











































































































































































































































































## 【举例】

# 创建 IPv6 VRRP 备份组 1, 配置 IPv6 VRRP 备份组 1 的虚拟 IPv6 地址为 fe80::10。为 IPv6 VRRP 备份组 1 添加一个虚拟 IPv6 地址 1::10。

```
<Sysname> system-view  
[Sysname] interface gigabitethernet 2/0/1  
[Sysname-GigabitEthernet2/0/1] vrrp ipv6 vrid 1 virtual-ip fe80::10 link-local  
[Sysname-GigabitEthernet2/0/1] vrrp ipv6 vrid 1 virtual-ip 1::10
```

## 【相关命令】

- **display vrrp ipv6**

# 目 录

1 BFD .....	1-1
1.1 BFD配置命令.....	1-1.....
1.1.1 bfd authentication-mode .....	1-1.....
1.1.2 bfd demand enable.....	1-1.....
1.1.3 bfd detect-interface.....	1-2.....
1.1.4 bfd detect-multiplier .....	1-3.....
1.1.5 bfd echo enable .....	1-4.....
1.1.6 bfd echo-source-ip.....	1-4.....
1.1.7 bfd echo-source-ipv6 .....	1-5.....
1.1.8 bfd min-echo-receive-interval .....	1-5.....
1.1.9 bfd min-receive-interval .....	1-6.....
1.1.10 bfd min-transmit-interval .....	1-6.....
1.1.11 bfd multi-hop authentication-mode .....	1-7.....
1.1.12 bfd multi-hop destination-port .....	1-8.....
1.1.13 bfd multi-hop detect-multiplier .....	1-8.....
1.1.14 bfd multi-hop min-receive-interval .....	1-9.....
1.1.15 bfd multi-hop min-transmit-interval .....	1-9.....
1.1.16 bfd init-fail timer .....	1-10.....
1.1.17 bfd session init-mode .....	1-11.....
1.1.18 bfd template .....	1-11.....
1.1.19 display bfd session .....	1-12.....
1.1.20 reset bfd session statistics .....	1-14.....
1.1.21 snmp-agent trap enable bfd .....	1-15.....

# 1 BFD

## 1.1 BFD配置命令

### 1.1.1 bfd authentication-mode

**bfd authentication-mode** 命令用来配置单跳 BFD 控制报文进行认证的方式。

**undo bfd authentication-mode** 命令用来恢复缺省情况。

#### 【命令】

```
bfd authentication-mode { m-md5 | m-sha1 | md5 | sha1 | simple } key-id { cipher cipher-string | plain plain-string }
```

```
undo bfd authentication-mode
```

#### 【缺省情况】

单跳 BFD 控制报文不进行认证。

#### 【视图】

接口视图/BFD 模板视图

#### 【缺省用户角色】

network-admin

#### 【参数】

**m-md5**: 采用 Meticulous MD5 算法进行认证。

**m-sha1**: 采用 Meticulous SHA1 算法进行认证。

**md5**: 采用 MD5 算法进行认证。

**sha1**: 采用 SHA1 算法进行认证。

**simple**: 采用简单认证。

**key-id**: 认证字标识符，取值范围为 1~255。

**cipher**: 表示输入的密码为密文。

**cipher-string**: 表示设置的密文密码，为 33~53 个字符的字符串，区分大小写。

**plain**: 表示输入的密码为明文。

**plain-string**: 表示设置的明文密码，为 1~16 个字符的字符串，区分大小写。

#### 【使用指导】

本命令主要为了提高 BFD 会话的安全性。

以明文或密文方式设置的密码，均以密文的方式保存在配置文件中。

#### 【举例】

```
# 配置接口 GigabitEthernet2/0/1 对单跳 BFD 控制报文进行简单明文认证，认证字标识符为 1，  
密码为 123456。  
<Sysname> system-view  
[Sysname] interface gigabitethernet 2/0/1  
[Sysname-GigabitEthernet2/0/1] bfd authentication-mode simple 1 plain 123456
```

### 1.1.2 bfd demand enable

**bfd demand enable** 命令用来配置 BFD 会话为查询模式。

**undo bfd demand enable** 命令用来恢复缺省情况。

### 【命令】

**bfd demand enable**  
**undo bfd demand enable**

### 【缺省情况】

BFD 会话为异步模式。

### 【视图】

接口视图

### 【缺省用户角色】

network-admin

### 【使用指导】

在查询模式下，设备周期性发送 BFD 控制报文，但是对端（缺省为异步模式）会停止周期性发送 BFD 控制报文。如果通信双方都是查询模式，则双方都停止周期性发送 BFD 控制报文。当需要验证连接性的时候，设备会以协商的周期连续发送几个 P 比特位置 1 的 BFD 控制报文。如果在检测时间内没有收到返回的报文，就认为会话 down；如果收到对方的回应 F 比特位置 1 的报文，就认为连通，停止发送报文，等待下一次触发查询。

在异步模式下，设备周期性地发送 BFD 控制报文，如果在检测时间内对端没有收到 BFD 控制报文，则认为会话 down。

### 【举例】

# 在接口 GigabitEthernet2/0/1 上配置 BFD 会话为查询模式。

```
<Sysname> system-view  
[Sysname] interface gigabitethernet 2/0/1  
[Sysname-GigabitEthernet2/0/1] bfd demand enable
```

## 1.1.3 bfd detect-interface

**bfd detect-interface source-ip** 命令用来创建一个检测本接口状态的 BFD 会话。

**undo bfd detect-interface** 命令用来删除创建的检测本接口状态的 BFD 会话。

### 【命令】

**bfd detect-interface source-ip ip-address**  
**undo bfd detect-interface**

### 【缺省情况】

没有创建检测本接口状态的 BFD 会话。

### 【视图】

接口视图

### 【缺省用户角色】

network-admin

### 【参数】

*ip-address*: BFD 控制报文的源 IP 地址。

## 【使用指导】



注意

请不要在三层以太网接口和该三层以太接口创建的子接口上同时使用该功能，否则，将导致主接口或子接口的 BFD 检测功能失效。

三层聚合接口的成员端口上没有 IP 地址，没有可以支持的快速检测机制。通过本功能可以快速检测成员链路的故障，帮助快速找出故障成员接口。

本功能同时支持普通三层以太网接口、以太网子接口故障快速检测，实现接口状态与 BFD 会话状态的快速联动，帮助上层路由协议实现快速收敛。

BFD 会话采用控制报文方式，两端都必须配置，配置规则如下：

- 报文的目的地址固定为 224.0.0.184，不支持配置。
- 报文的源 IP 地址建议配置为接口 IP 地址，如果接口没有 IP 地址，建议配置一个单播地址（0.0.0.0 除外）。

## 【举例】

```
# 配置检测 GigabitEthernet2/0/1 接口状态的 BFD 会话，其源地址为接口地址 20.1.1.1。  
<Sysname> system-view  
[Sysname] interface gigabitethernet 2/0/1  
[Sysname-GigabitEthernet2/0/1] bfd detect-interface source-ip 20.1.1.1
```

### 1.1.4 bfd detect-multiplier

**bfd detect-multiplier** 命令用来配置单跳 BFD 检测时间倍数。

**undo bfd detect-multiplier** 命令用来恢复缺省情况。

## 【命令】

```
bfd detect-multiplier value  
undo bfd detect-multiplier
```

## 【缺省情况】

单跳 BFD 检测时间倍数为 5。

## 【视图】

接口视图/BFD 模板视图

## 【缺省用户角色】

network-admin

## 【参数】

**value:** 单跳 BFD 检测时间倍数，取值范围为 3~50。

## 【使用指导】

检测时间倍数，即允许发送方发送 BFD 报文（包括 echo 报文和控制报文）的最大连续丢包数。对于 echo 报文方式，实际检测时间为发送方的检测时间倍数和发送方的实际发送时间的乘积；对于控制报文方式的异步模式，实际检测时间为接收方的检测时间倍数和接收方的实际发送时间的乘积；对于控制报文方式的查询模式，实际检测时间为发送方的检测时间倍数和发送方的实际发送时间的乘积。

## 【举例】

```
# 配置接口 GigabitEthernet2/0/1 的单跳 BFD 检测时间倍数为 6。
```

```
<Sysname> system-view  
[Sysname] interface gigabitethernet 2/0/1  
[Sysname-GigabitEthernet2/0/1] bfd detect-multiplier 6
```

### 1.1.5 bfd echo enable

**bfd echo enable** 命令用来使能 echo 功能。

**undo bfd echo enable** 命令用来恢复缺省情况。

#### 【命令】

**bfd echo enable**

**undo bfd echo enable**

#### 【缺省情况】

echo 功能处于关闭状态。

#### 【视图】

接口视图

#### 【缺省用户角色】

network-admin

#### 【使用指导】

本功能在发送控制报文的 BFD 会话时使用。使能 echo 功能并且会话 up 后，设备周期性发送 echo 报文检测链路连通性，同时降低控制报文的接收速率。

#### 【举例】

```
# 配置接口 GigabitEthernet2/0/1 使能 echo 功能。  
<Sysname> system-view  
[Sysname] interface gigabitethernet 2/0/1  
[Sysname-GigabitEthernet2/0/1] bfd echo enable
```

### 1.1.6 bfd echo-source-ip

**bfd echo-source-ip** 命令用来配置 echo 报文的源 IP 地址。

**undo bfd echo-source-ip** 命令用来删除 echo 报文的源 IP 地址。

#### 【命令】

**bfd echo-source-ip ip-address**

**undo bfd echo-source-ip**

#### 【缺省情况】

没有配置 echo 报文的源 IP 地址。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【参数】

*ip-address*: echo 报文的源 IP 地址。

## 【使用指导】

**echo** 报文的源 IP 地址用户可以任意指定。为了避免对端发送大量的 ICMP 重定向报文造成网络拥塞，建议配置 **echo** 报文的源 IP 地址不属于该设备任何一个接口所在网段。

## 【举例】

```
# 配置 echo 报文的源 IP 地址为 8.8.8.8。  
<Sysname> system-view  
[Sysname] bfd echo-source-ip 8.8.8.8
```

### 1.1.7 bfd echo-source-ipv6

**bfd echo-source-ipv6** 命令用来配置 echo 报文的源 IPv6 地址。

**undo bfd echo-source-ipv6** 命令用来删除 echo 报文的源 IPv6 地址。

## 【命令】

```
bfd echo-source-ipv6 ipv6-address  
undo bfd echo-source-ipv6
```

## 【缺省情况】

没有配置 echo 报文的源 IPv6 地址。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

*ipv6-address*: echo 报文的源 IPv6 地址。

## 【使用指导】

**echo** 报文源 IPv6 地址仅支持全球单播地址。

为了避免对端发送大量的 ICMPv6 重定向报文造成网络拥塞，建议不要将 **echo** 报文的源 IPv6 地址配置为属于该设备任何一个接口所在网段。

## 【举例】

```
# 配置 echo 报文的源 IPv6 地址为 80::2。  
<Sysname> system-view  
[Sysname] bfd echo-source-ipv6 80::2
```

### 1.1.8 bfd min-echo-receive-interval

**bfd min-echo-receive-interval** 命令用来配置接收 echo 报文的最短时间间隔。

**undo bfd min-echo-receive-interval** 命令用来恢复缺省情况。

## 【命令】

```
bfd min-echo-receive-interval value  
undo bfd min-echo-receive-interval
```

## 【缺省情况】

接收 echo 报文的最短时间间隔为 400 毫秒。

## 【视图】

接口视图

## 【缺省用户角色】

network-admin

## 【参数】

**value:** 接收 echo 报文的最小时间间隔，单位为毫秒。取值为 0 或 10~1000。

## 【使用指导】

使用本命令，设备能够控制接收两个 echo 报文之间的时间间隔，即 echo 报文实际发送时间间隔。

## 【举例】

```
# 配置接口 GigabitEthernet2/0/1 接收 echo 报文的最小时间间隔为 500 毫秒。  
<Sysname> system-view  
[Sysname] interface gigabitethernet 2/0/1  
[Sysname-GigabitEthernet2/0/1] bfd min-echo-receive-interval 500
```

## 1.1.9 bfd min-receive-interval

**bfd min-receive-interval** 命令用来配置接收单跳 BFD 控制报文的最小时间间隔。

**undo bfd min-receive-interval** 命令用来恢复缺省情况。

## 【命令】

```
bfd min-receive-interval value  
undo bfd min-receive-interval
```

## 【缺省情况】

接收单跳 BFD 控制报文的最小时间间隔为 400 毫秒。

## 【视图】

接口视图/BFD 模板视图

## 【缺省用户角色】

network-admin

## 【参数】

**value:** 接收单跳 BFD 控制报文的最小时间间隔，单位为毫秒，取值范围为 10~1000。

## 【使用指导】

本命令主要为了防止对端发送控制报文的速度超过本地接收控制报文的速度。

对端的控制报文实际发送时间为对端发送控制报文的最小时间间隔和本地接收控制报文的最小时间间隔之间的较大值。

## 【举例】

```
# 配置接口 GigabitEthernet2/0/1 接收单跳 BFD 控制报文的最小时间间隔为 500 毫秒。  
<Sysname> system-view  
[Sysname] interface gigabitethernet 2/0/1  
[Sysname-GigabitEthernet2/0/1] bfd min-receive-interval 500
```

## 1.1.10 bfd min-transmit-interval

**bfd min-transmit-interval** 命令用来配置发送单跳 BFD 控制报文的最小时间间隔。

**undo bfd min-transmit-interval** 命令用来恢复缺省情况。

## 【命令】

```
bfd min-transmit-interval value
```

**undo bfd min-transmit-interval**

### 【缺省情况】

发送单跳 BFD 控制报文的最小时间间隔为 400 毫秒。

### 【视图】

接口视图/BFD 模板视图

### 【缺省用户角色】

network-admin

### 【参数】

**value:** 发送单跳 BFD 控制报文的最小时间间隔，单位为毫秒，取值范围为 10~1000。

### 【使用指导】

本命令主要是为了保证发送 BFD 控制报文的速度不能超过设备发送报文的能力。本地实际发送 BFD 控制报文的时间间隔，为本地配置的发送 BFD 控制报文的最小时间间隔和对端接收 BFD 控制报文的最小时间间隔的最大值。

### 【举例】

```
# 配置接口 GigabitEthernet2/0/1 发送单跳 BFD 控制报文的最小时间间隔为 500 毫秒。  
<Sysname> system-view  
[Sysname] interface gigabitethernet 2/0/1  
[Sysname-GigabitEthernet2/0/1] bfd min-transmit-interval 500
```

## 1.1.11 bfd multi-hop authentication-mode

**bfd multi-hop authentication-mode** 命令用来配置多跳 BFD 控制报文进行认证的方式。

**undo bfd multi-hop authentication-mode** 命令用来恢复缺省情况。

### 【命令】

**bfd multi-hop authentication-mode { m-md5 | m-sha1 | md5 | sha1 | simple } key-id { cipher cipher-string | plain plain-string }**

**undo bfd multi-hop authentication-mode**

### 【缺省情况】

多跳 BFD 控制报文不进行认证。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

**m-md5:** 采用 Meticulous MD5 算法进行认证。

**m-sha1:** 采用 Meticulous SHA1 算法进行认证。

**md5:** 采用 MD5 算法进行认证。

**sha1:** 采用 SHA1 算法进行认证。

**simple:** 采用简单认证。

**key-id:** 认证字标识符，取值范围为 1~255。

**cipher:** 表示输入的密码为密文。

**cipher-string:** 表示设置的密文密码，为 33~53 个字符的字符串，区分大小写。

**plain:** 表示输入的密码为明文。

**plain-string:** 表示设置的明文密码，为 1~16 个字符的字符串，区分大小写。

## 【使用指导】

本命令主要为了提高 BFD 会话的安全性。

以明文或密文方式设置的密码，均以密文的方式保存在配置文件中。

## 【举例】

# 配置多跳 BFD 控制报文进行简单明文认证，认证字标识符为 1，密码为 123456。

```
<Sysname> system-view  
[Sysname] bfd multi-hop authentication-mode simple 1 plain 123456
```

## 1.1.12 bfd multi-hop destination-port

**bfd multi-hop destination-port** 命令用来配置多跳 BFD 控制报文的目的端口号。

**undo bfd multi-hop destination-port** 命令用来恢复缺省情况。

## 【命令】

**bfd multi-hop destination-port port-number**

**undo bfd multi-hop destination-port**

## 【缺省情况】

多跳 BFD 控制报文的目的端口号为 4784。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

**port-number:** 多跳 BFD 控制报文的目的端口号，取值可以为 3784 或者 4784。

## 【举例】

# 配置多跳 BFD 控制报文的目的端口号为 3784。

```
<Sysname> system-view  
[Sysname] bfd multi-hop destination-port 3784
```

## 1.1.13 bfd multi-hop detect-multiplier

**bfd multi-hop detect-multiplier** 命令用来配置多跳 BFD 检测时间倍数。

**undo bfd multi-hop detect-multiplier** 命令用来恢复缺省情况。

## 【命令】

**bfd multi-hop detect-multiplier value**

**undo bfd multi-hop detect-multiplier**

## 【缺省情况】

多跳 BFD 检测时间倍数为 5。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

**value:** 多跳 BFD 检测时间倍数，取值范围为 3~50。

## 【使用指导】

检测时间倍数，即接收方允许发送方发送 BFD 控制报文的最大连续丢包数。

对于控制报文方式的异步模式，实际检测时间为接收方的检测时间倍数和接收方的实际发送时间的乘积；对于控制报文方式的查询模式，实际检测时间为发送方的检测时间倍数和发送方的实际发送时间的乘积。

## 【举例】

# 配置多跳 BFD 检测时间倍数为 6。

```
<Sysname> system-view  
[Sysname] bfd multi-hop detect-multiplier 6
```

## 1.1.14 bfd multi-hop min-receive-interval

**bfd multi-hop min-receive-interval** 命令用来配置接收多跳 BFD 控制报文的最小时间间隔。

**undo bfd multi-hop min-receive-interval** 命令用来恢复缺省情况。

## 【命令】

**bfd multi-hop min-receive-interval value**

**undo bfd multi-hop min-receive-interval**

## 【缺省情况】

接收多跳 BFD 控制报文的最小时间间隔为 400 毫秒。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

**value:** 接收 BFD 控制报文的最小时间间隔，单位为毫秒，取值范围为 10~1000。

## 【使用指导】

本命令主要为了防止对端设备发送报文的速度超出本地接收报文的能力(接收 BFD 控制报文的最小时间间隔)，若超出，则对端设备将发送 BFD 控制报文的时间间隔动态调整为本地接收 BFD 控制报文的最小时间间隔。

## 【举例】

# 配置接收多跳 BFD 控制报文的最小时间间隔为 500 毫秒。

```
<Sysname> system-view  
[Sysname] bfd multi-hop min-receive-interval 500
```

## 1.1.15 bfd multi-hop min-transmit-interval

**bfd multi-hop min-transmit-interval** 命令用来配置发送多跳 BFD 控制报文的最小时间间隔。

**undo bfd multi-hop min-transmit-interval** 命令用来恢复缺省情况。

## 【命令】

```
bf  
d multi-hop min-transmit-interval value  
undo bf  
d multi-hop min-transmit-interval
```

## 【缺省情况】

发送多跳 BFD 控制报文的最小时间间隔为 400 毫秒。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

**value:** 发送 BFD 控制报文的最小时间间隔，单位为毫秒，取值范围为 10~1000。

## 【使用指导】

本命令主要是为了保证发送 BFD 控制报文的速度不能超过设备发送报文的能力。本地实际发送 BFD 控制报文的时间间隔，为本地配置的发送 BFD 控制报文的最小时间间隔和对端接收 BFD 控制报文的最小时间间隔的最大值。

## 【举例】

```
# 配置发送多跳 BFD 控制报文的最小时间间隔为 500 毫秒。  
<Sysname> system-view  
[Sysname] bf  
d multi-hop min-transmit-interval 500
```

## 1.1.16 bf d init-fail timer

**bf  
d init-fail-timer** 命令用来配置 BFD 会话无法建立时，通知上层协议 BFD 会话 down 的超时时间。

**undo bf  
d init-fail-timer** 命令用来恢复缺省情况。

## 【命令】

```
bf  
d init-fail-timer seconds  
undo bf  
d init-fail-timer
```

## 【缺省情况】

BFD 会话无法建立时，不会通知上层协议 BFD 会话 down。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

**seconds:** BFD 会话无法建立时，通知上层协议 BFD 会话 down 的超时时间，即在 **seconds** 时间后通知上层协议，取值范围为 5~600，单位为秒。

## 【使用指导】

缺省情况下，控制报文工作方式的 BFD 会话无法建立时，不会通知上层协议 BFD 会话 down。某些情况下，需要将 BFD 会话无法建立的消息通知给上层协议，以使上层协议作出正确的处理。比如在聚合链路中，由于链路故障等原因，BFD 会话无法进入 up 状态，从而导致聚合模块无法及时将成员端口的选中状态修改为非选中状态，此时，可以配置本命令。

配置本命令后，对于由于配置原因（比如对端设备没有使能 BFD，或者两端的 BFD 认证配置不一致等）造成 BFD 会话无法进入 up 状态的情况，如果配置了本定时器，会导致上层协议作出错误的处理，所以，请谨慎使用本命令。

对于 Echo 报文方式的 BFD 会话，本命令不生效。

### 【举例】

```
# 配置控制报文工作方式的 BFD 会话初始 down 事件定时器的超时时间为 10 秒。
```

```
<Sysname> system-view  
[Sysname] bfd init-fail-timer 10
```

## 1.1.17 bfd session init-mode

**bfd session init-mode** 命令用来配置 BFD 会话建立前的运行模式。

**undo bfd session init-mode** 命令用来恢复缺省情况。

### 【命令】

```
bfd session init-mode { active | passive }
```

```
undo bfd session init-mode
```

### 【缺省情况】

BFD 会话建立前的运行模式为主动模式。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

**active**: 主动模式。在建立会话前不管是否收到对端发来的 BFD 控制报文，都会主动向会话的对端发送 BFD 控制报文。

**passive**: 被动模式。在建立会话前不会主动向会话的对端发送 BFD 控制报文，只有等收到 BFD 控制报文后才会向对端发送 BFD 控制报文。

### 【使用指导】

通信双方至少要有一方运行在主动模式才能成功建立起 BFD 会话。

### 【举例】

```
# 配置 BFD 会话建立前的运行模式为被动模式。
```

```
<Sysname> system-view  
[Sysname] bfd session init-mode passive
```

## 1.1.18 bfd template

**bfd template** 命令用来创建 BFD 模板，并进入 BFD 模板视图。

**undo bfd template** 命令用来删除 BFD 模板。

### 【命令】

```
bfd template template-name
```

```
undo bfd template template-name
```

### 【缺省情况】

没有创建 BFD 模板。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

**template-name:** BFD 模板名称，为 1~63 个字符的字符串，区分大小写。

## 【举例】

# 创建 BFD 模板 bfd1，并进入 BFD 模板视图。

```
<Sysname> system-view  
[Sysname] bfd template bfd1  
[Sysname-bfd-template-bfd1]
```

## 1.1.19 display bfd session

**display bfd session** 命令用来显示 BFD 会话信息。

## 【命令】

**display bfd session [ discriminator value | verbose ]**

## 【视图】

任意视图

## 【缺省用户角色】

network-admin

network-operator

## 【参数】

**discriminator value:** 显示指定本地标识符的 BFD 会话信息。**value** 为本地标识符的值，取值范围为 1~4294967295。如果未指定本参数，将显示所有 BFD 会话概要信息。

**verbose:** 显示会话的详细信息。如果未指定本参数，将显示 BFD 会话概要信息。

## 【举例】

# 显示所有 BFD 会话的信息 (IPv4)。

```
<Sysname> display bfd session
```

```
Total Session Num: 1      Up Session Num: 1      Init Mode: Active
```

```
IPv4 Session Working Under Ctrl Mode:
```

LD/RD	SourceAddr	DestAddr	State	Holddtime	Interface
513/513	1.1.1.1	1.1.1.2	Up	2297ms	GE2/0/1

# 显示所有 BFD 会话的信息 (IPv6)。

```
<Sysname> display bfd session
```

```
Total Session Num: 1      Up Session Num: 1      Init Mode: Active
```

```
IPv6 Session Working Under Ctrl Mode:
```

Local Discr:	Remote Discr:
513	513
Source IP: FE80::20C:29FF:FE72:AC4D	Source IP: FE80::20C:29FF:FE72:AC4D
Destination IP: FE80::20C:29FF:FE72:AC4D	Destination IP: FE80::20C:29FF:FE72:AC4D

```

Session State: Up           Interface: GE2/0/2
Hold Time: 2142ms

# 显示 BFD 会话的详细信息 (IPv4)。
<Sysname> display bfd session verbose

Total Session Num: 1      Up Session Num: 1      Init Mode: Active

IPv4 Session Working Under Ctrl Mode:

    Local Discr: 513          Remote Discr: 513
    Source IP: 1.1.1.1        Destination IP: 1.1.1.2
    Session State: Up        Interface: GigabitEthernet2/0/1
    Min Tx Inter: 500ms      Act Tx Inter: 500ms
    Min Rx Inter: 500ms      Detect Inter: 2500ms
    Rx Count: 42            Tx Count: 43
    Connect Type: Direct    Running Up for: 00:00:20
    Hold Time: 2078ms       Auth mode: None
    Detect Mode: Async      Slot: 0
    Protocol: OSPF
    Diag Info: No Diagnostic
    Template name: abc

# 显示 BFD 会话的详细信息 (IPv6)。
<Sysname> display bfd session verbose

Total Session Num: 1      Up Session Num: 1      Init Mode: Active

IPv6 Session Working Under Ctrl Mode:

    Local Discr: 513          Remote Discr: 513
    Source IP: FE80::20C:29FF:FED4:7171
    Destination IP: FE80::20C:29FF:FE72:AC4D
    Session State: Up        Interface: GigabitEthernet2/0/2
    Min Tx Inter: 500ms      Act Tx Inter: 500ms
    Min Rx Inter: 500ms      Detect Inter: 2500ms
    Rx Count: 38            Tx Count: 38
    Connect Type: Direct    Running Up for: 00:00:15
    Hold Time: 2211ms       Auth mode: None
    Detect Mode: Async      Slot: 0
    Protocol: OSPFv3
    Diag Info: No Diagnostic
    Template name: abc

```

表1-1 display bfd session 命令显示信息描述表

字段	描述
Total Session Num	所有BFD会话的数目
Up Session Num	up的BFD会话的数目
Init Mode	BFD运行模式: ● Active: 主动模式 ● Passive: 被动模式
Session Working Under Ctrl Mode	BFD会话（有IPv4和IPv6两种）的工作方式: ● Ctrl: 控制报文方式

字段	描述
	<ul style="list-style-type: none"> <li>• Echo: echo 报文方式</li> </ul>
Local Discr/LD	会话的本地标识符
Remote Discr/RD	会话的远端标识符
Source IP/SourceAddr	会话的源IP地址
Destination IP/DestAddr	会话的目的IP地址
Session State/State	会话状态: Up和Down
Interface	会话所在的接口名
Min Tx Inter	最小发送时间间隔
Min Rx Inter	最小接收时间间隔
Act Tx Inter	实际发送间隔
Detect Inter	实际检测间隔
Rx Count	接收的报文数
Tx Count	发送的报文数
Hold Time/Holdtime	离会话检测时间超时的剩余时间
Auth mode	会话的认证模式, 目前只支持Simple
Connect Type	接口的连接类型: <ul style="list-style-type: none"> <li>• Direct: 直连</li> <li>• Indirect: 非直连</li> </ul>
Running up for	会话持续up的时间
Detect Mode	检测模式: <ul style="list-style-type: none"> <li>• Async: 异步模式</li> <li>• Demand: 查询模式</li> </ul>
Slot	槽号
Protocol	协议名
Diag Info	会话的诊断信息
Template name	会话指定模板的名称

### 1.1.20 reset bfd session statistics

**reset bfd session statistics** 命令用来清除所有 BFD 会话的统计信息。

#### 【命令】

**reset bfd session statistics**

#### 【视图】

用户视图

#### 【缺省用户角色】

network-admin

#### 【举例】

# 清除所有 BFD 会话的统计信息。

```
<Sysname> reset bfd session statistics
```

### 1.1.21 snmp-agent trap enable bfd

**snmp-agent trap enable bfd** 命令用来开启 BFD 的告警功能。

**undo snmp-agent trap enable bfd** 命令用来关闭 BFD 的告警功能。

#### 【命令】

**snmp-agent trap enable bfd**

**undo snmp-agent trap enable bfd**

#### 【缺省情况】

BFD 的告警功能处于开启状态。

#### 【视图】

系统视图

#### 【缺省用户角色】

network-admin

#### 【使用指导】

开启 BFD 模块的告警功能后，该模块会生成告警信息，用于报告该模块的重要事件。生成的告警信息将发送到设备的 SNMP 模块，通过设置 SNMP 中告警信息的发送参数，来决定告警信息输出的相关属性。（有关告警信息的详细介绍，请参见“网络管理和监控配置指导”中的“SNMP”。）

#### 【举例】

# 关闭 BFD 的告警功能。

```
<Sysname> system-view
```

```
[Sysname] undo snmp-agent trap enable bfd
```

# 目 录

1 Track.....	1-1
1.1 Track配置命令.....	1-1
1.1.1 display track.....	1-1
1.1.2 track bfd.....	1-2
1.1.3 track interface.....	1-3
1.1.4 track interface protocol.....	1-4
1.1.5 track nqa.....	1-6

# 1 Track

## 1.1 Track配置命令

### 1.1.1 display track

**display track** 命令用来显示 Track 项信息。

#### 【命令】

```
display track { track-entry-number | all }
```

#### 【视图】

任意视图

#### 【缺省用户角色】

network-admin

network-operator

#### 【参数】

**track-entry-number:** 显示指定 Track 项的信息。**track-entry-number** 为 Track 项的序号，取值范围为 1~1024。

**all:** 显示所有 Track 项的信息。

#### 【举例】

# 显示所有 Track 项的信息。

```
<Sysname> display track all
Track ID: 1
    State: Positive
    Duration: 0 days 0 hours 0 minutes 7 seconds
    Notification delay: Positive 20, Negative 30 (in seconds)
    Tracked object:
        NQA entry: admin test
        Reaction: 10
Track ID: 2
    State: NotReady
    Duration: 0 days 0 hours 0 minutes 32 seconds
    Notification delay: Positive 20, Negative 30 (in seconds)
    Tracked object:
        BFD session mode: Echo
        Outgoing interface: GigabitEthernet2/0/1
        VPN instance name: -
        Remote IP: 192.168.40.1
        Local IP: 192.168.40.2
Track ID: 3
    State: Negative
    Duration: 0 days 0 hours 0 minutes 32 seconds
    Notification delay: Positive 20, Negative 30 (in seconds)
    Tracked object:
```

```
Interface: GigabitEthernet2/0/2
Protocol: IPv4
```

表1-1 display track 命令输出信息描述

字段	描述
Track ID	Track项序号
State	Track项的状态, 取值包括: <ul style="list-style-type: none"><li>• Positive: 表示状态正常</li><li>• NotReady: 表示无效值</li><li>• Negative: 表示状态异常</li></ul>
Duration	Track项处于当前状态的持续时间
Notification delay: Positive 20, Negative 30 (in seconds)	通知延迟: <ul style="list-style-type: none"><li>• Track 项状态变为 Positive 后, 延迟 20 秒通知应用模块</li><li>• Track 项状态变为 Negative 后, 延迟 30 秒通知应用模块</li></ul>
Tracked object	Track项关联的对象
NQA entry	Track项关联的NQA测试组
Reaction	Track项关联的联动项
BFD session mode	BFD会话的模式, 当前只支持Echo模式
Outgoing interface	BFD会话报文的出接口
VPN instance name	BFD会话报文所属VPN实例的名称。如果属于公网, 则显示为“-”
Remote IP	BFD会话报文的远端IP地址
Local IP	BFD会话报文的本地IP地址
Interface	Track项关联的接口
Protocol	监视接口的链路状态或网络层协议状态, 取值包括: <ul style="list-style-type: none"><li>• None: 监视接口的链路状态</li><li>• IPv4: 监视三层接口的 IPv4 协议状态</li><li>• IPv6: 监视三层接口的 IPv6 协议状态</li></ul>

## 【相关命令】

- **track bfd**
- **track interface**
- **track interface protocol**
- **track nqa**

### 1.1.2 track bfd

**track bfd** 命令用来创建和 BFD 会话关联的 Track 项。

**undo track** 命令用来删除指定的 Track 项。

## 【命令】

```
track track-entry-number bfd echo interface interface-type interface-number remote ip  
remote-ip local ip local-ip [ delay { negative negative-time | positive positive-time } * ]  
undo track track-entry-number
```

## 【缺省情况】

设备上不存在任何 Track 项。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

**track-entry-number:** Track 项的序号，取值范围为 1~1024。

**interface interface-type interface-number :** BFD 会话报文的出接口。**interface-type** **interface-number** 为接口类型和接口编号。

**remote ip remote-ip:** BFD 会话探测的远端 IP 地址。

**local ip local-ip:** BFD 会话探测的本地 IP 地址。

**delay:** 指定 Track 项状态发生变化时，延迟通知应用模块。如果不指定该参数，则 Track 项状态变化后立即通知应用模块。

**negative negative-time:** 指定 Track 项状态变为 Negative 时，延迟通知应用模块的时间。  
**negative-time** 为延迟时间，取值范围为 1~300，单位为秒。

**positive positive-time:** 指定 Track 项状态变为 Positive 时，延迟通知应用模块的时间。  
**positive-time** 为延迟时间，取值范围为 1~300，单位为秒。

## 【使用指导】

- Track 项创建后，不能通过重复执行 **track** 命令的方式修改 Track 项关联的内容。只能删除 Track 项后，再重新创建 Track 项。
- Track 项创建后，可以通过再次执行 **track bfd delay** 命令的方式修改延迟通知应用模块的时间。
- 配置 Track 与 BFD 联动时，VRRP 备份组的虚拟 IP 地址不能作为 BFD 会话探测的本地地址和远端地址。

## 【举例】

```
# 创建与 BFD 会话关联的 Track 项 1。BFD 会话使用 Echo 报文进行探测，出接口为  
GigabitEthernet2/0/1，远端 IP 地址为 192.168.40.1，本地 IP 地址为 192.168.40.2。  
<Sysname> system-view  
[Sysname] track 1 bfd echo interface gigabitethernet 2/0/1 remote ip 192.168.40.1 local ip  
192.168.40.2
```

## 【相关命令】

- **display track**

### 1.1.3 track interface

**track interface** 命令用来创建与指定接口链路状态关联的 Track 项。

**undo track** 命令用来删除指定的 Track 项。

### 【命令】

```
track track-entry-number interface interface-type interface-number [ delay { negative negative-time | positive positive-time } * ]  
undo track track-entry-number
```

### 【缺省情况】

设备上不存在任何 Track 项。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

*track-entry-number*: Track 项的序号，取值范围为 1~1024。

*interface-type interface-number*: 监视的接口类型和接口编号。

**delay**: 指定 Track 项状态发生变化时，延迟通知应用模块。如果不指定该参数，则 Track 项状态变化后立即通知应用模块。

**negative negative-time**: 指定 Track 项状态变为 Negative 时，延迟通知应用模块的时间。

*negative-time* 为延迟时间，取值范围为 1~300，单位为秒。

**positive positive-time**: 指定 Track 项状态变为 Positive 时，延迟通知应用模块的时间。

*positive-time* 为延迟时间，取值范围为 1~300，单位为秒。

### 【使用指导】

创建与接口链路状态关联的 Track 项后，接口的链路状态为 up 时，Track 项的状态为 Positive；接口的链路状态为 down 时，Track 项的状态为 Negative。通过 **display ip interface brief** 命令可以查看接口的链路状态。

需要注意的是：

- Track 项创建后，不能通过重复执行 **track** 命令的方式修改 Track 项关联的内容。只能删除 Track 项后，再重新创建 Track 项。
- Track 项创建后，可以通过再次执行 **track interface delay** 命令的方式修改延迟通知应用模块的时间。

### 【举例】

```
# 创建与接口 GigabitEthernet2/0/1 的链路状态关联的 Track 项 1。  
<Sysname> system-view  
[Sysname] track 1 interface gigabitethernet 2/0/1
```

### 【相关命令】

- **display track**
- **display ip interface brief** (三层技术-IP 业务命令参考/IP 地址)

## 1.1.4 track interface protocol

**track interface protocol** 命令用来创建与指定接口网络层协议状态关联的 Track 项。

**undo track** 命令用来删除指定的 Track 项。

## 【命令】

```
track track-entry-number interface interface-type interface-number protocol { ipv4 | ipv6 }
[ delay { negative negative-time | positive positive-time } * ]
undo track track-entry-number
```

## 【缺省情况】

设备上不存在任何 Track 项。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

*track-entry-number*: Track 项的序号，取值范围为 1~1024。

*interface-type interface-number*: 监视的接口类型和接口编号。

**ipv4**: 监视接口的 IPv4 协议状态。接口的 IPv4 协议状态为 up 时，Track 项的状态为 Positive；接口的 IPv4 协议状态为 down 时，Track 项的状态为 Negative。通过 **display ip interface brief** 命令可以查看接口的 IPv4 协议状态。

**ipv6**: 监视接口的 IPv6 协议状态。接口的 IPv6 协议状态为 up 时，Track 项的状态为 Positive；接口的 IPv6 协议状态为 down 时，Track 项的状态为 Negative。通过 **display ipv6 interface brief** 命令可以查看接口的 IPv6 协议状态。

**delay**: 指定 Track 项状态发生变化时，延迟通知应用模块。如果不指定该参数，则 Track 项状态变化后立即通知应用模块。

**negative negative-time**: 指定 Track 项状态变为 Negative 时，延迟通知应用模块的时间。  
*negative-time* 为延迟时间，取值范围为 1~300，单位为秒。

**positive positive-time**: 指定 Track 项状态变为 Positive 时，延迟通知应用模块的时间。  
*positive-time* 为延迟时间，取值范围为 1~300，单位为秒。

## 【使用指导】

- Track 项创建后，不能通过重复执行 **track** 命令的方式修改 Track 项关联的内容。只能删除 Track 项后，再重新创建 Track 项。
- Track 项创建后，可以通过再次执行 **track interface protocol delay** 命令的方式修改延迟通知应用模块的时间。

## 【举例】

```
# 创建与接口 GigabitEthernet2/0/1 的 IPv4 协议状态关联的 Track 项 1。
<Sysname> system-view
[Sysname] track 1 interface gigabitethernet 2/0/1 protocol ipv4
```

## 【相关命令】

- **display ip interface brief** (三层技术-IP 业务命令参考/IP 地址)
- **display ipv6 interface brief** (三层技术-IP 业务命令参考/IPv6 基础)
- **display track**

## 1.1.5 track nqa

**track nqa** 命令用来创建与 NQA 测试组中指定联动项关联的 Track 项。

**undo track** 命令用来删除指定的 Track 项。

### 【命令】

```
track track-entry-number nqa entry admin-name operation-tag reaction item-number [ delay  
{ negative negative-time | positive positive-time } * ]  
undo track track-entry-number
```

### 【缺省情况】

设备上不存在任何 Track 项。

### 【视图】

系统视图

### 【缺省用户角色】

network-admin

### 【参数】

**track-entry-number:** Track 项的序号，取值范围为 1~1024。

**entry admin-name operation-tag:** 指定与 Track 项关联的 NQA 测试组。其中，*admin-name* 为创建 NQA 测试组的管理员的名字，为 1~32 个字符的字符串，不区分大小写；*operation-tag* 为 NQA 测试操作的标签，为 1~32 个字符的字符串，不区分大小写。

**reaction item-number:** 指定与 Track 项关联的联动项。其中，*item-number* 为联动项的序号，取值范围为 1~10。

**delay:** 指定 Track 项状态发生变化时，延迟通知应用模块。如果不指定该参数，则 Track 项状态变化后立即通知应用模块。

**negative negative-time:** 指定 Track 项状态变为 Negative 时，延迟通知应用模块的时间。  
*negative-time* 为延迟时间，取值范围为 1~300，单位为秒。

**positive positive-time:** 指定 Track 项状态变为 Positive 时，延迟通知应用模块的时间。  
*positive-time* 为延迟时间，取值范围为 1~300，单位为秒。

### 【使用指导】

- Track 项创建后，不能通过重复执行**track**命令的方式修改 Track 项关联的内容。只能删除 Track 项后，再重新创建 Track 项。
- Track 项创建后，可以通过再次执行**track nqa delay** 命令的方式修改延迟通知应用模块的时间。

### 【举例】

```
# 创建与 NQA 测试组 (admin-test) 中联动项 3 关联的 Track 项 1。  
<Sysname> system-view  
[Sysname] track 1 nqa entry admin test reaction 3
```

### 【相关命令】

- **display track**

# 目 录

1 进程分布优化 .....	1-1 .....
1.1 进程分布优化配置命令 .....	1-1 .....
1.1.1 affinity location-set .....	1-1 .....
1.1.2 affinity location-type .....	1-2 .....
1.1.3 affinity program .....	1-3 .....
1.1.4 affinity self .....	1-4 .....
1.1.5 display ha service-group .....	1-5 .....
1.1.6 display placement location .....	1-7 .....
1.1.7 display placement policy .....	1-9 .....
1.1.8 display placement program .....	1-10 .....
1.1.9 display placement reoptimize .....	1-11 .....
1.1.10 placement program .....	1-12 .....
1.1.11 placement reoptimize .....	1-13 .....

# 1 进程分布优化

## 1.1 进程分布优化配置命令

### 1.1.1 affinity location-set

**affinity location-set** 命令用来设置进程对于节点位置的偏好。

**undo affinity location-set** 命令用来取消设置。

#### 【命令】

独立运行模式:

```
affinity location-set { slot slot-number [ cpu cpu-number ] }&<1-5> { attract strength | default | none | repulse strength }
```

```
undo affinity location-set { slot slot-number [ cpu cpu-number ] }&<1-5>
```

IRF 模式:

```
affinity location-set { chassis chassis-number slot slot-number [ cpu cpu-number ] }&<1-5> { attract strength | default | none | repulse strength }
```

```
undo affinity location-set { chassis chassis-number slot slot-number [ cpu cpu-number ] }&<1-5>
```

#### 【缺省情况】

系统未配置进程对节点位置的偏好。

#### 【视图】

分布策略视图

#### 【缺省用户角色】

network-admin

#### 【参数】

{ **chassis chassis-number slot slot-number [ cpu cpu-number ]** }&<1-5>: 表示当前进程在指定 CPU 上运行的偏好。其中:

- **chassis chassis-number**: 表示设备在 IRF 中的成员编号。(IRF 模式)
- **slot slot-number**: 表示主控所在的槽位号。
- **cpu cpu-number**: 表示 CPU 的编号。取值只能为 0。
- **&<1-5>**: 表示前面的参数最多可以输入 5 次。

**attract strength**: 正向偏好程度, 表示希望运行在该位置。**strength** 表示偏好程度, 取值范围为 1~100000。值越大表示进程运行在该位置的可能性越大。

**default**: 缺省偏好, 取值为正向偏好 200。

**none**: 设置偏好为 0, 即主控进程对具体节点没有偏好, 主进程的运行位置由系统来决定。

**repulse strength**: 反向偏好程度, 表示不希望运行在该位置。**strength** 表示偏好程度, 取值范围为 1~100000。值越大表示进程运行在该位置的可能性越小。

## 【举例】

```
# 设置 staticroute 对于 3 号槽位的正向偏好为 500。(独立运行模式)
<Sysname> system-view
[Sysname] placement program staticroute
[Sysname-program-staticroute] affinity location-set slot 3 attract 500
# 设置 staticroute 对于 1 号成员设备的 3 号槽位的正向偏好为 500。(IRF 模式)
<Sysname> system-view
[Sysname] placement program staticroute
[Sysname-program-staticroute] affinity location-set chassis 1 slot 3 attract 500
```

## 1.1.2 affinity location-type

**affinity location-type** 命令用来设置进程对于位置类型的偏好。

**undo affinity location-type** 命令用来恢复缺省情况。

## 【命令】

```
affinity location-type { current | paired | primary } { attract strength | default | none | repulse strength }
undo affinity location-type { current | paired | primary }
```

## 【缺省情况】

系统未配置进程对位置类型的偏好。

## 【视图】

分布策略视图

## 【缺省用户角色】

network-admin

## 【参数】

**current**: 用来设置对主控进程当前运行位置的偏好。主控进程当前运行位置可以通过 **display placement program** 命令查看。

**paired**: 用来设置对所有备份进程当前运行位置的偏好。

**primary**: 用来设置对主用主控板的偏好。(独立运行模式)

**primary**: 用来设置对全局主用主控板的偏好。(IRF 模式)

**attract strength**: 正向偏好程度, 表示希望运行在该位置。**strength** 表示偏好程度, 取值范围为 1~100000。值越大表示进程运行在该位置类型的可能性越大。

**default**: 缺省偏好, 取值为正向偏好 200。

**none**: 设置偏好为 0, 即主进程对位置类型没有偏好, 主进程的运行位置由系统来决定。

**repulse strength**: 反向偏好程度, 表示不希望运行在该位置。**strength** 表示偏好程度, 取值范围为 1~100000。值越大表示进程运行在该位置类型的可能性越小。

## 【举例】

```
# 设置 staticroute 对于当前位置的正向偏好为 500。
```

```
<Sysname> system-view
[Sysname] placement program staticroute
```

```
[Sysname-program-staticroute] affinity location-type current attract 500
```

### 【相关命令】

- **affinity location-set**
- **affinity program**

### 1.1.3 affinity program

**affinity program** 命令用来设置本进程和其它进程运行在同一位置的偏好。

**undo affinity program** 命令用来取消设置。

### 【命令】

```
affinity program program-name { attract strength | default | none | repulse strength }
```

```
undo affinity program program-name
```

### 【缺省情况】

进程未配置和其它进程运行在同一位置的偏好。

### 【视图】

分布策略视图

### 【缺省用户角色】

network-admin

### 【参数】

**program-name:** 为当前设备上正在运行的进程的名称，为 1~15 个字符的字符串，不区分大小写。

用户可以通过 **display placement program all** 命令查看设备上正在运行的进程。

**attract strength:** 正向偏好程度，表示希望运行在该位置。**strength** 表示偏好程度，取值范围为 1~100000。值越大表示进程运行于同一位置的可能性越大。

**default:** 缺省偏好，取值为正向偏好 200。

**none:** 设置偏好为 0，即主进程对于是否和其它进程运行在同一位置没有偏好，主进程的运行位置由系统来决定。

**repulse strength:** 反向偏好程度，表示不希望运行在该位置。**strength** 表示偏好程度，取值范围为 1~100000。值越大表示进程运行于同一位置的可能性越小。

### 【使用指导】

该配置方式以其它进程通过进程分布策略计算出来的预测位置为参照物，配置的是本进程和其它进程运行在同一位置的偏好。

### 【举例】

```
# 设置 syslog 和 staticroute 运行于同一位置的偏好为反向 200。
<Sysname> system-view
[Sysname] placement program staticroute
[Sysname-program-ospf] affinity program syslog repulse 200
```

### 【相关命令】

- **affinity location-set**

- **affinity location-type**

#### 1.1.4 affinity self

**affinity self** 命令用来设置本进程所有实例运行于同一位置的偏好。

**undo affinity self** 命令用来取消设置。

##### 【命令】

**affinity self { attract strength | default | none | repulse strength }**

**undo affinity self**

##### 【缺省情况】

进程未配置所有实例运行于同一位置的偏好。

##### 【视图】

分布策略视图

##### 【缺省用户角色】

network-admin

##### 【参数】

**attract strength:** 正向偏好程度，表示希望运行在该位置。*strength* 表示偏好程度，取值范围为 1~100000。值越大表示进程运行于同一位置的可能性越大。

**default:** 缺省偏好，取值为正向偏好 200。

**none:** 设置偏好为 0，即进程对所有实例是否运行于同一位置没有偏好，运行位置由系统来决定。

**repulse strength:** 反向偏好程度，表示不希望运行在该位置。*strength* 表示偏好程度，取值范围为 1~100000。值越大表示进程运行于同一位置的可能性越小。

##### 【使用指导】

该配置用以决定一个进程的多个实例是否运行于同一个位置上，如果进程只有一个实例，则该配置不会产生作用。

本命令在进程的分布策略视图和进程任意实例的分布策略视图下配置效果相同，均对所有实例生效。多次配置该命令，最新配置生效。

进程是否包含多个实例可以通过 **display placement program all** 命令查看。

##### 【举例】

```
# 设置 staticroute 进程所有实例运行于同一位置的偏好为反向 200。
<Sysname> system-view
[Sysname] placement program staticroute
[Sysname-program-staticroute] affinity self repulse 200
```

##### 【相关命令】

- **affinity location-set**
- **affinity location-type**

## 1.1.5 display ha service-group

**display ha service-group** 命令用来显示服务组的信息。

### 【命令】

**display ha service-group { service-group-name [ instance instance-name ] | all }**

### 【视图】

任意视图

### 【缺省用户角色】

network-admin

network-operator

### 【参数】

**service-group-name:** 为当前设备上正在运行的服务组的名称，为 1~15 个字符的字符串，不区分大小写。

**instance instance-name:** 表示服务组的实例的名称，为 1~31 个字符的字符串，不区分大小写。一个服务组是否存在多个实例，由系统软件决定。

**all:** 表示当前设备上运行的所有服务组。

### 【使用指导】

服务组是进程的集合。目前，大部分服务组只包含一个进程。如果进程支持实例，则服务组也有相应的实例。

### 【举例】

# 显示所有服务组主控进程的位置和状态信息。(独立运行模式)

```
<Sysname> display ha service-group all
Service Group          Current Location      State
-----
syslog                1/0                  Realtime Backup
cryptomgr             1/0                  Realtime Backup
trange                1/0                  Realtime Backup
lagg                  1/0                  Realtime Backup
aaa                   1/0                  Realtime Backup
acl                   1/0                  Realtime Backup
copp                 1/0                  Realtime Backup
lauth                1/0                  Realtime Backup
ethbase               1/0                  Realtime Backup
oap                   1/0                  Realtime Backup
qos                   1/0                  Realtime Backup
comsh                1/0                  Realtime Backup
xmlcfg               1/0                  Realtime Backup
ppp                   1/0                  Realtime Backup
ipbase               1/0                  Realtime Backup
ipaddr               1/0                  Realtime Backup
rib                  1/0                  Realtime Backup
staticroute           1/0                  Realtime Backup
```

telnet	1/0	Realtime Backup
lldp	1/0	Realtime Backup
te	1/0	Realtime Backup
pce	1/0	Realtime Backup
rsvp	1/0	Realtime Backup
ip6base	1/0	Realtime Backup
ip6addr	1/0	Realtime Backup
rib6	1/0	Realtime Backup
staticroute6	1/0	Realtime Backup
snmp	1/0	Realtime Backup

# 显示指定服务组主控进程的位置和状态信息。(独立运行模式)

<Sysname> display ha service-group staticroute

Service Group	Current Location	State
staticroute	1/0 (Active)	Realtime Backup

Detailed information about services of the program:

Service	PID	Type	Location	State
ifm	214	Active	1/0	Realtime Backup
staticroute	214	Active	1/0	Realtime Backup

# 显示所有服务组主控进程的位置和状态信息。(IRF 模式)

<Sysname> display ha service-group all

Service Group	Current Location	State
syslog	1/1/0	Realtime Backup
cryptomgr	1/1/0	Realtime Backup
trange	1/1/0	Realtime Backup
lagg	1/1/0	Realtime Backup
aaa	1/1/0	Realtime Backup
acl	1/1/0	Realtime Backup
copp	1/1/0	Realtime Backup
lauth	1/1/0	Realtime Backup
ethbase	1/1/0	Realtime Backup
oap	1/1/0	Realtime Backup
qos	1/1/0	Realtime Backup
comsh	1/1/0	Realtime Backup
xmlcfg	1/1/0	Realtime Backup
ppp	1/1/0	Realtime Backup
ipbase	1/1/0	Realtime Backup
ipaddr	1/1/0	Realtime Backup
rib	1/1/0	Realtime Backup
staticroute	1/1/0	Realtime Backup
telnet	1/1/0	Realtime Backup
lldp	1/0	Realtime Backup
te	1/1/0	Realtime Backup
pce	1/1/0	Realtime Backup
rsvp	1/1/0	Realtime Backup

```

ip6base          1/1/0           Realtime Backup
ip6addr          1/1/0           Realtime Backup
rib6             1/1/0           Realtime Backup
staticroute6     1/1/0           Realtime Backup
snmp             1/1/0           Realtime Backup

# 显示指定进程主备身份及状态。(IRF 模式)
<Sysname>display ha service-group staticroute
Service Group      Current Location      State
-----
staticroute        1/1/0 (Active)       Realtime Backup

Detailed information about services of the program:
Service      PID   Type    Location      State
-----
ifm          214   Active   1/1/0       Realtime Backup
staticroute  214   Active   1/1/0       Realtime Backup

```

表1-1 display ha service-group 命令显示信息描述表

字段	描述
Service Group	服务组的名称
Current Location	服务组主控进程当前运行的位置
State	服务组主控进程和备份进程的备份状态
Detailed information about services of the program	服务组包含的所有进程（包括主进程和备进程）的详细信息
PID	进程的编号
Type	进程的主备身份，取值为： <ul style="list-style-type: none"> <li>• Active：表示服务组主控进程</li> <li>• Standby：表示服务组备用进程</li> </ul>
Service	服务组内的服务的名称
Location	当前服务主控进程的运行位置
State	进程的状态： <ul style="list-style-type: none"> <li>• Realtime Backup：实时备份状态</li> <li>• Batch Backup：批量备份状态</li> <li>• Stopping：停止状态</li> <li>• Degrading：降级状态</li> <li>• Upgrading：升级状态</li> </ul>

### 1.1.6 display placement location

**display placement location** 命令用来显示具体位置上正在运行的进程信息。

## 【命令】

独立运行模式:

```
display placement location { all | slot slot-number [ cpu cpu-number ] }
```

IRF 模式:

```
display placement location { all | chassis chassis-number slot slot-number [ cpu cpu-number ] }
```

## 【视图】

任意视图

## 【缺省用户角色】

network-admin

network-operator

## 【参数】

**all:** 表示当前设备上运行的所有进程。

**slot slot-number:** 表示主控所在的槽位号。(独立运行模式)

**chassis chassis-number slot slot-number:** 表示指定成员设备上的指定主控。*chassis-number* 表示设备在 IRF 中的成员编号, *slot-number* 表示主控所在的槽位号。不指定该参数时, 表示所有主控板。(IRF 模式)

**cpu cpu-number:** 表示 CPU 的编号, 取值始终为 0。

## 【举例】

# 显示 1 号槽主控上正在运行的进程信息。(独立运行模式)

```
<Sysname> display placement location slot 1
Program(s) placed at location: 1/0
    syslog
    cryptomgr
    trange
    lagg
    aaa
    acl
    copp
    lauth
    ethbase
    oap
    qos
    comsh
    xmlicfg
    ppp
    ipbase
    ipaddr
    rib
    staticroute
    telnet
    lldp
    te
```

```
pce
rsvp
ip6base
ip6addr
rib6
staticroute6
snmp

# 显示成员设备 1 的 0 号槽位主控上正在运行的进程信息。(IRF 模式)
<Sysname> display placement location chassis 1 slot 0
Program(s) placed at location: 1/1/0
    syslog
    cryptomgr
    trange
    lagg
    aaa
    acl
    copp
    lauth
    ethbase
    oap
    qos
    comsh
    xmldcfg
    ppp
    ipbase
    ipaddr
    rib
    staticroute
    telnet
    lldp
    te
    pce
    rsvp
    ip6base
    ip6addr
    rib6
    staticroute6
    snmp
```

### 1.1.7 display placement policy

**display placement policy** 命令用来显示进程的分布策略。

**【命令】**

**display placement policy program { program-name | all | default }**

**【视图】**

任意视图

## 【缺省用户角色】

network-admin  
network-operator

## 【参数】

**program-name:** 显示指定进程的分布策略，为 1~15 个字符的字符串，不区分大小写。

**all:** 显示所有配置的进程分布策略。

**default:** 显示用户配置的缺省分布策略的信息。如果没有通过 **placement program default** 配置，则没有显示信息。

## 【使用指导】

只有为进程成功配置分布策略后，才会输出相应的显示信息。

## 【举例】

```
# 显示缺省分布策略的信息。  
<Sysname> display placement policy program default  
Program: [default] : source  
-----  
    affinity location-set slot 0 cpu 0 attract : system [default]  
  
# 显示 aaa 进程的分布策略。  
<Sysname> display placement policy program aaa  
Program: aaa : source  
-----  
    affinity location-set slot 0 cpu 7 attract : system aaa  
    100  
    affinity location-set slot 0 cpu 1 attract : system aaa  
    100  
    affinity location-set slot 0 cpu 0 attract : system [default]  
    100
```

表1-2 display placement policy 命令显示信息描述表

字段	描述
Program	进程的名称以及进程的分布策略
source	进程分布策略的来源，其中：system [default]表示采用系统缺省分布策略，该策略是通过 <b>placement program default</b> 命令进入缺省分布策略视图后再配置的；system aaa表示采用AAA进程分布策略，该策略是通过 <b>placement program program-name</b> 命令进入AAA的分布策略视图后再配置的

## 1.1.8 display placement program

**display placement program** 命令用来显示主控进程的当前运行位置。

## 【命令】

**display placement program { program-name | all }**

## 【视图】

任意视图

## 【缺省用户角色】

network-admin  
network-operator

## 【参数】

**program-name:** 为当前设备上正在运行的进程的名称，为 1~15 个字符的字符串，不区分大小写。

**all:** 表示当前设备上运行的所有进程。

## 【举例】

```
# 显示 AAA 主控进程的当前运行位置。(独立运行模式)
<Sysname> display placement program aaa
Program           Placed at location
-----
aaa               0/0
# 显示 AAA 主控进程的当前运行位置。(IRF 模式)
<Sysname> display placement program aaa
Program           Placed at Location
-----
aaa               1/0/0
```

表1-3 display placement program 命令显示信息描述表

字段	描述
Program	进程的名称
Placed at location	主进程运行的位置 当显示为NA时表示该业务当前没有主进程（没有主进程的原因可能为：业务异常；主进程正在启动；主进程被关闭等）

## 1.1.9 display placement reoptimize

**display placement reoptimize** 命令用来显示进程分布优化后的预测位置。

## 【命令】

**display placement reoptimize program { program-name [ instance instance-name ] | all }**

## 【视图】

任意视图

## 【缺省用户角色】

network-admin  
network-operator

## 【参数】

**program-name:** 为当前设备上正在运行的、支持进程优化配置的进程的名称，为 1~15 个字符的字符串，不区分大小写。

**instance instance-name:** 表示实例名，为 1~31 个字符的字符串，不区分大小写。一个进程是否存在多个实例，由系统软件决定。

**all:** 表示当前设备上运行的、支持进程优化配置的所有进程。

### 【举例】

# 显示分布优化后所有进程的预测位置。(独立运行模式)

```
<Sysname> display placement reoptimize program all
Predicted changes to the placement
Program           Current location      New location
-----
rib                1/0                  1/0
staticroute        1/0                  1/0
te                 1/0                  1/0
pce                1/0                  1/0
rsvp               1/0                  1/0
rib6               1/0                  1/0
staticroute6       1/0                  1/0
```

以上显示信息中，**Program** 表示进程的名称，**Current location** 表示主进程当前运行的位置，**New location** 表示分布优化后，主进程将运行的位置。

# 显示分布优化后所有进程的预测位置。(IRF 模式)

```
<Sysname> display placement reoptimize program all
Predicted changes to the placement
Program           Current location      New location
-----
rib                1/1/0                1/1/0
staticroute        1/1/0                1/1/0
te                 1/1/0                1/1/0
pce                1/1/0                1/1/0
rsvp               1/1/0                1/1/0
rib6               1/1/0                1/1/0
staticroute6       1/1/0                1/1/0
```

以上显示信息中，**Program** 表示进程的名称，**Current location** 表示主进程当前运行的位置，**New location** 表示分布优化后，主进程将运行的位置。

## 1.1.10 placement program

**placement program** 命令用来进入指定进程的分布策略视图。

**undo placement program** 命令用来删除指定进程的分布策略。

### 【命令】

**placement program { program-name [ instance instance-name ] | default }**

**undo placement program { program-name [ instance instance-name ] | default }**

### 【缺省情况】

所有进程均未配置分布策略。当设备运行在独立运行模式时，所有进程的主控进程都在主用主控板上运行；当设备运行在 IRF 模式时，所有进程的主控进程都在全局主用主控板上运行。

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【参数】

**program-name:** 用来进入指定进程的分布策略视图。**program-name** 表示当前设备上正在运行的进程的名称，为 1~15 个字符的字符串，不区分大小写。

**instance instance-name:** 用来进入指定进程指定实例的分布策略视图。**instance-name** 表示实例名，为 1~31 个字符的字符串，不区分大小写。一个进程是否存在多个实例，由系统软件决定。

**default:** 用来进入缺省分布策略视图。进入该视图后，配置的是所有进程（所有实例）的缺省分布策略。

## 【使用指导】

为了提高系统的可靠性，系统在运行过程中会对进程进行 1:N 备份。当启动某个业务时，系统会自动同时为该业务运行一个主控进程和多个备份进程。

对于一些业务，其主控进程只能运行在主用主控板，这样的进程不支持进程分布优化配置（配置时会提示失败）。当主控进程异常时，系统会自动重启该主控进程。备份进程主要用于主备倒换和 ISSU 升级环境。

另一些业务，其主控进程可以运行在主用主控板上，也可以运行在备用主控板上。当主控进程异常时，需要从备份进程中选举一个新的主控进程，从而保证业务不受影响。在众多的备份进程中到底选用哪个作为新的主控进程，由该进程的分布策略决定。

分布策略的内容包括 **affinity location-type**、**affinity location-set**、**affinity program** 和 **affinity self**，这些命令从不同角度表达了用户对进程在某个位置运行的期望。

一个进程对应一个分布策略，所有的 **affinity** 命令可以同时设置。系统将根据用户的配置按照一定的算法，最后决定主控进程的预测位置（可以通过 **display placement reoptimize** 命令查看）。当发生主备倒换时，该位置的进程就能当选为主控进程，其它位置的进程则均为备份进程。

## 【举例】

```
# 进入 staticroute 分布策略视图。  
<Sysname> system-view  
[Sysname] placement program staticroute  
[Sysname-program-staticroute]  
  
# 进入缺省分布策略视图。  
<Sysname> system-view  
[Sysname] placement program default  
[Sysname-program-default]
```

### 1.1.11 placement reoptimize

**placement reoptimize** 命令用来优化进程运行位置，使进程分布策略生效。

## 【命令】

**placement reoptimize**

## 【视图】

系统视图

## 【缺省用户角色】

network-admin

## 【使用指导】

执行该命令后，系统会根据当前硬件的在位情况、主进程的运行位置和状态、分布策略的配置来综合计算主进程的新位置，并将该位置上的进程当选为主控进程，其它位置上的进程均为备份进程。如果新当选的主进程和原主进程不同，则会触发进程的主备倒换。在配置了 **NSR** 和 **GR** 的情况下，因为只是主备进程间角色的转换，进程不需要重启，所以进程的主备倒换不会造成业务中断。

执行此命令时请保持系统的稳定性，不建议在执行此命令的过程中进行任务涉及进程重启的操作。

如果待优化的进程配置了 **NSR**，需要确保 **NSR** 已完成备份同步，才能实现进程重分布时业务不中断。**NSR** 状态可通过 **display staticroute non-stop-routing status** 命令查看。

## 【举例】

# 手工进行进程分布优化。

```
<Sysname> system-view
[Sysname] placement reoptimize
Predicted changes to the placement
Program           Current location      New location
-----
rib                1/0                  1/0
staticroute        1/0                  1/0
te                 1/0                  1/0
pce                1/0                  1/0
rsvp               1/0                  1/0
rib6               1/0                  1/0
staticroute6       1/0                  1/0
Continue? [y/n]:y
Re-optimization of the placement start. You will be notified on completion.
Re-optimization of the placement complete. Use 'display placement' to view the new placement.
```