue Payment

For details, see [Service Suspension and Resource Release](#) and [Payment and Repayment](#).

3 How Can I Increase the Global Accelerator Quotas?

Global Accelerator resource quotas are displayed on the management console.
If you need to increase the quotas, [submit a service ticket](#).

4 How Can I Use Traffic Dial to Distribute Traffic?

You can set a traffic dial to control the percentage of traffic directed to an endpoint group. If a listener has multiple endpoint groups, traffic will be first distributed to the endpoint group with the lowest latency and then to other endpoint groups based on the traffic dial value you set.

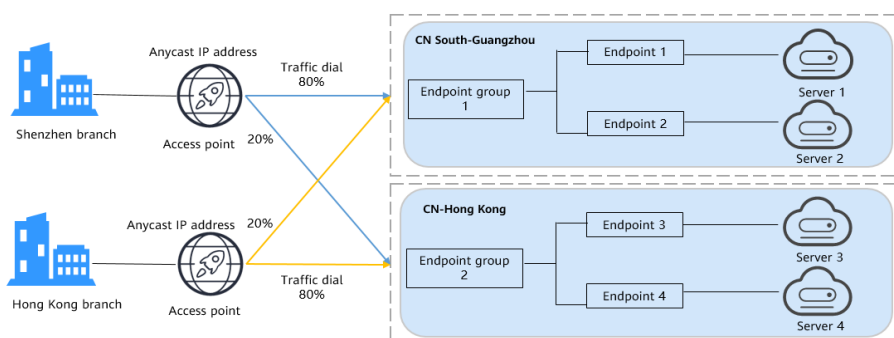
Example:

As shown in [Figure 4-1](#), a multinational enterprise has branches in Shenzhen and Hong Kong. The Shenzhen branch has deployed an application on two servers in the CN South-Guangzhou region, and the Hong Kong branch has deployed an application on two servers in the CN-Hong Kong region.

If the percentage of traffic directed to the endpoint group 1 and endpoint group 2 are set to 80%, requests from users in Shenzhen and Hong Kong are distributed as follows:

- 80% of the requests from users in Shenzhen will be sent to endpoint group 1, and the remaining 20% of the requests to the endpoint group 2.
- 80% of the requests from users in Hong Kong will be sent to the endpoint group 2, and the remaining 20% of the requests to the endpoint group 1.

Figure 4-1 Cross-border traffic dial



 **NOTE**

In this example, users in the Shenzhen branch have faster access to the application in the CN South-Guangzhou region than to that in the CN-Hong Kong region. Requests from users in the Shenzhen branch are preferentially sent to the CN South-Guangzhou region. For users in the Hong Kong branch, it is the other way around.

5 What Types of Endpoints Can Be Added to a Global Accelerator?

You can add an EIP as an endpoint.

6 Can I Use Global Accelerator in an Area That Is Not Listed Among the Acceleration Areas?

This depends on your scenario and requirements.

- You can use Global Accelerator if you require a unified IP address to ensure high reliability regardless of where your application is deployed.
- Do not use Global Accelerator if you require the optimal delay.

7 What Are the Statuses and Health Check Results of Endpoints?

Table 7-1 describes the statuses of endpoints.

Table 7-1 Endpoint statuses

Status	Description
Pending	The endpoint is being configured.
Running	The endpoint is working normally.
Abnormal	The endpoint is unavailable.
Deleting	The endpoint is being deleted.

Table 7-2 describes the health check results of endpoints.

Table 7-2 Endpoint health check results

Health Check Result	Description
Initial	The endpoint is being configured, and health check has not been performed.
Healthy	The endpoint is working normally.
Unhealthy	The endpoint is unhealthy and is unavailable to receive requests.
Not monitored	Health check is not enabled.

8 What Should I Do If an Endpoint Is Unhealthy?

Background

A global accelerator sends heartbeat requests to its endpoints to check their health. Before performing health checks, you need to ensure that TCP or UDP traffic is allowed from the global accelerator to endpoints over the listener ports.

If an endpoint is considered unhealthy, traffic will not be forwarded to it until the endpoint recovers. You can perform the following operations to verify health check settings:

Checking Health Check Settings

1. Go to the details page of the global accelerator and click the **Endpoint Groups** tab.
2. Click the name of the endpoint group. In the basic information area, click **Configure** next to **Health Check**. Check the following parameters:
 - Protocol
 - Port that is used by the endpoint and cannot be changed

Checking the Security Group Rules of the Endpoint

- **TCP listeners:** Verify that the security group of the endpoint has inbound rules that allow TCP traffic over the health check port.
 - If the port (port 80 as an example) for health check is same as that used by the endpoint, inbound security group rules must allow traffic over the port for health check.
 - If the port (port 80 as an example) for health check is different from that used by the endpoint (port 443 as an example), inbound security group rules must allow traffic over both ports.

NOTE

You can view the protocol and port in the **Basic Information** area of the endpoint group.

- **UDP listeners:** Verify that the security group of the endpoint has inbound rules that allow both the UDP traffic over the health check port and the ICMP traffic destined for the endpoint.

Checking Listener Settings for Endpoints

If the endpoint runs a Windows OS, use a browser to access *https://{Endpoint IP address}:{Health check port}*. If 2xx or 3xx is returned, the endpoint is running normally.

Run the following command on the endpoint to check whether the health check port (port 880 in this example) is listened on:

```
netstat -anlp | grep port
```

If figure [Figure 8-1](#) is displayed, TCP port 880 is listened on.

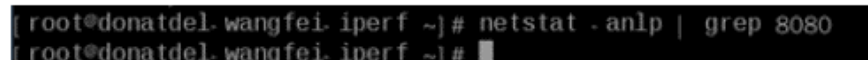
- If you do not specify a health check port, endpoint ports are used by default.

Figure 8-1 Endpoint port listened on



```
[root@ecs-elb-srv portable-nginx]# netstat -anlp | grep 880 | head
tcp        0      0 0.0.0.0:880          0.0.0.0:*        LISTEN
```

Figure 8-2 Endpoint port not listened on



```
[root@donatdel.wangfei.iperf ~]# netstat -anlp | grep 8080
[root@donatdel.wangfei.iperf ~]#
```

- If the health check port is not in the listening state, the endpoint is not listened on. You need to start the application on the endpoint and check whether the health check port is listened on.

9 Most Frequently Asked Questions

Why Do I Need a Cross-Border Permit?

In accordance with the laws and administrative regulations of the Ministry of Industry and Information Technology (MIIT) of the People's Republic of China, only three major operators in the Chinese mainland are allowed for cross-border network communications. To carry out business outside the Chinese mainland, you need to apply for a cross-border permit from China Unicom because Huawei Cloud cooperates with China Unicom to centrally manage user profiles for cross-border businesses. Huawei Cloud provides Global Accelerator, and China Unicom provides the cross-border circuit services.

When Do I Need to Apply for a Cross-Border Permit?

In accordance with the laws and administrative regulations of the Ministry of Industry and Information Technology (MIIT) of the People's Republic of China, only China Mobile, China Telecom, and China Unicom are allowed for cross-border network communications, and a cross-border permit is required if you carry out business activities outside the Chinese mainland.

To comply with laws and regulations on cross-border network communications, you need to apply for a cross-border permit in the following scenarios:

Cross-border communications are required in the following two scenarios:

The acceleration areas are in Europe, but the endpoints are running inside the Chinese mainland.

To apply for a cross-border permit, you need to prepare the required materials stamped with your company's official seal and submit an application on the console. China Unicom will review and approve the application within one working day.

How Can I Apply for a Cross-Border Permit?

Preparing for Materials

- A sealed copy of the company's business license
- A sealed copy of the *Huawei Cloud Cross-Border Circuit Service Agreement*

- A sealed copy of *China Unicom Letter of Commitment to Information Security of the Cross-Border Circuit Service*

Application Process


1. Log in to the management console.
2. Click  in the upper left corner and choose **Networking > Global Accelerator**.
The **Global Accelerator** page is displayed.
3. In the navigation pane on the left, click **Cross-Border Permits**.
4. Click **Request a Cross-Border Permit**.
5. On the application page, set related parameters and upload related materials.

Table 9-1 Online cross-border permit application

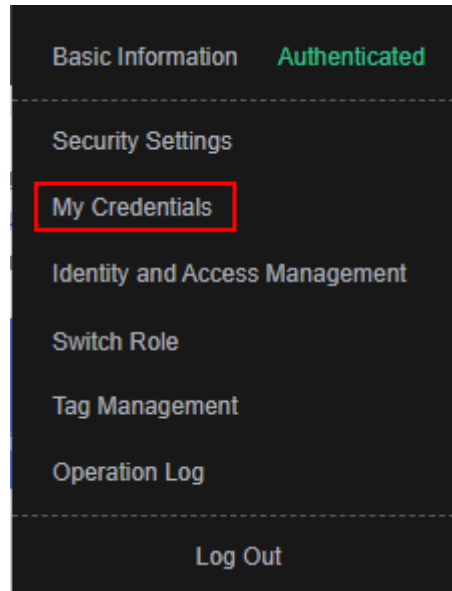
Parameter
Applicant Name
Huawei Cloud UID
Type of Product
Bandwidth (M)
Start Date
Termination Date
Customer Type
Country of the Customer
Contact Name
Contact Number
Type of ID
ID Number
Scope of Business
Number of Employees
Per Capita Bandwidth
Branch Location Country

 **NOTE**

HUAWEI ID is your account ID. You can take the following steps to obtain your account ID.

1. Log in to the management console.
2. Click the username and select **My Credentials** from the drop-down list.

Figure 9-1 My credentials



3. On the **API Credentials** page, view the **Account ID**.

Figure 9-2 Obtaining an account ID

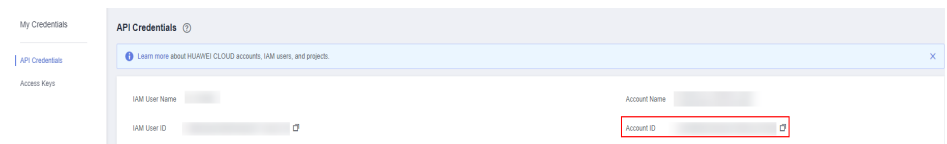


Table 9-2 Required materials

Material	Signature	Seal	Description
A scanned copy of your company's business license	-	√	See the template Huawei Cloud provides for the position of the seal.
A scanned copy of <i>Huawei Cloud Cross-Border Circuit Service Agreement</i>	√	√	<ul style="list-style-type: none"> • Sign the material on the signature block. • Stamp the seal over the signature.

Material	Signature	Seal	Description
A scanned copy of <i>China Unicom Letter of Commitment to Information Security of the Cross-Border Circuit Service</i>	√	√	<ul style="list-style-type: none">• Sign the material on the signature block.• Stamp the seal over the signature.• Specify the bandwidth you estimated and your company name.

6. Click **Submit**.
7. [Submit a service ticket](#) to request a cross-border permit.

How Long Will a Cross-Border Permit Be Approved?

Cross-border permits are approved within one working day.

Who Will Approve My request for a Cross-Border Permit?

China Unicom approves cross-border permits and provides you with the cross-border circuit services.

I Have Completed the Real-Name Authentication on Huawei Cloud. Why Do I Also Need to Apply for a Cross-Border Permit Before Using Global Accelerator?

- Huawei Cloud's real-name authentication is used for billing and issuing invoices.
- Global Accelerator's real-name authentication is required for China Unicom to approve your cross-border permit, which is required if you want to access an endpoint outside the Chinese mainland. China Unicom will archive your information for auditing if necessary.

Can I Modify the Content of the *Global Accelerator Cross-Border Circuit Service Agreement*?

No.

The *Huawei Cloud Cross-Border Circuit Service Agreement* is a standard contract confirmed with China Unicom Shenzhen Branch and cannot be modified.

Can I Download the Materials for My Cross-Border Permit Application on the Console After I Delete Them from My PC?

No, you cannot.

Keep your application materials safe and secure.

Does Huawei Cloud Need to Sign and Stamp the Seal on the Materials for Cross-Border Permit Application?

No.

Huawei Cloud works with China Unicom to enable network communications across borders. China Unicom provides the network circuit services and reviews and archives the application materials for cross-border permits under the requirements of the Ministry of Industry and Information Technology (MIIT).

10 Configuring the TOA Module

Scenario

Global Accelerator provides customized strategies for managing service access. Before these strategies can be customized, the clients' IP addresses contained in the requests are required. The TCP Option Address (TOA) kernel module is used to obtain the IP addresses of clients. It is installed on the server of the endpoint.

This section describes how you can compile the module in the OS if you use TCP to distribute IPv4 traffic.

The operations for Linux OSs with kernel version of 2.6.32 are different from those for Linux OSs with kernel version of 3.0 or later.

NOTE

- The TOA module cannot be used for UDP listeners.
- The TOA module can work properly in the following OSs, and the methods for installing other kernel versions are similar:
 - CentOS 6.8 (kernel version 2.6.32)
 - SUSE 11 SP3 (kernel version 3.0.76)
 - CentOS 7 or CentOS 7.2 (kernel version 3.10.0)
 - Ubuntu 16.04.3 (kernel version 4.4.0)
 - Ubuntu 18.04 (kernel version 4.15.0)
 - OpenSUSE 42.2 (kernel version 4.4.36)
 - Debian 8.2.0 (kernel version 3.16.0)

Constraints

- The development environment for compiling the module must be the same as that of the current kernel. For example, if the kernel version is kernel-3.10.0-693.11.1.el7, the kernel development package version must be kernel-devel-3.10.0-693.11.1.el7.
- The OS repositories are accessible to servers.
- Users other than **root** must have sudo permissions.

Procedure

- Linux kernel version 3.0 or later

1. Prepare the compilation environment.

 **NOTE**

- During the installation, download the required module development package from the Internet if it cannot be found in the source.
- If the kernel development package (kernel-devel) cannot be obtained, contact the image provider.

The following are operations for compiling the module in different Linux OSs.

– CentOS

- i. Install the gcc compiler.

sudo yum install gcc

- ii. Install the make tool.

sudo yum install make

- iii. Install the module development package (the package header and module library must have the same version as the kernel).

sudo yum install kernel-devel-`uname -r`

 **NOTE**

- Download the required module development package from the following address if it cannot be found in the source:
https://mirror.netcologne.de/oracle-linux-repos/ol7_latest/getPackage/
For example, run the following command to install 3.10.0-693.11.1.el7.x86_64:
rpm -ivh kernel-devel-3.10.0-693.11.1.el7.x86_64.rpm
- If the kernel development package (kernel-devel) cannot be obtained, contact the image provider.

– Ubuntu and Debian

- i. Install the gcc compiler.

sudo apt-get install gcc

- ii. Install the make tool.

sudo apt-get install make

- iii. Install the module development package (the package header and module library must have the same version as the kernel).

sudo apt-get install linux-headers-`uname -r`

– SUSE

- i. Install the gcc compiler.

sudo zypper install gcc

- ii. Install the make tool.

sudo zypper install make

- iii. Install the module development package (the package header and module library must have the same version as the kernel).

sudo zypper install kernel-default-devel

2. Compile the module.

- a. Enter the source code directory and compile the module.

```
cd src
```

```
make
```

If no warning or error information is prompted, the compilation is successful. Verify that the **toa.ko** file has generated in the current directory.

 **NOTE**

If error message "config_retpoline=y but not supported by the compiler, Compiler update recommended" is displayed, the GCC version is too old. Upgrade the GCC to a later version.

3. Load the module.

- a. Load the module.

```
sudo insmod toa.ko
```

- b. Check the module loading and view the kernel output information.

```
dmesg | grep TOA
```

If "TOA: toa loaded" is displayed in the command output, the module has been loaded.

 **NOTE**

After the CoreOS module is compiled in the container, copy it to the host system and then load it. The container for compiling the module shares the **/lib/modules** directory with the host system, so you can copy the module to this directory, allowing the host system to use it.

4. Set the script to enable the system to automatically load the module.

To make the module take effect when the system starts, add the command for loading the module to your startup script.

You can use either of the following methods to enable the module to automatically load:

- Add the command for the module to automatically load to the startup script as required.
- Perform the following operations to configure the startup script:

- i. Create the **toa.modules** file in the **/etc/sysconfig/modules/** directory. This file contains the module loading script.

The following is an example of the content in the **toa.modules** file.

```
#!/bin/sh
```

```
/sbin/modinfo -F filename /root/toa/toa.ko > /dev/null 2>&1
```

```
if [ $? -eq 0 ]; then
```

```
/sbin/insmod /root/toa/toa.ko
```

```
fi
```

/root/toa/toa.ko is the path of the module file. You need to replace it with their actual path.

- ii. Add execution permissions for the **toa.modules** startup script.

```
sudo chmod +x /etc/sysconfig/modules/toa.modules
```

 NOTE

If the kernel is upgraded, the current module will no longer match. Compile the module again.

5. Install the module on servers.

To load the module in the same OSs, copy the **toa.ko** file to VMs where the module is to be loaded and then perform the operations in [3](#).

After the module is loaded, the IP address of a client can be obtained.

 NOTE

The OS version of each server must be the same as that of the kernel.

6. Verify the module.

After the module is installed, the source IP address can be directly obtained. You can perform the following operations to verify:

Start SimpleHTTPServer on the server of the endpoint where Python is installed:

```
python -m SimpleHTTPServer port
```

The value of **port** must be a port used by the server, and the default value is **80**.

Access the anycast IP address provided by Global Accelerator. Access logs on the server are as follows:

```
192.168.0.90 -- [06/Aug/2020 14:24:21] "GET / HTTP/1.1" 200 -
```

 NOTE

192.168.0.90 is the source IP address and also the real IP address of the client that can be obtained by the backend server.

- **In the following operations, the Linux kernel version is 2.6.32.**

 NOTE

The TOA module supports OSs (CentOS 6.8 image) with a kernel of 2.6.32-xx.

1. Obtain the kernel source code package **Linux-2.6.32-220.23.1.el6.x86_64.rs.src.tar.gz** containing the module from the following link:

http://kb.linuxvirtualserver.org/images/3/34/Linux-2.6.32-220.23.1.el6.x86_64.rs.src.tar.gz

2. Decompress the kernel source code package.
3. Modify compilation parameters.
 - a. Open the **linux-2.6.32-220.23.1.el6.x86_64.rs** directory.
 - b. Edit the **net/toa/toa.h** file.
Change the value of **#define TCPOPT_TOA200** to **#define TCPOPT_TOA254**.
 - c. On the Shell page, run the following commands:

```
sed -i 's/CONFIG_IPV6=m/CONFIG_IPV6=y/g' .config  
echo -e '\n# toa\nCONFIG_TOA=m' >> .config
```

The IPv6 module has been compiled into the kernel. TOA is compiled into a separate module and can be independently started and stopped.

d. Edit **Makefile**.

You can add a description after **EXTRAVERSION =**. This description will be displayed in **uname -r**, for example, **-toa**.

4. Compile the software package.

```
make -j n
```

 **NOTE**

n indicates the number of vCPUs. For example, if there are four vCPUs, *n* must be set to 4.

5. Install the module.

```
make modules_install
```

Figure 10-1 shows the command output.

Figure 10-1 Installing the module

```
INSTALL /lib/firmware/kaweth/trigger_code_fix.bin
INSTALL /lib/firmware/ti_3410.fw
INSTALL /lib/firmware/ti_5052.fw
INSTALL /lib/firmware/mts_cdma.fw
INSTALL /lib/firmware/mts_gsm.fw
INSTALL /lib/firmware/mts_edge.fw
INSTALL /lib/firmware/edgeport/boot.fw
INSTALL /lib/firmware/edgeport/boot2.fw
INSTALL /lib/firmware/edgeport/down.fw
INSTALL /lib/firmware/edgeport/down2.fw
INSTALL /lib/firmware/edgeport/down3.bin
INSTALL /lib/firmware/whiteheat_loader.fw
INSTALL /lib/firmware/whiteheat.fw
INSTALL /lib/firmware/keyspan_pda/keyspan_pda.fw
INSTALL /lib/firmware/keyspan_pda/xircom_pgs.fw
DEPMOD 2.6.32-toa
```

6. Install the kernel.

```
make install
```

Figure 10-2 shows the command output.

Figure 10-2 Installing the kernel

```
INSTALL /lib/firmware/keyspan_pda/xircom_pgs.fw
DEPMOD 2.6.32-toa
[root@SZX1000167219 linux-2.6.32-220.23.1.el6.x86_64.rs]# make install
sh /root/humin/linux-2.6.32-220.23.1.el6.x86_64.rs/arch/x86/boot/install.sh 2.6.32-toa arch/x86/boot/bzImage \
    System.map "/boot"
ERROR: modinfo: could not find module xen_procfs
ERROR: modinfo: could not find module ipv6
ERROR: modinfo: could not find module xen_scxifront
ERROR: modinfo: could not find module xen_hcall
ERROR: modinfo: could not find module xen_balloon
[root@SZX1000167219 linux-2.6.32-220.23.1.el6.x86_64.rs]#
```

7. Open the **/boot/grub/grub.conf** file and configure the kernel to start up when the system starts.

- Change the default startup kernel from the first kernel to the zeroth kernel. To do so, change the value of **default** to **0**.
- Add the **nohz** parameter (set it to **off**) to the end of the line containing the **vmlinuz-2.6.32-toa** kernel. If **nohz** is not disabled, the CPU0 utilization may be high and overload the kernel.

Figure 10-3 Configuration file

```
default=1
timeout=5
splashimage=(hd0,1)/boot/grub/splash.xpm.gz
hiddenmenu
title Red Hat Enterprise Linux Server (2.6.32-toa)
    root (hd0,1)
    kernel /boot/vmlinuz-2.6.32-toa ro root=UUID=
et nohz=ofi
initrd /boot/initramfs-2.6.32-toa.img
```

- c. Save the modification and exit. Restart the OS.

During the restart, the system will load the **vmlinuz-2.6.32-toa** kernel.

8. After the restart, load the module.

modprobe toa

Add the **modprobe toa** command to both the startup script and the system scheduled monitoring script.

Figure 10-4 Adding the **modprobe toa** command

```
[root@SZX1000167219 ~]# modprobe toa
[root@SZX1000167219 ~]# lsmod |grep toa
toa                4203  0
[root@SZX1000167219 ~]#
```

After the module is loaded, query the kernel information.

Figure 10-5 Querying the kernel

```
[root@SZX1000167219 ~]# uname -a
Linux SZX1000167219 2.6.32-toa #1 SMP Sat Oct 15 11:50:05 CST 2016 x86_64 x86_64 x86_64 GNU/Linux
```

9. Verify the module.

After the module is installed, the source IP address can be directly obtained. You can perform the following operations to verify:

Start SimpleHTTPServer on the server of the endpoint where Python is installed:

```
python -m SimpleHTTPServer port
```

The value of **port** must be a port used by the server, and the default value is **80**.

Access the anycast IP address provided by Global Accelerator. Access logs on the server are as follows:

```
192.168.0.90 - - [06/Aug/2020 14:24:21] "GET / HTTP/1.1" 200 -
```

NOTE

192.168.0.90 is the source IP address and also the real IP address of the client that can be obtained by the backend server.